



# NVIDIA OptiX 7.4

## API Reference Manual

2 November 2021  
Version 7.4



# Contents

<b>1</b>	<b>Module Index</b>	<b>1</b>
1.1	Modules . . . . .	1
<b>2</b>	<b>Class Index</b>	<b>1</b>
2.1	Class List . . . . .	1
<b>3</b>	<b>Module Documentation</b>	<b>3</b>
3.1	Device API . . . . .	3
3.2	Host API . . . . .	106
3.3	Error handling . . . . .	107
3.4	Device context . . . . .	108
3.5	Pipelines . . . . .	113
3.6	Modules . . . . .	115
3.7	Tasks . . . . .	118
3.8	Program groups . . . . .	119
3.9	Launches . . . . .	121
3.10	Acceleration structures . . . . .	123
3.11	Denoiser . . . . .	128
3.12	Types . . . . .	134
3.13	Function Table . . . . .	174
3.14	Utilities . . . . .	175
<b>4</b>	<b>Namespace Documentation</b>	<b>182</b>
4.1	optix_impl Namespace Reference . . . . .	182
<b>5</b>	<b>Class Documentation</b>	<b>186</b>
5.1	OptixAabb Struct Reference . . . . .	186
5.2	OptixAccelBufferSizes Struct Reference . . . . .	187
5.3	OptixAccelBuildOptions Struct Reference . . . . .	188
5.4	OptixAccelEmitDesc Struct Reference . . . . .	189
5.5	OptixAccelRelocationInfo Struct Reference . . . . .	189
5.6	OptixBuildInput Struct Reference . . . . .	190
5.7	OptixBuildInputCurveArray Struct Reference . . . . .	191
5.8	OptixBuildInputCustomPrimitiveArray Struct Reference . . . . .	193
5.9	OptixBuildInputInstanceArray Struct Reference . . . . .	194

5.10	OptixBuildInputTriangleArray Struct Reference	195
5.11	OptixBuiltinISOOptions Struct Reference	197
5.12	OptixDenoiserGuideLayer Struct Reference	198
5.13	OptixDenoiserLayer Struct Reference	199
5.14	OptixDenoiserOptions Struct Reference	199
5.15	OptixDenoiserParams Struct Reference	200
5.16	OptixDenoiserSizes Struct Reference	201
5.17	OptixDeviceContextOptions Struct Reference	201
5.18	OptixFunctionTable Struct Reference	202
5.19	OptixImage2D Struct Reference	211
5.20	OptixInstance Struct Reference	212
5.21	OptixMatrixMotionTransform Struct Reference	213
5.22	OptixModuleCompileBoundValueEntry Struct Reference	214
5.23	OptixModuleCompileOptions Struct Reference	215
5.24	OptixMotionOptions Struct Reference	216
5.25	OptixPayloadType Struct Reference	217
5.26	OptixPipelineCompileOptions Struct Reference	217
5.27	OptixPipelineLinkOptions Struct Reference	218
5.28	OptixProgramGroupCallables Struct Reference	219
5.29	OptixProgramGroupDesc Struct Reference	220
5.30	OptixProgramGroupHitgroup Struct Reference	221
5.31	OptixProgramGroupOptions Struct Reference	222
5.32	OptixProgramGroupSingleModule Struct Reference	223
5.33	OptixShaderBindingTable Struct Reference	223
5.34	OptixSRTData Struct Reference	225
5.35	OptixSRTMotionTransform Struct Reference	227
5.36	OptixStackSizes Struct Reference	228
5.37	OptixStaticTransform Struct Reference	230
5.38	OptixUtilDenoiserImageTile Struct Reference	230
<b>6</b>	<b>File Documentation</b>	<b>231</b>
6.1	optix.h File Reference	231
6.2	optix_7_device.h File Reference	232
6.3	optix_7_device_impl.h File Reference	251
6.4	optix_7_device_impl_exception.h File Reference	331

6.5	optix_7_device_impl_transformations.h File Reference . . . . .	332
6.6	optix_7_host.h File Reference . . . . .	334
6.7	optix_7_types.h File Reference . . . . .	336
6.8	optix_denoiser_tiling.h File Reference . . . . .	346
6.9	optix_device.h File Reference . . . . .	347
6.10	optix_function_table.h File Reference . . . . .	347
6.11	optix_function_table_definition.h File Reference . . . . .	348
6.12	optix_host.h File Reference . . . . .	348
6.13	optix_stack_size.h File Reference . . . . .	349
6.14	optix_stubs.h File Reference . . . . .	349
6.15	optix_types.h File Reference . . . . .	350

<b>Index</b>	<b>352</b>
--------------	------------

# 1 Module Index

## 1.1 Modules

Here is a list of all modules:

Device API	3
Host API	106
Error handling	107
Device context	108
Pipelines	113
Modules	115
Tasks	118
Program groups	119
Launches	121
Acceleration structures	123
Denoiser	128
Types	134
Function Table	174
Utilities	175

# 2 Class Index

## 2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

<a href="#">OptixAabb</a>	
AABB inputs	186
<a href="#">OptixAccelBufferSizes</a>	
Struct for querying builder allocation requirements	187
<a href="#">OptixAccelBuildOptions</a>	
Build options for acceleration structures	188
<a href="#">OptixAccelEmitDesc</a>	
Specifies a type and output destination for emitted post-build properties	189
<a href="#">OptixAccelRelocationInfo</a>	
Used to store information related to relocation of acceleration structures	189
<a href="#">OptixBuildInput</a>	
Build inputs	190
<a href="#">OptixBuildInputCurveArray</a>	
Curve inputs	191

<a href="#">OptixBuildInputCustomPrimitiveArray</a>	
Custom primitive inputs	193
<a href="#">OptixBuildInputInstanceArray</a>	
Instance and instance pointer inputs	194
<a href="#">OptixBuildInputTriangleArray</a>	
Triangle inputs	195
<a href="#">OptixBuiltinISOOptions</a>	
Specifies the options for retrieving an intersection program for a built-in primitive type. The primitive type must not be <code>OPTIX_PRIMITIVE_TYPE_CUSTOM</code>	197
<a href="#">OptixDenoiserGuideLayer</a>	
Guide layer for the denoiser	198
<a href="#">OptixDenoiserLayer</a>	
Input/Output layers for the denoiser	199
<a href="#">OptixDenoiserOptions</a>	
Options used by the denoiser	199
<a href="#">OptixDenoiserParams</a>	
Various parameters used by the denoiser	200
<a href="#">OptixDenoiserSizes</a>	
Various sizes related to the denoiser	201
<a href="#">OptixDeviceContextOptions</a>	
Parameters used for <code>optixDeviceContextCreate()</code>	201
<a href="#">OptixFunctionTable</a>	
The function table containing all API functions	202
<a href="#">OptixImage2D</a>	
Image descriptor used by the denoiser	211
<a href="#">OptixInstance</a>	
Instances	212
<a href="#">OptixMatrixMotionTransform</a>	
Represents a matrix motion transformation	213
<a href="#">OptixModuleCompileBoundValueEntry</a>	
Struct for specifying specializations for pipelineParams as specified in <code>OptixPipelineCompileOptions::pipelineLaunchParamsVariableName</code>	214
<a href="#">OptixModuleCompileOptions</a>	
Compilation options for module	215
<a href="#">OptixMotionOptions</a>	
Motion options	216
<a href="#">OptixPayloadType</a>	
Specifies a single payload type	217
<a href="#">OptixPipelineCompileOptions</a>	
Compilation options for all modules of a pipeline	217

<a href="#">OptixPipelineLinkOptions</a>	
Link options for a pipeline	218
<a href="#">OptixProgramGroupCallables</a>	
Program group representing callables	219
<a href="#">OptixProgramGroupDesc</a>	
Descriptor for program groups	220
<a href="#">OptixProgramGroupHitgroup</a>	
Program group representing the hitgroup	221
<a href="#">OptixProgramGroupOptions</a>	
Program group options	222
<a href="#">OptixProgramGroupSingleModule</a>	
Program group representing a single module	223
<a href="#">OptixShaderBindingTable</a>	
Describes the shader binding table (SBT)	223
<a href="#">OptixSRTData</a>	
Represents an SRT transformation	225
<a href="#">OptixSRTMotionTransform</a>	
Represents an SRT motion transformation	227
<a href="#">OptixStackSizes</a>	
Describes the stack size requirements of a program group	228
<a href="#">OptixStaticTransform</a>	
Static transform	230
<a href="#">OptixUtilDenoiserImageTile</a>	
Tile definition	230

## 3 Module Documentation

### 3.1 Device API

#### Functions

- static `__forceinline__`  
`__device__ void optixTrace (OptixTraversableHandle handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, OptixVisibilityMask visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex)`
- static `__forceinline__`  
`__device__ void optixTrace (OptixTraversableHandle handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, OptixVisibilityMask visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex, unsigned int &p0)`
- static `__forceinline__`  
`__device__ void optixTrace (OptixTraversableHandle handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, OptixVisibilityMask visibilityMask, unsigned int rayFlags,`

[illegible]



- [illegible]



```
unsigned int &p6, unsigned int &p7, unsigned int &p8, unsigned int &p9, unsigned int &p10,
unsigned int &p11, unsigned int &p12, unsigned int &p13, unsigned int &p14, unsigned int &p15,
unsigned int &p16, unsigned int &p17, unsigned int &p18, unsigned int &p19, unsigned int &p20,
unsigned int &p21, unsigned int &p22)
```

- [illegible]









- [illegible]







&p9, unsigned int &p10, unsigned int &p11, unsigned int &p12, unsigned int &p13, unsigned int &p14, unsigned int &p15, unsigned int &p16, unsigned int &p17, unsigned int &p18, unsigned int &p19, unsigned int &p20, unsigned int &p21, unsigned int &p22, unsigned int &p23, unsigned int &p24, unsigned int &p25, unsigned int &p26, unsigned int &p27, unsigned int &p28, unsigned int &p29, unsigned int &p30)

- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixTrace](#) ([OptixPayloadTypeID](#) type, [OptixTraversableHandle](#) handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, [OptixVisibilityMask](#) visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex, unsigned int &p0, unsigned int &p1, unsigned int &p2, unsigned int &p3, unsigned int &p4, unsigned int &p5, unsigned int &p6, unsigned int &p7, unsigned int &p8, unsigned int &p9, unsigned int &p10, unsigned int &p11, unsigned int &p12, unsigned int &p13, unsigned int &p14, unsigned int &p15, unsigned int &p16, unsigned int &p17, unsigned int &p18, unsigned int &p19, unsigned int &p20, unsigned int &p21, unsigned int &p22, unsigned int &p23, unsigned int &p24, unsigned int &p25, unsigned int &p26, unsigned int &p27, unsigned int &p28, unsigned int &p29, unsigned int &p30, unsigned int &p31)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_0](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_1](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_2](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_3](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_4](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_5](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_6](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_7](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_8](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_9](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_10](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_11](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_12](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_13](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_14](#) (unsigned int p)

- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_15](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_16](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_17](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_18](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_19](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_20](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_21](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_22](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_23](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_24](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_25](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_26](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_27](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_28](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_29](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_30](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_31](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_0](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_1](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_2](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_3](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_4](#) ()

- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_5 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_6 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_7 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_8 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_9 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_10 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_11 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_12 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_13 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_14 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_15 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_16 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_17 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_18 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_19 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_20 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_21 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_22 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_23 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_24 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_25 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_26 \(\)](#)

- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_27](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_28](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_29](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_30](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_31](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayloadTypes](#) (unsigned int typeMask)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixUndefinedValue](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 [optixGetWorldRayOrigin](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 [optixGetWorldRayDirection](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 [optixGetObjectRayOrigin](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 [optixGetObjectRayDirection](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float [optixGetRayTmin](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float [optixGetRayTmax](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float [optixGetRayTime](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetRayFlags](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetRayVisibilityMask](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_  
[OptixTraversableHandle](#) [optixGetInstanceTraversableFromIAS](#) ([OptixTraversableHandle](#) ias,  
unsigned int instIdx)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixGetTriangleVertexData](#) ([OptixTraversableHandle](#) gas, unsigned int primIdx,  
unsigned int sbtGASIndex, float time, float3 data[3])
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixGetLinearCurveVertexData](#) ([OptixTraversableHandle](#) gas, unsigned int  
primIdx, unsigned int sbtGASIndex, float time, float4 data[2])
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixGetQuadraticBSplineVertexData](#) ([OptixTraversableHandle](#) gas, unsigned int  
primIdx, unsigned int sbtGASIndex, float time, float4 data[3])

- static \_\_forceinline\_\_  
\_\_device\_\_ void `optixGetCubicBSplineVertexData` (`OptixTraversableHandle` gas, unsigned int primIdx, unsigned int sbtGASIndex, float time, float4 data[4])
- static \_\_forceinline\_\_  
\_\_device\_\_ void `optixGetCatmullRomVertexData` (`OptixTraversableHandle` gas, unsigned int primIdx, unsigned int sbtGASIndex, float time, float4 data[4])
- static \_\_forceinline\_\_  
\_\_device\_\_  
`OptixTraversableHandle` `optixGetGASTraversableHandle` ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float `optixGetGASMotionTimeBegin` (`OptixTraversableHandle` gas)
- static \_\_forceinline\_\_  
\_\_device\_\_ float `optixGetGASMotionTimeEnd` (`OptixTraversableHandle` gas)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int `optixGetGASMotionStepCount` (`OptixTraversableHandle` gas)
- static \_\_forceinline\_\_  
\_\_device\_\_ void `optixGetWorldToObjectTransformMatrix` (float m[12])
- static \_\_forceinline\_\_  
\_\_device\_\_ void `optixGetObjectToWorldTransformMatrix` (float m[12])
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 `optixTransformPointFromWorldToObjectSpace` (float3 point)
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 `optixTransformVectorFromWorldToObjectSpace` (float3 vec)
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 `optixTransformNormalFromWorldToObjectSpace` (float3 normal)
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 `optixTransformPointFromObjectToWorldSpace` (float3 point)
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 `optixTransformVectorFromObjectToWorldSpace` (float3 vec)
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 `optixTransformNormalFromObjectToWorldSpace` (float3 normal)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int `optixGetTransformListSize` ()
- static \_\_forceinline\_\_  
\_\_device\_\_  
`OptixTraversableHandle` `optixGetTransformListHandle` (unsigned int index)
- static \_\_forceinline\_\_  
\_\_device\_\_ `OptixTransformType` `optixGetTransformTypeFromHandle` (`OptixTraversableHandle` handle)
- static \_\_forceinline\_\_  
\_\_device\_\_ const  
`OptixStaticTransform` \* `optixGetStaticTransformFromHandle` (`OptixTraversableHandle` handle)
- static \_\_forceinline\_\_  
\_\_device\_\_ const  
`OptixSRTMotionTransform` \* `optixGetSRTMotionTransformFromHandle` (`OptixTraversableHandle` handle)

- static \_\_forceinline\_\_  
\_\_device\_\_ const  
OptixMatrixMotionTransform \* optixGetMatrixMotionTransformFromHandle  
(OptixTraversableHandle handle)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int optixGetInstanceIdFromHandle (OptixTraversableHandle handle)
- static \_\_forceinline\_\_  
\_\_device\_\_  
OptixTraversableHandle optixGetInstanceChildFromHandle (OptixTraversableHandle handle)
- static \_\_forceinline\_\_  
\_\_device\_\_ const float4 \* optixGetInstanceTransformFromHandle (OptixTraversableHandle  
handle)
- static \_\_forceinline\_\_  
\_\_device\_\_ const float4 \* optixGetInstanceInverseTransformFromHandle  
(OptixTraversableHandle handle)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool optixReportIntersection (float hitT, unsigned int hitKind)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool optixReportIntersection (float hitT, unsigned int hitKind, unsigned int a0)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool optixReportIntersection (float hitT, unsigned int hitKind, unsigned int a0,  
unsigned int a1)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool optixReportIntersection (float hitT, unsigned int hitKind, unsigned int a0,  
unsigned int a1, unsigned int a2)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool optixReportIntersection (float hitT, unsigned int hitKind, unsigned int a0,  
unsigned int a1, unsigned int a2, unsigned int a3)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool optixReportIntersection (float hitT, unsigned int hitKind, unsigned int a0,  
unsigned int a1, unsigned int a2, unsigned int a3, unsigned int a4)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool optixReportIntersection (float hitT, unsigned int hitKind, unsigned int a0,  
unsigned int a1, unsigned int a2, unsigned int a3, unsigned int a4, unsigned int a5)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool optixReportIntersection (float hitT, unsigned int hitKind, unsigned int a0,  
unsigned int a1, unsigned int a2, unsigned int a3, unsigned int a4, unsigned int a5, unsigned int  
a6)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool optixReportIntersection (float hitT, unsigned int hitKind, unsigned int a0,  
unsigned int a1, unsigned int a2, unsigned int a3, unsigned int a4, unsigned int a5, unsigned int  
a6, unsigned int a7)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int optixGetAttribute\_0 ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int optixGetAttribute\_1 ()

- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetAttribute\\_2](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetAttribute\\_3](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetAttribute\\_4](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetAttribute\\_5](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetAttribute\\_6](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetAttribute\\_7](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixTerminateRay](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixIgnoreIntersection](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPrimitiveIndex](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetSbtGASIndex](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetInstanceId](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetInstanceIndex](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetHitKind](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ [OptixPrimitiveType](#) [optixGetPrimitiveType](#) (unsigned int hitKind)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixIsFrontFaceHit](#) (unsigned int hitKind)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixIsBackFaceHit](#) (unsigned int hitKind)
- static \_\_forceinline\_\_  
\_\_device\_\_ [OptixPrimitiveType](#) [optixGetPrimitiveType](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixIsFrontFaceHit](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixIsBackFaceHit](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixIsTriangleHit](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixIsTriangleFrontFaceHit](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixIsTriangleBackFaceHit](#) ()



- static \_\_forceinline\_\_  
\_\_device\_\_ float2 [optixGetTriangleBarycentrics](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float [optixGetCurveParameter](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ uint3 [optixGetLaunchIndex](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ uint3 [optixGetLaunchDimensions](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ [CUdeviceptr](#) [optixGetSbtDataPointer](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode, unsigned int exceptionDetail0)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode, unsigned int exceptionDetail0,  
unsigned int exceptionDetail1)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode, unsigned int exceptionDetail0,  
unsigned int exceptionDetail1, unsigned int exceptionDetail2)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode, unsigned int exceptionDetail0,  
unsigned int exceptionDetail1, unsigned int exceptionDetail2, unsigned int exceptionDetail3)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode, unsigned int exceptionDetail0,  
unsigned int exceptionDetail1, unsigned int exceptionDetail2, unsigned int exceptionDetail3,  
unsigned int exceptionDetail4)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode, unsigned int exceptionDetail0,  
unsigned int exceptionDetail1, unsigned int exceptionDetail2, unsigned int exceptionDetail3,  
unsigned int exceptionDetail4, unsigned int exceptionDetail5)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode, unsigned int exceptionDetail0,  
unsigned int exceptionDetail1, unsigned int exceptionDetail2, unsigned int exceptionDetail3,  
unsigned int exceptionDetail4, unsigned int exceptionDetail5, unsigned int exceptionDetail6)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode, unsigned int exceptionDetail0,  
unsigned int exceptionDetail1, unsigned int exceptionDetail2, unsigned int exceptionDetail3,  
unsigned int exceptionDetail4, unsigned int exceptionDetail5, unsigned int exceptionDetail6,  
unsigned int exceptionDetail7)
- static \_\_forceinline\_\_  
\_\_device\_\_ int [optixGetExceptionCode](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetExceptionDetail\\_0](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetExceptionDetail\\_1](#) ()

- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetExceptionDetail\\_2](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetExceptionDetail\\_3](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetExceptionDetail\\_4](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetExceptionDetail\\_5](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetExceptionDetail\\_6](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetExceptionDetail\\_7](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_  
[OptixTraversableHandle](#) [optixGetExceptionInvalidTraversable](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ int [optixGetExceptionInvalidSbtOffset](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_  
[OptixInvalidRayExceptionDetails](#) [optixGetExceptionInvalidRay](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_  
[OptixParameterMismatchExceptionDetails](#) [optixGetExceptionParameterMismatch](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ char \* [optixGetExceptionLineInfo](#) ()
- template<typename ReturnT , typename... ArgTypes>  
static \_\_forceinline\_\_  
\_\_device\_\_ ReturnT [optixDirectCall](#) (unsigned int sbtIndex, ArgTypes...args)
- template<typename ReturnT , typename... ArgTypes>  
static \_\_forceinline\_\_  
\_\_device\_\_ ReturnT [optixContinuationCall](#) (unsigned int sbtIndex, ArgTypes...args)
- static \_\_forceinline\_\_  
\_\_device\_\_ uint4 [optixTexFootprint2D](#) (unsigned long long tex, unsigned int texInfo, float x, float y,  
unsigned int \*singleMipLevel)
- static \_\_forceinline\_\_  
\_\_device\_\_ uint4 [optixTexFootprint2DLod](#) (unsigned long long tex, unsigned int texInfo, float x,  
float y, float level, bool coarse, unsigned int \*singleMipLevel)
- static \_\_forceinline\_\_  
\_\_device\_\_ uint4 [optixTexFootprint2DGrad](#) (unsigned long long tex, unsigned int texInfo, float x,  
float y, float dPdx\_x, float dPdx\_y, float dPdy\_x, float dPdy\_y, bool coarse, unsigned int  
\*singleMipLevel)

### 3.1.1 Detailed Description

OptiX Device API.

### 3.1.2 Function Documentation

#### 3.1.2.1 `template<typename ReturnT , typename... ArgTypes> static __forceinline__ __device__ ReturnT optixContinuationCall (     unsigned int sbtIndex,     ArgTypes... args ) [static]`

Creates a call to the continuation callable program at the specified SBT entry.

This will call the program that was specified in the `OptixProgramGroupCallables::entryFunctionNameCC` in the module specified by `OptixProgramGroupCallables::moduleCC`. The address of the SBT entry is calculated by `OptixShaderBindingTable::callablesRecordBase + ( OptixShaderBindingTable::callablesRecordStrideInBytes * sbtIndex )`. As opposed to direct callable programs, continuation callable programs are allowed to call `optixTrace` recursively.

Behavior is undefined if there is no continuation callable program at the specified SBT entry.

Behavior is undefined if the number of arguments that are being passed in does not match the number of parameters expected by the program that is called. In that case an exception of type `OPTIX_EXCEPTION_CODE_CALLABLE_PARAMETER_MISMATCH` will be thrown if `OPTIX_EXCEPTION_FLAG_DEBUG` was specified for the `OptixPipelineCompileOptions::exceptionFlags`.

#### Parameters

in	<i>sbtIndex</i>	The offset of the SBT entry of the continuation callable program to call relative to <code>OptixShaderBindingTable::callablesRecordBase</code> .
in	<i>args</i>	The arguments to pass to the continuation callable program.

#### 3.1.2.2 `template<typename ReturnT , typename... ArgTypes> static __forceinline__ __device__ ReturnT optixDirectCall (     unsigned int sbtIndex,     ArgTypes... args ) [static]`

Creates a call to the direct callable program at the specified SBT entry.

This will call the program that was specified in the `OptixProgramGroupCallables::entryFunctionNameDC` in the module specified by `OptixProgramGroupCallables::moduleDC`. The address of the SBT entry is calculated by `OptixShaderBindingTable::callablesRecordBase + ( OptixShaderBindingTable::callablesRecordStrideInBytes * sbtIndex )`.

Behavior is undefined if there is no direct callable program at the specified SBT entry.

Behavior is undefined if the number of arguments that are being passed in does not match the number of parameters expected by the program that is called. In that case an exception of type `OPTIX_EXCEPTION_CODE_CALLABLE_PARAMETER_MISMATCH` will be thrown if `OPTIX_EXCEPTION_FLAG_DEBUG` was specified for the `OptixPipelineCompileOptions::exceptionFlags`.

### Parameters

in	<i>sbtIndex</i>	The offset of the SBT entry of the direct callable program to call relative to <a href="#">OptixShaderBindingTable::callablesRecordBase</a> .
in	<i>args</i>	The arguments to pass to the direct callable program.

**3.1.2.3 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetAttribute\_0 ( ) [static]**

Returns the attribute at slot 0.

**3.1.2.4 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetAttribute\_1 ( ) [static]**

Returns the attribute at slot 1.

**3.1.2.5 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetAttribute\_2 ( ) [static]**

Returns the attribute at slot 2.

**3.1.2.6 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetAttribute\_3 ( ) [static]**

Returns the attribute at slot 3.

**3.1.2.7 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetAttribute\_4 ( ) [static]**

Returns the attribute at slot 4.

**3.1.2.8 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetAttribute\_5 ( ) [static]**

Returns the attribute at slot 5.

**3.1.2.9 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetAttribute\_6 ( ) [static]**

Returns the attribute at slot 6.

**3.1.2.10 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetAttribute\_7 ( ) [static]**

Returns the attribute at slot 7.

**3.1.2.11 static \_\_forceinline\_\_ \_\_device\_\_ void optixGetCatmullRomVertexData (**  
**OptixTraversableHandle *gas*,**  
**unsigned int *primIdx*,**  
**unsigned int *sbtGASIndex*,**  
**float *time*,**  
**float4 *data*[4] ) [static]**

Return the object space curve control vertex data of a CatmullRom spline curve in a Geometry Acceleration Structure (GAS) at a given motion time. To access vertex data, the GAS must be built using the flag OPTIX\_BUILD\_FLAG\_ALLOW\_RANDOM\_VERTEX\_ACCESS.

*data*[*i*] = {*x*,*y*,*z*,*w*} with {*x*,*y*,*z*} the position and *w* the radius of control vertex *i*. If motion is disabled via

[OptixPipelineCompileOptions::usesMotionBlur](#), or the GAS does not contain motion, the time parameter is ignored.

```
3.1.2.12 static __forceinline__ __device__ void optixGetCubicBSplineVertexData (  

    OptixTraversableHandle gas,  

    unsigned int primIdx,  

    unsigned int sbtGASIndex,  

    float time,  

    float4 data[4] ) [static]
```

Return the object space curve control vertex data of a cubic BSpline curve in a Geometry Acceleration Structure (GAS) at a given motion time. To access vertex data, the GAS must be built using the flag OPTIX\_BUILD\_FLAG\_ALLOW\_RANDOM\_VERTEX\_ACCESS.

$data[i] = \{x,y,z,w\}$  with  $\{x,y,z\}$  the position and  $w$  the radius of control vertex  $i$ . If motion is disabled via [OptixPipelineCompileOptions::usesMotionBlur](#), or the GAS does not contain motion, the time parameter is ignored.

```
3.1.2.13 static __forceinline__ __device__ float optixGetCurveParameter ( ) [static]
```

Convenience function that returns the curve parameter.

When using [OptixBuildInputCurveArray](#) objects, during intersection the curve parameter is stored into the first attribute register.

```
3.1.2.14 static __forceinline__ __device__ int optixGetExceptionCode ( ) [static]
```

Returns the exception code.

Only available in EX.

```
3.1.2.15 static __forceinline__ __device__ unsigned int optixGetExceptionDetail_0 ( )  

    [static]
```

Returns the 32-bit exception detail at slot 0.

The behavior is undefined if the exception is not a user exception, or the used overload [optixThrowException\(\)](#) did not provide the queried exception detail.

Only available in EX.

```
3.1.2.16 static __forceinline__ __device__ unsigned int optixGetExceptionDetail_1 ( )  

    [static]
```

Returns the 32-bit exception detail at slot 1.

See Also

[optixGetExceptionDetail\\_0\(\)](#)

**3.1.2.17** `static __forceinline__ __device__ unsigned int optixGetExceptionDetail_2 ( )`  
`[static]`

Returns the 32-bit exception detail at slot 2.

See Also

[optixGetExceptionDetail\\_0\(\)](#)

**3.1.2.18** `static __forceinline__ __device__ unsigned int optixGetExceptionDetail_3 ( )`  
`[static]`

Returns the 32-bit exception detail at slot 3.

See Also

[optixGetExceptionDetail\\_0\(\)](#)

**3.1.2.19** `static __forceinline__ __device__ unsigned int optixGetExceptionDetail_4 ( )`  
`[static]`

Returns the 32-bit exception detail at slot 4.

See Also

[optixGetExceptionDetail\\_0\(\)](#)

**3.1.2.20** `static __forceinline__ __device__ unsigned int optixGetExceptionDetail_5 ( )`  
`[static]`

Returns the 32-bit exception detail at slot 5.

See Also

[optixGetExceptionDetail\\_0\(\)](#)

**3.1.2.21** `static __forceinline__ __device__ unsigned int optixGetExceptionDetail_6 ( )`  
`[static]`

Returns the 32-bit exception detail at slot 6.

See Also

[optixGetExceptionDetail\\_0\(\)](#)

**3.1.2.22** `static __forceinline__ __device__ unsigned int optixGetExceptionDetail_7 ( )`  
`[static]`

Returns the 32-bit exception detail at slot 7.

See Also

[optixGetExceptionDetail\\_0\(\)](#)

**3.1.2.23** `static __forceinline__ __device__ OptixInvalidRayExceptionDetails`  
`optixGetExceptionInvalidRay ( ) [static]`

Returns the invalid ray for exceptions with exception code `OPTIX_EXCEPTION_CODE_INVALID_RAY`. Exceptions of type `OPTIX_EXCEPTION_CODE_INVALID_RAY` are thrown when one or more values that were passed into `optixTrace` are either `inf` or `nan`.

`OptixInvalidRayExceptionDetails::rayTime` will always be 0 if

[OptixPipelineCompileOptions::usesMotionBlur](#) is 0. Values in the returned struct are all zero for all other exception codes.

Only available in EX.

**3.1.2.24** `static __forceinline__ __device__ int optixGetExceptionInvalidSbtOffset ( ) [static]`

Returns the invalid sbt offset for exceptions with exception code `OPTIX_EXCEPTION_CODE_TRAVERSAL_INVALID_MISS_SBT` and `OPTIX_EXCEPTION_CODE_TRAVERSAL_INVALID_HIT_SBT`.

Returns zero for all other exception codes.

Only available in EX.

**3.1.2.25** `static __forceinline__ __device__ OptixTraversableHandle`  
`optixGetExceptionInvalidTraversable ( ) [static]`

Returns the invalid traversable handle for exceptions with exception code `OPTIX_EXCEPTION_CODE_TRAVERSAL_INVALID_TRAVERSABLE`.

Returns zero for all other exception codes.

Only available in EX.

**3.1.2.26** `static __forceinline__ __device__ char* optixGetExceptionLineInfo ( ) [static]`

Returns a string that includes information about the source location that caused the current exception.

The source location is only available for exceptions of type

`OPTIX_EXCEPTION_CODE_CALLABLE_PARAMETER_MISMATCH`, `OPTIX_EXCEPTION_CODE_UNSUPPORTED_PRIMITIVE_TYPE`, `OPTIX_EXCEPTION_CODE_INVALID_RAY`, and for user exceptions. Line information needs to be present in the input PTX and [OptixModuleCompileOptions::debugLevel](#) may not be set to `OPTIX_COMPILE_DEBUG_LEVEL_NONE`.

Returns a NULL pointer if no line information is available.

Only available in EX.

### 3.1.2.27 **static \_\_forceinline\_\_ \_\_device\_\_ OptixParameterMismatchExceptionDetails optixGetExceptionParameterMismatch ( ) [static]**

Returns information about an exception with code  
OPTIX\_EXCEPTION\_CODE\_CALLABLE\_PARAMETER\_MISMATCH.

Exceptions of type OPTIX\_EXCEPTION\_CODE\_CALLABLE\_PARAMETER\_MISMATCH are called when the number of arguments that were passed into a call to `optixDirectCall` or `optixContinuationCall` does not match the number of parameters of the callable that is called. Note that the parameters are packed by OptiX into individual 32 bit values, so the number of expected and passed values may not correspond to the number of arguments passed into `optixDirectCall` or `optixContinuationCall`.

Values in the returned struct are all zero for all other exception codes.

Only available in EX.

### 3.1.2.28 **static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetGASMotionStepCount ( OptixTraversableHandle *gas* ) [static]**

Returns the number of motion steps of a GAS (see [OptixMotionOptions](#))

### 3.1.2.29 **static \_\_forceinline\_\_ \_\_device\_\_ float optixGetGASMotionTimeBegin ( OptixTraversableHandle *gas* ) [static]**

Returns the motion begin time of a GAS (see [OptixMotionOptions](#))

### 3.1.2.30 **static \_\_forceinline\_\_ \_\_device\_\_ float optixGetGASMotionTimeEnd ( OptixTraversableHandle *gas* ) [static]**

Returns the motion end time of a GAS (see [OptixMotionOptions](#))

### 3.1.2.31 **static \_\_forceinline\_\_ \_\_device\_\_ OptixTraversableHandle optixGetGASTraversableHandle ( ) [static]**

Returns the traversable handle for the Geometry Acceleration Structure (GAS) containing the current hit. May be called from IS, AH and CH.

### 3.1.2.32 **static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetHitKind ( ) [static]**

Returns the 8 bit hit kind associated with the current hit.

Use [optixGetPrimitiveType\(\)](#) to interpret the hit kind. For custom intersections (primitive type OPTIX\_PRIMITIVE\_TYPE\_CUSTOM), this is the 7-bit hitKind passed to [optixReportIntersection\(\)](#). Hit kinds greater than 127 are reserved for built-in primitives.

Available only in AH and CH.

### 3.1.2.33 **static \_\_forceinline\_\_ \_\_device\_\_ OptixTraversableHandle optixGetInstanceChildFromHandle ( OptixTraversableHandle *handle* ) [static]**

Returns child traversable handle from an [OptixInstance](#) traversable.



Returns 0 if the traversable handle does not reference an [OptixInstance](#).

#### 3.1.2.34 **static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetInstanceId ( ) [static]**

Returns the [OptixInstance::instanceId](#) of the instance within the top level acceleration structure associated with the current intersection.

When building an acceleration structure using [OptixBuildInputInstanceArray](#) each [OptixInstance](#) has a user supplied instanceId. [OptixInstance](#) objects reference another acceleration structure. During traversal the acceleration structures are visited top down. In the IS and AH programs the [OptixInstance::instanceId](#) corresponding to the most recently visited [OptixInstance](#) is returned when calling [optixGetInstanceId\(\)](#). In CH [optixGetInstanceId\(\)](#) returns the [OptixInstance::instanceId](#) when the hit was recorded with [optixReportIntersection](#). In the case where there is no [OptixInstance](#) visited, [optixGetInstanceId](#) returns ~0u

#### 3.1.2.35 **static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetInstanceIdFromHandle ( OptixTraversableHandle *handle* ) [static]**

Returns instanceId from an [OptixInstance](#) traversable.

Returns 0 if the traversable handle does not reference an [OptixInstance](#).

#### 3.1.2.36 **static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetInstanceIndex ( ) [static]**

Returns the zero-based index of the instance within its instance acceleration structure associated with the current intersection.

In the IS and AH programs the index corresponding to the most recently visited [OptixInstance](#) is returned when calling [optixGetInstanceIndex\(\)](#). In CH [optixGetInstanceIndex\(\)](#) returns the index when the hit was recorded with [optixReportIntersection](#). In the case where there is no [OptixInstance](#) visited, [optixGetInstanceIndex](#) returns 0

#### 3.1.2.37 **static \_\_forceinline\_\_ \_\_device\_\_ const float4\* optixGetInstanceInverseTransform- FromHandle ( OptixTraversableHandle *handle* ) [static]**

Returns world-to-object transform from an [OptixInstance](#) traversable.

Returns 0 if the traversable handle does not reference an [OptixInstance](#).

#### 3.1.2.38 **static \_\_forceinline\_\_ \_\_device\_\_ const float4\* optixGetInstanceTransformFromHandle ( OptixTraversableHandle *handle* ) [static]**

Returns object-to-world transform from an [OptixInstance](#) traversable.

Returns 0 if the traversable handle does not reference an [OptixInstance](#).

#### 3.1.2.39 **static \_\_forceinline\_\_ \_\_device\_\_ OptixTraversableHandle optixGetInstanceTraversableFromIAS ( OptixTraversableHandle *ias*,**

**unsigned int *instIdx* ) [static]**

Return the traversable handle of a given instance in an Instance Acceleration Structure (IAS)

**3.1.2.40 static \_\_forceinline\_\_ \_\_device\_\_ uint3 optixGetLaunchDimensions ( ) [static]**

Available in any program, it returns the dimensions of the current launch specified by `optixLaunch` on the host.

**3.1.2.41 static \_\_forceinline\_\_ \_\_device\_\_ uint3 optixGetLaunchIndex ( ) [static]**

Available in any program, it returns the current launch index within the launch dimensions specified by `optixLaunch` on the host.

The raygen program is typically only launched once per launch index.

**3.1.2.42 static \_\_forceinline\_\_ \_\_device\_\_ void optixGetLinearCurveVertexData (**  
**OptixTraversableHandle *gas*,**  
**unsigned int *primIdx*,**  
**unsigned int *sbtGASIndex*,**  
**float *time*,**  
**float4 *data*[2] ) [static]**

Return the object space curve control vertex data of a linear curve in a Geometry Acceleration Structure (GAS) at a given motion time. To access vertex data, the GAS must be built using the flag `OPTIX_BUILD_FLAG_ALLOW_RANDOM_VERTEX_ACCESS`.

`data[i] = {x,y,z,w}` with `{x,y,z}` the position and `w` the radius of control vertex `i`. If motion is disabled via `OptixPipelineCompileOptions::usesMotionBlur`, or the GAS does not contain motion, the time parameter is ignored.

**3.1.2.43 static \_\_forceinline\_\_ \_\_device\_\_ const OptixMatrixMotionTransform\***  
**optixGetMatrixMotionTransformFromHandle (**  
**OptixTraversableHandle *handle* ) [static]**

Returns a pointer to a `OptixMatrixMotionTransform` from its traversable handle.

Returns 0 if the traversable is not of type `OPTIX_TRANSFORM_TYPE_MATRIX_MOTION_TRANSFORM`.

**3.1.2.44 static \_\_forceinline\_\_ \_\_device\_\_ float3 optixGetObjectRayDirection ( ) [static]**

Returns the current object space ray direction based on the current transform stack.

Only available in IS and AH.

**3.1.2.45 static \_\_forceinline\_\_ \_\_device\_\_ float3 optixGetObjectRayOrigin ( ) [static]**

Returns the current object space ray origin based on the current transform stack.

Only available in IS and AH.

**3.1.2.46 static \_\_forceinline\_\_ \_\_device\_\_ void optixGetObjectToWorldTransformMatrix (**

**float m[12] ) [static]**

Returns the object-to-world transformation matrix resulting from the current active transformation list.  
The cost of this function may be proportional to the size of the transformation list.

**3.1.2.47 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_0 ( ) [static]**

Reads the 32-bit payload value at slot 0.

**3.1.2.48 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_1 ( ) [static]**

Reads the 32-bit payload value at slot 1.

**3.1.2.49 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_10 ( ) [static]**

Reads the 32-bit payload value at slot 10.

**3.1.2.50 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_11 ( ) [static]**

Reads the 32-bit payload value at slot 11.

**3.1.2.51 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_12 ( ) [static]**

Reads the 32-bit payload value at slot 12.

**3.1.2.52 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_13 ( ) [static]**

Reads the 32-bit payload value at slot 13.

**3.1.2.53 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_14 ( ) [static]**

Reads the 32-bit payload value at slot 14.

**3.1.2.54 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_15 ( ) [static]**

Reads the 32-bit payload value at slot 15.

**3.1.2.55 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_16 ( ) [static]**

Reads the 32-bit payload value at slot 16.

**3.1.2.56 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_17 ( ) [static]**

Reads the 32-bit payload value at slot 17.

**3.1.2.57 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_18 ( ) [static]**

Reads the 32-bit payload value at slot 18.

**3.1.2.58 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_19 ( ) [static]**

Reads the 32-bit payload value at slot 19.

**3.1.2.59 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_2 ( ) [static]**

Reads the 32-bit payload value at slot 2.

**3.1.2.60 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_20 ( ) [static]**

Reads the 32-bit payload value at slot 20.

**3.1.2.61 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_21 ( ) [static]**

Reads the 32-bit payload value at slot 21.

**3.1.2.62 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_22 ( ) [static]**

Reads the 32-bit payload value at slot 22.

**3.1.2.63 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_23 ( ) [static]**

Reads the 32-bit payload value at slot 23.

**3.1.2.64 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_24 ( ) [static]**

Reads the 32-bit payload value at slot 24.

**3.1.2.65 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_25 ( ) [static]**

Reads the 32-bit payload value at slot 25.

**3.1.2.66 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_26 ( ) [static]**

Reads the 32-bit payload value at slot 26.

**3.1.2.67 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_27 ( ) [static]**

Reads the 32-bit payload value at slot 27.

**3.1.2.68 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_28 ( ) [static]**

Reads the 32-bit payload value at slot 28.

**3.1.2.69 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_29 ( ) [static]**

Reads the 32-bit payload value at slot 29.

**3.1.2.70 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_3 ( ) [static]**

Reads the 32-bit payload value at slot 3.

**3.1.2.71 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_30 ( ) [static]**

Reads the 32-bit payload value at slot 30.

**3.1.2.72 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_31 ( ) [static]**

Reads the 32-bit payload value at slot 31.

**3.1.2.73 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_4 ( ) [static]**

Reads the 32-bit payload value at slot 4.

**3.1.2.74 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_5 ( ) [static]**

Reads the 32-bit payload value at slot 5.

**3.1.2.75 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_6 ( ) [static]**

Reads the 32-bit payload value at slot 6.

**3.1.2.76 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_7 ( ) [static]**

Reads the 32-bit payload value at slot 7.

**3.1.2.77 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_8 ( ) [static]**

Reads the 32-bit payload value at slot 8.

**3.1.2.78 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_9 ( ) [static]**

Reads the 32-bit payload value at slot 9.

**3.1.2.79 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPrimitiveIndex ( ) [static]**

For a given [OptixBuildInputTriangleArray](#) the number of primitives is defined as

"(OptixBuildInputTriangleArray::indexBuffer == 0) ? OptixBuildInputTriangleArray::numVertices/3 : OptixBuildInputTriangleArray::numIndexTriplets;". For a given [OptixBuildInputCustomPrimitiveArray](#) the number of primitives is defined as numAabbs.

The primitive index returns the index into the array of primitives plus the primitiveIndexOffset.

In IS and AH this corresponds to the currently intersected primitive. In CH this corresponds to the primitive index of the closest intersected primitive.

**3.1.2.80 static \_\_forceinline\_\_ \_\_device\_\_ OptixPrimitiveType optixGetPrimitiveType ( unsigned int *hitKind* ) [static]**

Function interpreting the result of [optixGetHitKind\(\)](#).

**3.1.2.81 static \_\_forceinline\_\_ \_\_device\_\_ OptixPrimitiveType optixGetPrimitiveType ( ) [static]**

Function interpreting the hit kind associated with the current optixReportIntersection.

**3.1.2.82 static \_\_forceinline\_\_ \_\_device\_\_ void optixGetQuadraticBSplineVertexData ( OptixTraversableHandle *gas*,**

```

unsigned int primIdx,
unsigned int sbtGASIndex,
float time,
float4 data[3] ) [static]

```

Return the object space curve control vertex data of a quadratic BSpline curve in a Geometry Acceleration Structure (GAS) at a given motion time. To access vertex data, the GAS must be built using the flag OPTIX\_BUILD\_FLAG\_ALLOW\_RANDOM\_VERTEX\_ACCESS.

data[i] = {x,y,z,w} with {x,y,z} the position and w the radius of control vertex i. If motion is disabled via [OptixPipelineCompileOptions::usesMotionBlur](#), or the GAS does not contain motion, the time parameter is ignored.

**3.1.2.83 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetRayFlags ( ) [static]**

Returns the rayFlags passed into optixTrace.

Only available in IS, AH, CH, MS

**3.1.2.84 static \_\_forceinline\_\_ \_\_device\_\_ float optixGetRayTime ( ) [static]**

Returns the rayTime passed into optixTrace.

Will return 0 if motion is disabled. Only available in IS, AH, CH, MS

**3.1.2.85 static \_\_forceinline\_\_ \_\_device\_\_ float optixGetRayTmax ( ) [static]**

In IS and CH returns the current smallest reported hitT or the tmax passed into optixTrace if no hit has been reported In AH returns the hitT value as passed in to optixReportIntersection In MS returns the tmax passed into optixTrace Only available in IS, AH, CH, MS.

**3.1.2.86 static \_\_forceinline\_\_ \_\_device\_\_ float optixGetRayTmin ( ) [static]**

Returns the tmin passed into optixTrace.

Only available in IS, AH, CH, MS

**3.1.2.87 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetRayVisibilityMask ( ) [static]**

Returns the visibilityMask passed into optixTrace.

Only available in IS, AH, CH, MS

**3.1.2.88 static \_\_forceinline\_\_ \_\_device\_\_ CUdeviceptr optixGetSbtDataPointer ( ) [static]**

Returns the generic memory space pointer to the data region (past the header) of the currently active SBT record corresponding to the current program.

**3.1.2.89 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetSbtGASIndex ( ) [static]**

Returns the Sbt GAS index of the primitive associated with the current intersection.

In IS and AH this corresponds to the currently intersected primitive. In CH this corresponds to the Sbt

GAS index of the closest intersected primitive. In EX with exception code `OPTIX_EXCEPTION_CODE_TRAVERSAL_INVALID_HIT_SBT` corresponds to the sbt index within the hit GAS. Returns zero for all other exceptions.

**3.1.2.90** `static __forceinline__ __device__ const OptixSRTMotionTransform*  
optixGetSRTMotionTransformFromHandle (  
OptixTraversableHandle handle ) [static]`

Returns a pointer to a [OptixSRTMotionTransform](#) from its traversable handle.

Returns 0 if the traversable is not of type `OPTIX_TRANSFORM_TYPE_SRT_MOTION_TRANSFORM`.

**3.1.2.91** `static __forceinline__ __device__ const OptixStaticTransform*  
optixGetStaticTransformFromHandle (  
OptixTraversableHandle handle ) [static]`

Returns a pointer to a [OptixStaticTransform](#) from its traversable handle.

Returns 0 if the traversable is not of type `OPTIX_TRANSFORM_TYPE_STATIC_TRANSFORM`.

**3.1.2.92** `static __forceinline__ __device__ OptixTraversableHandle  
optixGetTransformListHandle (  
unsigned int index ) [static]`

Returns the traversable handle for a transform on the current transform list.

Only available in IS, AH, CH, EX

**3.1.2.93** `static __forceinline__ __device__ unsigned int optixGetTransformListSize ( )  
[static]`

Returns the number of transforms on the current transform list.

Only available in IS, AH, CH, EX

**3.1.2.94** `static __forceinline__ __device__ OptixTransformType optixGetTransformType-  
FromHandle (  
OptixTraversableHandle handle ) [static]`

Returns the transform type of a traversable handle from a transform list.

**3.1.2.95** `static __forceinline__ __device__ float2 optixGetTriangleBarycentrics ( ) [static]`

Convenience function that returns the first two attributes as floats.

When using [OptixBuildInputTriangleArray](#) objects, during intersection the barycentric coordinates are stored into the first two attribute registers.

**3.1.2.96** `static __forceinline__ __device__ void optixGetTriangleVertexData (  
OptixTraversableHandle gas,  
unsigned int primIdx,  
unsigned int sbtGASIndex,`

**float time,**  
**float3 data[3] ) [static]**

Return the object space triangle vertex positions of a given triangle in a Geometry Acceleration Structure (GAS) at a given motion time. To access vertex data, the GAS must be built using the flag OPTIX\_BUILD\_FLAG\_ALLOW\_RANDOM\_VERTEX\_ACCESS.

If motion is disabled via [OptixPipelineCompileOptions::usesMotionBlur](#), or the GAS does not contain motion, the time parameter is ignored.

**3.1.2.97 static \_\_forceinline\_\_ \_\_device\_\_ float3 optixGetWorldRayDirection ( ) [static]**

Returns the rayDirection passed into optixTrace.

May be more expensive to call in IS and AH than their object space counterparts, so effort should be made to use the object space ray in those programs. Only available in IS, AH, CH, MS

**3.1.2.98 static \_\_forceinline\_\_ \_\_device\_\_ float3 optixGetWorldRayOrigin ( ) [static]**

Returns the rayOrigin passed into optixTrace.

May be more expensive to call in IS and AH than their object space counterparts, so effort should be made to use the object space ray in those programs. Only available in IS, AH, CH, MS

**3.1.2.99 static \_\_forceinline\_\_ \_\_device\_\_ void optixGetWorldToObjectTransformMatrix ( float m[12] ) [static]**

Returns the world-to-object transformation matrix resulting from the current active transformation list.

The cost of this function may be proportional to the size of the transformation list.

**3.1.2.100 static \_\_forceinline\_\_ \_\_device\_\_ void optixIgnoreIntersection ( ) [static]**

Discards the hit, and returns control to the calling optixReportIntersection or built-in intersection routine.

Available only in AH.

**3.1.2.101 static \_\_forceinline\_\_ \_\_device\_\_ bool optixIsBackFaceHit ( unsigned int hitKind ) [static]**

Function interpreting the result of [optixGetHitKind\(\)](#).

**3.1.2.102 static \_\_forceinline\_\_ \_\_device\_\_ bool optixIsBackFaceHit ( ) [static]**

Function interpreting the hit kind associated with the current optixReportIntersection.

**3.1.2.103 static \_\_forceinline\_\_ \_\_device\_\_ bool optixIsFrontFaceHit ( unsigned int hitKind ) [static]**

Function interpreting the result of [optixGetHitKind\(\)](#).

**3.1.2.104 static \_\_forceinline\_\_ \_\_device\_\_ bool optixIsFrontFaceHit ( ) [static]**

Function interpreting the hit kind associated with the current optixReportIntersection.



**3.1.2.105** `static __forceinline__ __device__ bool optixIsTriangleBackFaceHit ( ) [static]`

Convenience function interpreting the result of [optixGetHitKind\(\)](#).

**3.1.2.106** `static __forceinline__ __device__ bool optixIsTriangleFrontFaceHit ( ) [static]`

Convenience function interpreting the result of [optixGetHitKind\(\)](#).

**3.1.2.107** `static __forceinline__ __device__ bool optixIsTriangleHit ( ) [static]`

Convenience function interpreting the result of [optixGetHitKind\(\)](#).

**3.1.2.108** `static __forceinline__ __device__ bool optixReportIntersection (`  
`float hitT,`  
`unsigned int hitKind ) [static]`

Reports an intersections (overload without attributes).

If [optixGetRayTmin\(\)](#) <= *hitT* <= [optixGetRayTmax\(\)](#), the any hit program associated with this intersection program (via the SBT entry) is called. The AH program can do one of three things:

1. call [optixIgnoreIntersection](#) - no hit is recorded, [optixReportIntersection](#) returns false
2. call [optixTerminateRay](#) - hit is recorded, [optixReportIntersection](#) does not return, no further traversal occurs, and the associated closest hit program is called
3. neither - hit is recorded, [optixReportIntersection](#) returns true *hitKind* - Only the 7 least significant bits should be written [0..127]. Any values above 127 are reserved for built in intersection. The value can be queried with [optixGetHitKind\(\)](#) in AH and CH.

The attributes specified with *a0*..*a7* are available in the AH and CH programs. Note that the attributes available in the CH program correspond to the closest recorded intersection. The number of attributes in registers and memory can be configured in the pipeline.

#### Parameters

in	<i>hitT</i>	
in	<i>hitKind</i>	

**3.1.2.109** `static __forceinline__ __device__ bool optixReportIntersection (`  
`float hitT,`  
`unsigned int hitKind,`  
`unsigned int a0 ) [static]`

Reports an intersection (overload with 1 attribute register).

See Also

[optixReportIntersection\(float,unsigned int\)](#)

**3.1.2.110** `static __forceinline__ __device__ bool optixReportIntersection (`  
     `float hitT,`  
     `unsigned int hitKind,`  
     `unsigned int a0,`  
     `unsigned int a1 ) [static]`

Reports an intersection (overload with 2 attribute registers).

See Also

[optixReportIntersection\(float,unsigned int\)](#)

**3.1.2.111** `static __forceinline__ __device__ bool optixReportIntersection (`  
     `float hitT,`  
     `unsigned int hitKind,`  
     `unsigned int a0,`  
     `unsigned int a1,`  
     `unsigned int a2 ) [static]`

Reports an intersection (overload with 3 attribute registers).

See Also

[optixReportIntersection\(float,unsigned int\)](#)

**3.1.2.112** `static __forceinline__ __device__ bool optixReportIntersection (`  
     `float hitT,`  
     `unsigned int hitKind,`  
     `unsigned int a0,`  
     `unsigned int a1,`  
     `unsigned int a2,`  
     `unsigned int a3 ) [static]`

Reports an intersection (overload with 4 attribute registers).

See Also

[optixReportIntersection\(float,unsigned int\)](#)

**3.1.2.113** `static __forceinline__ __device__ bool optixReportIntersection (`  
     `float hitT,`  
     `unsigned int hitKind,`  
     `unsigned int a0,`  
     `unsigned int a1,`  
     `unsigned int a2,`  
     `unsigned int a3,`

**unsigned int *a4* ) [static]**

Reports an intersection (overload with 5 attribute registers).

See Also

[optixReportIntersection\(float,unsigned int\)](#)

**3.1.2.114 static \_\_forceinline\_\_ \_\_device\_\_ bool optixReportIntersection (**  
**float *hitT*,**  
**unsigned int *hitKind*,**  
**unsigned int *a0*,**  
**unsigned int *a1*,**  
**unsigned int *a2*,**  
**unsigned int *a3*,**  
**unsigned int *a4*,**  
**unsigned int *a5* ) [static]**

Reports an intersection (overload with 6 attribute registers).

See Also

[optixReportIntersection\(float,unsigned int\)](#)

**3.1.2.115 static \_\_forceinline\_\_ \_\_device\_\_ bool optixReportIntersection (**  
**float *hitT*,**  
**unsigned int *hitKind*,**  
**unsigned int *a0*,**  
**unsigned int *a1*,**  
**unsigned int *a2*,**  
**unsigned int *a3*,**  
**unsigned int *a4*,**  
**unsigned int *a5*,**  
**unsigned int *a6* ) [static]**

Reports an intersection (overload with 7 attribute registers).

See Also

[optixReportIntersection\(float,unsigned int\)](#)

**3.1.2.116 static \_\_forceinline\_\_ \_\_device\_\_ bool optixReportIntersection (**  
**float *hitT*,**  
**unsigned int *hitKind*,**  
**unsigned int *a0*,**  
**unsigned int *a1*,**  
**unsigned int *a2*,**

```

    unsigned int a3,
    unsigned int a4,
    unsigned int a5,
    unsigned int a6,
    unsigned int a7 ) [static]

```

Reports an intersection (overload with 8 attribute registers).

See Also

[optixReportIntersection\(float,unsigned int\)](#)

```

3.1.2.117 static __forceinline__ __device__ void optixSetPayload_0 (
    unsigned int p ) [static]

```

Writes the 32-bit payload value at slot 0.

```

3.1.2.118 static __forceinline__ __device__ void optixSetPayload_1 (
    unsigned int p ) [static]

```

Writes the 32-bit payload value at slot 1.

```

3.1.2.119 static __forceinline__ __device__ void optixSetPayload_10 (
    unsigned int p ) [static]

```

Writes the 32-bit payload value at slot 10.

```

3.1.2.120 static __forceinline__ __device__ void optixSetPayload_11 (
    unsigned int p ) [static]

```

Writes the 32-bit payload value at slot 11.

```

3.1.2.121 static __forceinline__ __device__ void optixSetPayload_12 (
    unsigned int p ) [static]

```

Writes the 32-bit payload value at slot 12.

```

3.1.2.122 static __forceinline__ __device__ void optixSetPayload_13 (
    unsigned int p ) [static]

```

Writes the 32-bit payload value at slot 13.

```

3.1.2.123 static __forceinline__ __device__ void optixSetPayload_14 (
    unsigned int p ) [static]

```

Writes the 32-bit payload value at slot 14.

```

3.1.2.124 static __forceinline__ __device__ void optixSetPayload_15 (

```

**unsigned int *p* ) [static]**

Writes the 32-bit payload value at slot 15.

**3.1.2.125 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_16 (**  
**unsigned int *p* ) [static]**

Writes the 32-bit payload value at slot 16.

**3.1.2.126 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_17 (**  
**unsigned int *p* ) [static]**

Writes the 32-bit payload value at slot 17.

**3.1.2.127 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_18 (**  
**unsigned int *p* ) [static]**

Writes the 32-bit payload value at slot 18.

**3.1.2.128 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_19 (**  
**unsigned int *p* ) [static]**

Writes the 32-bit payload value at slot 19.

**3.1.2.129 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_2 (**  
**unsigned int *p* ) [static]**

Writes the 32-bit payload value at slot 2.

**3.1.2.130 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_20 (**  
**unsigned int *p* ) [static]**

Writes the 32-bit payload value at slot 20.

**3.1.2.131 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_21 (**  
**unsigned int *p* ) [static]**

Writes the 32-bit payload value at slot 21.

**3.1.2.132 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_22 (**  
**unsigned int *p* ) [static]**

Writes the 32-bit payload value at slot 22.

**3.1.2.133 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_23 (**  
**unsigned int *p* ) [static]**

Writes the 32-bit payload value at slot 23.

**3.1.2.134 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_24 (**

**unsigned int *p* ) [static]**

Writes the 32-bit payload value at slot 24.

**3.1.2.135 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_25 (**  
**unsigned int *p* ) [static]**

Writes the 32-bit payload value at slot 25.

**3.1.2.136 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_26 (**  
**unsigned int *p* ) [static]**

Writes the 32-bit payload value at slot 26.

**3.1.2.137 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_27 (**  
**unsigned int *p* ) [static]**

Writes the 32-bit payload value at slot 27.

**3.1.2.138 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_28 (**  
**unsigned int *p* ) [static]**

Writes the 32-bit payload value at slot 28.

**3.1.2.139 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_29 (**  
**unsigned int *p* ) [static]**

Writes the 32-bit payload value at slot 29.

**3.1.2.140 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_3 (**  
**unsigned int *p* ) [static]**

Writes the 32-bit payload value at slot 3.

**3.1.2.141 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_30 (**  
**unsigned int *p* ) [static]**

Writes the 32-bit payload value at slot 30.

**3.1.2.142 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_31 (**  
**unsigned int *p* ) [static]**

Writes the 32-bit payload value at slot 31.

**3.1.2.143 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_4 (**  
**unsigned int *p* ) [static]**

Writes the 32-bit payload value at slot 4.

**3.1.2.144 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_5 (**

**unsigned int *p* ) [static]**

Writes the 32-bit payload value at slot 5.

**3.1.2.145 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_6 (**  
**unsigned int *p* ) [static]**

Writes the 32-bit payload value at slot 6.

**3.1.2.146 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_7 (**  
**unsigned int *p* ) [static]**

Writes the 32-bit payload value at slot 7.

**3.1.2.147 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_8 (**  
**unsigned int *p* ) [static]**

Writes the 32-bit payload value at slot 8.

**3.1.2.148 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_9 (**  
**unsigned int *p* ) [static]**

Writes the 32-bit payload value at slot 9.

**3.1.2.149 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayloadTypes (**  
**unsigned int *typeMask* ) [static]**

Specify the supported payload types for a program.

The supported types are specified as a bitwise combination of payload types. (See OptixPayloadTypeID) May only be called once per program. Must be called at the top of the program. Only available in IS, AH, CH, MS

**3.1.2.150 static \_\_forceinline\_\_ \_\_device\_\_ void optixTerminateRay ( ) [static]**

Record the hit, stops traversal, and proceeds to CH.

Available only in AH.

**3.1.2.151 static \_\_forceinline\_\_ \_\_device\_\_ uint4 optixTexFootprint2D (**  
**unsigned long long *tex*,**  
**unsigned int *texInfo*,**  
**float *x*,**  
**float *y*,**  
**unsigned int \* *singleMipLevel* ) [static]**

optixTexFootprint2D calculates the footprint of a corresponding 2D texture fetch (non-mipmapped).

On Turing and subsequent architectures, a texture footprint instruction allows user programs to determine the set of texels that would be accessed by an equivalent filtered texture lookup.

## Parameters

in	<i>tex</i>	CUDA texture object (cast to 64-bit integer)
in	<i>texInfo</i>	Texture info packed into 32-bit integer, described below.
in	<i>x</i>	Texture coordinate
in	<i>y</i>	Texture coordinate
out	<i>singleMipLevel</i>	Result indicating whether the footprint spans only a single miplevel.

The texture info argument is a packed 32-bit integer with the following layout:

texInfo[31:29] = reserved (3 bits) texInfo[28:24] = miplevel count (5 bits) texInfo[23:20] = log2 of tile width (4 bits) texInfo[19:16] = log2 of tile height (4 bits) texInfo[15:10] = reserved (6 bits) texInfo[9:8] = horizontal wrap mode (2 bits) (CUaddress\_mode) texInfo[7:6] = vertical wrap mode (2 bits) (CUaddress\_mode) texInfo[5] = mipmap filter mode (1 bit) (CUfilter\_mode) texInfo[4:0] = maximum anisotropy (5 bits)

Returns a 16-byte structure (as a uint4) that stores the footprint of a texture request at a particular "granularity", which has the following layout:

```
struct Texture2DFootprint { unsigned long long mask; unsigned int tileY : 12; unsigned int reserved1 : 4;
unsigned int dx : 3; unsigned int dy : 3; unsigned int reserved2 : 2; unsigned int granularity : 4;
unsigned int reserved3 : 4; unsigned int tileX : 12; unsigned int level : 4; unsigned int reserved4 : 16; };
```

The granularity indicates the size of texel groups that are represented by an 8x8 bitmask. For example, a granularity of 12 indicates texel groups that are 128x64 texels in size. In a footprint call, The returned granularity will either be the actual granularity of the result, or 0 if the footprint call was able to honor the requested granularity (the usual case).

level is the mip level of the returned footprint. Two footprint calls are needed to get the complete footprint when a texture call spans multiple mip levels.

mask is an 8x8 bitmask of texel groups that are covered, or partially covered, by the footprint. tileX and tileY give the starting position of the mask in 8x8 texel-group blocks. For example, suppose a granularity of 12 (128x64 texels), and tileX=3 and tileY=4. In this case, bit 0 of the mask (the low order bit) corresponds to texel group coordinates (3\*8, 4\*8), and texel coordinates (3\*8\*128, 4\*8\*64), within the specified mip level.

If nonzero, dx and dy specify a "toroidal rotation" of the bitmask. Toroidal rotation of a coordinate in the mask simply means that its value is reduced by 8. Continuing the example from above, if dx=0 and dy=0 the mask covers texel groups (3\*8, 4\*8) to (3\*8+7, 4\*8+7) inclusive. If, on the other hand, dx=2, the rightmost 2 columns in the mask have their x coordinates reduced by 8, and similarly for dy.

See the OptiX SDK for sample code that illustrates how to unpack the result.

```
3.1.2.152 static __forceinline__ __device__ uint4 optixTexFootprint2DGrad (
    unsigned long long tex,
    unsigned int texInfo,
    float x,
    float y,
    float dPdx_x,
    float dPdx_y,
```



```

float dPdy_x,
float dPdy_y,
bool coarse,
unsigned int * singleMipLevel ) [static]

```

optixTexFootprint2DGrad calculates the footprint of a corresponding 2D texture fetch (tex2DGrad)

#### Parameters

in	<i>tex</i>	CUDA texture object (cast to 64-bit integer)
in	<i>texInfo</i>	Texture info packed into 32-bit integer, described below.
in	<i>x</i>	Texture coordinate
in	<i>y</i>	Texture coordinate
in	<i>dPdx_x</i>	Derivative of x coordinte, which determines level of detail.
in	<i>dPdx_y</i>	Derivative of x coordinte, which determines level of detail.
in	<i>dPdy_x</i>	Derivative of y coordinte, which determines level of detail.
in	<i>dPdy_y</i>	Derivative of y coordinte, which determines level of detail.
in	<i>coarse</i>	Requests footprint from coarse miplevel, when the footprint spans two levels.
out	<i>singleMipLevel</i>	Result indicating whether the footprint spans only a single miplevel.

See Also

[optixTexFootprint2D\(unsigned long long,unsigned int,float,float,unsigned int\\*\)](#)

```

3.1.2.153 static __forceinline__ __device__ uint4 optixTexFootprint2DLod (
    unsigned long long tex,
    unsigned int texInfo,
    float x,
    float y,
    float level,
    bool coarse,
    unsigned int * singleMipLevel ) [static]

```

optixTexFootprint2DLod calculates the footprint of a corresponding 2D texture fetch (tex2DLod)

#### Parameters

in	<i>tex</i>	CUDA texture object (cast to 64-bit integer)
in	<i>texInfo</i>	Texture info packed into 32-bit integer, described below.
in	<i>x</i>	Texture coordinate
in	<i>y</i>	Texture coordinate
in	<i>level</i>	Level of detail (lod)
in	<i>coarse</i>	Requests footprint from coarse miplevel, when the footprint spans two levels.

### Parameters

out	<i>singleMipLevel</i>	Result indicating whether the footprint spans only a single miplevel.
-----	-----------------------	---

See Also

[optixTexFootprint2D\(unsigned long long,unsigned int,float,float,unsigned int\\*\)](#)

**3.1.2.154 static \_\_forceinline\_\_ \_\_device\_\_ void optixThrowException (**  
**int *exceptionCode* ) [static]**

Throws a user exception with the given exception code (overload without exception details).

The exception code must be in the range from 0 to  $2^{30} - 1$ . Up to 8 optional exception details can be passed. They can be queried in the EX program using [optixGetExceptionDetail\\_0\(\)](#) to ...\_8().

The exception details must not be used to encode pointers to the stack since the current stack is not preserved in the EX program.

Not available in EX.

### Parameters

in	<i>exceptionCode</i>	The exception code to be thrown.
----	----------------------	----------------------------------

**3.1.2.155 static \_\_forceinline\_\_ \_\_device\_\_ void optixThrowException (**  
**int *exceptionCode*,**  
**unsigned int *exceptionDetail0* ) [static]**

Throws a user exception with the given exception code (overload with 1 exception detail).

See Also

[optixThrowException\(int\)](#)

**3.1.2.156 static \_\_forceinline\_\_ \_\_device\_\_ void optixThrowException (**  
**int *exceptionCode*,**  
**unsigned int *exceptionDetail0*,**  
**unsigned int *exceptionDetail1* ) [static]**

Throws a user exception with the given exception code (overload with 2 exception details).

See Also

[optixThrowException\(int\)](#)

**3.1.2.157 static \_\_forceinline\_\_ \_\_device\_\_ void optixThrowException (**  
**int *exceptionCode*,**  
**unsigned int *exceptionDetail0*,**

```

    unsigned int exceptionDetail1,
    unsigned int exceptionDetail2 ) [static]

```

Throws a user exception with the given exception code (overload with 3 exception details).

See Also

[optixThrowException\(int\)](#)

```

3.1.2.158 static __forceinline__ __device__ void optixThrowException (
    int exceptionCode,
    unsigned int exceptionDetail0,
    unsigned int exceptionDetail1,
    unsigned int exceptionDetail2,
    unsigned int exceptionDetail3 ) [static]

```

Throws a user exception with the given exception code (overload with 4 exception details).

See Also

[optixThrowException\(int\)](#)

```

3.1.2.159 static __forceinline__ __device__ void optixThrowException (
    int exceptionCode,
    unsigned int exceptionDetail0,
    unsigned int exceptionDetail1,
    unsigned int exceptionDetail2,
    unsigned int exceptionDetail3,
    unsigned int exceptionDetail4 ) [static]

```

Throws a user exception with the given exception code (overload with 5 exception details).

See Also

[optixThrowException\(int\)](#)

```

3.1.2.160 static __forceinline__ __device__ void optixThrowException (
    int exceptionCode,
    unsigned int exceptionDetail0,
    unsigned int exceptionDetail1,
    unsigned int exceptionDetail2,
    unsigned int exceptionDetail3,
    unsigned int exceptionDetail4,
    unsigned int exceptionDetail5 ) [static]

```

Throws a user exception with the given exception code (overload with 6 exception details).

See Also

[optixThrowException\(int\)](#)

```
3.1.2.161 static __forceinline__ __device__ void optixThrowException (
    int exceptionCode,
    unsigned int exceptionDetail0,
    unsigned int exceptionDetail1,
    unsigned int exceptionDetail2,
    unsigned int exceptionDetail3,
    unsigned int exceptionDetail4,
    unsigned int exceptionDetail5,
    unsigned int exceptionDetail6 ) [static]
```

Throws a user exception with the given exception code (overload with 7 exception details).

See Also

[optixThrowException\(int\)](#)

```
3.1.2.162 static __forceinline__ __device__ void optixThrowException (
    int exceptionCode,
    unsigned int exceptionDetail0,
    unsigned int exceptionDetail1,
    unsigned int exceptionDetail2,
    unsigned int exceptionDetail3,
    unsigned int exceptionDetail4,
    unsigned int exceptionDetail5,
    unsigned int exceptionDetail6,
    unsigned int exceptionDetail7 ) [static]
```

Throws a user exception with the given exception code (overload with 8 exception details).

See Also

[optixThrowException\(int\)](#)

```
3.1.2.163 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
```

```

unsigned int SBTOffset,
unsigned int SBTstride,
unsigned int missSBTIndex ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload without payload).

#### Parameters

in	<i>handle</i>	
in	<i>rayOrigin</i>	
in	<i>rayDirection</i>	
in	<i>tmin</i>	
in	<i>tmax</i>	
in	<i>rayTime</i>	
in	<i>visibilityMask</i>	really only 8 bits
in	<i>rayFlags</i>	really only 8 bits, combination of OptixRayFlags
in	<i>SBTOffset</i>	really only 8 bits
in	<i>SBTstride</i>	really only 8 bits
in	<i>missSBTIndex</i>	specifies the miss program invoked on a miss

```

3.1.2.164 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 1 payload registers).

See Also

[optixTrace\(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned int,unsigned int,unsigned int,unsigned int\)](#)

```

3.1.2.165 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,

```

```

float3 rayOrigin,
float3 rayDirection,
float tmin,
float tmax,
float rayTime,
OptixVisibilityMask visibilityMask,
unsigned int rayFlags,
unsigned int SBTOffset,
unsigned int SBTstride,
unsigned int missSBTIndex,
unsigned int & p0,
unsigned int & p1 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 2 payload registers).

See Also

```

optixTrace(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

```

3.1.2.166 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 3 payload registers).

See Also

```

optixTrace(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

```

3.1.2.167 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,

```

```

float3 rayOrigin,
float3 rayDirection,
float tmin,
float tmax,
float rayTime,
OptixVisibilityMask visibilityMask,
unsigned int rayFlags,
unsigned int SBTOffset,
unsigned int SBTstride,
unsigned int missSBTIndex,
unsigned int & p0,
unsigned int & p1,
unsigned int & p2,
unsigned int & p3 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 4 payload registers).

See Also

```

optixTrace\(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int\)

```

**3.1.2.168** `static __forceinline__ __device__ void optixTrace (`  
`OptixTraversableHandle handle,`  
`float3 rayOrigin,`  
`float3 rayDirection,`  
`float tmin,`  
`float tmax,`  
`float rayTime,`  
`OptixVisibilityMask visibilityMask,`  
`unsigned int rayFlags,`  
`unsigned int SBTOffset,`  
`unsigned int SBTstride,`  
`unsigned int missSBTIndex,`  
`unsigned int & p0,`  
`unsigned int & p1,`  
`unsigned int & p2,`  
`unsigned int & p3,`  
`unsigned int & p4 ) [static]`

Initiates a ray tracing query starting with the given traversable (overload with 5 payload registers).

See Also

```

optixTrace\(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int\)

```

```

3.1.2.169 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 6 payload registers).

See Also

[optixTrace\(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned int,unsigned int,unsigned int,unsigned int\)](#)

```

3.1.2.170 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,

```



**unsigned int & p6 ) [static]**

Initiates a ray tracing query starting with the given traversable (overload with 7 payload registers).

See Also

[optixTrace\(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned int,unsigned int,unsigned int,unsigned int\)](#)

**3.1.2.171 static \_\_forceinline\_\_ \_\_device\_\_ void optixTrace (**  
**OptixTraversableHandle *handle*,**  
**float3 *rayOrigin*,**  
**float3 *rayDirection*,**  
**float *tmin*,**  
**float *tmax*,**  
**float *rayTime*,**  
**OptixVisibilityMask *visibilityMask*,**  
**unsigned int *rayFlags*,**  
**unsigned int *SBTOffset*,**  
**unsigned int *SBTstride*,**  
**unsigned int *missSBTIndex*,**  
**unsigned int & *p0*,**  
**unsigned int & *p1*,**  
**unsigned int & *p2*,**  
**unsigned int & *p3*,**  
**unsigned int & *p4*,**  
**unsigned int & *p5*,**  
**unsigned int & *p6*,**  
**unsigned int & *p7* ) [static]**

Initiates a ray tracing query starting with the given traversable (overload with 8 payload registers).

See Also

[optixTrace\(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned int,unsigned int,unsigned int,unsigned int\)](#)

**3.1.2.172 static \_\_forceinline\_\_ \_\_device\_\_ void optixTrace (**  
**OptixTraversableHandle *handle*,**  
**float3 *rayOrigin*,**  
**float3 *rayDirection*,**  
**float *tmin*,**  
**float *tmax*,**  
**float *rayTime*,**  
**OptixVisibilityMask *visibilityMask*,**

```

    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 9 payload registers).

See Also

```

optixTrace(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

```

3.1.2.173 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,

```

**unsigned int & p9 ) [static]**

Initiates a ray tracing query starting with the given traversable (overload with 10 payload registers).

See Also

[optixTrace\(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned int,unsigned int,unsigned int,unsigned int\)](#)

**3.1.2.174 static \_\_forceinline\_\_ \_\_device\_\_ void optixTrace (**  
**OptixTraversableHandle *handle*,**  
**float3 *rayOrigin*,**  
**float3 *rayDirection*,**  
**float *tmin*,**  
**float *tmax*,**  
**float *rayTime*,**  
**OptixVisibilityMask *visibilityMask*,**  
**unsigned int *rayFlags*,**  
**unsigned int *SBTOffset*,**  
**unsigned int *SBTstride*,**  
**unsigned int *missSBTIndex*,**  
**unsigned int & *p0*,**  
**unsigned int & *p1*,**  
**unsigned int & *p2*,**  
**unsigned int & *p3*,**  
**unsigned int & *p4*,**  
**unsigned int & *p5*,**  
**unsigned int & *p6*,**  
**unsigned int & *p7*,**  
**unsigned int & *p8*,**  
**unsigned int & *p9*,**  
**unsigned int & *p10* ) [static]**

Initiates a ray tracing query starting with the given traversable (overload with 11 payload registers).

See Also

[optixTrace\(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned int,unsigned int,unsigned int,unsigned int\)](#)

**3.1.2.175 static \_\_forceinline\_\_ \_\_device\_\_ void optixTrace (**  
**OptixTraversableHandle *handle*,**  
**float3 *rayOrigin*,**  
**float3 *rayDirection*,**  
**float *tmin*,**

```

float tmax,
float rayTime,
OptixVisibilityMask visibilityMask,
unsigned int rayFlags,
unsigned int SBTOffset,
unsigned int SBTstride,
unsigned int missSBTIndex,
unsigned int & p0,
unsigned int & p1,
unsigned int & p2,
unsigned int & p3,
unsigned int & p4,
unsigned int & p5,
unsigned int & p6,
unsigned int & p7,
unsigned int & p8,
unsigned int & p9,
unsigned int & p10,
unsigned int & p11 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 12 payload registers).

See Also

[optixTrace\(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned int,unsigned int,unsigned int,unsigned int\)](#)

```

3.1.2.176 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,

```

```

    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 13 payload registers).

See Also

[optixTrace\(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned int,unsigned int,unsigned int,unsigned int\)](#)

```

3.1.2.177 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 14 payload registers).

See Also

`optixTrace(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned int,unsigned int,unsigned int,unsigned int)`

```
3.1.2.178 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14 ) [static]
```

Initiates a ray tracing query starting with the given traversable (overload with 15 payload registers).

See Also

`optixTrace(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned int,unsigned int,unsigned int,unsigned int)`

```
3.1.2.179 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
```

```

float tmax,
float rayTime,
OptixVisibilityMask visibilityMask,
unsigned int rayFlags,
unsigned int SBTOffset,
unsigned int SBTstride,
unsigned int missSBTIndex,
unsigned int & p0,
unsigned int & p1,
unsigned int & p2,
unsigned int & p3,
unsigned int & p4,
unsigned int & p5,
unsigned int & p6,
unsigned int & p7,
unsigned int & p8,
unsigned int & p9,
unsigned int & p10,
unsigned int & p11,
unsigned int & p12,
unsigned int & p13,
unsigned int & p14,
unsigned int & p15 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 16 payload registers).

See Also

```

optixTrace(OptixTraversableHandle, float3, float3, float, float, float, OptixVisibilityMask, unsigned
int, unsigned int, unsigned int, unsigned int)

```

```

3.1.2.180 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,

```

```

    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 17 payload registers).

See Also

```

optixTrace(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

```

3.1.2.181 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,

```



```

    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 18 payload registers).

See Also

```

optixTrace(OptixTraversableHandle, float3, float3, float, float, float, OptixVisibilityMask, unsigned
int, unsigned int, unsigned int, unsigned int)

```

```

3.1.2.182 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,

```

```

    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 19 payload registers).

See Also

```

optixTrace(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

```

3.1.2.183 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,

```

**unsigned int & *p19* ) [static]**

Initiates a ray tracing query starting with the given traversable (overload with 20 payload registers).

See Also

[optixTrace\(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned int,unsigned int,unsigned int,unsigned int\)](#)

**3.1.2.184 static \_\_forceinline\_\_ \_\_device\_\_ void optixTrace (**  
**OptixTraversableHandle *handle*,**  
**float3 *rayOrigin*,**  
**float3 *rayDirection*,**  
**float *tmin*,**  
**float *tmax*,**  
**float *rayTime*,**  
**OptixVisibilityMask *visibilityMask*,**  
**unsigned int *rayFlags*,**  
**unsigned int *SBTOffset*,**  
**unsigned int *SBTstride*,**  
**unsigned int *missSBTIndex*,**  
**unsigned int & *p0*,**  
**unsigned int & *p1*,**  
**unsigned int & *p2*,**  
**unsigned int & *p3*,**  
**unsigned int & *p4*,**  
**unsigned int & *p5*,**  
**unsigned int & *p6*,**  
**unsigned int & *p7*,**  
**unsigned int & *p8*,**  
**unsigned int & *p9*,**  
**unsigned int & *p10*,**  
**unsigned int & *p11*,**  
**unsigned int & *p12*,**  
**unsigned int & *p13*,**  
**unsigned int & *p14*,**  
**unsigned int & *p15*,**  
**unsigned int & *p16*,**  
**unsigned int & *p17*,**  
**unsigned int & *p18*,**  
**unsigned int & *p19*,**  
**unsigned int & *p20* ) [static]**

Initiates a ray tracing query starting with the given traversable (overload with 21 payload registers).

See Also

`optixTrace(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned int,unsigned int,unsigned int,unsigned int)`

```

3.1.2.185 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 22 payload registers).

See Also

`optixTrace(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned int,unsigned int,unsigned int,unsigned int)`

```

3.1.2.186 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 23 payload registers).

See Also

`optixTrace(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned int,unsigned int,unsigned int,unsigned int)`

```

3.1.2.187 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 24 payload registers).

See Also

`optixTrace(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned int,unsigned int,unsigned int,unsigned int)`

```

3.1.2.188 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23,

```

**unsigned int & p24 ) [static]**

Initiates a ray tracing query starting with the given traversable (overload with 25 payload registers).

See Also

[optixTrace\(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned int,unsigned int,unsigned int,unsigned int\)](#)

```

3.1.2.189 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,

```



```

    unsigned int & p23,
    unsigned int & p24,
    unsigned int & p25 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 26 payload registers).

See Also

```

optixTrace(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

### 3.1.2.190 static \_\_forceinline\_\_ \_\_device\_\_ void optixTrace (

```

    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,

```

```

    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23,
    unsigned int & p24,
    unsigned int & p25,
    unsigned int & p26 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 27 payload registers).

See Also

```

optixTrace(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

**3.1.2.191** `static __forceinline__ __device__ void optixTrace (`

```

    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,

```

```

    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23,
    unsigned int & p24,
    unsigned int & p25,
    unsigned int & p26,
    unsigned int & p27 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 28 payload registers).

See Also

```

optixTrace(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

**3.1.2.192** `static __forceinline__ __device__ void optixTrace (`

```

    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,

```

```

    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23,
    unsigned int & p24,
    unsigned int & p25,
    unsigned int & p26,
    unsigned int & p27,
    unsigned int & p28 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 29 payload registers).

See Also

```

optixTrace(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

```

3.1.2.193 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,

```

```

    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23,
    unsigned int & p24,
    unsigned int & p25,
    unsigned int & p26,
    unsigned int & p27,
    unsigned int & p28,
    unsigned int & p29 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 30 payload registers).

See Also

```

optixTrace(OptixTraversableHandle, float3, float3, float, float, float, OptixVisibilityMask, unsigned
int, unsigned int, unsigned int, unsigned int)

```

```

3.1.2.194 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,

```

```

    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23,
    unsigned int & p24,
    unsigned int & p25,
    unsigned int & p26,
    unsigned int & p27,
    unsigned int & p28,
    unsigned int & p29,
    unsigned int & p30 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 31 payload registers).

See Also

```

optixTrace(OptixTraversableHandle, float3, float3, float, float, float, OptixVisibilityMask, unsigned
int, unsigned int, unsigned int, unsigned int)

```

```

3.1.2.195 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,

```

```

unsigned int rayFlags,
unsigned int SBTOffset,
unsigned int SBTstride,
unsigned int missSBTIndex,
unsigned int & p0,
unsigned int & p1,
unsigned int & p2,
unsigned int & p3,
unsigned int & p4,
unsigned int & p5,
unsigned int & p6,
unsigned int & p7,
unsigned int & p8,
unsigned int & p9,
unsigned int & p10,
unsigned int & p11,
unsigned int & p12,
unsigned int & p13,
unsigned int & p14,
unsigned int & p15,
unsigned int & p16,
unsigned int & p17,
unsigned int & p18,
unsigned int & p19,
unsigned int & p20,
unsigned int & p21,
unsigned int & p22,
unsigned int & p23,
unsigned int & p24,
unsigned int & p25,
unsigned int & p26,
unsigned int & p27,
unsigned int & p28,
unsigned int & p29,
unsigned int & p30,
unsigned int & p31 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 32 payload registers).

See Also

```

optixTrace\(OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int\)

```

```

3.1.2.196 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload without payload).

#### Parameters

in	<i>type</i>	
in	<i>handle</i>	
in	<i>rayOrigin</i>	
in	<i>rayDirection</i>	
in	<i>tmin</i>	
in	<i>tmax</i>	
in	<i>rayTime</i>	
in	<i>visibilityMask</i>	really only 8 bits
in	<i>rayFlags</i>	really only 8 bits, combination of OptixRayFlags
in	<i>SBTOffset</i>	really only 8 bits
in	<i>SBTstride</i>	really only 8 bits
in	<i>missSBTIndex</i>	specifies the miss program invoked on a miss

```

3.1.2.197 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,

```



```

unsigned int SBTOffset,
unsigned int SBTstride,
unsigned int missSBTIndex,
unsigned int & p0 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 1 payload registers).

See Also

```

optixTrace\(OptixPayloadTypeID,OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int\)

```

```

3.1.2.198 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 2 payload registers).

See Also

```

optixTrace\(OptixPayloadTypeID,OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int\)

```

```

3.1.2.199 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,

```

```

    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 3 payload registers).

See Also

```

optixTrace(OptixPayloadTypeID,OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

```

3.1.2.200 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 4 payload registers).

See Also

```

optixTrace(OptixPayloadTypeID,OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

```

3.1.2.201 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,

```

```

float3 rayDirection,
float tmin,
float tmax,
float rayTime,
OptixVisibilityMask visibilityMask,
unsigned int rayFlags,
unsigned int SBTOffset,
unsigned int SBTstride,
unsigned int missSBTIndex,
unsigned int & p0,
unsigned int & p1,
unsigned int & p2,
unsigned int & p3,
unsigned int & p4 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 5 payload registers).

See Also

```

optixTrace(OptixPayloadTypeID,OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

```

3.1.2.202 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 6 payload registers).

See Also

`optixTrace(OptixPayloadTypeID, OptixTraversableHandle, float3, float3, float, float, float, float, OptixVisibilityMask, unsigned int, unsigned int, unsigned int, unsigned int)`

```
3.1.2.203 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6 ) [static]
```

Initiates a ray tracing query starting with the given traversable (overload with 7 payload registers).

See Also

`optixTrace(OptixPayloadTypeID, OptixTraversableHandle, float3, float3, float, float, float, float, OptixVisibilityMask, unsigned int, unsigned int, unsigned int, unsigned int)`

```
3.1.2.204 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
```

```

unsigned int SBTOffset,
unsigned int SBTstride,
unsigned int missSBTIndex,
unsigned int & p0,
unsigned int & p1,
unsigned int & p2,
unsigned int & p3,
unsigned int & p4,
unsigned int & p5,
unsigned int & p6,
unsigned int & p7 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 8 payload registers).

See Also

```

optixTrace(OptixPayloadTypeID,OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

```

3.1.2.205 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 9 payload registers).

See Also

`optixTrace(OptixPayloadTypeID, OptixTraversableHandle, float3, float3, float, float, float, float, OptixVisibilityMask, unsigned int, unsigned int, unsigned int, unsigned int)`

```
3.1.2.206 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9 ) [static]
```

Initiates a ray tracing query starting with the given traversable (overload with 10 payload registers).

See Also

`optixTrace(OptixPayloadTypeID, OptixTraversableHandle, float3, float3, float, float, float, float, OptixVisibilityMask, unsigned int, unsigned int, unsigned int, unsigned int)`

```
3.1.2.207 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
```

```

float rayTime,
OptixVisibilityMask visibilityMask,
unsigned int rayFlags,
unsigned int SBTOffset,
unsigned int SBTstride,
unsigned int missSBTIndex,
unsigned int & p0,
unsigned int & p1,
unsigned int & p2,
unsigned int & p3,
unsigned int & p4,
unsigned int & p5,
unsigned int & p6,
unsigned int & p7,
unsigned int & p8,
unsigned int & p9,
unsigned int & p10 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 11 payload registers).

See Also

```

optixTrace(OptixPayloadTypeID,OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

```

3.1.2.208 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,

```

```

    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 12 payload registers).

See Also

```

optixTrace(OptixPayloadTypeID,OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

```

3.1.2.209 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 13 payload registers).



See Also

`optixTrace(OptixPayloadTypeID, OptixTraversableHandle, float3, float3, float, float, float, float, OptixVisibilityMask, unsigned int, unsigned int, unsigned int, unsigned int)`

```
3.1.2.210 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13 ) [static]
```

Initiates a ray tracing query starting with the given traversable (overload with 14 payload registers).

See Also

`optixTrace(OptixPayloadTypeID, OptixTraversableHandle, float3, float3, float, float, float, float, OptixVisibilityMask, unsigned int, unsigned int, unsigned int, unsigned int)`

```
3.1.2.211 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
```

```

float3 rayOrigin,
float3 rayDirection,
float tmin,
float tmax,
float rayTime,
OptixVisibilityMask visibilityMask,
unsigned int rayFlags,
unsigned int SBTOffset,
unsigned int SBTstride,
unsigned int missSBTIndex,
unsigned int & p0,
unsigned int & p1,
unsigned int & p2,
unsigned int & p3,
unsigned int & p4,
unsigned int & p5,
unsigned int & p6,
unsigned int & p7,
unsigned int & p8,
unsigned int & p9,
unsigned int & p10,
unsigned int & p11,
unsigned int & p12,
unsigned int & p13,
unsigned int & p14 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 15 payload registers).

See Also

[optixTrace\(OptixPayloadTypeID,OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned int,unsigned int,unsigned int,unsigned int\)](#)

**3.1.2.212** `static __forceinline__ __device__ void optixTrace (`  
`OptixPayloadTypeID type,`  
`OptixTraversableHandle handle,`  
`float3 rayOrigin,`  
`float3 rayDirection,`  
`float tmin,`  
`float tmax,`  
`float rayTime,`  
`OptixVisibilityMask visibilityMask,`  
`unsigned int rayFlags,`

```

unsigned int SBToffset,
unsigned int SBTstride,
unsigned int missSBTIndex,
unsigned int & p0,
unsigned int & p1,
unsigned int & p2,
unsigned int & p3,
unsigned int & p4,
unsigned int & p5,
unsigned int & p6,
unsigned int & p7,
unsigned int & p8,
unsigned int & p9,
unsigned int & p10,
unsigned int & p11,
unsigned int & p12,
unsigned int & p13,
unsigned int & p14,
unsigned int & p15 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 16 payload registers).

See Also

```

optixTrace(OptixPayloadTypeID,OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

```

3.1.2.213 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBToffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,

```

```

    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 17 payload registers).

See Also

```

optixTrace(OptixPayloadTypeID,OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

```

3.1.2.214 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,

```

```

    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 18 payload registers).

See Also

```

optixTrace(OptixPayloadTypeID,OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

```

3.1.2.215 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,

```

```

    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 19 payload registers).

See Also

```

optixTrace(OptixPayloadTypeID,OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

```

3.1.2.216 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,

```

```

    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 20 payload registers).

See Also

```

optixTrace(OptixPayloadTypeID,OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

**3.1.2.217** static \_\_forceinline\_\_ \_\_device\_\_ void optixTrace (

```

    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,

```

```

    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 21 payload registers).

See Also

```

optixTrace(OptixPayloadTypeID,OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

```

3.1.2.218 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,

```



```

    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 22 payload registers).

See Also

```

optixTrace(OptixPayloadTypeID,OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

```

3.1.2.219 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,

```

```

    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 23 payload registers).

See Also

```

optixTrace(OptixPayloadTypeID,OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

**3.1.2.220** `static __forceinline__ __device__ void optixTrace (`

```

    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,

```

```

    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 24 payload registers).

See Also

```

optixTrace(OptixPayloadTypeID,OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

```

3.1.2.221 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,

```

```

    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23,
    unsigned int & p24 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 25 payload registers).

See Also

```

optixTrace(OptixPayloadTypeID,OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

```

3.1.2.222 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,

```

```

    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23,
    unsigned int & p24,
    unsigned int & p25 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 26 payload registers).

See Also

```

optixTrace(OptixPayloadTypeID,OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

```

3.1.2.223 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,

```

```

    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23,
    unsigned int & p24,
    unsigned int & p25,
    unsigned int & p26 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 27 payload registers).

See Also

```

optixTrace(OptixPayloadTypeID,OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

```

3.1.2.224 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,

```

```

    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23,
    unsigned int & p24,
    unsigned int & p25,
    unsigned int & p26,
    unsigned int & p27 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 28 payload registers).

See Also

```

optixTrace(OptixPayloadTypeID,OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

```

3.1.2.225 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,

```

```

unsigned int SBTOffset,
unsigned int SBTstride,
unsigned int missSBTIndex,
unsigned int & p0,
unsigned int & p1,
unsigned int & p2,
unsigned int & p3,
unsigned int & p4,
unsigned int & p5,
unsigned int & p6,
unsigned int & p7,
unsigned int & p8,
unsigned int & p9,
unsigned int & p10,
unsigned int & p11,
unsigned int & p12,
unsigned int & p13,
unsigned int & p14,
unsigned int & p15,
unsigned int & p16,
unsigned int & p17,
unsigned int & p18,
unsigned int & p19,
unsigned int & p20,
unsigned int & p21,
unsigned int & p22,
unsigned int & p23,
unsigned int & p24,
unsigned int & p25,
unsigned int & p26,
unsigned int & p27,
unsigned int & p28 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 29 payload registers).

See Also

```

optixTrace(OptixPayloadTypeID,OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

```

3.1.2.226 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,

```



```

float3 rayOrigin,
float3 rayDirection,
float tmin,
float tmax,
float rayTime,
OptixVisibilityMask visibilityMask,
unsigned int rayFlags,
unsigned int SBTOffset,
unsigned int SBTstride,
unsigned int missSBTIndex,
unsigned int & p0,
unsigned int & p1,
unsigned int & p2,
unsigned int & p3,
unsigned int & p4,
unsigned int & p5,
unsigned int & p6,
unsigned int & p7,
unsigned int & p8,
unsigned int & p9,
unsigned int & p10,
unsigned int & p11,
unsigned int & p12,
unsigned int & p13,
unsigned int & p14,
unsigned int & p15,
unsigned int & p16,
unsigned int & p17,
unsigned int & p18,
unsigned int & p19,
unsigned int & p20,
unsigned int & p21,
unsigned int & p22,
unsigned int & p23,
unsigned int & p24,
unsigned int & p25,
unsigned int & p26,
unsigned int & p27,
unsigned int & p28,
unsigned int & p29 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 30 payload registers).

See Also

`optixTrace(OptixPayloadTypeID, OptixTraversableHandle, float3, float3, float, float, float, OptixVisibilityMask, unsigned int, unsigned int, unsigned int, unsigned int)`

**3.1.2.227** `static __forceinline__ __device__ void optixTrace (`  
     *OptixPayloadTypeID* *type*,  
     *OptixTraversableHandle* *handle*,  
     float3 *rayOrigin*,  
     float3 *rayDirection*,  
     float *tmin*,  
     float *tmax*,  
     float *rayTime*,  
     OptixVisibilityMask *visibilityMask*,  
     unsigned int *rayFlags*,  
     unsigned int *SBTOffset*,  
     unsigned int *SBTstride*,  
     unsigned int *missSBTIndex*,  
     unsigned int & *p0*,  
     unsigned int & *p1*,  
     unsigned int & *p2*,  
     unsigned int & *p3*,  
     unsigned int & *p4*,  
     unsigned int & *p5*,  
     unsigned int & *p6*,  
     unsigned int & *p7*,  
     unsigned int & *p8*,  
     unsigned int & *p9*,  
     unsigned int & *p10*,  
     unsigned int & *p11*,  
     unsigned int & *p12*,  
     unsigned int & *p13*,  
     unsigned int & *p14*,  
     unsigned int & *p15*,  
     unsigned int & *p16*,  
     unsigned int & *p17*,  
     unsigned int & *p18*,  
     unsigned int & *p19*,  
     unsigned int & *p20*,  
     unsigned int & *p21*,  
     unsigned int & *p22*,  
     unsigned int & *p23*,

```

    unsigned int & p24,
    unsigned int & p25,
    unsigned int & p26,
    unsigned int & p27,
    unsigned int & p28,
    unsigned int & p29,
    unsigned int & p30 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 31 payload registers).

See Also

`optixTrace(OptixPayloadTypeID,OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned int,unsigned int,unsigned int,unsigned int)`

```

3.1.2.228 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,

```

```

    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23,
    unsigned int & p24,
    unsigned int & p25,
    unsigned int & p26,
    unsigned int & p27,
    unsigned int & p28,
    unsigned int & p29,
    unsigned int & p30,
    unsigned int & p31 ) [static]

```

Initiates a ray tracing query starting with the given traversable (overload with 32 payload registers).

See Also

```

optixTrace(OptixPayloadTypeID,OptixTraversableHandle,float3,float3,float,float,float,OptixVisibilityMask,unsigned
int,unsigned int,unsigned int,unsigned int)

```

**3.1.2.229 static \_\_forceinline\_\_ \_\_device\_\_ float3 optixTransformNormalFromObjectToWorldSpace (**  
**float3 normal ) [static]**

Transforms the normal using object-to-world transformation matrix resulting from the current active transformation list.

The cost of this function may be proportional to the size of the transformation list.

**3.1.2.230 static \_\_forceinline\_\_ \_\_device\_\_ float3 optixTransformNormalFromWorldToObjectSpace (**  
**float3 normal ) [static]**

Transforms the normal using world-to-object transformation matrix resulting from the current active transformation list.

The cost of this function may be proportional to the size of the transformation list.

**3.1.2.231 static \_\_forceinline\_\_ \_\_device\_\_ float3 optixTransformPointFromObjectToWorldSpace (**

**float3 point ) [static]**

Transforms the point using object-to-world transformation matrix resulting from the current active transformation list.

The cost of this function may be proportional to the size of the transformation list.

**3.1.2.232 static \_\_forceinline\_\_ \_\_device\_\_ float3 optixTransformPointFromWorldToObjectSpace ( float3 point ) [static]**

Transforms the point using world-to-object transformation matrix resulting from the current active transformation list.

The cost of this function may be proportional to the size of the transformation list.

**3.1.2.233 static \_\_forceinline\_\_ \_\_device\_\_ float3 optixTransformVectorFromObjectToWorldSpace ( float3 vec ) [static]**

Transforms the vector using object-to-world transformation matrix resulting from the current active transformation list.

The cost of this function may be proportional to the size of the transformation list.

**3.1.2.234 static \_\_forceinline\_\_ \_\_device\_\_ float3 optixTransformVectorFromWorldToObjectSpace ( float3 vec ) [static]**

Transforms the vector using world-to-object transformation matrix resulting from the current active transformation list.

The cost of this function may be proportional to the size of the transformation list.

**3.1.2.235 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixUndefinedValue ( ) [static]**

Returns an undefined value.

## 3.2 Host API

### Modules

- Error handling
- Device context
- Pipelines
- Modules
- Tasks
- Program groups
- Launches
- Acceleration structures
- Denoiser

### 3.2.1 Detailed Description

OptiX Host API.

## 3.3 Error handling

### Functions

- `const char * optixGetErrorName (OptixResult result)`
- `const char * optixGetErrorString (OptixResult result)`

#### 3.3.1 Detailed Description

#### 3.3.2 Function Documentation

##### 3.3.2.1 `const char* optixGetErrorName (OptixResult result )`

Returns a string containing the name of an error code in the enum.

Output is a string representation of the enum. For example "OPTIX\_SUCCESS" for OPTIX\_SUCCESS and "OPTIX\_ERROR\_INVALID\_VALUE" for OPTIX\_ERROR\_INVALID\_VALUE.

If the error code is not recognized, "Unrecognized OptixResult code" is returned.

#### Parameters

in	<i>result</i>	OptixResult enum to generate string name for
----	---------------	--

See Also

[optixGetErrorString](#)

##### 3.3.2.2 `const char* optixGetErrorString (OptixResult result )`

Returns the description string for an error code.

Output is a string description of the enum. For example "Success" for OPTIX\_SUCCESS and "Invalid value" for OPTIX\_ERROR\_INVALID\_VALUE.

If the error code is not recognized, "Unrecognized OptixResult code" is returned.

#### Parameters

in	<i>result</i>	OptixResult enum to generate string description for
----	---------------	---

See Also

[optixGetErrorName](#)

## 3.4 Device context

### Functions

- `OptixResult optixDeviceContextCreate` (CUcontext fromContext, const OptixDeviceContextOptions \*options, OptixDeviceContext \*context)
- `OptixResult optixDeviceContextDestroy` (OptixDeviceContext context)
- `OptixResult optixDeviceContextGetProperty` (OptixDeviceContext context, OptixDeviceProperty property, void \*value, size\_t sizeInBytes)
- `OptixResult optixDeviceContextSetLogCallback` (OptixDeviceContext context, OptixLogCallback callbackFunction, void \*callbackData, unsigned int callbackLevel)
- `OptixResult optixDeviceContextSetCacheEnabled` (OptixDeviceContext context, int enabled)
- `OptixResult optixDeviceContextSetCacheLocation` (OptixDeviceContext context, const char \*location)
- `OptixResult optixDeviceContextSetCacheDatabaseSizes` (OptixDeviceContext context, size\_t lowWaterMark, size\_t highWaterMark)
- `OptixResult optixDeviceContextGetCacheEnabled` (OptixDeviceContext context, int \*enabled)
- `OptixResult optixDeviceContextGetCacheLocation` (OptixDeviceContext context, char \*location, size\_t locationSize)
- `OptixResult optixDeviceContextGetCacheDatabaseSizes` (OptixDeviceContext context, size\_t \*lowWaterMark, size\_t \*highWaterMark)

### 3.4.1 Detailed Description

### 3.4.2 Function Documentation

#### 3.4.2.1 OptixResult optixDeviceContextCreate (

**CUcontext** *fromContext*,

**const OptixDeviceContextOptions** \* *options*,

**OptixDeviceContext** \* *context* )

Create a device context associated with the CUDA context specified with 'fromContext'.

If zero is specified for 'fromContext', OptiX will use the current CUDA context. The CUDA context should be initialized before calling optixDeviceContextCreate.

#### Parameters

in	<i>fromContext</i>	
in	<i>options</i>	
out	<i>context</i>	

#### Returns

- `OPTIX_ERROR_CUDA_NOT_INITIALIZED` If using zero for 'fromContext' and CUDA has not been initialized yet on the calling thread.
- `OPTIX_ERROR_CUDA_ERROR` CUDA operation failed.



- OPTIX\_ERROR\_HOST\_OUT\_OF\_MEMORY Heap allocation failed.
- OPTIX\_ERROR\_INTERNAL\_ERROR Internal error

### 3.4.2.2 OptixResult optixDeviceContextDestroy ( OptixDeviceContext *context* )

Destroys all CPU and GPU state associated with the device.

It will attempt to block on CUDA streams that have launch work outstanding.

Any API objects, such as OptixModule and OptixPipeline, not already destroyed will be destroyed.

Thread safety: A device context must not be destroyed while it is still in use by concurrent API calls in other threads.

### 3.4.2.3 OptixResult optixDeviceContextGetCacheDatabaseSizes ( OptixDeviceContext *context*, size\_t \* *lowWaterMark*, size\_t \* *highWaterMark* )

Returns the low and high water marks for disk cache garbage collection. If the cache has been disabled by setting the environment variable OPTIX\_CACHE\_MAXSIZE=0, this function will return 0 for the low and high water marks.

#### Parameters

in	<i>context</i>	the device context
out	<i>lowWaterMark</i>	the low water mark
out	<i>highWaterMark</i>	the high water mark

### 3.4.2.4 OptixResult optixDeviceContextGetCacheEnabled ( OptixDeviceContext *context*, int \* *enabled* )

Indicates whether the disk cache is enabled or disabled.

#### Parameters

in	<i>context</i>	the device context
out	<i>enabled</i>	1 if enabled, 0 if disabled

### 3.4.2.5 OptixResult optixDeviceContextGetCacheLocation ( OptixDeviceContext *context*, char \* *location*,

### **size\_t locationSize )**

Returns the location of the disk cache. If the cache has been disabled by setting the environment variable OPTIX\_CACHE\_MAXSIZE=0, this function will return an empty string.

#### **Parameters**

in	<i>context</i>	the device context
out	<i>location</i>	directory of disk cache, null terminated if locationSize > 0
in	<i>locationSize</i>	locationSize

#### **3.4.2.6 OptixResult optixDeviceContextGetProperty (**

**OptixDeviceContext *context*,**  
**OptixDeviceProperty *property*,**  
**void \* *value*,**  
**size\_t *sizeInBytes* )**

Query properties of a device context.

#### **Parameters**

in	<i>context</i>	the device context to query the property for
in	<i>property</i>	the property to query
out	<i>value</i>	pointer to the returned
in	<i>sizeInBytes</i>	size of output

#### **3.4.2.7 OptixResult optixDeviceContextSetCacheDatabaseSizes (**

**OptixDeviceContext *context*,**  
**size\_t *lowWaterMark*,**  
**size\_t *highWaterMark* )**

Sets the low and high water marks for disk cache garbage collection.

Garbage collection is triggered when a new entry is written to the cache and the current cache data size plus the size of the cache entry that is about to be inserted exceeds the high water mark. Garbage collection proceeds until the size reaches the low water mark. Garbage collection will always free enough space to insert the new entry without exceeding the low water mark. Setting either limit to zero will disable garbage collection. An error will be returned if both limits are non-zero and the high water mark is smaller than the low water mark.

Note that garbage collection is performed only on writes to the disk cache. No garbage collection is triggered on disk cache initialization or immediately when calling this function, but on subsequent inserting of data into the database.

If the size of a compiled module exceeds the value configured for the high water mark and garbage collection is enabled, the module will not be added to the cache and a warning will be added to the log.

The high water mark can be overridden with the environment variable OPTIX\_CACHE\_MAXSIZE. The

environment variable takes precedence over the function parameters. The low water mark will be set to half the value of `OPTIX_CACHE_MAXSIZE`. Setting `OPTIX_CACHE_MAXSIZE` to 0 will disable the disk cache, but will not alter the contents of the cache. Negative and non-integer values will be ignored.

### Parameters

in	<i>context</i>	the device context
in	<i>lowWaterMark</i>	the low water mark
in	<i>highWaterMark</i>	the high water mark

#### 3.4.2.8 **OptixResult optixDeviceContextSetCacheEnabled (** **OptixDeviceContext *context*,** **int *enabled* )**

Enables or disables the disk cache.

If caching was previously disabled, enabling it will attempt to initialize the disk cache database using the currently configured cache location. An error will be returned if initialization fails.

Note that no in-memory cache is used, so no caching behavior will be observed if the disk cache is disabled.

The cache can be disabled by setting the environment variable `OPTIX_CACHE_MAXSIZE=0`. The environment variable takes precedence over this setting. See [optixDeviceContextSetCacheDatabaseSizes](#) for additional information.

Note that the disk cache can be disabled by the environment variable, but it cannot be enabled via the environment if it is disabled via the API.

### Parameters

in	<i>context</i>	the device context
in	<i>enabled</i>	1 to enabled, 0 to disable

#### 3.4.2.9 **OptixResult optixDeviceContextSetCacheLocation (** **OptixDeviceContext *context*,** **const char \* *location* )**

Sets the location of the disk cache.

The location is specified by a directory. This directory should not be used for other purposes and will be created if it does not exist. An error will be returned if it is not possible to create the disk cache at the specified location for any reason (e.g., the path is invalid or the directory is not writable). Caching will be disabled if the disk cache cannot be initialized in the new location. If caching is disabled, no error will be returned until caching is enabled. If the disk cache is located on a network file share, behavior is undefined.

The location of the disk cache can be overridden with the environment variable `OPTIX_CACHE_PATH`. The environment variable takes precedence over this setting.

The default location depends on the operating system:

- Windows: LOCALAPPDATA%\NVIDIA\OptixCache
- Linux: /var/tmp/OptixCache\_<username> (or /tmp/OptixCache\_<username> if the first choice is not usable), the underscore and username suffix are omitted if the username cannot be obtained
- MacOS X: /Library/Application Support/NVIDIA/OptixCache

### Parameters

in	<i>context</i>	the device context
in	<i>location</i>	directory of disk cache

**3.4.2.10 OptixResult optixDeviceContextSetLogCallback (**  
**OptixDeviceContext *context*,**  
**OptixLogCallback *callbackFunction*,**  
**void \* *callbackData*,**  
**unsigned int *callbackLevel* )**

Sets the current log callback method.

See [OptixLogCallback](#) for more details.

Thread safety: It is guaranteed that the callback itself (*callbackFunction* and *callbackData*) are updated atomically. It is not guaranteed that the callback itself (*callbackFunction* and *callbackData*) and the *callbackLevel* are updated atomically. It is unspecified when concurrent API calls using the same context start to make use of the new callback method.

### Parameters

in	<i>context</i>	the device context
in	<i>callbackFunction</i>	the callback function to call
in	<i>callbackData</i>	pointer to data passed to callback function while invoking it
in	<i>callbackLevel</i>	callback level

## 3.5 Pipelines

### Functions

- `OptixResult optixPipelineCreate` (`OptixDeviceContext` context, const `OptixPipelineCompileOptions` \*pipelineCompileOptions, const `OptixPipelineLinkOptions` \*pipelineLinkOptions, const `OptixProgramGroup` \*programGroups, unsigned int numProgramGroups, char \*logString, size\_t \*logStringSize, `OptixPipeline` \*pipeline)
- `OptixResult optixPipelineDestroy` (`OptixPipeline` pipeline)
- `OptixResult optixPipelineSetStackSize` (`OptixPipeline` pipeline, unsigned int directCallableStackSizeFromTraversal, unsigned int directCallableStackSizeFromState, unsigned int continuationStackSize, unsigned int maxTraversableGraphDepth)

### 3.5.1 Detailed Description

### 3.5.2 Function Documentation

**3.5.2.1 `OptixResult optixPipelineCreate` (**  
**`OptixDeviceContext` context,**  
**const `OptixPipelineCompileOptions` \* pipelineCompileOptions,**  
**const `OptixPipelineLinkOptions` \* pipelineLinkOptions,**  
**const `OptixProgramGroup` \* programGroups,**  
**unsigned int numProgramGroups,**  
**char \* logString,**  
**size\_t \* logStringSize,**  
**`OptixPipeline` \* pipeline )**

logString is an optional buffer that contains compiler feedback and errors. This information is also passed to the context logger (if enabled), however it may be difficult to correlate output to the logger to specific API invocations when using multiple threads. The output to logString will only contain feedback for this specific invocation of this API call.

logStringSize as input should be a pointer to the number of bytes backing logString. Upon return it contains the length of the log message (including the null terminator) which may be greater than the input value. In this case, the log message will be truncated to fit into logString.

If logString or logStringSize are NULL, no output is written to logString. If logStringSize points to a value that is zero, no output is written. This does not affect output to the context logger if enabled.

### Parameters

in	<i>context</i>	
in	<i>pipelineCompileOptions</i>	
in	<i>pipelineLinkOptions</i>	
in	<i>programGroups</i>	array of ProgramGroup objects
in	<i>numProgramGroups</i>	number of ProgramGroup objects

## Parameters

out	<i>logString</i>	Information will be written to this string. If <i>logStringSize</i> > 0 <i>logString</i> will be null terminated.
in,out	<i>logStringSize</i>	
out	<i>pipeline</i>	

### 3.5.2.2 OptixResult optixPipelineDestroy ( OptixPipeline *pipeline* )

Thread safety: A pipeline must not be destroyed while it is still in use by concurrent API calls in other threads.

### 3.5.2.3 OptixResult optixPipelineSetStackSize ( OptixPipeline *pipeline*, unsigned int *directCallableStackSizeFromTraversal*, unsigned int *directCallableStackSizeFromState*, unsigned int *continuationStackSize*, unsigned int *maxTraversableGraphDepth* )

Sets the stack sizes for a pipeline.

Users are encouraged to see the programming guide and the implementations of the helper functions to understand how to construct the stack sizes based on their particular needs.

If this method is not used, an internal default implementation is used. The default implementation is correct (but not necessarily optimal) as long as the maximum depth of call trees of CC and DC programs is at most 2 and no motion transforms are used.

The *maxTraversableGraphDepth* responds to the maximal number of traversables visited when calling trace. Every acceleration structure and motion transform count as one level of traversal. E.g., for a simple IAS (instance acceleration structure) -> GAS (geometry acceleration structure) traversal graph, the *maxTraversableGraphDepth* is two. For IAS -> MT (motion transform) -> GAS, the *maxTraversableGraphDepth* is three. Note that it does not matter whether a IAS or GAS has motion or not, it always counts as one. Launching optix with exceptions turned on (see [OPTIX\\_EXCEPTION\\_FLAG\\_TRACE\\_DEPTH](#)) will throw an exception if the specified *maxTraversableGraphDepth* is too small.

## Parameters

in	<i>pipeline</i>	The pipeline to configure the stack size for.
in	<i>directCallableStackSizeFromTraversal</i>	The direct stack size requirement for direct callables invoked from IS or AH.
in	<i>directCallableStackSizeFromState</i>	The direct stack size requirement for direct callables invoked from RG, MS, or CH.
in	<i>continuationStackSize</i>	The continuation stack requirement.
in	<i>maxTraversableGraphDepth</i>	The maximum depth of a traversable graph passed to trace.

## 3.6 Modules

### Functions

- `OptixResult optixModuleCreateFromPTX` (`OptixDeviceContext` context, const `OptixModuleCompileOptions` \*moduleCompileOptions, const `OptixPipelineCompileOptions` \*pipelineCompileOptions, const char \*PTX, size\_t PTXsize, char \*logString, size\_t \*logStringSize, `OptixModule` \*module)
- `OptixResult optixModuleCreateFromPTXWithTasks` (`OptixDeviceContext` context, const `OptixModuleCompileOptions` \*moduleCompileOptions, const `OptixPipelineCompileOptions` \*pipelineCompileOptions, const char \*PTX, size\_t PTXsize, char \*logString, size\_t \*logStringSize, `OptixModule` \*module, `OptixTask` \*firstTask)
- `OptixResult optixModuleGetCompilationState` (`OptixModule` module, `OptixModuleCompileState` \*state)
- `OptixResult optixModuleDestroy` (`OptixModule` module)
- `OptixResult optixBuiltinISModuleGet` (`OptixDeviceContext` context, const `OptixModuleCompileOptions` \*moduleCompileOptions, const `OptixPipelineCompileOptions` \*pipelineCompileOptions, const `OptixBuiltinISOptions` \*builtinISOptions, `OptixModule` \*builtinModule)

### 3.6.1 Detailed Description

### 3.6.2 Function Documentation

#### 3.6.2.1 `OptixResult optixBuiltinISModuleGet` (

`OptixDeviceContext` context,  
const `OptixModuleCompileOptions` \* *moduleCompileOptions*,  
const `OptixPipelineCompileOptions` \* *pipelineCompileOptions*,  
const `OptixBuiltinISOptions` \* *builtinISOptions*,  
`OptixModule` \* *builtinModule* )

Returns a module containing the intersection program for the built-in primitive type specified by the `builtinISOptions`. This module must be used as the moduleIS for the `OptixProgramGroupHitgroup` in any SBT record for that primitive type. (The `entryFunctionNameIS` should be null.)

#### 3.6.2.2 `OptixResult optixModuleCreateFromPTX` (

`OptixDeviceContext` context,  
const `OptixModuleCompileOptions` \* *moduleCompileOptions*,  
const `OptixPipelineCompileOptions` \* *pipelineCompileOptions*,  
const char \* *PTX*,  
size\_t *PTXsize*,  
char \* *logString*,  
size\_t \* *logStringSize*,

### **OptixModule \* *module* )**

logString is an optional buffer that contains compiler feedback and errors. This information is also passed to the context logger (if enabled), however it may be difficult to correlate output to the logger to specific API invocations when using multiple threads. The output to logString will only contain feedback for this specific invocation of this API call.

logStringSize as input should be a pointer to the number of bytes backing logString. Upon return it contains the length of the log message (including the null terminator) which may be greater than the input value. In this case, the log message will be truncated to fit into logString.

If logString or logStringSize are NULL, no output is written to logString. If logStringSize points to a value that is zero, no output is written. This does not affect output to the context logger if enabled.

#### **Parameters**

in	<i>context</i>	
in	<i>moduleCompileOptions</i>	
in	<i>pipelineCompileOptions</i>	All modules in a pipeline need to use the same values for the pipeline compile options.
in	<i>PTX</i>	Pointer to the PTX input string.
in	<i>PTXsize</i>	Parsing proceeds up to PTXsize characters, or the first NUL byte, whichever occurs first.
out	<i>logString</i>	Information will be written to this string. If logStringSize > 0 logString will be null terminated.
in,out	<i>logStringSize</i>	
out	<i>module</i>	

#### **Returns**

OPTIX\_ERROR\_INVALID\_VALUE - context is 0, moduleCompileOptions is 0, pipelineCompileOptions is 0, PTX is 0, module is 0.

#### **3.6.2.3 OptixResult optixModuleCreateFromPTXWithTasks (**

**OptixDeviceContext *context*,**  
**const OptixModuleCompileOptions \* *moduleCompileOptions*,**  
**const OptixPipelineCompileOptions \* *pipelineCompileOptions*,**  
**const char \* *PTX*,**  
**size\_t *PTXsize*,**  
**char \* *logString*,**  
**size\_t \* *logStringSize*,**  
**OptixModule \* *module*,**  
**OptixTask \* *firstTask* )**

This function is designed to do just enough work to create the OptixTask return parameter and is expected to be fast enough run without needing parallel execution. A single thread could generate all the OptixTask objects for further processing in a work pool.



Options are similar to [optixModuleCreateFromPTX\(\)](#), aside from the return parameter, firstTask.

The memory used to hold the PTX should be live until all tasks are finished.

It is illegal to call [optixModuleDestroy\(\)](#) if any OptixTask objects are currently being executed. In that case OPTIX\_ERROR\_ILLEGAL\_DURING\_TASK\_EXECUTE will be returned.

If an invocation of [optixTaskExecute](#) fails, the OptixModule will be marked as OPTIX\_MODULE\_COMPILE\_STATE\_PENDING\_FAILURE if there are outstanding tasks or OPTIX\_MODULE\_COMPILE\_STATE\_FAILURE if there are no outstanding tasks. Subsequent calls to [optixTaskExecute\(\)](#) may execute additional work to collect compilation errors generated from the input. Currently executing tasks will not necessarily be terminated immediately but at the next opportunity. Logging will continue to be directed to the logger installed with the OptixDeviceContext. If logString is provided to [optixModuleCreateFromPTXWithTasks\(\)](#), it will contain all the compiler feedback from all executed tasks. The lifetime of the memory pointed to by logString should extend from calling [optixModuleCreateFromPTXWithTasks\(\)](#) to when the compilation state is either OPTIX\_MODULE\_COMPILE\_STATE\_FAILURE or OPTIX\_MODULE\_COMPILE\_STATE\_COMPLETED. OptiX will not write to the logString outside of execution of [optixModuleCreateFromPTXWithTasks\(\)](#) or [optixTaskExecute\(\)](#). If the compilation state is OPTIX\_MODULE\_COMPILE\_STATE\_PENDING\_FAILURE and no further execution of [optixTaskExecute\(\)](#) is performed the logString may be reclaimed by the application before calling [optixModuleDestroy\(\)](#). The contents of logString will contain output from currently completed tasks. All OptixTask objects associated with a given OptixModule will be cleaned up when [optixModuleDestroy\(\)](#) is called regardless of whether the compilation was successful or not. If the compilation state is OPTIX\_MODULE\_COMPILE\_STATE\_PENDING\_FAILURE, any unstarted OptixTask objects do not need to be executed though there is no harm doing so.

See Also

[optixModuleCreateFromPTX](#)

#### 3.6.2.4 OptixResult optixModuleDestroy ( OptixModule *module* )

Call for OptixModule objects created with [optixModuleCreateFromPTX](#) and [optixModuleDeserialize](#).

Modules must not be destroyed while they are still used by any program group.

Thread safety: A module must not be destroyed while it is still in use by concurrent API calls in other threads.

#### 3.6.2.5 OptixResult optixModuleGetCompilationState ( OptixModule *module*, OptixModuleCompileState \* *state* )

When creating a module with tasks, the current state of the module can be queried using this function.

Thread safety: Safe to call from any thread until [optixModuleDestroy](#) is called.

See Also

[optixModuleCreateFromPTXWithTasks](#)

## 3.7 Tasks

### Functions

- [OptixResult optixTaskExecute](#) ([OptixTask](#) task, [OptixTask \\*](#)additionalTasks, unsigned int maxNumAdditionalTasks, unsigned int \*numAdditionalTasksCreated)

#### 3.7.1 Detailed Description

#### 3.7.2 Function Documentation

**3.7.2.1 OptixResult optixTaskExecute (**  
**OptixTask task,**  
**OptixTask \* additionalTasks,**  
**unsigned int maxNumAdditionalTasks,**  
**unsigned int \* numAdditionalTasksCreated )**

Each OptixTask should be executed with [optixTaskExecute\(\)](#). If additional parallel work is found, new OptixTask objects will be returned in additionalTasks along with the number of additional tasks in numAdditionalTasksCreated. The parameter additionalTasks should point to a user allocated array of minimum size maxNumAdditionalTasks. OptiX can generate upto maxNumAdditionalTasks additional tasks.

Each task can be executed in parallel and in any order.

Thread safety: Safe to call from any thread until [optixModuleDestroy\(\)](#) is called for any associated task.

See Also

[optixModuleCreateFromPTXWithTasks](#)

### Parameters

in	<i>task</i>	the OptixTask to execute
in	<i>additionalTasks</i>	pointer to array of OptixTask objects to be filled in
in	<i>maxNumAdditionalTasks</i>	maximum number of additional OptixTask objects
out	<i>numAdditionalTasksCreated</i>	number of OptixTask objects created by OptiX and written into #additionalTasks

## 3.8 Program groups

### Functions

- `OptixResult optixProgramGroupGetStackSize (OptixProgramGroup programGroup, OptixStackSizes *stackSizes)`
- `OptixResult optixProgramGroupCreate (OptixDeviceContext context, const OptixProgramGroupDesc *programDescriptions, unsigned int numProgramGroups, const OptixProgramGroupOptions *options, char *logString, size_t *logStringSize, OptixProgramGroup *programGroups)`
- `OptixResult optixProgramGroupDestroy (OptixProgramGroup programGroup)`

### 3.8.1 Detailed Description

### 3.8.2 Function Documentation

**3.8.2.1 OptixResult optixProgramGroupCreate (**  
**OptixDeviceContext *context*,**  
**const OptixProgramGroupDesc \* *programDescriptions*,**  
**unsigned int *numProgramGroups*,**  
**const OptixProgramGroupOptions \* *options*,**  
**char \* *logString*,**  
**size\_t \* *logStringSize*,**  
**OptixProgramGroup \* *programGroups* )**

`logString` is an optional buffer that contains compiler feedback and errors. This information is also passed to the context logger (if enabled), however it may be difficult to correlate output to the logger to specific API invocations when using multiple threads. The output to `logString` will only contain feedback for this specific invocation of this API call.

`logStringSize` as input should be a pointer to the number of bytes backing `logString`. Upon return it contains the length of the log message (including the null terminator) which may be greater than the input value. In this case, the log message will be truncated to fit into `logString`.

If `logString` or `logStringSize` are NULL, no output is written to `logString`. If `logStringSize` points to a value that is zero, no output is written. This does not affect output to the context logger if enabled.

Creates `numProgramGroups` OptiXProgramGroup objects from the specified `OptixProgramGroupDesc` array. The size of the arrays must match.

### Parameters

in	<i>context</i>	
in	<i>programDescriptions</i>	N * <code>OptixProgramGroupDesc</code>
in	<i>numProgramGroups</i>	N
in	<i>options</i>	
out	<i>logString</i>	Information will be written to this string. If <code>logStringSize &gt; 0</code> <code>logString</code> will be null terminated.

**Parameters**

in, out	<i>logStringSize</i>	
out	<i>programGroups</i>	

**3.8.2.2 OptixResult optixProgramGroupDestroy (**  
**OptixProgramGroup *programGroup* )**

Thread safety: A program group must not be destroyed while it is still in use by concurrent API calls in other threads.

**3.8.2.3 OptixResult optixProgramGroupGetStackSize (**  
**OptixProgramGroup *programGroup*,**  
**OptixStackSizes \* *stackSizes* )**

Returns the stack sizes for the given program group.

**Parameters**

in	<i>programGroup</i>	the program group
out	<i>stackSizes</i>	the corresponding stack sizes

## 3.9 Launches

### Functions

- [OptixResult optixLaunch](#) ([OptixPipeline](#) pipeline, [CUstream](#) stream, [CUdeviceptr](#) pipelineParams, [size\\_t](#) pipelineParamsSize, const [OptixShaderBindingTable](#) \*sbt, unsigned int width, unsigned int height, unsigned int depth)
- [OptixResult optixSbtRecordPackHeader](#) ([OptixProgramGroup](#) programGroup, void \*sbtRecordHeaderHostPointer)

#### 3.9.1 Detailed Description

#### 3.9.2 Function Documentation

**3.9.2.1 OptixResult optixLaunch (**  
**OptixPipeline *pipeline*,**  
**CUstream *stream*,**  
**CUdeviceptr *pipelineParams*,**  
**size\_t *pipelineParamsSize*,**  
**const OptixShaderBindingTable \* *sbt*,**  
**unsigned int *width*,**  
**unsigned int *height*,**  
**unsigned int *depth* )**

Where the magic happens.

The stream and pipeline must belong to the same device context. Multiple launches may be issues in parallel from multiple threads to different streams.

pipelineParamsSize number of bytes are copied from the device memory pointed to by pipelineParams before launch. It is an error if pipelineParamsSize is greater than the size of the variable declared in modules and identified by [OptixPipelineCompileOptions::pipelineLaunchParamsVariableName](#). If the launch params variable was optimized out or not found in the modules linked to the pipeline then the pipelineParams and pipelineParamsSize parameters are ignored.

sbt points to the shader binding table, which defines shader groupings and their resources. See the SBT spec.

### Parameters

in	<i>pipeline</i>	
in	<i>stream</i>	
in	<i>pipelineParams</i>	
in	<i>pipelineParamsSize</i>	
in	<i>sbt</i>	
in	<i>width</i>	number of elements to compute
in	<i>height</i>	number of elements to compute

**Parameters**

in	<i>depth</i>	number of elements to compute
----	--------------	-------------------------------

Thread safety: In the current implementation concurrent launches to the same pipeline are not supported. Concurrent launches require separate OptixPipeline objects.

**3.9.2.2 OptixResult optixSbtRecordPackHeader (**  
**OptixProgramGroup *programGroup*,**  
**void \* *sbtRecordHeaderHostPointer* )**

**Parameters**

in	<i>programGroup</i>	the program group containing the program(s)
out	<i>sbtRecordHeaderHostPointer</i>	the result sbt record header

## 3.10 Acceleration structures

### Functions

- `OptixResult optixAccelComputeMemoryUsage` (`OptixDeviceContext` context, const `OptixAccelBuildOptions` \*accelOptions, const `OptixBuildInput` \*buildInputs, unsigned int numBuildInputs, `OptixAccelBufferSizes` \*bufferSizes)
- `OptixResult optixAccelBuild` (`OptixDeviceContext` context, `CUstream` stream, const `OptixAccelBuildOptions` \*accelOptions, const `OptixBuildInput` \*buildInputs, unsigned int numBuildInputs, `CUdeviceptr` tempBuffer, `size_t` tempBufferSizeInBytes, `CUdeviceptr` outputBuffer, `size_t` outputBufferSizeInBytes, `OptixTraversableHandle` \*outputHandle, const `OptixAccelEmitDesc` \*emittedProperties, unsigned int numEmittedProperties)
- `OptixResult optixAccelGetRelocationInfo` (`OptixDeviceContext` context, `OptixTraversableHandle` handle, `OptixAccelRelocationInfo` \*info)
- `OptixResult optixAccelCheckRelocationCompatibility` (`OptixDeviceContext` context, const `OptixAccelRelocationInfo` \*info, int \*compatible)
- `OptixResult optixAccelRelocate` (`OptixDeviceContext` context, `CUstream` stream, const `OptixAccelRelocationInfo` \*info, `CUdeviceptr` instanceTraversableHandles, `size_t` numInstanceTraversableHandles, `CUdeviceptr` targetAccel, `size_t` targetAccelSizeInBytes, `OptixTraversableHandle` \*targetHandle)
- `OptixResult optixAccelCompact` (`OptixDeviceContext` context, `CUstream` stream, `OptixTraversableHandle` inputHandle, `CUdeviceptr` outputBuffer, `size_t` outputBufferSizeInBytes, `OptixTraversableHandle` \*outputHandle)
- `OptixResult optixConvertPointerToTraversableHandle` (`OptixDeviceContext` onDevice, `CUdeviceptr` pointer, `OptixTraversableType` traversableType, `OptixTraversableHandle` \*traversableHandle)

#### 3.10.1 Detailed Description

#### 3.10.2 Function Documentation

**3.10.2.1 `OptixResult optixAccelBuild` (**  
     `OptixDeviceContext` *context*,  
     `CUstream` *stream*,  
     const `OptixAccelBuildOptions` \* *accelOptions*,  
     const `OptixBuildInput` \* *buildInputs*,  
     unsigned int *numBuildInputs*,  
     `CUdeviceptr` *tempBuffer*,  
     `size_t` *tempBufferSizeInBytes*,  
     `CUdeviceptr` *outputBuffer*,  
     `size_t` *outputBufferSizeInBytes*,  
     `OptixTraversableHandle` \* *outputHandle*,  
     const `OptixAccelEmitDesc` \* *emittedProperties*,  
     unsigned int *numEmittedProperties* )

**Parameters**

in	<i>context</i>	
in	<i>stream</i>	
in	<i>accelOptions</i>	accel options
in	<i>buildInputs</i>	an array of <a href="#">OptixBuildInput</a> objects
in	<i>numBuildInputs</i>	must be $\geq 1$ for GAS, and $= 1$ for IAS
in	<i>tempBuffer</i>	must be a multiple of <code>OPTIX_ACCEL_BUFFER_BYTE_ALIGNMENT</code>
in	<i>tempBufferSizeInBytes</i>	
in	<i>outputBuffer</i>	must be a multiple of <code>OPTIX_ACCEL_BUFFER_BYTE_ALIGNMENT</code>
in	<i>outputBufferSizeInBytes</i>	
out	<i>outputHandle</i>	
out	<i>emittedProperties</i>	types of requested properties and output buffers
in	<i>numEmittedProperties</i>	number of post-build properties to populate (may be zero)

**3.10.2.2 OptixResult optixAccelCheckRelocationCompatibility (**

**OptixDeviceContext *context*,**  
**const OptixAccelRelocationInfo \* *info*,**  
**int \* *compatible* )**

Checks if an acceleration structure built using another OptixDeviceContext (that was used to fill in 'info') is compatible with the OptixDeviceContext specified in the 'context' parameter.

Any device is always compatible with itself.

**Parameters**

in	<i>context</i>	
in	<i>info</i>	
out	<i>compatible</i>	If <code>OPTIX_SUCCESS</code> is returned ' <i>compatible</i> ' will have the value of either: <ul style="list-style-type: none"> <li>• 0: This context is not compatible with acceleration structure data associated with 'info'.</li> <li>• 1: This context is compatible.</li> </ul>

**3.10.2.3 OptixResult optixAccelCompact (**

**OptixDeviceContext *context*,**  
**CUstream *stream*,**  
**OptixTraversableHandle *inputHandle*,**  
**CUdeviceptr *outputBuffer*,**  
**size\_t *outputBufferSizeInBytes*,**



**OptixTraversableHandle \* *outputHandle* )**

After building an acceleration structure, it can be copied in a compacted form to reduce memory. In order to be compacted, `OPTIX_BUILD_FLAG_ALLOW_COMPACTION` must be supplied in `OptixAccelBuildOptions::buildFlags` passed to `optixAccelBuild`.

'outputBuffer' is the pointer to where the compacted acceleration structure will be written. This pointer must be a multiple of `OPTIX_ACCEL_BUFFER_BYTE_ALIGNMENT`.

The size of the memory specified in 'outputBufferSizeInBytes' should be at least the value computed using the `OPTIX_PROPERTY_TYPE_COMPACTED_SIZE` that was reported during `optixAccelBuild`.

#### Parameters

in	<i>context</i>	
in	<i>stream</i>	
in	<i>inputHandle</i>	
in	<i>outputBuffer</i>	
in	<i>outputBufferSizeInBytes</i>	
out	<i>outputHandle</i>	

**3.10.2.4 OptixResult optixAccelComputeMemoryUsage (**  
**OptixDeviceContext *context*,**  
**const OptixAccelBuildOptions \* *accelOptions*,**  
**const OptixBuildInput \* *buildInputs*,**  
**unsigned int *numBuildInputs*,**  
**OptixAccelBufferSizes \* *bufferSizes* )**

#### Parameters

in	<i>context</i>	
in	<i>accelOptions</i>	options for the accel build
in	<i>buildInputs</i>	an array of <code>OptixBuildInput</code> objects
in	<i>numBuildInputs</i>	number of elements in buildInputs (must be at least 1)
out	<i>bufferSizes</i>	fills in buffer sizes

**3.10.2.5 OptixResult optixAccelGetRelocationInfo (**  
**OptixDeviceContext *context*,**  
**OptixTraversableHandle *handle*,**  
**OptixAccelRelocationInfo \* *info* )**

Obtain relocation information, stored in `OptixAccelRelocationInfo`, for a given context and acceleration structure's traversable handle.

The relocation information can be passed to `optixAccelCheckRelocationCompatibility` to determine if an

acceleration structure, referenced by 'handle', can be relocated to a different device's memory space (see [optixAccelCheckRelocationCompatibility](#)).

When used with `optixAccelRelocate`, it provides data necessary for doing the relocation.

If the acceleration structure data associated with 'handle' is copied multiple times, the same [OptixAccelRelocationInfo](#) can also be used on all copies.

### Parameters

in	<i>context</i>	
in	<i>handle</i>	
out	<i>info</i>	

### Returns

`OPTIX_ERROR_INVALID_VALUE` will be returned for traversable handles that are not from acceleration structure builds.

**3.10.2.6 OptixResult optixAccelRelocate (**  
**OptixDeviceContext *context*,**  
**CUstream *stream*,**  
**const OptixAccelRelocationInfo \* *info*,**  
**CUdeviceptr *instanceTraversableHandles*,**  
**size\_t *numInstanceTraversableHandles*,**  
**CUdeviceptr *targetAccel*,**  
**size\_t *targetAccelSizeInBytes*,**  
**OptixTraversableHandle \* *targetHandle* )**

`optixAccelRelocate` is called to update the acceleration structure after it has been relocated. Relocation is necessary when the acceleration structure's location in device memory has changed.

`optixAccelRelocate` does not copy the memory. This function only operates on the relocated memory whose new location is specified by 'targetAccel'. `optixAccelRelocate` also returns the new `OptixTraversableHandle` associated with 'targetAccel'. The original memory (source) is not required to be valid, only the [OptixAccelRelocationInfo](#).

Before copying the data and calling `optixAccelRelocate`, `optixAccelCheckRelocationCompatibility` should be called to ensure the copy will be compatible with the destination device context.

The memory pointed to by 'targetAccel' should be allocated with the same size as the source acceleration. Similar to the 'outputBuffer' used in `optixAccelBuild`, this pointer must be a multiple of `OPTIX_ACCEL_BUFFER_BYTE_ALIGNMENT`.

The memory in 'targetAccel' must be allocated as long as the accel is in use.

When relocating an accel that contains instances, 'instanceTraversableHandles' and 'numInstanceTraversableHandles' should be supplied. These are the traversable handles of the instances. These can be used when also relocating the instances. No updates to the bounds are performed. Use `optixAccelBuild` to update the bounds. 'instanceTraversableHandles' and 'numInstanceTraversableHandles' may be zero when relocating bottom level accel (i.e. an accel with no instances).

**Parameters**

in	<i>context</i>	
in	<i>stream</i>	
in	<i>info</i>	
in	<i>instanceTraversableHandles</i>	
in	<i>numInstanceTraversableHandles</i>	
in	<i>targetAccel</i>	
in	<i>targetAccelSizeInBytes</i>	
out	<i>targetHandle</i>	

**3.10.2.7 OptixResult optixConvertPointerToTraversableHandle (**  
**OptixDeviceContext *onDevice*,**  
**CUdeviceptr *pointer*,**  
**OptixTraversableType *traversableType*,**  
**OptixTraversableHandle \* *traversableHandle* )**

**Parameters**

in	<i>onDevice</i>	
in	<i>pointer</i>	pointer to traversable allocated in OptixDeviceContext. This pointer must be a multiple of OPTIX_TRANSFORM_BYTE_ALIGNMENT
in	<i>traversableType</i>	Type of OptixTraversableHandle to create
out	<i>traversableHandle</i>	traversable handle. traversableHandle must be in host memory

## 3.11 Denoiser

### Functions

- `OptixResult optixDenoiserCreate` (`OptixDeviceContext` context, `OptixDenoiserModelKind` modelKind, const `OptixDenoiserOptions` \*options, `OptixDenoiser` \*denoiser)
- `OptixResult optixDenoiserCreateWithUserModel` (`OptixDeviceContext` context, const void \*userData, `size_t` userDataSizeInBytes, `OptixDenoiser` \*denoiser)
- `OptixResult optixDenoiserDestroy` (`OptixDenoiser` denoiser)
- `OptixResult optixDenoiserComputeMemoryResources` (const `OptixDenoiser` denoiser, unsigned int outputWidth, unsigned int outputHeight, `OptixDenoiserSizes` \*returnSizes)
- `OptixResult optixDenoiserSetup` (`OptixDenoiser` denoiser, `CUstream` stream, unsigned int inputWidth, unsigned int inputHeight, `CUdeviceptr` denoiserState, `size_t` denoiserStateSizeInBytes, `CUdeviceptr` scratch, `size_t` scratchSizeInBytes)
- `OptixResult optixDenoiserInvoke` (`OptixDenoiser` denoiser, `CUstream` stream, const `OptixDenoiserParams` \*params, `CUdeviceptr` denoiserState, `size_t` denoiserStateSizeInBytes, const `OptixDenoiserGuideLayer` \*guideLayer, const `OptixDenoiserLayer` \*layers, unsigned int numLayers, unsigned int inputOffsetX, unsigned int inputOffsetY, `CUdeviceptr` scratch, `size_t` scratchSizeInBytes)
- `OptixResult optixDenoiserComputeIntensity` (`OptixDenoiser` denoiser, `CUstream` stream, const `OptixImage2D` \*inputImage, `CUdeviceptr` outputIntensity, `CUdeviceptr` scratch, `size_t` scratchSizeInBytes)
- `OptixResult optixDenoiserComputeAverageColor` (`OptixDenoiser` denoiser, `CUstream` stream, const `OptixImage2D` \*inputImage, `CUdeviceptr` outputAverageColor, `CUdeviceptr` scratch, `size_t` scratchSizeInBytes)

#### 3.11.1 Detailed Description

#### 3.11.2 Function Documentation

##### 3.11.2.1 `OptixResult optixDenoiserComputeAverageColor` (

`OptixDenoiser denoiser`,  
`CUstream stream`,  
const `OptixImage2D` \* `inputImage`,  
`CUdeviceptr outputAverageColor`,  
`CUdeviceptr scratch`,  
`size_t scratchSizeInBytes` )

Compute average logarithmic for each of the first three channels for the given image. When denoising tiles the intensity of the entire image should be computed, i.e. not per tile to get consistent results. This function needs scratch memory with a size of at least `sizeof( int ) * ( 3 + 3 * inputImage::width * inputImage::height )`. When denoising entire images (no tiling) the same scratch memory as passed to `optixDenoiserInvoke` could be used.

data type unsigned char is not supported for 'inputImage', it must be 3 or 4 component half/float.

### Parameters

in	<i>denoiser</i>	
in	<i>stream</i>	
in	<i>inputImage</i>	
out	<i>outputAverageColor</i>	three floats
in	<i>scratch</i>	
in	<i>scratchSizeInBytes</i>	

#### 3.11.2.2 OptixResult optixDenoiserComputeIntensity (

**OptixDenoiser** *denoiser*,  
**CUstream** *stream*,  
**const OptixImage2D \*** *inputImage*,  
**CUdeviceptr** *outputIntensity*,  
**CUdeviceptr** *scratch*,  
**size\_t** *scratchSizeInBytes* )

Computes the logarithmic average intensity of the given image. The returned value 'outputIntensity' is multiplied with the RGB values of the input image/tile in optixDenoiserInvoke if given in the parameter [OptixDenoiserParams::hdrIntensity](#) (otherwise 'hdrIntensity' must be a null pointer). This is useful for denoising HDR images which are very dark or bright. When denoising tiles the intensity of the entire image should be computed, i.e. not per tile to get consistent results.

For each RGB pixel in the inputImage the intensity is calculated and summed if it is greater than 1e-8f:  $\text{intensity} = \log(r * 0.212586f + g * 0.715170f + b * 0.072200f)$ . The function returns  $0.18 / \exp(\text{sum of intensities} / \text{number of summed pixels})$ . More details could be found in the Reinhard tonemapping paper: [http://www.cmap.polytechnique.fr/~peyre/cours/x2005signal/hdr\\_photographic.pdf](http://www.cmap.polytechnique.fr/~peyre/cours/x2005signal/hdr_photographic.pdf)

This function needs scratch memory with a size of at least  $\text{sizeof(int)} * (2 + \text{inputImage::width} * \text{inputImage::height})$ . When denoising entire images (no tiling) the same scratch memory as passed to optixDenoiserInvoke could be used. data type unsigned char is not supported for 'inputImage', it must be 3 or 4 component half/float.

### Parameters

in	<i>denoiser</i>	
in	<i>stream</i>	
in	<i>inputImage</i>	
out	<i>outputIntensity</i>	single float
in	<i>scratch</i>	
in	<i>scratchSizeInBytes</i>	

#### 3.11.2.3 OptixResult optixDenoiserComputeMemoryResources (

**const OptixDenoiser** *denoiser*,

**unsigned int *outputWidth*,**  
**unsigned int *outputHeight*,**  
**OptixDenoiserSizes \* *returnSizes* )**

Computes the GPU memory resources required to execute the denoiser.

Memory for state and scratch buffers must be allocated with the sizes in 'returnSizes' and scratch memory passed to `optixDenoiserSetup`, `optixDenoiserInvoke`, `optixDenoiserComputeIntensity` and `optixDenoiserComputeAverageColor`. For tiled denoising an overlap area must be added to each tile on all sides which increases the amount of memory needed to denoise a tile. In case of tiling use `withOverlapScratchSizeInBytes`. If only full resolution images are denoised, `withoutOverlapScratchSizeInBytes` can be used which is always smaller than `withOverlapScratchSizeInBytes`.

'outputWidth' and 'outputHeight' is the dimension of the image to be denoised (without overlap in case tiling is being used). 'outputWidth' and 'outputHeight' must be greater than or equal to the dimensions passed to `optixDenoiserSetup`.

#### Parameters

in	<i>denoiser</i>	
in	<i>outputWidth</i>	
in	<i>outputHeight</i>	
out	<i>returnSizes</i>	

**3.11.2.4 OptixResult optixDenoiserCreate (**  
**OptixDeviceContext *context*,**  
**OptixDenoiserModelKind *modelKind*,**  
**const OptixDenoiserOptions \* *options*,**  
**OptixDenoiser \* *denoiser* )**

Creates a denoiser object with the given options, using built-in inference models.

'modelKind' selects the model used for inference. Inference for the built-in models can be guided (giving hints to improve image quality) with albedo and normal vector images in the guide layer (see 'optixDenoiserInvoke'). Use of these images must be enabled in '[OptixDenoiserOptions](#)'.

#### Parameters

in	<i>context</i>	
in	<i>modelKind</i>	
in	<i>options</i>	
out	<i>denoiser</i>	

**3.11.2.5 OptixResult optixDenoiserCreateWithUserModel (**  
**OptixDeviceContext *context*,**

```

const void * userData,
size_t userDataSizeInBytes,
OptixDenoiser * denoiser )

```

Creates a denoiser object with the given options, using a provided inference model.

'userData' and 'userDataSizeInBytes' provide a user model for inference. The memory passed in userData will be accessed only during the invocation of this function and can be freed after it returns. The user model must export only one weight set which determines both the model kind and the required set of guide images.

#### Parameters

in	<i>context</i>	
in	<i>userData</i>	
in	<i>userDataSizeInBytes</i>	
out	<i>denoiser</i>	

#### 3.11.2.6 OptixResult optixDenoiserDestroy ( OptixDenoiser *denoiser* )

Destroys the denoiser object and any associated host resources.

#### 3.11.2.7 OptixResult optixDenoiserInvoke ( OptixDenoiser *denoiser*, CUstream *stream*, const OptixDenoiserParams \* *params*, CUdeviceptr *denoiserState*, size\_t *denoiserStateSizeInBytes*, const OptixDenoiserGuideLayer \* *guideLayer*, const OptixDenoiserLayer \* *layers*, unsigned int *numLayers*, unsigned int *inputOffsetX*, unsigned int *inputOffsetY*, CUdeviceptr *scratch*, size\_t *scratchSizeInBytes* )

Invokes denoiser on a set of input data and produces at least one output image. State memory must be available during the execution of the denoiser (or until optixDenoiserSetup is called with a new state memory pointer). Scratch memory passed is used only for the duration of this function. Scratch and state memory sizes must have a size greater than or equal to the sizes as returned by optixDenoiserComputeMemoryResources.

'inputOffsetX' and 'inputOffsetY' are pixel offsets in the 'inputLayers' image specifying the beginning of the image without overlap. When denoising an entire image without tiling there is no overlap and 'inputOffsetX' and 'inputOffsetY' must be zero. When denoising a tile which is adjacent to one of the four sides of the entire image the corresponding offsets must also be zero since there is no overlap at

the side adjacent to the image border.

'guideLayer' provides additional information to the denoiser. When providing albedo and normal vector guide images, the corresponding fields in the '[OptixDenoiserOptions](#)' must be enabled, see [optixDenoiserCreate](#). 'guideLayer' must not be null. If a guide image in '[OptixDenoiserOptions](#)' is not enabled, the corresponding image in '[OptixDenoiserGuideLayer](#)' is ignored.

If `OPTIX_DENOISER_MODEL_KIND_TEMPORAL` or `OPTIX_DENOISER_MODEL_KIND_TEMPORAL_AOV` is selected, a 2d flow image must be given in '[OptixDenoiserGuideLayer](#)'. It describes for each pixel the flow from the previous to the current frame (a 2d vector in pixel space). The denoised beauty/AOV of the previous frame must be given in 'previousOutput'. If this image is not available in the first frame of a sequence, the noisy beauty/AOV from the first frame and zero flow vectors could be given as a substitute. For non-temporal model kinds the flow image in '[OptixDenoiserGuideLayer](#)' is ignored. 'previousOutput' and 'output' may refer to the same buffer, i.e. 'previousOutput' is first read by this function and later overwritten with the denoised result. 'output' can be passed as 'previousOutput' to the next frame. In other model kinds (not temporal) 'previousOutput' is ignored.

The beauty layer must be given as the first entry in 'layers'. In AOV type model kinds (`OPTIX_DENOISER_MODEL_KIND_AOV` or in user defined models implementing kernel-prediction) additional layers for the AOV images can be given. In each layer the noisy input image is given in 'input', the denoised output is written into the 'output' image. input and output images may refer to the same buffer, with the restriction that the pixel formats must be identical for input and output when the blend mode is selected (see [OptixDenoiserParams](#)).

If `OPTIX_DENOISER_MODEL_KIND_TEMPORAL` or `OPTIX_DENOISER_MODEL_KIND_TEMPORAL_AOV` is selected, the denoised image from the previous frame must be given in 'previousOutput' in the layer. 'previousOutput' and 'output' may refer to the same buffer, i.e. 'previousOutput' is first read by this function and later overwritten with the denoised result. 'output' can be passed as 'previousOutput' to the next frame. In other model kinds (not temporal) 'previousOutput' is ignored.

If `OPTIX_DENOISER_MODEL_KIND_TEMPORAL` or `OPTIX_DENOISER_MODEL_KIND_TEMPORAL_AOV` is selected, the normal vector guide image must be given as 3d vectors in camera space. In the other models only the x and y channels are used and other channels are ignored.

## Parameters

in	<i>denoiser</i>	
in	<i>stream</i>	
in	<i>params</i>	
in	<i>denoiserState</i>	
in	<i>denoiserStateSizeInBytes</i>	
in	<i>guideLayer</i>	
in	<i>layers</i>	
in	<i>numLayers</i>	
in	<i>inputOffsetX</i>	
in	<i>inputOffsetY</i>	



**Parameters**

in	<i>scratch</i>	
in	<i>scratchSizeInBytes</i>	

**3.11.2.8 OptixResult optixDenoiserSetup (**  
**OptixDenoiser *denoiser*,**  
**CUstream *stream*,**  
**unsigned int *inputWidth*,**  
**unsigned int *inputHeight*,**  
**CUdeviceptr *denoiserState*,**  
**size\_t *denoiserStateSizeInBytes*,**  
**CUdeviceptr *scratch*,**  
**size\_t *scratchSizeInBytes* )**

Initializes the state required by the denoiser.

'inputWidth' and 'inputHeight' must include overlap on both sides of the image if tiling is being used. The overlap is returned by [optixDenoiserComputeMemoryResources](#). For subsequent calls to [optixDenoiserInvoke](#) 'inputWidth' and 'inputHeight' are the maximum dimensions of the input layers. Dimensions of the input layers passed to [optixDenoiserInvoke](#) may be different in each invocation however they always must be smaller than 'inputWidth' and 'inputHeight' passed to [optixDenoiserSetup](#).

**Parameters**

in	<i>denoiser</i>	
in	<i>stream</i>	
in	<i>inputWidth</i>	
in	<i>inputHeight</i>	
in	<i>denoiserState</i>	
in	<i>denoiserStateSizeInBytes</i>	
in	<i>scratch</i>	
in	<i>scratchSizeInBytes</i>	

## 3.12 Types

### Classes

- struct [OptixDeviceContextOptions](#)
- struct [OptixBuildInputTriangleArray](#)
- struct [OptixBuildInputCurveArray](#)
- struct [OptixAabb](#)
- struct [OptixBuildInputCustomPrimitiveArray](#)
- struct [OptixBuildInputInstanceArray](#)
- struct [OptixBuildInput](#)
- struct [OptixInstance](#)
- struct [OptixMotionOptions](#)
- struct [OptixAccelBuildOptions](#)
- struct [OptixAccelBufferSizes](#)
- struct [OptixAccelEmitDesc](#)
- struct [OptixAccelRelocationInfo](#)
- struct [OptixStaticTransform](#)
- struct [OptixMatrixMotionTransform](#)
- struct [OptixSRTData](#)
- struct [OptixSRTMotionTransform](#)
- struct [OptixImage2D](#)
- struct [OptixDenoiserOptions](#)
- struct [OptixDenoiserGuideLayer](#)
- struct [OptixDenoiserLayer](#)
- struct [OptixDenoiserParams](#)
- struct [OptixDenoiserSizes](#)
- struct [OptixModuleCompileBoundValueEntry](#)
- struct [OptixPayloadType](#)
- struct [OptixModuleCompileOptions](#)
- struct [OptixProgramGroupSingleModule](#)
- struct [OptixProgramGroupHitgroup](#)
- struct [OptixProgramGroupCallables](#)
- struct [OptixProgramGroupDesc](#)
- struct [OptixProgramGroupOptions](#)
- struct [OptixPipelineCompileOptions](#)
- struct [OptixPipelineLinkOptions](#)
- struct [OptixShaderBindingTable](#)
- struct [OptixStackSizes](#)
- struct [OptixBuiltinISOOptions](#)

## Macros

- `#define OPTIX_SBT_RECORD_HEADER_SIZE ( (size_t)32 )`
- `#define OPTIX_SBT_RECORD_ALIGNMENT 16ull`
- `#define OPTIX_ACCEL_BUFFER_BYTE_ALIGNMENT 128ull`
- `#define OPTIX_INSTANCE_BYTE_ALIGNMENT 16ull`
- `#define OPTIX_AABB_BUFFER_BYTE_ALIGNMENT 8ull`
- `#define OPTIX_GEOMETRY_TRANSFORM_BYTE_ALIGNMENT 16ull`
- `#define OPTIX_TRANSFORM_BYTE_ALIGNMENT 64ull`
- `#define OPTIX_COMPILE_DEFAULT_MAX_REGISTER_COUNT 0`
- `#define OPTIX_COMPILE_DEFAULT_MAX_PAYLOAD_TYPE_COUNT 8`
- `#define OPTIX_COMPILE_DEFAULT_MAX_PAYLOAD_VALUE_COUNT 32`

## Typedefs

- `typedef unsigned long long CUdeviceptr`
- `typedef struct`  
`OptixDeviceContext_t * OptixDeviceContext`
- `typedef struct OptixModule_t * OptixModule`
- `typedef struct`  
`OptixProgramGroup_t * OptixProgramGroup`
- `typedef struct OptixPipeline_t * OptixPipeline`
- `typedef struct OptixDenoiser_t * OptixDenoiser`
- `typedef struct OptixTask_t * OptixTask`
- `typedef unsigned long long OptixTraversableHandle`
- `typedef unsigned int OptixVisibilityMask`
- `typedef enum OptixResult OptixResult`
- `typedef enum OptixDeviceProperty OptixDeviceProperty`
- `typedef void(* OptixLogCallback )(unsigned int level, const char *tag, const char *message, void *cbdata)`
- `typedef enum`  
`OptixDeviceContextValidationMode OptixDeviceContextValidationMode`
- `typedef struct`  
`OptixDeviceContextOptions OptixDeviceContextOptions`
- `typedef enum OptixGeometryFlags OptixGeometryFlags`
- `typedef enum OptixHitKind OptixHitKind`
- `typedef enum OptixIndicesFormat OptixIndicesFormat`
- `typedef enum OptixVertexFormat OptixVertexFormat`
- `typedef enum OptixTransformFormat OptixTransformFormat`
- `typedef struct`  
`OptixBuildInputTriangleArray OptixBuildInputTriangleArray`
- `typedef enum OptixPrimitiveType OptixPrimitiveType`

- typedef enum  
OptixPrimitiveTypeFlags OptixPrimitiveTypeFlags
- typedef enum OptixCurveEndcapFlags OptixCurveEndcapFlags
- typedef struct  
OptixBuildInputCurveArray OptixBuildInputCurveArray
- typedef struct OptixAabb OptixAabb
- typedef struct  
OptixBuildInputCustomPrimitiveArray OptixBuildInputCustomPrimitiveArray
- typedef struct  
OptixBuildInputInstanceArray OptixBuildInputInstanceArray
- typedef enum OptixBuildInputType OptixBuildInputType
- typedef struct OptixBuildInput OptixBuildInput
- typedef enum OptixInstanceFlags OptixInstanceFlags
- typedef struct OptixInstance OptixInstance
- typedef enum OptixBuildFlags OptixBuildFlags
- typedef enum OptixBuildOperation OptixBuildOperation
- typedef enum OptixMotionFlags OptixMotionFlags
- typedef struct OptixMotionOptions OptixMotionOptions
- typedef struct  
OptixAccelBuildOptions OptixAccelBuildOptions
- typedef struct  
OptixAccelBufferSizes OptixAccelBufferSizes
- typedef enum OptixAccelPropertyType OptixAccelPropertyType
- typedef struct OptixAccelEmitDesc OptixAccelEmitDesc
- typedef struct  
OptixAccelRelocationInfo OptixAccelRelocationInfo
- typedef struct OptixStaticTransform OptixStaticTransform
- typedef struct  
OptixMatrixMotionTransform OptixMatrixMotionTransform
- typedef struct OptixSRTData OptixSRTData
- typedef struct  
OptixSRTMotionTransform OptixSRTMotionTransform
- typedef enum OptixTraversableType OptixTraversableType
- typedef enum OptixPixelFormat OptixPixelFormat
- typedef struct OptixImage2D OptixImage2D
- typedef enum OptixDenoiserModelKind OptixDenoiserModelKind
- typedef struct OptixDenoiserOptions OptixDenoiserOptions
- typedef struct  
OptixDenoiserGuideLayer OptixDenoiserGuideLayer
- typedef struct OptixDenoiserLayer OptixDenoiserLayer
- typedef struct OptixDenoiserParams OptixDenoiserParams
- typedef struct OptixDenoiserSizes OptixDenoiserSizes

- typedef enum `OptixRayFlags` `OptixRayFlags`
- typedef enum `OptixTransformType` `OptixTransformType`
- typedef enum  
    `OptixTraversableGraphFlags` `OptixTraversableGraphFlags`
- typedef enum  
    `OptixCompileOptimizationLevel` `OptixCompileOptimizationLevel`
- typedef enum `OptixCompileDebugLevel` `OptixCompileDebugLevel`
- typedef enum  
    `OptixModuleCompileState` `OptixModuleCompileState`
- typedef struct  
    `OptixModuleCompileBoundValueEntry` `OptixModuleCompileBoundValueEntry`
- typedef enum `OptixPayloadTypeID` `OptixPayloadTypeID`
- typedef enum `OptixPayloadSemantics` `OptixPayloadSemantics`
- typedef struct `OptixPayloadType` `OptixPayloadType`
- typedef struct  
    `OptixModuleCompileOptions` `OptixModuleCompileOptions`
- typedef enum `OptixProgramGroupKind` `OptixProgramGroupKind`
- typedef enum `OptixProgramGroupFlags` `OptixProgramGroupFlags`
- typedef struct  
    `OptixProgramGroupSingleModule` `OptixProgramGroupSingleModule`
- typedef struct  
    `OptixProgramGroupHitgroup` `OptixProgramGroupHitgroup`
- typedef struct  
    `OptixProgramGroupCallables` `OptixProgramGroupCallables`
- typedef struct  
    `OptixProgramGroupDesc` `OptixProgramGroupDesc`
- typedef struct  
    `OptixProgramGroupOptions` `OptixProgramGroupOptions`
- typedef enum `OptixExceptionCodes` `OptixExceptionCodes`
- typedef enum `OptixExceptionFlags` `OptixExceptionFlags`
- typedef struct  
    `OptixPipelineCompileOptions` `OptixPipelineCompileOptions`
- typedef struct  
    `OptixPipelineLinkOptions` `OptixPipelineLinkOptions`
- typedef struct  
    `OptixShaderBindingTable` `OptixShaderBindingTable`
- typedef struct `OptixStackSizes` `OptixStackSizes`
- typedef enum  
    `OptixQueryFunctionTableOptions` `OptixQueryFunctionTableOptions`
- typedef `OptixResult( OptixQueryFunctionTable_t )(int abild, unsigned int numOptions,  
    OptixQueryFunctionTableOptions *, const void **, void *functionTable, size_t sizeofTable)`
- typedef struct  
    `OptixBuiltinISOOptions` `OptixBuiltinISOOptions`

## Enumerations

- enum OptixResult {
  - OPTIX\_SUCCESS = 0,
  - OPTIX\_ERROR\_INVALID\_VALUE = 7001,
  - OPTIX\_ERROR\_HOST\_OUT\_OF\_MEMORY = 7002,
  - OPTIX\_ERROR\_INVALID\_OPERATION = 7003,
  - OPTIX\_ERROR\_FILE\_IO\_ERROR = 7004,
  - OPTIX\_ERROR\_INVALID\_FILE\_FORMAT = 7005,
  - OPTIX\_ERROR\_DISK\_CACHE\_INVALID\_PATH = 7010,
  - OPTIX\_ERROR\_DISK\_CACHE\_PERMISSION\_ERROR = 7011,
  - OPTIX\_ERROR\_DISK\_CACHE\_DATABASE\_ERROR = 7012,
  - OPTIX\_ERROR\_DISK\_CACHE\_INVALID\_DATA = 7013,
  - OPTIX\_ERROR\_LAUNCH\_FAILURE = 7050,
  - OPTIX\_ERROR\_INVALID\_DEVICE\_CONTEXT = 7051,
  - OPTIX\_ERROR\_CUDA\_NOT\_INITIALIZED = 7052,
  - OPTIX\_ERROR\_VALIDATION\_FAILURE = 7053,
  - OPTIX\_ERROR\_INVALID\_PTX = 7200,
  - OPTIX\_ERROR\_INVALID\_LAUNCH\_PARAMETER = 7201,
  - OPTIX\_ERROR\_INVALID\_PAYLOAD\_ACCESS = 7202,
  - OPTIX\_ERROR\_INVALID\_ATTRIBUTE\_ACCESS = 7203,
  - OPTIX\_ERROR\_INVALID\_FUNCTION\_USE = 7204,
  - OPTIX\_ERROR\_INVALID\_FUNCTION\_ARGUMENTS = 7205,
  - OPTIX\_ERROR\_PIPELINE\_OUT\_OF\_CONSTANT\_MEMORY = 7250,
  - OPTIX\_ERROR\_PIPELINE\_LINK\_ERROR = 7251,
  - OPTIX\_ERROR\_ILLEGAL\_DURING\_TASK\_EXECUTE = 7270,
  - OPTIX\_ERROR\_INTERNAL\_COMPILER\_ERROR = 7299,
  - OPTIX\_ERROR\_DENOISER\_MODEL\_NOT\_SET = 7300,
  - OPTIX\_ERROR\_DENOISER\_NOT\_INITIALIZED = 7301,
  - OPTIX\_ERROR\_ACCEL\_NOT\_COMPATIBLE = 7400,
  - OPTIX\_ERROR\_PAYLOAD\_TYPE\_MISMATCH = 7500,
  - OPTIX\_ERROR\_PAYLOAD\_TYPE\_RESOLUTION\_FAILED = 7501,
  - OPTIX\_ERROR\_PAYLOAD\_TYPE\_ID\_INVALID = 7502,
  - OPTIX\_ERROR\_NOT\_SUPPORTED = 7800,
  - OPTIX\_ERROR\_UNSUPPORTED\_ABI\_VERSION = 7801,
  - OPTIX\_ERROR\_FUNCTION\_TABLE\_SIZE\_MISMATCH = 7802,
  - OPTIX\_ERROR\_INVALID\_ENTRY\_FUNCTION\_OPTIONS = 7803,
  - OPTIX\_ERROR\_LIBRARY\_NOT\_FOUND = 7804,
  - OPTIX\_ERROR\_ENTRY\_SYMBOL\_NOT\_FOUND = 7805,
  - OPTIX\_ERROR\_LIBRARY\_UNLOAD\_FAILURE = 7806,
  - OPTIX\_ERROR\_CUDA\_ERROR = 7900,
  - OPTIX\_ERROR\_INTERNAL\_ERROR = 7990,
  - OPTIX\_ERROR\_UNKNOWN = 7999 }

- enum OptixDeviceProperty {  
OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_TRACE\_DEPTH = 0x2001,  
OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_TRAVERSABLE\_GRAPH\_DEPTH = 0x2002,  
OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_PRIMITIVES\_PER\_GAS = 0x2003,  
OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_INSTANCES\_PER\_IAS = 0x2004,  
OPTIX\_DEVICE\_PROPERTY\_RTCORE\_VERSION = 0x2005,  
OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_INSTANCE\_ID = 0x2006,  
OPTIX\_DEVICE\_PROPERTY\_LIMIT\_NUM\_BITS\_INSTANCE\_VISIBILITY\_MASK = 0x2007,  
OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_SBT\_RECORDS\_PER\_GAS = 0x2008,  
OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_SBT\_OFFSET = 0x2009 }
- enum OptixDeviceContextValidationMode {  
OPTIX\_DEVICE\_CONTEXT\_VALIDATION\_MODE\_OFF = 0,  
OPTIX\_DEVICE\_CONTEXT\_VALIDATION\_MODE\_ALL = 0xFFFFFFFF }
- enum OptixGeometryFlags {  
OPTIX\_GEOMETRY\_FLAG\_NONE = 0,  
OPTIX\_GEOMETRY\_FLAG\_DISABLE\_ANYHIT = 1u << 0,  
OPTIX\_GEOMETRY\_FLAG\_REQUIRE\_SINGLE\_ANYHIT\_CALL = 1u << 1 }
- enum OptixHitKind {  
OPTIX\_HIT\_KIND\_TRIANGLE\_FRONT\_FACE = 0xFE,  
OPTIX\_HIT\_KIND\_TRIANGLE\_BACK\_FACE = 0xFF }
- enum OptixIndicesFormat {  
OPTIX\_INDICES\_FORMAT\_NONE = 0,  
OPTIX\_INDICES\_FORMAT\_UNSIGNED\_SHORT3 = 0x2102,  
OPTIX\_INDICES\_FORMAT\_UNSIGNED\_INT3 = 0x2103 }
- enum OptixVertexFormat {  
OPTIX\_VERTEX\_FORMAT\_NONE = 0,  
OPTIX\_VERTEX\_FORMAT\_FLOAT3 = 0x2121,  
OPTIX\_VERTEX\_FORMAT\_FLOAT2 = 0x2122,  
OPTIX\_VERTEX\_FORMAT\_HALF3 = 0x2123,  
OPTIX\_VERTEX\_FORMAT\_HALF2 = 0x2124,  
OPTIX\_VERTEX\_FORMAT\_SNORM16\_3 = 0x2125,  
OPTIX\_VERTEX\_FORMAT\_SNORM16\_2 = 0x2126 }
- enum OptixTransformFormat {  
OPTIX\_TRANSFORM\_FORMAT\_NONE = 0,  
OPTIX\_TRANSFORM\_FORMAT\_MATRIX\_FLOAT12 = 0x21E1 }
- enum OptixPrimitiveType {  
OPTIX\_PRIMITIVE\_TYPE\_CUSTOM = 0x2500,  
OPTIX\_PRIMITIVE\_TYPE\_ROUND\_QUADRATIC\_BSPLINE = 0x2501,  
OPTIX\_PRIMITIVE\_TYPE\_ROUND\_CUBIC\_BSPLINE = 0x2502,  
OPTIX\_PRIMITIVE\_TYPE\_ROUND\_LINEAR = 0x2503,  
OPTIX\_PRIMITIVE\_TYPE\_ROUND\_CATMULLROM = 0x2504,  
OPTIX\_PRIMITIVE\_TYPE\_TRIANGLE = 0x2531 }

- enum OptixPrimitiveTypeFlags {  
OPTIX\_PRIMITIVE\_TYPE\_FLAGS\_CUSTOM = 1 << 0,  
OPTIX\_PRIMITIVE\_TYPE\_FLAGS\_ROUND\_QUADRATIC\_BSPLINE = 1 << 1,  
OPTIX\_PRIMITIVE\_TYPE\_FLAGS\_ROUND\_CUBIC\_BSPLINE = 1 << 2,  
OPTIX\_PRIMITIVE\_TYPE\_FLAGS\_ROUND\_LINEAR = 1 << 3,  
OPTIX\_PRIMITIVE\_TYPE\_FLAGS\_ROUND\_CATMULLROM = 1 << 4,  
OPTIX\_PRIMITIVE\_TYPE\_FLAGS\_TRIANGLE = 1 << 31 }
- enum OptixCurveEndcapFlags {  
OPTIX\_CURVE\_ENDCAP\_DEFAULT = 0,  
OPTIX\_CURVE\_ENDCAP\_ON = 1 << 0 }
- enum OptixBuildInputType {  
OPTIX\_BUILD\_INPUT\_TYPE\_TRIANGLES = 0x2141,  
OPTIX\_BUILD\_INPUT\_TYPE\_CUSTOM\_PRIMITIVES = 0x2142,  
OPTIX\_BUILD\_INPUT\_TYPE\_INSTANCES = 0x2143,  
OPTIX\_BUILD\_INPUT\_TYPE\_INSTANCE\_POINTERS = 0x2144,  
OPTIX\_BUILD\_INPUT\_TYPE\_CURVES = 0x2145 }
- enum OptixInstanceFlags {  
OPTIX\_INSTANCE\_FLAG\_NONE = 0,  
OPTIX\_INSTANCE\_FLAG\_DISABLE\_TRIANGLE\_FACE\_CULLING = 1u << 0,  
OPTIX\_INSTANCE\_FLAG\_FLIP\_TRIANGLE\_FACING = 1u << 1,  
OPTIX\_INSTANCE\_FLAG\_DISABLE\_ANYHIT = 1u << 2,  
OPTIX\_INSTANCE\_FLAG\_ENFORCE\_ANYHIT = 1u << 3 }
- enum OptixBuildFlags {  
OPTIX\_BUILD\_FLAG\_NONE = 0,  
OPTIX\_BUILD\_FLAG\_ALLOW\_UPDATE = 1u << 0,  
OPTIX\_BUILD\_FLAG\_ALLOW\_COMPACTION = 1u << 1,  
OPTIX\_BUILD\_FLAG\_PREFER\_FAST\_TRACE = 1u << 2,  
OPTIX\_BUILD\_FLAG\_PREFER\_FAST\_BUILD = 1u << 3,  
OPTIX\_BUILD\_FLAG\_ALLOW\_RANDOM\_VERTEX\_ACCESS = 1u << 4,  
OPTIX\_BUILD\_FLAG\_ALLOW\_RANDOM\_INSTANCE\_ACCESS = 1u << 5 }
- enum OptixBuildOperation {  
OPTIX\_BUILD\_OPERATION\_BUILD = 0x2161,  
OPTIX\_BUILD\_OPERATION\_UPDATE = 0x2162 }
- enum OptixMotionFlags {  
OPTIX\_MOTION\_FLAG\_NONE = 0,  
OPTIX\_MOTION\_FLAG\_START\_VANISH = 1u << 0,  
OPTIX\_MOTION\_FLAG\_END\_VANISH = 1u << 1 }
- enum OptixAccelPropertyType {  
OPTIX\_PROPERTY\_TYPE\_COMPACTED\_SIZE = 0x2181,  
OPTIX\_PROPERTY\_TYPE\_AABBS = 0x2182 }
- enum OptixTraversableType {  
OPTIX\_TRAVERSABLE\_TYPE\_STATIC\_TRANSFORM = 0x21C1,  
OPTIX\_TRAVERSABLE\_TYPE\_MATRIX\_MOTION\_TRANSFORM = 0x21C2,  
OPTIX\_TRAVERSABLE\_TYPE\_SRT\_MOTION\_TRANSFORM = 0x21C3 }



- enum OptixPixelFormat {  
OPTIX\_PIXEL\_FORMAT\_HALF2 = 0x2207,  
OPTIX\_PIXEL\_FORMAT\_HALF3 = 0x2201,  
OPTIX\_PIXEL\_FORMAT\_HALF4 = 0x2202,  
OPTIX\_PIXEL\_FORMAT\_FLOAT2 = 0x2208,  
OPTIX\_PIXEL\_FORMAT\_FLOAT3 = 0x2203,  
OPTIX\_PIXEL\_FORMAT\_FLOAT4 = 0x2204,  
OPTIX\_PIXEL\_FORMAT\_UCHAR3 = 0x2205,  
OPTIX\_PIXEL\_FORMAT\_UCHAR4 = 0x2206 }
- enum OptixDenoiserModelKind {  
OPTIX\_DENOISER\_MODEL\_KIND\_LDR = 0x2322,  
OPTIX\_DENOISER\_MODEL\_KIND\_HDR = 0x2323,  
OPTIX\_DENOISER\_MODEL\_KIND\_AOV = 0x2324,  
OPTIX\_DENOISER\_MODEL\_KIND\_TEMPORAL = 0x2325,  
OPTIX\_DENOISER\_MODEL\_KIND\_TEMPORAL\_AOV = 0x2326 }
- enum OptixRayFlags {  
OPTIX\_RAY\_FLAG\_NONE = 0u,  
OPTIX\_RAY\_FLAG\_DISABLE\_ANYHIT = 1u << 0,  
OPTIX\_RAY\_FLAG\_ENFORCE\_ANYHIT = 1u << 1,  
OPTIX\_RAY\_FLAG\_TERMINATE\_ON\_FIRST\_HIT = 1u << 2,  
OPTIX\_RAY\_FLAG\_DISABLE\_CLOSESTHIT = 1u << 3,  
OPTIX\_RAY\_FLAG\_CULL\_BACK\_FACING\_TRIANGLES = 1u << 4,  
OPTIX\_RAY\_FLAG\_CULL\_FRONT\_FACING\_TRIANGLES = 1u << 5,  
OPTIX\_RAY\_FLAG\_CULL\_DISABLED\_ANYHIT = 1u << 6,  
OPTIX\_RAY\_FLAG\_CULL\_ENFORCED\_ANYHIT = 1u << 7 }
- enum OptixTransformType {  
OPTIX\_TRANSFORM\_TYPE\_NONE = 0,  
OPTIX\_TRANSFORM\_TYPE\_STATIC\_TRANSFORM = 1,  
OPTIX\_TRANSFORM\_TYPE\_MATRIX\_MOTION\_TRANSFORM = 2,  
OPTIX\_TRANSFORM\_TYPE\_SRT\_MOTION\_TRANSFORM = 3,  
OPTIX\_TRANSFORM\_TYPE\_INSTANCE = 4 }
- enum OptixTraversableGraphFlags {  
OPTIX\_TRAVERSABLE\_GRAPH\_FLAG\_ALLOW\_ANY = 0,  
OPTIX\_TRAVERSABLE\_GRAPH\_FLAG\_ALLOW\_SINGLE\_GAS = 1u << 0,  
OPTIX\_TRAVERSABLE\_GRAPH\_FLAG\_ALLOW\_SINGLE\_LEVEL\_INSTANCING = 1u << 1 }
- enum OptixCompileOptimizationLevel {  
OPTIX\_COMPILE\_OPTIMIZATION\_DEFAULT = 0,  
OPTIX\_COMPILE\_OPTIMIZATION\_LEVEL\_0 = 0x2340,  
OPTIX\_COMPILE\_OPTIMIZATION\_LEVEL\_1 = 0x2341,  
OPTIX\_COMPILE\_OPTIMIZATION\_LEVEL\_2 = 0x2342,  
OPTIX\_COMPILE\_OPTIMIZATION\_LEVEL\_3 = 0x2343 }
- enum OptixCompileDebugLevel {  
OPTIX\_COMPILE\_DEBUG\_LEVEL\_DEFAULT = 0,  
OPTIX\_COMPILE\_DEBUG\_LEVEL\_NONE = 0x2350,  
OPTIX\_COMPILE\_DEBUG\_LEVEL\_MINIMAL = 0x2351,  
OPTIX\_COMPILE\_DEBUG\_LEVEL\_MODERATE = 0x2353,  
OPTIX\_COMPILE\_DEBUG\_LEVEL\_FULL = 0x2352 }

- enum OptixModuleCompileState {  
OPTIX\_MODULE\_COMPILE\_STATE\_NOT\_STARTED = 0x2360,  
OPTIX\_MODULE\_COMPILE\_STATE\_STARTED = 0x2361,  
OPTIX\_MODULE\_COMPILE\_STATE\_IMPENDING\_FAILURE = 0x2362,  
OPTIX\_MODULE\_COMPILE\_STATE\_FAILED = 0x2363,  
OPTIX\_MODULE\_COMPILE\_STATE\_COMPLETED = 0x2364 }
- enum OptixPayloadTypeID {  
OPTIX\_PAYLOAD\_TYPE\_DEFAULT = 0,  
OPTIX\_PAYLOAD\_TYPE\_ID\_0 = (1 << 0u),  
OPTIX\_PAYLOAD\_TYPE\_ID\_1 = (1 << 1u),  
OPTIX\_PAYLOAD\_TYPE\_ID\_2 = (1 << 2u),  
OPTIX\_PAYLOAD\_TYPE\_ID\_3 = (1 << 3u),  
OPTIX\_PAYLOAD\_TYPE\_ID\_4 = (1 << 4u),  
OPTIX\_PAYLOAD\_TYPE\_ID\_5 = (1 << 5u),  
OPTIX\_PAYLOAD\_TYPE\_ID\_6 = (1 << 6u),  
OPTIX\_PAYLOAD\_TYPE\_ID\_7 = (1 << 7u) }
- enum OptixPayloadSemantics {  
OPTIX\_PAYLOAD\_SEMANTICS\_TRACE\_CALLER\_NONE = 0,  
OPTIX\_PAYLOAD\_SEMANTICS\_TRACE\_CALLER\_READ = 1u << 0,  
OPTIX\_PAYLOAD\_SEMANTICS\_TRACE\_CALLER\_WRITE = 2u << 0,  
OPTIX\_PAYLOAD\_SEMANTICS\_TRACE\_CALLER\_READ\_WRITE = 3u << 0,  
OPTIX\_PAYLOAD\_SEMANTICS\_CH\_NONE = 0,  
OPTIX\_PAYLOAD\_SEMANTICS\_CH\_READ = 1u << 2,  
OPTIX\_PAYLOAD\_SEMANTICS\_CH\_WRITE = 2u << 2,  
OPTIX\_PAYLOAD\_SEMANTICS\_CH\_READ\_WRITE = 3u << 2,  
OPTIX\_PAYLOAD\_SEMANTICS\_MS\_NONE = 0,  
OPTIX\_PAYLOAD\_SEMANTICS\_MS\_READ = 1u << 4,  
OPTIX\_PAYLOAD\_SEMANTICS\_MS\_WRITE = 2u << 4,  
OPTIX\_PAYLOAD\_SEMANTICS\_MS\_READ\_WRITE = 3u << 4,  
OPTIX\_PAYLOAD\_SEMANTICS\_AH\_NONE = 0,  
OPTIX\_PAYLOAD\_SEMANTICS\_AH\_READ = 1u << 6,  
OPTIX\_PAYLOAD\_SEMANTICS\_AH\_WRITE = 2u << 6,  
OPTIX\_PAYLOAD\_SEMANTICS\_AH\_READ\_WRITE = 3u << 6,  
OPTIX\_PAYLOAD\_SEMANTICS\_IS\_NONE = 0,  
OPTIX\_PAYLOAD\_SEMANTICS\_IS\_READ = 1u << 8,  
OPTIX\_PAYLOAD\_SEMANTICS\_IS\_WRITE = 2u << 8,  
OPTIX\_PAYLOAD\_SEMANTICS\_IS\_READ\_WRITE = 3u << 8 }
- enum OptixProgramGroupKind {  
OPTIX\_PROGRAM\_GROUP\_KIND\_RAYGEN = 0x2421,  
OPTIX\_PROGRAM\_GROUP\_KIND\_MISS = 0x2422,  
OPTIX\_PROGRAM\_GROUP\_KIND\_EXCEPTION = 0x2423,  
OPTIX\_PROGRAM\_GROUP\_KIND\_HITGROUP = 0x2424,  
OPTIX\_PROGRAM\_GROUP\_KIND\_CALLABLES = 0x2425 }
- enum OptixProgramGroupFlags { OPTIX\_PROGRAM\_GROUP\_FLAGS\_NONE = 0 }

- enum OptixExceptionCodes {
  - OPTIX\_EXCEPTION\_CODE\_STACK\_OVERFLOW = -1,
  - OPTIX\_EXCEPTION\_CODE\_TRACE\_DEPTH\_EXCEEDED = -2,
  - OPTIX\_EXCEPTION\_CODE\_TRAVERSAL\_DEPTH\_EXCEEDED = -3,
  - OPTIX\_EXCEPTION\_CODE\_TRAVERSAL\_INVALID\_TRAVERSABLE = -5,
  - OPTIX\_EXCEPTION\_CODE\_TRAVERSAL\_INVALID\_MISS\_SBT = -6,
  - OPTIX\_EXCEPTION\_CODE\_TRAVERSAL\_INVALID\_HIT\_SBT = -7,
  - OPTIX\_EXCEPTION\_CODE\_UNSUPPORTED\_PRIMITIVE\_TYPE = -8,
  - OPTIX\_EXCEPTION\_CODE\_INVALID\_RAY = -9,
  - OPTIX\_EXCEPTION\_CODE\_CALLABLE\_PARAMETER\_MISMATCH = -10,
  - OPTIX\_EXCEPTION\_CODE\_BUILTIN\_IS\_MISMATCH = -11,
  - OPTIX\_EXCEPTION\_CODE\_CALLABLE\_INVALID\_SBT = -12,
  - OPTIX\_EXCEPTION\_CODE\_CALLABLE\_NO\_DC\_SBT\_RECORD = -13,
  - OPTIX\_EXCEPTION\_CODE\_CALLABLE\_NO\_CC\_SBT\_RECORD = -14,
  - OPTIX\_EXCEPTION\_CODE\_UNSUPPORTED\_SINGLE\_LEVEL\_GAS = -15,
  - OPTIX\_EXCEPTION\_CODE\_INVALID\_VALUE\_ARGUMENT\_0 = -16,
  - OPTIX\_EXCEPTION\_CODE\_INVALID\_VALUE\_ARGUMENT\_1 = -17,
  - OPTIX\_EXCEPTION\_CODE\_INVALID\_VALUE\_ARGUMENT\_2 = -18,
  - OPTIX\_EXCEPTION\_CODE\_UNSUPPORTED\_DATA\_ACCESS = -32,
  - OPTIX\_EXCEPTION\_CODE\_PAYLOAD\_TYPE\_MISMATCH = -33 }
- enum OptixExceptionFlags {
  - OPTIX\_EXCEPTION\_FLAG\_NONE = 0,
  - OPTIX\_EXCEPTION\_FLAG\_STACK\_OVERFLOW = 1u << 0,
  - OPTIX\_EXCEPTION\_FLAG\_TRACE\_DEPTH = 1u << 1,
  - OPTIX\_EXCEPTION\_FLAG\_USER = 1u << 2,
  - OPTIX\_EXCEPTION\_FLAG\_DEBUG = 1u << 3 }
- enum OptixQueryFunctionTableOptions {
  - OPTIX\_QUERY\_FUNCTION\_TABLE\_OPTION\_DUMMY = 0 }

### 3.12.1 Detailed Description

OptiX Types.

### 3.12.2 Macro Definition Documentation

#### 3.12.2.1 #define OPTIX\_AABB\_BUFFER\_BYTE\_ALIGNMENT 8ull

Alignment requirement for `OptixBuildInputCustomPrimitiveArray::aabbBuffers`.

#### 3.12.2.2 #define OPTIX\_ACCEL\_BUFFER\_BYTE\_ALIGNMENT 128ull

Alignment requirement for output and temporary buffers for acceleration structures.

#### 3.12.2.3 #define OPTIX\_COMPILE\_DEFAULT\_MAX\_PAYLOAD\_TYPE\_COUNT 8

Maximum number of payload types allowed.

### 3.12.2.4 **#define OPTIX\_COMPILE\_DEFAULT\_MAX\_PAYLOAD\_VALUE\_COUNT 32**

Maximum number of payload values allowed.

### 3.12.2.5 **#define OPTIX\_COMPILE\_DEFAULT\_MAX\_REGISTER\_COUNT 0**

Maximum number of registers allowed. Defaults to no explicit limit.

### 3.12.2.6 **#define OPTIX\_GEOMETRY\_TRANSFORM\_BYTE\_ALIGNMENT 16ull**

Alignment requirement for [OptixBuildInputTriangleArray::preTransform](#).

### 3.12.2.7 **#define OPTIX\_INSTANCE\_BYTE\_ALIGNMENT 16ull**

Alignment requirement for [OptixBuildInputInstanceArray::instances](#).

### 3.12.2.8 **#define OPTIX\_SBT\_RECORD\_ALIGNMENT 16ull**

Alignment requirement for device pointers in [OptixShaderBindingTable](#).

### 3.12.2.9 **#define OPTIX\_SBT\_RECORD\_HEADER\_SIZE ( (size\_t)32 )**

Size of the SBT record headers.

### 3.12.2.10 **#define OPTIX\_TRANSFORM\_BYTE\_ALIGNMENT 64ull**

Alignment requirement for [OptixStaticTransform](#), [OptixMatrixMotionTransform](#), [OptixSRTMotionTransform](#).

## 3.12.3 Typedef Documentation

### 3.12.3.1 **typedef unsigned long long CUdeviceptr**

CUDA device pointer.

### 3.12.3.2 **typedef struct OptixAabb OptixAabb**

AABB inputs.

### 3.12.3.3 **typedef struct OptixAccelBufferSizes OptixAccelBufferSizes**

Struct for querying builder allocation requirements.

Once queried the sizes should be used to allocate device memory of at least these sizes.

See Also

[optixAccelComputeMemoryUsage\(\)](#)

### 3.12.3.4 **typedef struct OptixAccelBuildOptions OptixAccelBuildOptions**

Build options for acceleration structures.

See Also

[optixAccelComputeMemoryUsage\(\)](#), [optixAccelBuild\(\)](#)

### 3.12.3.5 **typedef struct OptixAccelEmitDesc OptixAccelEmitDesc**

Specifies a type and output destination for emitted post-build properties.

See Also

[optixAccelBuild\(\)](#)

### 3.12.3.6 **typedef enum OptixAccelPropertyType OptixAccelPropertyType**

Properties which can be emitted during acceleration structure build.

See Also

[OptixAccelEmitDesc::type](#).

### 3.12.3.7 **typedef struct OptixAccelRelocationInfo OptixAccelRelocationInfo**

Used to store information related to relocation of acceleration structures.

See Also

[optixAccelGetRelocationInfo\(\)](#), [optixAccelCheckRelocationCompatibility\(\)](#), [optixAccelRelocate\(\)](#)

### 3.12.3.8 **typedef enum OptixBuildFlags OptixBuildFlags**

Builder Options.

Used for [OptixAccelBuildOptions::buildFlags](#). Can be or'ed together.

### 3.12.3.9 **typedef struct OptixBuildInput OptixBuildInput**

Build inputs.

All of them support motion and the size of the data arrays needs to match the number of motion steps

See Also

[optixAccelComputeMemoryUsage\(\)](#), [optixAccelBuild\(\)](#)

### 3.12.3.10 **typedef struct OptixBuildInputCurveArray OptixBuildInputCurveArray**

Curve inputs.

A curve is a swept surface defined by a 3D spline curve and a varying width (radius). A curve (or "strand") of degree  $d$  (3=cubic, 2=quadratic, 1=linear) is represented by  $N > d$  vertices and  $N$  width values, and comprises  $N - d$  segments. Each segment is defined by  $d+1$  consecutive vertices. Each curve may have a different number of vertices.

OptiX describes the curve array as a list of curve segments. The primitive id is the segment number. It is the user's responsibility to maintain a mapping between curves and curve segments. Each index buffer entry  $i = \text{indexBuffer}[\text{primid}]$  specifies the start of a curve segment, represented by  $d+1$  consecutive vertices in the vertex buffer, and  $d+1$  consecutive widths in the width buffer. Width is interpolated the same way vertices are interpolated, that is, using the curve basis.

Each curves build input has only one SBT record. To create curves with different materials in the same BVH, use multiple build inputs.

See Also

[OptixBuildInput::curveArray](#)

### 3.12.3.11 **typedef struct OptixBuildInputCustomPrimitiveArray OptixBuildInputCustomPrimitiveArray**

Custom primitive inputs.

See Also

[OptixBuildInput::customPrimitiveArray](#)

### 3.12.3.12 **typedef struct OptixBuildInputInstanceArray OptixBuildInputInstanceArray**

Instance and instance pointer inputs.

See Also

[OptixBuildInput::instanceArray](#)

### 3.12.3.13 **typedef struct OptixBuildInputTriangleArray OptixBuildInputTriangleArray**

Triangle inputs.

See Also

[OptixBuildInput::triangleArray](#)

### 3.12.3.14 **typedef enum OptixBuildInputType OptixBuildInputType**

Enum to distinguish the different build input types.

See Also

[OptixBuildInput::type](#)

### 3.12.3.15 **typedef enum OptixBuildOperation OptixBuildOperation**

Enum to specify the acceleration build operation.

Used in [OptixAccelBuildOptions](#), which is then passed to `optixAccelBuild` and `optixAccelComputeMemoryUsage`, this enum indicates whether to do a build or an update of the acceleration structure.

Acceleration structure updates utilize the same acceleration structure, but with updated bounds. Updates are typically much faster than builds, however, large perturbations can degrade the quality of the acceleration structure.

See Also

[optixAccelComputeMemoryUsage\(\)](#), [optixAccelBuild\(\)](#), [OptixAccelBuildOptions](#)

#### 3.12.3.16 **typedef struct OptixBuiltinISOptions OptixBuiltinISOptions**

Specifies the options for retrieving an intersection program for a built-in primitive type. The primitive type must not be `OPTIX_PRIMITIVE_TYPE_CUSTOM`.

See Also

[optixBuiltinISModuleGet\(\)](#)

#### 3.12.3.17 **typedef enum OptixCompileDebugLevel OptixCompileDebugLevel**

Debug levels.

See Also

[OptixModuleCompileOptions::debugLevel](#)

#### 3.12.3.18 **typedef enum OptixCompileOptimizationLevel OptixCompileOptimizationLevel**

Optimization levels.

See Also

[OptixModuleCompileOptions::optLevel](#)

#### 3.12.3.19 **typedef enum OptixCurveEndcapFlags OptixCurveEndcapFlags**

Curve end cap types, for non-linear curves.

#### 3.12.3.20 **typedef struct OptixDenoiser\_t\* OptixDenoiser**

Opaque type representing a denoiser instance.

#### 3.12.3.21 **typedef struct OptixDenoiserGuideLayer OptixDenoiserGuideLayer**

Guide layer for the denoiser.

See Also

[optixDenoiserInvoke\(\)](#)

#### 3.12.3.22 **typedef struct OptixDenoiserLayer OptixDenoiserLayer**

Input/Output layers for the denoiser.

See Also

[optixDenoiserInvoke\(\)](#)

### 3.12.3.23 **typedef enum OptixDenoiserModelKind OptixDenoiserModelKind**

Model kind used by the denoiser.

See Also

[optixDenoiserCreate](#)

### 3.12.3.24 **typedef struct OptixDenoiserOptions OptixDenoiserOptions**

Options used by the denoiser.

See Also

[optixDenoiserCreate\(\)](#)

### 3.12.3.25 **typedef struct OptixDenoiserParams OptixDenoiserParams**

Various parameters used by the denoiser.

See Also

[optixDenoiserInvoke\(\)](#)  
[optixDenoiserComputeIntensity\(\)](#)  
[optixDenoiserComputeAverageColor\(\)](#)

### 3.12.3.26 **typedef struct OptixDenoiserSizes OptixDenoiserSizes**

Various sizes related to the denoiser.

See Also

[optixDenoiserComputeMemoryResources\(\)](#)

### 3.12.3.27 **typedef struct OptixDeviceContext\_t\* OptixDeviceContext**

Opaque type representing a device context.

### 3.12.3.28 **typedef struct OptixDeviceContextOptions OptixDeviceContextOptions**

Parameters used for [optixDeviceContextCreate\(\)](#)

See Also

[optixDeviceContextCreate\(\)](#)



### 3.12.3.29 **typedef enum OptixDeviceContextValidationMode OptixDeviceContextValidationMode**

Validation mode settings.

When enabled, certain device code utilities will be enabled to provide as good debug and error checking facilities as possible.

See Also

[optixDeviceContextCreate\(\)](#)

### 3.12.3.30 **typedef enum OptixDeviceProperty OptixDeviceProperty**

Parameters used for [optixDeviceContextGetProperty\(\)](#)

See Also

[optixDeviceContextGetProperty\(\)](#)

### 3.12.3.31 **typedef enum OptixExceptionCodes OptixExceptionCodes**

The following values are used to indicate which exception was thrown.

### 3.12.3.32 **typedef enum OptixExceptionFlags OptixExceptionFlags**

Exception flags.

See Also

[OptixPipelineCompileOptions::exceptionFlags](#), [OptixExceptionCodes](#)

### 3.12.3.33 **typedef enum OptixGeometryFlags OptixGeometryFlags**

Flags used by [OptixBuildInputTriangleArray::flags](#) and [#OptixBuildInput::flag](#) and [OptixBuildInputCustomPrimitiveArray::flags](#).

### 3.12.3.34 **typedef enum OptixHitKind OptixHitKind**

Legacy type: A subset of the hit kinds for built-in primitive intersections. It is preferred to use [optixGetPrimitiveType\(\)](#), together with [optixIsFrontFaceHit\(\)](#) or [optixIsBackFaceHit\(\)](#).

See Also

[optixGetHitKind\(\)](#)

### 3.12.3.35 **typedef struct OptixImage2D OptixImage2D**

Image descriptor used by the denoiser.

See Also

[optixDenoiserInvoke\(\)](#), [optixDenoiserComputeIntensity\(\)](#)

### 3.12.3.36 **typedef enum OptixIndicesFormat OptixIndicesFormat**

Format of indices used in `OptixBuildInputTriangleArray::indexFormat`.

### 3.12.3.37 **typedef struct OptixInstance OptixInstance**

Instances.

See Also

`OptixBuildInputInstanceArray::instances`

### 3.12.3.38 **typedef enum OptixInstanceFlags OptixInstanceFlags**

Flags set on the `OptixInstance::flags`.

These can be or'ed together to combine multiple flags.

### 3.12.3.39 **typedef void( \* OptixLogCallback)(unsigned int level, const char \*tag, const char \*message, void \*cbdata)**

Type of the callback function used for log messages.

#### Parameters

in	<i>level</i>	The log level indicates the severity of the message. See below for possible values.
in	<i>tag</i>	A terse message category description (e.g., 'SCENE STAT').
in	<i>message</i>	Null terminated log message (without newline at the end).
in	<i>cbdata</i>	Callback data that was provided with the callback pointer.

It is the users responsibility to ensure thread safety within this function.

The following log levels are defined.

0 disable Setting the callback level will disable all messages. The callback function will not be called in this case. 1 fatal A non-recoverable error. The context and/or OptiX itself might no longer be in a usable state. 2 error A recoverable error, e.g., when passing invalid call parameters. 3 warning Hints that OptiX might not behave exactly as requested by the user or may perform slower than expected. 4 print Status or progress messages.

Higher levels might occur.

See Also

`optixDeviceContextSetLogCallback()`, `OptixDeviceContextOptions`

### 3.12.3.40 **typedef struct OptixMatrixMotionTransform OptixMatrixMotionTransform**

Represents a matrix motion transformation.

The device address of instances of this type must be a multiple of `OPTIX_TRANSFORM_BYTE_ALIGNMENT`.

This struct, as defined here, handles only N=2 motion keys due to the fixed array length of its transform member. The following example shows how to create instances for an arbitrary number N of motion keys:

```
float matrixData[N][12];
... // setup matrixData

size_t transformSizeInBytes = sizeof( OptixMatrixMotionTransform ) + ( N-2 ) * 12
    * sizeof( float );
OptixMatrixMotionTransform* matrixMoptionTransform = (
    OptixMatrixMotionTransform*) malloc( transformSizeInBytes );
memset( matrixMoptionTransform, 0, transformSizeInBytes );

... // setup other members of matrixMoptionTransform
matrixMoptionTransform->motionOptions.numKeys
memcpy( matrixMoptionTransform->transform, matrixData, N * 12 * sizeof( float ) );

... // copy matrixMoptionTransform to device memory
free( matrixMoptionTransform )
```

See Also

[optixConvertPointerToTraversableHandle\(\)](#)

### 3.12.3.41 typedef struct OptixModule\_t\* OptixModule

Opaque type representing a module.

### 3.12.3.42 typedef struct OptixModuleCompileBoundValueEntry OptixModuleCompileBound-ValueEntry

Struct for specifying specializations for pipelineParams as specified in [OptixPipelineCompileOptions::pipelineLaunchParamsVariableName](#).

The bound values are supposed to represent a constant value in the pipelineParams. OptiX will attempt to locate all loads from the pipelineParams and correlate them to the appropriate bound value, but there are cases where OptiX cannot safely or reliably do this. For example if the pointer to the pipelineParams is passed as an argument to a non-inline function or the offset of the load to the pipelineParams cannot be statically determined (e.g. accessed in a loop). No module should rely on the value being specialized in order to work correctly. The values in the pipelineParams specified on optixLaunch should match the bound value. If validation mode is enabled on the context, OptiX will verify that the bound values specified matches the values in pipelineParams specified to optixLaunch.

These values are compiled in to the module as constants. Once the constants are inserted into the code, an optimization pass will be run that will attempt to propagate the consants and remove unreachable code.

If caching is enabled, changes in these values will result in newly compiled modules.

The pipelineParamOffset and sizeInBytes must be within the bounds of the pipelineParams variable. OPTIX\_ERROR\_INVALID\_VALUE will be returned from optixModuleCreateFromPTX otherwise.

If more than one bound value overlaps or the size of a bound value is equal to 0, an OPTIX\_ERROR\_INVALID\_VALUE will be returned from optixModuleCreateFromPTX.

The same set of bound values do not need to be used for all modules in a pipeline, but overlapping values between modules must have the same value. OPTIX\_ERROR\_INVALID\_VALUE will be returned from optixPipelineCreate otherwise.

See Also

[OptixModuleCompileOptions](#)

### 3.12.3.43 typedef struct OptixModuleCompileOptions OptixModuleCompileOptions

Compilation options for module.

See Also

[optixModuleCreateFromPTX\(\)](#)

### 3.12.3.44 typedef enum OptixModuleCompileState OptixModuleCompileState

Module compilation state.

See Also

[optixModuleGetCompilationState\(\)](#), [optixModuleCreateFromPTXWithTasks\(\)](#)

### 3.12.3.45 typedef enum OptixMotionFlags OptixMotionFlags

Enum to specify motion flags.

See Also

[OptixMotionOptions::flags.](#)

### 3.12.3.46 typedef struct OptixMotionOptions OptixMotionOptions

Motion options.

See Also

[OptixAccelBuildOptions::motionOptions](#), [OptixMatrixMotionTransform::motionOptions](#),  
[OptixSRTMotionTransform::motionOptions](#)

### 3.12.3.47 typedef enum OptixPayloadSemantics OptixPayloadSemantics

Semantic flags for a single payload word.

Used to specify the semantics of a payload word per shader type. "read": Shader of this type may read the payload word. "write": Shader of this type may write the payload word.

"trace\_caller\_write": Shaders may consume the value of the payload word passed to `optixTrace` by the caller. "trace\_caller\_read": The caller to `optixTrace` may read the payload word after the call to `optixTrace`.

Semantics can be bitwise combined. Combining "read" and "write" is equivalent to specifying "read\_write". A payload needs to be writable by the caller or at least one shader type. A payload needs to be readable by the caller or at least one shader type after a being writable.

#### **3.12.3.48 typedef struct OptixPayloadType OptixPayloadType**

Specifies a single payload type.

#### **3.12.3.49 typedef enum OptixPayloadTypeID OptixPayloadTypeID**

Payload type identifiers.

#### **3.12.3.50 typedef struct OptixPipeline\_t\* OptixPipeline**

Opaque type representing a pipeline.

#### **3.12.3.51 typedef struct OptixPipelineCompileOptions OptixPipelineCompileOptions**

Compilation options for all modules of a pipeline.

Similar to [OptixModuleCompileOptions](#), but these options here need to be equal for all modules of a pipeline.

See Also

[optixModuleCreateFromPTX\(\)](#), [optixPipelineCreate\(\)](#)

#### **3.12.3.52 typedef struct OptixPipelineLinkOptions OptixPipelineLinkOptions**

Link options for a pipeline.

See Also

[optixPipelineCreate\(\)](#)

#### **3.12.3.53 typedef enum OptixPixelFormat OptixPixelFormat**

Pixel formats used by the denoiser.

See Also

[OptixImage2D::format](#)

#### **3.12.3.54 typedef enum OptixPrimitiveType OptixPrimitiveType**

Builtin primitive types.

#### **3.12.3.55 typedef enum OptixPrimitiveTypeFlags OptixPrimitiveTypeFlags**

Builtin flags may be bitwise combined.

See Also

[OptixPipelineCompileOptions::usesPrimitiveTypeFlags](#)

### 3.12.3.56 **typedef struct OptixProgramGroup\_t\* OptixProgramGroup**

Opaque type representing a program group.

### 3.12.3.57 **typedef struct OptixProgramGroupCallables OptixProgramGroupCallables**

Program group representing callables.

Module and entry function name need to be valid for at least one of the two callables.

See Also

[#OptixProgramGroupDesc::callables](#)

### 3.12.3.58 **typedef struct OptixProgramGroupDesc OptixProgramGroupDesc**

Descriptor for program groups.

### 3.12.3.59 **typedef enum OptixProgramGroupFlags OptixProgramGroupFlags**

Flags for program groups.

### 3.12.3.60 **typedef struct OptixProgramGroupHitgroup OptixProgramGroupHitgroup**

Program group representing the hitgroup.

For each of the three program types, module and entry function name might both be nullptr.

See Also

[OptixProgramGroupDesc::hitgroup](#)

### 3.12.3.61 **typedef enum OptixProgramGroupKind OptixProgramGroupKind**

Distinguishes different kinds of program groups.

### 3.12.3.62 **typedef struct OptixProgramGroupOptions OptixProgramGroupOptions**

Program group options.

See Also

[optixProgramGroupCreate\(\)](#)

### 3.12.3.63 **typedef struct OptixProgramGroupSingleModule OptixProgramGroupSingleModule**

Program group representing a single module.

Used for raygen, miss, and exception programs. In case of raygen and exception programs, module and entry function name need to be valid. For miss programs, module and entry function name might both be nullptr.

See Also

[OptixProgramGroupDesc::raygen](#), [OptixProgramGroupDesc::miss](#),  
[OptixProgramGroupDesc::exception](#)

**3.12.3.64 typedef OptixResult( OptixQueryFunctionTable\_t)(int abild, unsigned int numOptions, OptixQueryFunctionTableOptions \*, const void \*\*, void \*functionTable, size\_t sizeOfTable)**

Type of the function [optixQueryFunctionTable\(\)](#)

**3.12.3.65 typedef enum OptixQueryFunctionTableOptions OptixQueryFunctionTableOptions**

Options that can be passed to [optixQueryFunctionTable\(\)](#)

**3.12.3.66 typedef enum OptixRayFlags OptixRayFlags**

Ray flags passed to the device function [optixTrace\(\)](#). These affect the behavior of traversal per invocation.

See Also

[optixTrace\(\)](#)

**3.12.3.67 typedef enum OptixResult OptixResult**

Result codes returned from API functions.

All host side API functions return OptixResult with the exception of [optixGetErrorName](#) and [optixGetErrorString](#). When successful OPTIX\_SUCCESS is returned. All return codes except for OPTIX\_SUCCESS should be assumed to be errors as opposed to a warning.

See Also

[optixGetErrorName\(\)](#), [optixGetErrorString\(\)](#)

**3.12.3.68 typedef struct OptixShaderBindingTable OptixShaderBindingTable**

Describes the shader binding table (SBT)

See Also

[optixLaunch\(\)](#)

**3.12.3.69 typedef struct OptixSRTData OptixSRTData**

Represents an SRT transformation.

An SRT transformation can represent a smooth rotation with fewer motion keys than a matrix transformation. Each motion key is constructed from elements taken from a matrix S, a quaternion R, and a translation T.

The scaling matrix  $S = \begin{bmatrix} sx & a & b & pvx \\ 0 & sy & c & pvy \\ 0 & 0 & sz & pvz \end{bmatrix}$  defines an affine transformation that can include scale,

shear, and a translation. The translation allows to define the pivot point for the subsequent rotation.

The quaternion  $R = [qx, qy, qz, qw]$  describes a rotation with angular component  $qw = \cos(\theta/2)$  and other components  $[qx, qy, qz] = \sin(\theta/2) * [ax, ay, az]$  where the axis  $[ax, ay, az]$  is normalized.

The translation matrix  $T = \begin{bmatrix} 1 & 0 & 0 & tx \\ 0 & 1 & 0 & ty \\ 0 & 0 & 1 & tz \end{bmatrix}$  defines another translation that is applied after the rotation.

Typically, this translation includes the inverse translation from the matrix  $S$  to reverse the translation for the pivot point for  $R$ .

To obtain the effective transformation at time  $t$ , the elements of the components of  $S$ ,  $R$ , and  $T$  will be interpolated linearly. The components are then multiplied to obtain the combined transformation  $C = T * R * S$ . The transformation  $C$  is the effective object-to-world transformations at time  $t$ , and  $C^{-1}$  is the effective world-to-object transformation at time  $t$ .

See Also

[OptixSRTMotionTransform::srtData](#), [optixConvertPointerToTraversableHandle\(\)](#)

### 3.12.3.70 typedef struct OptixSRTMotionTransform OptixSRTMotionTransform

Represents an SRT motion transformation.

The device address of instances of this type must be a multiple of `OPTIX_TRANSFORM_BYTE_ALIGNMENT`.

This struct, as defined here, handles only  $N=2$  motion keys due to the fixed array length of its `srtData` member. The following example shows how to create instances for an arbitrary number  $N$  of motion keys:

```
OptixSRTData srtData[N];
... // setup srtData

size_t transformSizeInBytes = sizeof( OptixSRTMotionTransform ) + ( N-2 ) * sizeof(
    OptixSRTData );
OptixSRTMotionTransform* srtMotionTransform = (
    OptixSRTMotionTransform*) malloc( transformSizeInBytes );
memset( srtMotionTransform, 0, transformSizeInBytes );

... // setup other members of srtMotionTransform
srtMotionTransform->motionOptions.numKeys = N;
memcpy( srtMotionTransform->srtData, srtData, N * sizeof( OptixSRTData ) );

... // copy srtMotionTransform to device memory
free( srtMotionTransform )
```



See Also

[optixConvertPointerToTraversableHandle\(\)](#)

### 3.12.3.71 **typedef struct OptixStackSizes OptixStackSizes**

Describes the stack size requirements of a program group.

See Also

[optixProgramGroupGetStackSize\(\)](#)

### 3.12.3.72 **typedef struct OptixStaticTransform OptixStaticTransform**

Static transform.

The device address of instances of this type must be a multiple of `OPTIX_TRANSFORM_BYTE_ALIGNMENT`.

See Also

[optixConvertPointerToTraversableHandle\(\)](#)

### 3.12.3.73 **typedef struct OptixTask\_t\* OptixTask**

Opaque type representing a work task.

### 3.12.3.74 **typedef enum OptixTransformFormat OptixTransformFormat**

Format of transform used in [OptixBuildInputTriangleArray::transformFormat](#).

### 3.12.3.75 **typedef enum OptixTransformType OptixTransformType**

Transform.

`OptixTransformType` is used by the device function [optixGetTransformTypeFromHandle\(\)](#) to determine the type of the `OptixTraversableHandle` returned from [optixGetTransformListHandle\(\)](#).

### 3.12.3.76 **typedef enum OptixTraversableGraphFlags OptixTraversableGraphFlags**

Specifies the set of valid traversable graphs that may be passed to invocation of [optixTrace\(\)](#). Flags may be bitwise combined.

### 3.12.3.77 **typedef unsigned long long OptixTraversableHandle**

Traversable handle.

### 3.12.3.78 **typedef enum OptixTraversableType OptixTraversableType**

Traversable Handles.

See Also

[optixConvertPointerToTraversableHandle\(\)](#)

### 3.12.3.79 typedef enum OptixVertexFormat OptixVertexFormat

Format of vertices used in `OptixBuildInputTriangleArray::vertexFormat`.

### 3.12.3.80 typedef unsigned int OptixVisibilityMask

Visibility mask.

## 3.12.4 Enumeration Type Documentation

### 3.12.4.1 enum OptixAccelPropertyType

Properties which can be emitted during acceleration structure build.

See Also

`OptixAccelEmitDesc::type`.

Enumerator

**OPTIX\_PROPERTY\_TYPE\_COMPACTED\_SIZE** Size of a compacted acceleration structure.  
The device pointer points to a uint64.

**OPTIX\_PROPERTY\_TYPE\_AABBS** `OptixAabb * numMotionSteps`.

### 3.12.4.2 enum OptixBuildFlags

Builder Options.

Used for `OptixAccelBuildOptions::buildFlags`. Can be or'ed together.

Enumerator

**OPTIX\_BUILD\_FLAG\_NONE** No special flags set.

**OPTIX\_BUILD\_FLAG\_ALLOW\_UPDATE** Allow updating the build with new vertex positions  
with subsequent calls to `optixAccelBuild`.

**OPTIX\_BUILD\_FLAG\_ALLOW\_COMPACTION**

**OPTIX\_BUILD\_FLAG\_PREFER\_FAST\_TRACE**

**OPTIX\_BUILD\_FLAG\_PREFER\_FAST\_BUILD**

**OPTIX\_BUILD\_FLAG\_ALLOW\_RANDOM\_VERTEX\_ACCESS** Allow random access to build  
input vertices See `optixGetTriangleVertexData` `optixGetLinearCurveVertexData`  
`optixGetQuadraticBSplineVertexData` `optixGetCubicBSplineVertexData`  
`optixGetCatmullRomVertexData`.

**OPTIX\_BUILD\_FLAG\_ALLOW\_RANDOM\_INSTANCE\_ACCESS** Allow random access to  
instances See `optixGetInstanceTraversableFromIAS`.

### 3.12.4.3 enum OptixBuildInputType

Enum to distinguish the different build input types.

See Also

[OptixBuildInput::type](#)

Enumerator

**OPTIX\_BUILD\_INPUT\_TYPE\_TRIANGLES** Triangle inputs.

See Also

[OptixBuildInputTriangleArray](#)

**OPTIX\_BUILD\_INPUT\_TYPE\_CUSTOM\_PRIMITIVES** Custom primitive inputs.

See Also

[OptixBuildInputCustomPrimitiveArray](#)

**OPTIX\_BUILD\_INPUT\_TYPE\_INSTANCES** Instance inputs.

See Also

[OptixBuildInputInstanceArray](#)

**OPTIX\_BUILD\_INPUT\_TYPE\_INSTANCE\_POINTERS** Instance pointer inputs.

See Also

[OptixBuildInputInstanceArray](#)

**OPTIX\_BUILD\_INPUT\_TYPE\_CURVES** Curve inputs.

See Also

[OptixBuildInputCurveArray](#)

#### 3.12.4.4 enum OptixBuildOperation

Enum to specify the acceleration build operation.

Used in [OptixAccelBuildOptions](#), which is then passed to [optixAccelBuild](#) and [optixAccelComputeMemoryUsage](#), this enum indicates whether to do a build or an update of the acceleration structure.

Acceleration structure updates utilize the same acceleration structure, but with updated bounds. Updates are typically much faster than builds, however, large perturbations can degrade the quality of the acceleration structure.

See Also

[optixAccelComputeMemoryUsage\(\)](#), [optixAccelBuild\(\)](#), [OptixAccelBuildOptions](#)

Enumerator

**OPTIX\_BUILD\_OPERATION\_BUILD** Perform a full build operation.

**OPTIX\_BUILD\_OPERATION\_UPDATE** Perform an update using new bounds.

#### 3.12.4.5 enum OptixCompileDebugLevel

Debug levels.

See Also

[OptixModuleCompileOptions::debugLevel](#)

Enumerator

**OPTIX\_COMPILE\_DEBUG\_LEVEL\_DEFAULT** Default currently is minimal.

**OPTIX\_COMPILE\_DEBUG\_LEVEL\_NONE** No debug information.

**OPTIX\_COMPILE\_DEBUG\_LEVEL\_MINIMAL** Generate information that does not impact performance. Note this replaces OPTIX\_COMPILE\_DEBUG\_LEVEL\_LINEINFO.

**OPTIX\_COMPILE\_DEBUG\_LEVEL\_MODERATE** Generate some debug information with slight performance cost.

**OPTIX\_COMPILE\_DEBUG\_LEVEL\_FULL** Generate full debug information.

#### 3.12.4.6 enum OptixCompileOptimizationLevel

Optimization levels.

See Also

[OptixModuleCompileOptions::optLevel](#)

Enumerator

**OPTIX\_COMPILE\_OPTIMIZATION\_DEFAULT** Default is to run all optimizations.

**OPTIX\_COMPILE\_OPTIMIZATION\_LEVEL\_0** No optimizations.

**OPTIX\_COMPILE\_OPTIMIZATION\_LEVEL\_1** Some optimizations.

**OPTIX\_COMPILE\_OPTIMIZATION\_LEVEL\_2** Most optimizations.

**OPTIX\_COMPILE\_OPTIMIZATION\_LEVEL\_3** All optimizations.

#### 3.12.4.7 enum OptixCurveEndcapFlags

Curve end cap types, for non-linear curves.

Enumerator

**OPTIX\_CURVE\_ENDCAP\_DEFAULT** Default end caps. Round end caps for linear, no end caps for quadratic/cubic.

**OPTIX\_CURVE\_ENDCAP\_ON** Flat end caps at both ends of quadratic/cubic curve segments. Not valid for linear.

#### 3.12.4.8 enum OptixDenoiserModelKind

Model kind used by the denoiser.

See Also

[optixDenoiserCreate](#)

Enumerator

**OPTIX\_DENOISER\_MODEL\_KIND\_LDR** Use the built-in model appropriate for low dynamic range input.

**OPTIX\_DENOISER\_MODEL\_KIND\_HDR** Use the built-in model appropriate for high dynamic range input.

**OPTIX\_DENOISER\_MODEL\_KIND\_AOV** Use the built-in model appropriate for high dynamic range input and support for AOVs.

**OPTIX\_DENOISER\_MODEL\_KIND\_TEMPORAL** Use the built-in model appropriate for high dynamic range input, temporally stable.

**OPTIX\_DENOISER\_MODEL\_KIND\_TEMPORAL\_AOV** Use the built-in model appropriate for high dynamic range input and support for AOVs, temporally stable.

#### 3.12.4.9 enum OptixDeviceContextValidationMode

Validation mode settings.

When enabled, certain device code utilities will be enabled to provide as good debug and error checking facilities as possible.

See Also

[optixDeviceContextCreate\(\)](#)

Enumerator

**OPTIX\_DEVICE\_CONTEXT\_VALIDATION\_MODE\_OFF**

**OPTIX\_DEVICE\_CONTEXT\_VALIDATION\_MODE\_ALL**

#### 3.12.4.10 enum OptixDeviceProperty

Parameters used for [optixDeviceContextGetProperty\(\)](#)

See Also

[optixDeviceContextGetProperty\(\)](#)

Enumerator

**OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_TRACE\_DEPTH** Maximum value for [OptixPipelineLinkOptions::maxTraceDepth](#). sizeof( unsigned int )

**OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_TRAVERSABLE\_GRAPH\_DEPTH** Maximum value to pass into [optixPipelineSetStackSize](#) for parameter [maxTraversableGraphDepth.v](#) sizeof( unsigned int )

**OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_PRIMITIVES\_PER\_GAS** The maximum number of primitives (over all build inputs) as input to a single Geometry Acceleration Structure (GAS). sizeof( unsigned int )

**OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_INSTANCES\_PER\_IAS** The maximum number of instances (over all build inputs) as input to a single Instance Acceleration Structure (IAS). sizeof( unsigned int )

**OPTIX\_DEVICE\_PROPERTY\_RTCORE\_VERSION** The RT core version supported by the device (0 for no support, 10 for version 1.0). sizeof( unsigned int )

**OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_INSTANCE\_ID** The maximum value for [OptixInstance::instanceId](#). sizeof( unsigned int )

**OPTIX\_DEVICE\_PROPERTY\_LIMIT\_NUM\_BITS\_INSTANCE\_VISIBILITY\_MASK** The number of bits available for the [OptixInstance::visibilityMask](#). Higher bits must be set to zero. sizeof( unsigned int )

**OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_SBT\_RECORDS\_PER\_GAS** The maximum number of instances that can be added to a single Instance Acceleration Structure (IAS). sizeof( unsigned int )

**OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_SBT\_OFFSET** The maximum value for [OptixInstance::sbtOffset](#). sizeof( unsigned int )

#### 3.12.4.11 enum OptixExceptionCodes

The following values are used to indicate which exception was thrown.

Enumerator

**OPTIX\_EXCEPTION\_CODE\_STACK\_OVERFLOW** Stack overflow of the continuation stack. no exception details.

**OPTIX\_EXCEPTION\_CODE\_TRACE\_DEPTH\_EXCEEDED** The trace depth is exceeded. no exception details.

**OPTIX\_EXCEPTION\_CODE\_TRAVERSAL\_DEPTH\_EXCEEDED** The traversal depth is exceeded. Exception details: [optixGetTransformListSize\(\)](#) [optixGetTransformListHandle\(\)](#)

**OPTIX\_EXCEPTION\_CODE\_TRAVERSAL\_INVALID\_TRAVERSABLE** Traversal encountered an invalid traversable type. Exception details: [optixGetTransformListSize\(\)](#) [optixGetTransformListHandle\(\)](#) [optixGetExceptionInvalidTraversable\(\)](#)

**OPTIX\_EXCEPTION\_CODE\_TRAVERSAL\_INVALID\_MISS\_SBT** The miss SBT record index is out of bounds A miss SBT record index is valid within the range [0, [OptixShaderBindingTable::missRecordCount](#)) (See [optixLaunch](#)) Exception details: [optixGetExceptionInvalidSbtOffset\(\)](#)

**OPTIX\_EXCEPTION\_CODE\_TRAVERSAL\_INVALID\_HIT\_SBT** The traversal hit SBT record index out of bounds. A traversal hit SBT record index is valid within the range [0, [OptixShaderBindingTable::hitgroupRecordCount](#)) (See [optixLaunch](#)) The following formula relates the sbt-geometry-acceleration-structure-index (See [optixGetSbtGASIndex\(\)](#)), sbt-stride-from-trace-call and sbt-offset-from-trace-call (See [optixTrace](#))  

$$\text{sbt-index} = \text{sbt-instance-offset} + (\text{sbt-geometry-acceleration-structure-index} * \text{sbt-stride-from-trace-call}) + \text{sbt-offset-from-trace-call}$$
Exception details: [optixGetTransformListSize\(\)](#) [optixGetTransformListHandle\(\)](#) [optixGetExceptionInvalidSbtOffset\(\)](#) [optixGetSbtGASIndex\(\)](#)

**OPTIX\_EXCEPTION\_CODE\_UNSUPPORTED\_PRIMITIVE\_TYPE** The shader encountered an unsupported primitive type (See [OptixPipelineCompileOptions::usesPrimitiveTypeFlags](#)). no exception details.

**OPTIX\_EXCEPTION\_CODE\_INVALID\_RAY** The shader encountered a call to `optixTrace` with at least one of the float arguments being inf or nan. Exception details:

`optixGetExceptionInvalidRay()`

**OPTIX\_EXCEPTION\_CODE\_CALLABLE\_PARAMETER\_MISMATCH** The shader encountered a call to either `optixDirectCall` or `optixCallableCall` where the argument count does not match the parameter count of the callable program which is called. Exception details:

`optixGetExceptionParameterMismatch()`

**OPTIX\_EXCEPTION\_CODE\_BUILTIN\_IS\_MISMATCH** The invoked builtin IS does not match the current GAS.

**OPTIX\_EXCEPTION\_CODE\_CALLABLE\_INVALID\_SBT** Tried to call a callable program using an SBT offset that is larger than the number of passed in callable SBT records. Exception details:

`optixGetExceptionInvalidSbtOffset()`

**OPTIX\_EXCEPTION\_CODE\_CALLABLE\_NO\_DC\_SBT\_RECORD** Tried to call a direct callable using an SBT offset of a record that was built from a program group that did not include a direct callable.

**OPTIX\_EXCEPTION\_CODE\_CALLABLE\_NO\_CC\_SBT\_RECORD** Tried to call a continuation callable using an SBT offset of a record that was built from a program group that did not include a continuation callable.

**OPTIX\_EXCEPTION\_CODE\_UNSUPPORTED\_SINGLE\_LEVEL\_GAS** Tried to directly traverse a single gas while single gas traversable graphs are not enabled (see `OptixTraversableGraphFlags::OPTIX_TRAVERSABLE_GRAPH_FLAG_ALLOW_SINGLE_GAS`). Exception details:

`optixGetTransformListSize()` `optixGetTransformListHandle()`  
`optixGetExceptionInvalidTraversable()`

**OPTIX\_EXCEPTION\_CODE\_INVALID\_VALUE\_ARGUMENT\_0** argument passed to an optix call is not within an acceptable range of values.

**OPTIX\_EXCEPTION\_CODE\_INVALID\_VALUE\_ARGUMENT\_1**

**OPTIX\_EXCEPTION\_CODE\_INVALID\_VALUE\_ARGUMENT\_2**

**OPTIX\_EXCEPTION\_CODE\_UNSUPPORTED\_DATA\_ACCESS** Tried to access data on an AS without random data access support (See `OptixBuildFlags`).

**OPTIX\_EXCEPTION\_CODE\_PAYLOAD\_TYPE\_MISMATCH** The program payload type doesn't match the trace payload type.

#### 3.12.4.12 enum OptixExceptionFlags

Exception flags.

See Also

`OptixPipelineCompileOptions::exceptionFlags`, `OptixExceptionCodes`

Enumerator

**OPTIX\_EXCEPTION\_FLAG\_NONE** No exception are enabled.

**OPTIX\_EXCEPTION\_FLAG\_STACK\_OVERFLOW** Enables exceptions check related to the continuation stack.

**OPTIX\_EXCEPTION\_FLAG\_TRACE\_DEPTH** Enables exceptions check related to trace depth.

**OPTIX\_EXCEPTION\_FLAG\_USER** Enables user exceptions via [optixThrowException\(\)](#). This flag must be specified for all modules in a pipeline if any module calls [optixThrowException\(\)](#).

**OPTIX\_EXCEPTION\_FLAG\_DEBUG** Enables various exceptions check related to traversal.

#### 3.12.4.13 enum OptixGeometryFlags

Flags used by [OptixBuildInputTriangleArray::flags](#) and [#OptixBuildInput::flag](#) and [OptixBuildInputCustomPrimitiveArray::flags](#).

Enumerator

**OPTIX\_GEOMETRY\_FLAG\_NONE** No flags set.

**OPTIX\_GEOMETRY\_FLAG\_DISABLE\_ANYHIT** Disables the invocation of the anyhit program. Can be overridden by [OPTIX\\_INSTANCE\\_FLAG\\_ENFORCE\\_ANYHIT](#) and [OPTIX\\_RAY\\_FLAG\\_ENFORCE\\_ANYHIT](#).

**OPTIX\_GEOMETRY\_FLAG\_REQUIRE\_SINGLE\_ANYHIT\_CALL** If set, an intersection with the primitive will trigger one and only one invocation of the anyhit program. Otherwise, the anyhit program may be invoked more than once.

#### 3.12.4.14 enum OptixHitKind

Legacy type: A subset of the hit kinds for built-in primitive intersections. It is preferred to use [optixGetPrimitiveType\(\)](#), together with [optixIsFrontFaceHit\(\)](#) or [optixIsBackFaceHit\(\)](#).

See Also

[optixGetHitKind\(\)](#)

Enumerator

**OPTIX\_HIT\_KIND\_TRIANGLE\_FRONT\_FACE** Ray hit the triangle on the front face.

**OPTIX\_HIT\_KIND\_TRIANGLE\_BACK\_FACE** Ray hit the triangle on the back face.

#### 3.12.4.15 enum OptixIndicesFormat

Format of indices used in [OptixBuildInputTriangleArray::indexFormat](#).

Enumerator

**OPTIX\_INDICES\_FORMAT\_NONE** No indices, this format must only be used in combination with triangle soups, i.e., [numIndexTriplets](#) must be zero.

**OPTIX\_INDICES\_FORMAT\_UNSIGNED\_SHORT3** Three shorts.

**OPTIX\_INDICES\_FORMAT\_UNSIGNED\_INT3** Three ints.

#### 3.12.4.16 enum OptixInstanceFlags

Flags set on the [OptixInstance::flags](#).

These can be or'ed together to combine multiple flags.



Enumerator

**OPTIX\_INSTANCE\_FLAG\_NONE** No special flag set.

**OPTIX\_INSTANCE\_FLAG\_DISABLE\_TRIANGLE\_FACE\_CULLING** Prevent triangles from getting culled due to their orientation. Effectively ignores ray flags **OPTIX\_RAY\_FLAG\_CULL\_BACK\_FACING\_TRIANGLES** and **OPTIX\_RAY\_FLAG\_CULL\_FRONT\_FACING\_TRIANGLES**.

**OPTIX\_INSTANCE\_FLAG\_FLIP\_TRIANGLE\_FACING** Flip triangle orientation. This affects front/backface culling as well as the reported face in case of a hit.

**OPTIX\_INSTANCE\_FLAG\_DISABLE\_ANYHIT** Disable anyhit programs for all geometries of the instance. Can be overridden by **OPTIX\_RAY\_FLAG\_ENFORCE\_ANYHIT**. This flag is mutually exclusive with **OPTIX\_INSTANCE\_FLAG\_ENFORCE\_ANYHIT**.

**OPTIX\_INSTANCE\_FLAG\_ENFORCE\_ANYHIT** Enables anyhit programs for all geometries of the instance. Overrides **OPTIX\_GEOMETRY\_FLAG\_DISABLE\_ANYHIT**. Can be overridden by **OPTIX\_RAY\_FLAG\_DISABLE\_ANYHIT**. This flag is mutually exclusive with **OPTIX\_INSTANCE\_FLAG\_DISABLE\_ANYHIT**.

#### 3.12.4.17 enum OptixModuleCompileState

Module compilation state.

See Also

[optixModuleGetCompilationState\(\)](#), [optixModuleCreateFromPTXWithTasks\(\)](#)

Enumerator

**OPTIX\_MODULE\_COMPILE\_STATE\_NOT\_STARTED** No OptixTask objects have started.

**OPTIX\_MODULE\_COMPILE\_STATE\_STARTED** Started, but not all OptixTask objects have completed. No detected failures.

**OPTIX\_MODULE\_COMPILE\_STATE\_IMPENDING\_FAILURE** Not all OptixTask objects have completed, but at least one has failed.

**OPTIX\_MODULE\_COMPILE\_STATE\_FAILED** All OptixTask objects have completed, and at least one has failed.

**OPTIX\_MODULE\_COMPILE\_STATE\_COMPLETED** All OptixTask objects have completed. The OptixModule is ready to be used.

#### 3.12.4.18 enum OptixMotionFlags

Enum to specify motion flags.

See Also

[OptixMotionOptions::flags](#).

Enumerator

**OPTIX\_MOTION\_FLAG\_NONE**

**OPTIX\_MOTION\_FLAG\_START\_VANISH**

**OPTIX\_MOTION\_FLAG\_END\_VANISH**

### 3.12.4.19 enum OptixPayloadSemantics

Semantic flags for a single payload word.

Used to specify the semantics of a payload word per shader type. "read": Shader of this type may read the payload word. "write": Shader of this type may write the payload word.

"trace\_caller\_write": Shaders may consume the value of the payload word passed to optixTrace by the caller. "trace\_caller\_read": The caller to optixTrace may read the payload word after the call to optixTrace.

Semantics can be bitwise combined. Combining "read" and "write" is equivalent to specifying "read\_write". A payload needs to be writable by the caller or at least one shader type. A payload needs to be readable by the caller or at least one shader type after a being writable.

Enumerator

```

OPTIX_PAYLOAD_SEMANTICS_TRACE_CALLER_NONE
OPTIX_PAYLOAD_SEMANTICS_TRACE_CALLER_READ
OPTIX_PAYLOAD_SEMANTICS_TRACE_CALLER_WRITE
OPTIX_PAYLOAD_SEMANTICS_TRACE_CALLER_READ_WRITE
OPTIX_PAYLOAD_SEMANTICS_CH_NONE
OPTIX_PAYLOAD_SEMANTICS_CH_READ
OPTIX_PAYLOAD_SEMANTICS_CH_WRITE
OPTIX_PAYLOAD_SEMANTICS_CH_READ_WRITE
OPTIX_PAYLOAD_SEMANTICS_MS_NONE
OPTIX_PAYLOAD_SEMANTICS_MS_READ
OPTIX_PAYLOAD_SEMANTICS_MS_WRITE
OPTIX_PAYLOAD_SEMANTICS_MS_READ_WRITE
OPTIX_PAYLOAD_SEMANTICS_AH_NONE
OPTIX_PAYLOAD_SEMANTICS_AH_READ
OPTIX_PAYLOAD_SEMANTICS_AH_WRITE
OPTIX_PAYLOAD_SEMANTICS_AH_READ_WRITE
OPTIX_PAYLOAD_SEMANTICS_IS_NONE
OPTIX_PAYLOAD_SEMANTICS_IS_READ
OPTIX_PAYLOAD_SEMANTICS_IS_WRITE
OPTIX_PAYLOAD_SEMANTICS_IS_READ_WRITE

```

### 3.12.4.20 enum OptixPayloadTypeID

Payload type identifiers.

Enumerator

```

OPTIX_PAYLOAD_TYPE_DEFAULT
OPTIX_PAYLOAD_TYPE_ID_0
OPTIX_PAYLOAD_TYPE_ID_1

```

***OPTIX\_PAYLOAD\_TYPE\_ID\_2***  
***OPTIX\_PAYLOAD\_TYPE\_ID\_3***  
***OPTIX\_PAYLOAD\_TYPE\_ID\_4***  
***OPTIX\_PAYLOAD\_TYPE\_ID\_5***  
***OPTIX\_PAYLOAD\_TYPE\_ID\_6***  
***OPTIX\_PAYLOAD\_TYPE\_ID\_7***

#### 3.12.4.21 enum OptixPixelFormat

Pixel formats used by the denoiser.

See Also

[OptixImage2D::format](#)

Enumerator

***OPTIX\_PIXEL\_FORMAT\_HALF2*** two halves, XY  
***OPTIX\_PIXEL\_FORMAT\_HALF3*** three halves, RGB  
***OPTIX\_PIXEL\_FORMAT\_HALF4*** four halves, RGBA  
***OPTIX\_PIXEL\_FORMAT\_FLOAT2*** two floats, XY  
***OPTIX\_PIXEL\_FORMAT\_FLOAT3*** three floats, RGB  
***OPTIX\_PIXEL\_FORMAT\_FLOAT4*** four floats, RGBA  
***OPTIX\_PIXEL\_FORMAT\_UCHAR3*** three unsigned chars, RGB  
***OPTIX\_PIXEL\_FORMAT\_UCHAR4*** four unsigned chars, RGBA

#### 3.12.4.22 enum OptixPrimitiveType

Builtin primitive types.

Enumerator

***OPTIX\_PRIMITIVE\_TYPE\_CUSTOM*** Custom primitive.  
***OPTIX\_PRIMITIVE\_TYPE\_ROUND\_QUADRATIC\_BSPLINE*** B-spline curve of degree 2 with circular cross-section.  
***OPTIX\_PRIMITIVE\_TYPE\_ROUND\_CUBIC\_BSPLINE*** B-spline curve of degree 3 with circular cross-section.  
***OPTIX\_PRIMITIVE\_TYPE\_ROUND\_LINEAR*** Piecewise linear curve with circular cross-section.  
***OPTIX\_PRIMITIVE\_TYPE\_ROUND\_CATMULLROM*** CatmullRom curve with circular cross-section.  
***OPTIX\_PRIMITIVE\_TYPE\_TRIANGLE*** Triangle.

### 3.12.4.23 enum OptixPrimitiveTypeFlags

Builtin flags may be bitwise combined.

See Also

[OptixPipelineCompileOptions::usesPrimitiveTypeFlags](#)

Enumerator

**OPTIX\_PRIMITIVE\_TYPE\_FLAGS\_CUSTOM** Custom primitive.

**OPTIX\_PRIMITIVE\_TYPE\_FLAGS\_ROUND\_QUADRATIC\_BSPLINE** B-spline curve of degree 2 with circular cross-section.

**OPTIX\_PRIMITIVE\_TYPE\_FLAGS\_ROUND\_CUBIC\_BSPLINE** B-spline curve of degree 3 with circular cross-section.

**OPTIX\_PRIMITIVE\_TYPE\_FLAGS\_ROUND\_LINEAR** Piecewise linear curve with circular cross-section.

**OPTIX\_PRIMITIVE\_TYPE\_FLAGS\_ROUND\_CATMULLROM** CatmullRom curve with circular cross-section.

**OPTIX\_PRIMITIVE\_TYPE\_FLAGS\_TRIANGLE** Triangle.

### 3.12.4.24 enum OptixProgramGroupFlags

Flags for program groups.

Enumerator

**OPTIX\_PROGRAM\_GROUP\_FLAGS\_NONE** Currently there are no flags.

### 3.12.4.25 enum OptixProgramGroupKind

Distinguishes different kinds of program groups.

Enumerator

**OPTIX\_PROGRAM\_GROUP\_KIND\_RAYGEN** Program group containing a raygen (RG) program.

See Also

[OptixProgramGroupSingleModule](#), [OptixProgramGroupDesc::raygen](#)

**OPTIX\_PROGRAM\_GROUP\_KIND\_MISS** Program group containing a miss (MS) program.

See Also

[OptixProgramGroupSingleModule](#), [OptixProgramGroupDesc::miss](#)

**OPTIX\_PROGRAM\_GROUP\_KIND\_EXCEPTION** Program group containing an exception (EX) program.

See Also

[OptixProgramGroupHitgroup](#), [OptixProgramGroupDesc::exception](#)

**OPTIX\_PROGRAM\_GROUP\_KIND\_HITGROUP** Program group containing an intersection (IS), any hit (AH), and/or closest hit (CH) program.

See Also

[OptixProgramGroupSingleModule](#), [OptixProgramGroupDesc::hitgroup](#)

**OPTIX\_PROGRAM\_GROUP\_KIND\_CALLABLES** Program group containing a direct (DC) or continuation (CC) callable program.

See Also

[OptixProgramGroupCallables](#), [OptixProgramGroupDesc::callables](#)

#### 3.12.4.26 enum OptixQueryFunctionTableOptions

Options that can be passed to [optixQueryFunctionTable\(\)](#)

Enumerator

**OPTIX\_QUERY\_FUNCTION\_TABLE\_OPTION\_DUMMY** Placeholder (there are no options yet)

#### 3.12.4.27 enum OptixRayFlags

Ray flags passed to the device function [optixTrace\(\)](#). These affect the behavior of traversal per invocation.

See Also

[optixTrace\(\)](#)

Enumerator

**OPTIX\_RAY\_FLAG\_NONE** No change from the behavior configured for the individual AS.

**OPTIX\_RAY\_FLAG\_DISABLE\_ANYHIT** Disables anyhit programs for the ray. Overrides [OPTIX\\_INSTANCE\\_FLAG\\_ENFORCE\\_ANYHIT](#). This flag is mutually exclusive with [OPTIX\\_RAY\\_FLAG\\_ENFORCE\\_ANYHIT](#), [OPTIX\\_RAY\\_FLAG\\_CULL\\_DISABLED\\_ANYHIT](#), [OPTIX\\_RAY\\_FLAG\\_CULL\\_ENFORCED\\_ANYHIT](#).

**OPTIX\_RAY\_FLAG\_ENFORCE\_ANYHIT** Forces anyhit program execution for the ray. Overrides [OPTIX\\_GEOMETRY\\_FLAG\\_DISABLE\\_ANYHIT](#) as well as [OPTIX\\_INSTANCE\\_FLAG\\_DISABLE\\_ANYHIT](#). This flag is mutually exclusive with [OPTIX\\_RAY\\_FLAG\\_DISABLE\\_ANYHIT](#), [OPTIX\\_RAY\\_FLAG\\_CULL\\_DISABLED\\_ANYHIT](#), [OPTIX\\_RAY\\_FLAG\\_CULL\\_ENFORCED\\_ANYHIT](#).

**OPTIX\_RAY\_FLAG\_TERMINATE\_ON\_FIRST\_HIT** Terminates the ray after the first hit and executes the closesthit program of that hit.

**OPTIX\_RAY\_FLAG\_DISABLE\_CLOSESTHIT** Disables closesthit programs for the ray, but still executes miss program in case of a miss.

**OPTIX\_RAY\_FLAG\_CULL\_BACK\_FACING\_TRIANGLES** Do not intersect triangle back faces (respects a possible face change due to instance flag [OPTIX\\_INSTANCE\\_FLAG\\_FLIP\\_TRIANGLE\\_FACING](#)). This flag is mutually exclusive with [OPTIX\\_RAY\\_FLAG\\_CULL\\_FRONT\\_FACING\\_TRIANGLES](#).

**OPTIX\_RAY\_FLAG\_CULL\_FRONT\_FACING\_TRIANGLES** Do not intersect triangle front faces (respects a possible face change due to instance flag [OPTIX\\_INSTANCE\\_FLAG\\_FLIP\\_TRIANGLE\\_FACING](#)). This flag is mutually exclusive with [OPTIX\\_RAY\\_FLAG\\_CULL\\_BACK\\_FACING\\_TRIANGLES](#).

**OPTIX\_RAY\_FLAG\_CULL\_DISABLED\_ANYHIT** Do not intersect geometry which disables anyhit programs (due to setting geometry flag OPTIX\_GEOMETRY\_FLAG\_DISABLE\_ANYHIT or instance flag OPTIX\_INSTANCE\_FLAG\_DISABLE\_ANYHIT). This flag is mutually exclusive with OPTIX\_RAY\_FLAG\_CULL\_ENFORCED\_ANYHIT, OPTIX\_RAY\_FLAG\_ENFORCE\_ANYHIT, OPTIX\_RAY\_FLAG\_DISABLE\_ANYHIT.

**OPTIX\_RAY\_FLAG\_CULL\_ENFORCED\_ANYHIT** Do not intersect geometry which have an enabled anyhit program (due to not setting geometry flag OPTIX\_GEOMETRY\_FLAG\_DISABLE\_ANYHIT or setting instance flag OPTIX\_INSTANCE\_FLAG\_ENFORCE\_ANYHIT). This flag is mutually exclusive with OPTIX\_RAY\_FLAG\_CULL\_DISABLED\_ANYHIT, OPTIX\_RAY\_FLAG\_ENFORCE\_ANYHIT, OPTIX\_RAY\_FLAG\_DISABLE\_ANYHIT.

### 3.12.4.28 enum OptixResult

Result codes returned from API functions.

All host side API functions return OptixResult with the exception of optixGetErrorName and optixGetErrorString. When successful OPTIX\_SUCCESS is returned. All return codes except for OPTIX\_SUCCESS should be assumed to be errors as opposed to a warning.

See Also

[optixGetErrorName\(\)](#), [optixGetErrorString\(\)](#)

Enumerator

**OPTIX\_SUCCESS**  
**OPTIX\_ERROR\_INVALID\_VALUE**  
**OPTIX\_ERROR\_HOST\_OUT\_OF\_MEMORY**  
**OPTIX\_ERROR\_INVALID\_OPERATION**  
**OPTIX\_ERROR\_FILE\_IO\_ERROR**  
**OPTIX\_ERROR\_INVALID\_FILE\_FORMAT**  
**OPTIX\_ERROR\_DISK\_CACHE\_INVALID\_PATH**  
**OPTIX\_ERROR\_DISK\_CACHE\_PERMISSION\_ERROR**  
**OPTIX\_ERROR\_DISK\_CACHE\_DATABASE\_ERROR**  
**OPTIX\_ERROR\_DISK\_CACHE\_INVALID\_DATA**  
**OPTIX\_ERROR\_LAUNCH\_FAILURE**  
**OPTIX\_ERROR\_INVALID\_DEVICE\_CONTEXT**  
**OPTIX\_ERROR\_CUDA\_NOT\_INITIALIZED**  
**OPTIX\_ERROR\_VALIDATION\_FAILURE**  
**OPTIX\_ERROR\_INVALID\_PTX**  
**OPTIX\_ERROR\_INVALID\_LAUNCH\_PARAMETER**  
**OPTIX\_ERROR\_INVALID\_PAYLOAD\_ACCESS**  
**OPTIX\_ERROR\_INVALID\_ATTRIBUTE\_ACCESS**  
**OPTIX\_ERROR\_INVALID\_FUNCTION\_USE**

**OPTIX\_ERROR\_INVALID\_FUNCTION\_ARGUMENTS**  
**OPTIX\_ERROR\_PIPELINE\_OUT\_OF\_CONSTANT\_MEMORY**  
**OPTIX\_ERROR\_PIPELINE\_LINK\_ERROR**  
**OPTIX\_ERROR\_ILLEGAL\_DURING\_TASK\_EXECUTE**  
**OPTIX\_ERROR\_INTERNAL\_COMPILER\_ERROR**  
**OPTIX\_ERROR\_DENOISER\_MODEL\_NOT\_SET**  
**OPTIX\_ERROR\_DENOISER\_NOT\_INITIALIZED**  
**OPTIX\_ERROR\_ACCEL\_NOT\_COMPATIBLE**  
**OPTIX\_ERROR\_PAYLOAD\_TYPE\_MISMATCH**  
**OPTIX\_ERROR\_PAYLOAD\_TYPE\_RESOLUTION\_FAILED**  
**OPTIX\_ERROR\_PAYLOAD\_TYPE\_ID\_INVALID**  
**OPTIX\_ERROR\_NOT\_SUPPORTED**  
**OPTIX\_ERROR\_UNSUPPORTED\_ABI\_VERSION**  
**OPTIX\_ERROR\_FUNCTION\_TABLE\_SIZE\_MISMATCH**  
**OPTIX\_ERROR\_INVALID\_ENTRY\_FUNCTION\_OPTIONS**  
**OPTIX\_ERROR\_LIBRARY\_NOT\_FOUND**  
**OPTIX\_ERROR\_ENTRY\_SYMBOL\_NOT\_FOUND**  
**OPTIX\_ERROR\_LIBRARY\_UNLOAD\_FAILURE**  
**OPTIX\_ERROR\_CUDA\_ERROR**  
**OPTIX\_ERROR\_INTERNAL\_ERROR**  
**OPTIX\_ERROR\_UNKNOWN**

#### 3.12.4.29 enum OptixTransformFormat

Format of transform used in [OptixBuildInputTriangleArray::transformFormat](#).

Enumerator

**OPTIX\_TRANSFORM\_FORMAT\_NONE** no transform, default for zero initialization  
**OPTIX\_TRANSFORM\_FORMAT\_MATRIX\_FLOAT12** 3x4 row major affine matrix

#### 3.12.4.30 enum OptixTransformType

Transform.

OptixTransformType is used by the device function [optixGetTransformTypeFromHandle\(\)](#) to determine the type of the OptixTraversableHandle returned from [optixGetTransformListHandle\(\)](#).

Enumerator

**OPTIX\_TRANSFORM\_TYPE\_NONE** Not a transformation.

See Also

**OPTIX\_TRANSFORM\_TYPE\_STATIC\_TRANSFORM** [OptixStaticTransform](#)

See Also

**OPTIX\_TRANSFORM\_TYPE\_MATRIX\_MOTION\_TRANSFORM**

[OptixMatrixMotionTransform](#)

See Also

**OPTIX\_TRANSFORM\_TYPE\_SRT\_MOTION\_TRANSFORM**

[OptixSRTMotionTransform](#)

See Also

**OPTIX\_TRANSFORM\_TYPE\_INSTANCE**

[OptixInstance](#)

### 3.12.4.31 enum OptixTraversableGraphFlags

Specifies the set of valid traversable graphs that may be passed to invocation of [optixTrace\(\)](#). Flags may be bitwise combined.

Enumerator

**OPTIX\_TRAVERSABLE\_GRAPH\_FLAG\_ALLOW\_ANY** Used to signal that any traversable graphs is valid. This flag is mutually exclusive with all other flags.

**OPTIX\_TRAVERSABLE\_GRAPH\_FLAG\_ALLOW\_SINGLE\_GAS** Used to signal that a traversable graph of a single Geometry Acceleration Structure (GAS) without any transforms is valid. This flag may be combined with other flags except for **OPTIX\_TRAVERSABLE\_GRAPH\_FLAG\_ALLOW\_ANY**.

**OPTIX\_TRAVERSABLE\_GRAPH\_FLAG\_ALLOW\_SINGLE\_LEVEL\_INSTANCING** Used to signal that a traversable graph of a single Instance Acceleration Structure (IAS) directly connected to Geometry Acceleration Structure (GAS) traversables without transform traversables in between is valid. This flag may be combined with other flags except for **OPTIX\_TRAVERSABLE\_GRAPH\_FLAG\_ALLOW\_ANY**.

### 3.12.4.32 enum OptixTraversableType

Traversable Handles.

See Also

[optixConvertPointerToTraversableHandle\(\)](#)

Enumerator

**OPTIX\_TRAVERSABLE\_TYPE\_STATIC\_TRANSFORM** Static transforms.

See Also

[OptixStaticTransform](#)

**OPTIX\_TRAVERSABLE\_TYPE\_MATRIX\_MOTION\_TRANSFORM** Matrix motion transform.

See Also

[OptixMatrixMotionTransform](#)

**OPTIX\_TRAVERSABLE\_TYPE\_SRT\_MOTION\_TRANSFORM** SRT motion transform.

See Also

[OptixSRTMotionTransform](#)



### 3.12.4.33 enum OptixVertexFormat

Format of vertices used in [OptixBuildInputTriangleArray::vertexFormat](#).

Enumerator

***OPTIX\_VERTEX\_FORMAT\_NONE*** No vertices.

***OPTIX\_VERTEX\_FORMAT\_FLOAT3*** Vertices are represented by three floats.

***OPTIX\_VERTEX\_FORMAT\_FLOAT2*** Vertices are represented by two floats.

***OPTIX\_VERTEX\_FORMAT\_HALF3*** Vertices are represented by three halves.

***OPTIX\_VERTEX\_FORMAT\_HALF2*** Vertices are represented by two halves.

***OPTIX\_VERTEX\_FORMAT\_SNORM16\_3***

***OPTIX\_VERTEX\_FORMAT\_SNORM16\_2***

## 3.13 Function Table

### Classes

- struct [OptixFunctionTable](#)

### Typedefs

- typedef struct [OptixFunctionTable](#) [OptixFunctionTable](#)

### Variables

- [OptixFunctionTable](#) [g\\_optixFunctionTable](#)

#### 3.13.1 Detailed Description

OptiX Function Table.

#### 3.13.2 Typedef Documentation

##### 3.13.2.1 typedef struct [OptixFunctionTable](#) [OptixFunctionTable](#)

The function table containing all API functions.

See [optixInit\(\)](#) and [optixInitWithHandle\(\)](#).

#### 3.13.3 Variable Documentation

##### 3.13.3.1 [OptixFunctionTable](#) [g\\_optixFunctionTable](#)

If the stubs in [optix\\_stubs.h](#) are used, then the function table needs to be defined in exactly one translation unit. This can be achieved by including this header file in that translation unit.

## 3.14 Utilities

### Classes

- struct `OptixUtilDenoiserImageTile`

### Functions

- `OptixResult optixUtilAccumulateStackSizes` (`OptixProgramGroup` programGroup, `OptixStackSizes` \*stackSizes)
- `OptixResult optixUtilComputeStackSizes` (const `OptixStackSizes` \*stackSizes, unsigned int maxTraceDepth, unsigned int maxCCDepth, unsigned int maxDCDepth, unsigned int \*directCallableStackSizeFromTraversal, unsigned int \*directCallableStackSizeFromState, unsigned int \*continuationStackSize)
- `OptixResult optixUtilComputeStackSizesDCSplit` (const `OptixStackSizes` \*stackSizes, unsigned int dssDCFromTraversal, unsigned int dssDCFromState, unsigned int maxTraceDepth, unsigned int maxCCDepth, unsigned int maxDCDepthFromTraversal, unsigned int maxDCDepthFromState, unsigned int \*directCallableStackSizeFromTraversal, unsigned int \*directCallableStackSizeFromState, unsigned int \*continuationStackSize)
- `OptixResult optixUtilComputeStackSizesCssCCTree` (const `OptixStackSizes` \*stackSizes, unsigned int cssCCTree, unsigned int maxTraceDepth, unsigned int maxDCDepth, unsigned int \*directCallableStackSizeFromTraversal, unsigned int \*directCallableStackSizeFromState, unsigned int \*continuationStackSize)
- `OptixResult optixUtilComputeStackSizesSimplePathTracer` (`OptixProgramGroup` programGroupRG, `OptixProgramGroup` programGroupMS1, const `OptixProgramGroup` \*programGroupCH1, unsigned int programGroupCH1Count, `OptixProgramGroup` programGroupMS2, const `OptixProgramGroup` \*programGroupCH2, unsigned int programGroupCH2Count, unsigned int \*directCallableStackSizeFromTraversal, unsigned int \*directCallableStackSizeFromState, unsigned int \*continuationStackSize)
- unsigned int `optixUtilGetPixelStride` (const `OptixImage2D` &image)
- `OptixResult optixUtilDenoiserSplitImage` (const `OptixImage2D` &input, const `OptixImage2D` &output, unsigned int overlapWindowSizeInPixels, unsigned int tileWidth, unsigned int tileHeight, std::vector< `OptixUtilDenoiserImageTile` > &tiles)
- `OptixResult optixUtilDenoiserInvokeTiled` (`OptixDenoiser` denoiser, `CUstream` stream, const `OptixDenoiserParams` \*params, `CUdeviceptr` denoiserState, size\_t denoiserStateSizeInBytes, const `OptixDenoiserGuideLayer` \*guideLayer, const `OptixDenoiserLayer` \*layers, unsigned int numLayers, `CUdeviceptr` scratch, size\_t scratchSizeInBytes, unsigned int overlapWindowSizeInPixels, unsigned int tileWidth, unsigned int tileHeight)
- `OptixResult optixInitWithHandle` (void \*\*handlePtr)
- `OptixResult optixInit` (void)
- `OptixResult optixUninitWithHandle` (void \*handle)

#### 3.14.1 Detailed Description

OptiX Utilities.

### 3.14.2 Function Documentation

#### 3.14.2.1 **OptixResult optixInit (** **void ) [inline]**

Loads the OptiX library and initializes the function table used by the stubs below.

A variant of [optixInitWithHandle\(\)](#) that does not make the handle to the loaded library available.

#### 3.14.2.2 **OptixResult optixInitWithHandle (** **void \*\* *handlePtr* ) [inline]**

Loads the OptiX library and initializes the function table used by the stubs below.

If *handlePtr* is not nullptr, an OS-specific handle to the library will be returned in *\*handlePtr*.

See Also

[optixUninitWithHandle](#)

#### 3.14.2.3 **OptixResult optixUninitWithHandle (** **void \* *handle* ) [inline]**

Unloads the OptiX library and zeros the function table used by the stubs below. Takes the handle returned by [optixInitWithHandle](#). All `OptixDeviceContext` objects must be destroyed before calling this function, or the behavior is undefined.

See Also

[optixInitWithHandle](#)

#### 3.14.2.4 **OptixResult optixUtilAccumulateStackSizes (** **OptixProgramGroup *programGroup*,** **OptixStackSizes \* *stackSizes* ) [inline]**

Retrieves direct and continuation stack sizes for each program in the program group and accumulates the upper bounds in the corresponding output variables based on the semantic type of the program.

Before the first invocation of this function with a given instance of [OptixStackSizes](#), the members of that instance should be set to 0.

#### 3.14.2.5 **OptixResult optixUtilComputeStackSizes (** **const OptixStackSizes \* *stackSizes*,** **unsigned int *maxTraceDepth*,** **unsigned int *maxCCDepth*,** **unsigned int *maxDCDepth*,** **unsigned int \* *directCallableStackSizeFromTraversal*,** **unsigned int \* *directCallableStackSizeFromState*,** **unsigned int \* *continuationStackSize* ) [inline]**

Computes the stack size values needed to configure a pipeline.

See the programming guide for an explanation of the formula.

### Parameters

in	<i>stackSizes</i>	Accumulated stack sizes of all programs in the call graph.
in	<i>maxTraceDepth</i>	Maximum depth of <a href="#">optixTrace()</a> calls.
in	<i>maxCCDepth</i>	Maximum depth of calls trees of continuation callables.
in	<i>maxDCDepth</i>	Maximum depth of calls trees of direct callables.
out	<i>directCallableStackSizeFromTraversal</i>	Direct stack size requirement for direct callables invoked from IS or AH.
out	<i>directCallableStackSizeFromState</i>	Direct stack size requirement for direct callables invoked from RG, MS, or CH.
out	<i>continuationStackSize</i>	Continuation stack requirement.

#### 3.14.2.6 OptixResult optixUtilComputeStackSizesCssCCTree (

```

    const OptixStackSizes * stackSizes,
    unsigned int cssCCTree,
    unsigned int maxTraceDepth,
    unsigned int maxDCDepth,
    unsigned int * directCallableStackSizeFromTraversal,
    unsigned int * directCallableStackSizeFromState,
    unsigned int * continuationStackSize ) [inline]

```

Computes the stack size values needed to configure a pipeline.

This variant is similar to [optixUtilComputeStackSizes\(\)](#), except that it expects the value *cssCCTree* instead of *cssCC* and *maxCCDepth*.

See programming guide for an explanation of the formula.

### Parameters

in	<i>stackSizes</i>	Accumulated stack sizes of all programs in the call graph.
in	<i>cssCCTree</i>	Maximum stack size used by calls trees of continuation callables.
in	<i>maxTraceDepth</i>	Maximum depth of <a href="#">optixTrace()</a> calls.
in	<i>maxDCDepth</i>	Maximum depth of calls trees of direct callables.
out	<i>directCallableStackSizeFromTraversal</i>	Direct stack size requirement for direct callables invoked from IS or AH.
out	<i>directCallableStackSizeFromState</i>	Direct stack size requirement for direct callables invoked from RG, MS, or CH.
out	<i>continuationStackSize</i>	Continuation stack requirement.

### 3.14.2.7 OptixResult optixUtilComputeStackSizesDCSplit (

```

    const OptixStackSizes * stackSizes,
    unsigned int dssDCFromTraversal,
    unsigned int dssDCFromState,
    unsigned int maxTraceDepth,
    unsigned int maxCCDepth,
    unsigned int maxDCDepthFromTraversal,
    unsigned int maxDCDepthFromState,
    unsigned int * directCallableStackSizeFromTraversal,
    unsigned int * directCallableStackSizeFromState,
    unsigned int * continuationStackSize ) [inline]

```

Computes the stack size values needed to configure a pipeline.

This variant is similar to [optixUtilComputeStackSizes\(\)](#), except that it expects the values *dssDC* and *maxDCDepth* split by call site semantic.

See programming guide for an explanation of the formula.

#### Parameters

in	<i>stackSizes</i>	Accumulated stack sizes of all programs in the call graph.
in	<i>dssDCFromTraversal</i>	Accumulated direct stack size of all DC programs invoked from IS or AH.
in	<i>dssDCFromState</i>	Accumulated direct stack size of all DC programs invoked from RG, MS, or CH.
in	<i>maxTraceDepth</i>	Maximum depth of <a href="#">optixTrace()</a> calls.
in	<i>maxCCDepth</i>	Maximum depth of calls trees of continuation callables.
in	<i>maxDCDepthFromTraversal</i>	Maximum depth of calls trees of direct callables invoked from IS or AH.
in	<i>maxDCDepthFromState</i>	Maximum depth of calls trees of direct callables invoked from RG, MS, or CH.
out	<i>directCallableStackSizeFromTraversal</i>	Direct stack size requirement for direct callables invoked from IS or AH.
out	<i>directCallableStackSizeFromState</i>	Direct stack size requirement for direct callables invoked from RG, MS, or CH.
out	<i>continuationStackSize</i>	Continuation stack requirement.

### 3.14.2.8 OptixResult optixUtilComputeStackSizesSimplePathTracer (

```

    OptixProgramGroup programGroupRG,
    OptixProgramGroup programGroupMS1,
    const OptixProgramGroup * programGroupCH1,
    unsigned int programGroupCH1Count,

```

```

OptixProgramGroup programGroupMS2,
const OptixProgramGroup * programGroupCH2,
unsigned int programGroupCH2Count,
unsigned int * directCallableStackSizeFromTraversal,
unsigned int * directCallableStackSizeFromState,
unsigned int * continuationStackSize ) [inline]

```

Computes the stack size values needed to configure a pipeline.

This variant is a specialization of [optixUtilComputeStackSizes\(\)](#) for a simple path tracer with the following assumptions: There are only two ray types, camera rays and shadow rays. There are only RG, MS, and CH programs, and no AH, IS, CC, or DC programs. The camera rays invoke only the miss and closest hit programs MS1 and CH1, respectively. The CH1 program might trace shadow rays, which invoke only the miss and closest hit programs MS2 and CH2, respectively.

For flexibility, we allow for each of CH1 and CH2 not just one single program group, but an array of programs groups, and compute the maximas of the stack size requirements per array.

See programming guide for an explanation of the formula.

#### 3.14.2.9 OptixResult optixUtilDenoiserInvokeTiled (

```

OptixDenoiser denoiser,
CUstream stream,
const OptixDenoiserParams * params,
CUdeviceptr denoiserState,
size_t denoiserStateSizeInBytes,
const OptixDenoiserGuideLayer * guideLayer,
const OptixDenoiserLayer * layers,
unsigned int numLayers,
CUdeviceptr scratch,
size_t scratchSizeInBytes,
unsigned int overlapWindowSizeInPixels,
unsigned int tileWidth,
unsigned int tileHeight ) [inline]

```

Run denoiser on input layers see [optixDenoiserInvoke](#) additional parameters:

Runs the denoiser on the input layers on a single GPU and stream using [optixDenoiserInvoke](#). If the input layers' dimensions are larger than the specified tile size, the image is divided into tiles using [optixUtilDenoiserSplitImage](#), and multiple back-to-back invocations are performed in order to reuse the scratch space. Multiple tiles can be invoked concurrently if [optixUtilDenoiserSplitImage](#) is used directly and multiple scratch allocations for each concurrent invocation are used. The input parameters are the same as [optixDenoiserInvoke](#) except for the addition of the maximum tile size.

#### Parameters

in	<i>denoiser</i>	
in	<i>stream</i>	

### Parameters

in	<i>params</i>	
in	<i>denoiserState</i>	
in	<i>denoiserStateSizeInBytes</i>	
in	<i>guideLayer</i>	
in	<i>layers</i>	
in	<i>numLayers</i>	
in	<i>scratch</i>	
in	<i>scratchSizeInBytes</i>	
in	<i>overlapWindowSizeInPixels</i>	
in	<i>tileWidth</i>	
in	<i>tileHeight</i>	

**3.14.2.10 OptixResult optixUtilDenoiserSplitImage (**  
**const OptixImage2D & *input*,**  
**const OptixImage2D & *output*,**  
**unsigned int *overlapWindowSizeInPixels*,**  
**unsigned int *tileWidth*,**  
**unsigned int *tileHeight*,**  
**std::vector< OptixUtilDenoiserImageTile > & *tiles* ) [inline]**

Split image into 2D tiles given horizontal and vertical tile size.

### Parameters

in	<i>input</i>	full resolution input image to be split
in	<i>output</i>	full resolution output image
in	<i>overlapWindowSizeInPixels</i>	see <a href="#">OptixDenoiserSizes</a> , <a href="#">optixDenoiserComputeMemoryResources</a>
in	<i>tileWidth</i>	maximum width of tiles
in	<i>tileHeight</i>	maximum height of tiles
out	<i>tiles</i>	list of tiles covering the input image

**3.14.2.11 unsigned int optixUtilGetPixelStride (**  
**const OptixImage2D & *image* ) [inline]**

Return pixel stride in bytes for the given pixel format if the pixelStrideInBytes member of the image is zero. Otherwise return pixelStrideInBytes from the image.



**Parameters**

in	<i>image</i>	Image containing the pixel stride
----	--------------	-----------------------------------

## 4 Namespace Documentation

### 4.1 optix\_impl Namespace Reference

#### Functions

- static `__forceinline__`  
`__device__ void optixDumpStaticTransformFromHandle (OptixTraversableHandle handle)`
- static `__forceinline__`  
`__device__ void optixDumpMotionMatrixTransformFromHandle (OptixTraversableHandle handle)`
- static `__forceinline__`  
`__device__ void optixDumpSrtMatrixTransformFromHandle (OptixTraversableHandle handle)`
- static `__forceinline__`  
`__device__ void optixDumpInstanceFromHandle (OptixTraversableHandle handle)`
- static `__forceinline__`  
`__device__ void optixDumpTransform (OptixTraversableHandle handle)`
- static `__forceinline__`  
`__device__ void optixDumpTransformList ()`
- static `__forceinline__`  
`__device__ void optixDumpExceptionDetails ()`
- static `__forceinline__`  
`__device__ float4 optixAddFloat4 (const float4 &a, const float4 &b)`
- static `__forceinline__`  
`__device__ float4 optixMulFloat4 (const float4 &a, float b)`
- static `__forceinline__`  
`__device__ uint4 optixLdg (unsigned long long addr)`
- `template<class T >`  
`static __forceinline__ __device__ T optixLoadReadOnlyAlign16 (const T *ptr)`
- static `__forceinline__`  
`__device__ float4 optixMultiplyRowMatrix (const float4 vec, const float4 m0, const float4 m1, const float4 m2)`
- static `__forceinline__`  
`__device__ void optixGetMatrixFromSrt (float4 &m0, float4 &m1, float4 &m2, const OptixSRTData &srt)`
- static `__forceinline__`  
`__device__ void optixInvertMatrix (float4 &m0, float4 &m1, float4 &m2)`
- static `__forceinline__`  
`__device__ void optixLoadInterpolatedMatrixKey (float4 &m0, float4 &m1, float4 &m2, const float4 *matrix, const float t1)`
- static `__forceinline__`  
`__device__ void optixLoadInterpolatedSrtKey (float4 &srt0, float4 &srt1, float4 &srt2, float4 &srt3, const float4 *srt, const float t1)`
- static `__forceinline__`  
`__device__ void optixResolveMotionKey (float &localt, int &key, const OptixMotionOptions &options, const float globalt)`

- static `__forceinline__`  
`__device__ void optixGetInterpolatedTransformation` (float4 &trf0, float4 &trf1, float4 &trf2, const `OptixMatrixMotionTransform` \*transformData, const float time)
- static `__forceinline__`  
`__device__ void optixGetInterpolatedTransformation` (float4 &trf0, float4 &trf1, float4 &trf2, const `OptixSRTMotionTransform` \*transformData, const float time)
- static `__forceinline__`  
`__device__ void optixGetInterpolatedTransformationFromHandle` (float4 &trf0, float4 &trf1, float4 &trf2, const `OptixTraversableHandle` handle, const float time, const bool objectToWorld)
- static `__forceinline__`  
`__device__ void optixGetWorldToObjectTransformMatrix` (float4 &m0, float4 &m1, float4 &m2)
- static `__forceinline__`  
`__device__ void optixGetObjectToWorldTransformMatrix` (float4 &m0, float4 &m1, float4 &m2)
- static `__forceinline__`  
`__device__ float3 optixTransformPoint` (const float4 &m0, const float4 &m1, const float4 &m2, const float3 &p)
- static `__forceinline__`  
`__device__ float3 optixTransformVector` (const float4 &m0, const float4 &m1, const float4 &m2, const float3 &v)
- static `__forceinline__`  
`__device__ float3 optixTransformNormal` (const float4 &m0, const float4 &m1, const float4 &m2, const float3 &n)

#### 4.1.1 Function Documentation

- 4.1.1.1** static `__forceinline__ __device__ float4 optix_impl::optixAddFloat4` (  
     const float4 & *a*,  
     const float4 & *b* ) [static]
- 4.1.1.2** static `__forceinline__ __device__ void optix_impl::optixDumpExceptionDetails` ( )  
 [static]
- 4.1.1.3** static `__forceinline__ __device__ void optix_impl::optixDumpInstanceFromHandle` (  
     `OptixTraversableHandle` *handle* ) [static]
- 4.1.1.4** static `__forceinline__ __device__ void op-`  
`tix_impl::optixDumpMotionMatrixTransformFromHandle` (  
     `OptixTraversableHandle` *handle* ) [static]
- 4.1.1.5** static `__forceinline__ __device__ void op-`  
`tix_impl::optixDumpSrtMatrixTransformFromHandle` (  
     `OptixTraversableHandle` *handle* ) [static]
- 4.1.1.6** static `__forceinline__ __device__ void op-`  
`tix_impl::optixDumpStaticTransformFromHandle` (

**OptixTraversableHandle *handle* ) [static]**

**4.1.1.7 static \_\_forceinline\_\_ \_\_device\_\_ void optix\_impl::optixDumpTransform (**  
**OptixTraversableHandle *handle* ) [static]**

**4.1.1.8 static \_\_forceinline\_\_ \_\_device\_\_ void optix\_impl::optixDumpTransformList ( )**  
**[static]**

**4.1.1.9 static \_\_forceinline\_\_ \_\_device\_\_ void optix\_impl::optixGetInterpolatedTransformation**  
**(**  
**float4 & *trf0*,**  
**float4 & *trf1*,**  
**float4 & *trf2*,**  
**const OptixMatrixMotionTransform \* *transformData*,**  
**const float *time* ) [static]**

**4.1.1.10 static \_\_forceinline\_\_ \_\_device\_\_ void optix\_impl::optixGetInterpolatedTransformation**  
**(**  
**float4 & *trf0*,**  
**float4 & *trf1*,**  
**float4 & *trf2*,**  
**const OptixSRTMotionTransform \* *transformData*,**  
**const float *time* ) [static]**

**4.1.1.11 static \_\_forceinline\_\_ \_\_device\_\_ void op-**  
**tix\_impl::optixGetInterpolatedTransformationFromHandle**  
**(**  
**float4 & *trf0*,**  
**float4 & *trf1*,**  
**float4 & *trf2*,**  
**const OptixTraversableHandle *handle*,**  
**const float *time*,**  
**const bool *objectToWorld* ) [static]**

**4.1.1.12 static \_\_forceinline\_\_ \_\_device\_\_ void optix\_impl::optixGetMatrixFromSrt (**  
**float4 & *m0*,**  
**float4 & *m1*,**  
**float4 & *m2*,**  
**const OptixSRTData & *srt* ) [static]**

**4.1.1.13 static \_\_forceinline\_\_ \_\_device\_\_ void op-**  
**tix\_impl::optixGetObjectToWorldTransformMatrix (**  
**float4 & *m0*,**  
**float4 & *m1*,**

**float4 & *m2* ) [static]**

**4.1.1.14 static \_\_forceinline\_\_ \_\_device\_\_ void optix\_impl::optixGetWorldToObjectTransformMatrix (**  
**float4 & *m0*,**  
**float4 & *m1*,**  
**float4 & *m2* ) [static]**

**4.1.1.15 static \_\_forceinline\_\_ \_\_device\_\_ void optix\_impl::optixInvertMatrix (**  
**float4 & *m0*,**  
**float4 & *m1*,**  
**float4 & *m2* ) [static]**

**4.1.1.16 static \_\_forceinline\_\_ \_\_device\_\_ uint4 optix\_impl::optixLdg (**  
**unsigned long long *addr* ) [static]**

**4.1.1.17 static \_\_forceinline\_\_ \_\_device\_\_ void optix\_impl::optixLoadInterpolatedMatrixKey (**  
**float4 & *m0*,**  
**float4 & *m1*,**  
**float4 & *m2*,**  
**const float4 \* *matrix*,**  
**const float *t1* ) [static]**

**4.1.1.18 static \_\_forceinline\_\_ \_\_device\_\_ void optix\_impl::optixLoadInterpolatedSrtKey (**  
**float4 & *srt0*,**  
**float4 & *srt1*,**  
**float4 & *srt2*,**  
**float4 & *srt3*,**  
**const float4 \* *srt*,**  
**const float *t1* ) [static]**

**4.1.1.19 template<class T > static \_\_forceinline\_\_ \_\_device\_\_ T**  
**optix\_impl::optixLoadReadOnlyAlign16 (**  
**const T \* *ptr* ) [static]**

**4.1.1.20 static \_\_forceinline\_\_ \_\_device\_\_ float4 optix\_impl::optixMulFloat4 (**  
**const float4 & *a*,**  
**float *b* ) [static]**

**4.1.1.21 static \_\_forceinline\_\_ \_\_device\_\_ float4 optix\_impl::optixMultiplyRowMatrix (**  
**const float4 *vec*,**  
**const float4 *m0*,**  
**const float4 *m1*,**

```
const float4 m2 ) [static]
```

```
4.1.1.22 static __forceinline__ __device__ void optix_impl::optixResolveMotionKey (
    float & localt,
    int & key,
    const OptixMotionOptions & options,
    const float globalt ) [static]
```

```
4.1.1.23 static __forceinline__ __device__ float3 optix_impl::optixTransformNormal (
    const float4 & m0,
    const float4 & m1,
    const float4 & m2,
    const float3 & n ) [static]
```

```
4.1.1.24 static __forceinline__ __device__ float3 optix_impl::optixTransformPoint (
    const float4 & m0,
    const float4 & m1,
    const float4 & m2,
    const float3 & p ) [static]
```

```
4.1.1.25 static __forceinline__ __device__ float3 optix_impl::optixTransformVector (
    const float4 & m0,
    const float4 & m1,
    const float4 & m2,
    const float3 & v ) [static]
```

## 5 Class Documentation

### 5.1 OptixAabb Struct Reference

#### Public Attributes

- float [minX](#)
- float [minY](#)
- float [minZ](#)
- float [maxX](#)
- float [maxY](#)
- float [maxZ](#)

#### 5.1.1 Detailed Description

AABB inputs.

### 5.1.2 Member Data Documentation

#### 5.1.2.1 float OptixAabb::maxX

Upper extent in X direction.

#### 5.1.2.2 float OptixAabb::maxY

Upper extent in Y direction.

#### 5.1.2.3 float OptixAabb::maxZ

Upper extent in Z direction.

#### 5.1.2.4 float OptixAabb::minX

Lower extent in X direction.

#### 5.1.2.5 float OptixAabb::minY

Lower extent in Y direction.

#### 5.1.2.6 float OptixAabb::minZ

Lower extent in Z direction.

## 5.2 OptixAccelBufferSizes Struct Reference

### Public Attributes

- [size\\_t outputSizeInBytes](#)
- [size\\_t tempSizeInBytes](#)
- [size\\_t tempUpdateSizeInBytes](#)

### 5.2.1 Detailed Description

Struct for querying builder allocation requirements.

Once queried the sizes should be used to allocate device memory of at least these sizes.

See Also

[optixAccelComputeMemoryUsage\(\)](#)

### 5.2.2 Member Data Documentation

#### 5.2.2.1 size\_t OptixAccelBufferSizes::outputSizeInBytes

The size in bytes required for the outputBuffer parameter to optixAccelBuild when doing a build (OPTIX\_BUILD\_OPERATION\_BUILD).

### 5.2.2.2 `size_t` `OptixAccelBufferSizes::tempSizeInBytes`

The size in bytes required for the `tempBuffer` parameter to `optixAccelBuild` when doing a build (`OPTIX_BUILD_OPERATION_BUILD`).

### 5.2.2.3 `size_t` `OptixAccelBufferSizes::tempUpdateSizeInBytes`

The size in bytes required for the `tempBuffer` parameter to `optixAccelBuild` when doing an update (`OPTIX_BUILD_OPERATION_UPDATE`). This value can be different than `tempSizeInBytes` used for a full build. Only non-zero if `OPTIX_BUILD_FLAG_ALLOW_UPDATE` flag is set in [OptixAccelBuildOptions](#).

## 5.3 OptixAccelBuildOptions Struct Reference

### Public Attributes

- unsigned int [buildFlags](#)
- [OptixBuildOperation](#) `operation`
- [OptixMotionOptions](#) `motionOptions`

### 5.3.1 Detailed Description

Build options for acceleration structures.

See Also

[optixAccelComputeMemoryUsage\(\)](#), [optixAccelBuild\(\)](#)

### 5.3.2 Member Data Documentation

#### 5.3.2.1 unsigned int `OptixAccelBuildOptions::buildFlags`

Combinations of [OptixBuildFlags](#).

#### 5.3.2.2 `OptixMotionOptions` `OptixAccelBuildOptions::motionOptions`

Options for motion.

#### 5.3.2.3 `OptixBuildOperation` `OptixAccelBuildOptions::operation`

If `OPTIX_BUILD_OPERATION_UPDATE` the output buffer is assumed to contain the result of a full build with `OPTIX_BUILD_FLAG_ALLOW_UPDATE` set and using the same number of primitives. It is updated incrementally to reflect the current position of the primitives.



## 5.4 OptixAccelEmitDesc Struct Reference

### Public Attributes

- [CUdeviceptr](#) result
- [OptixAccelPropertyType](#) type

#### 5.4.1 Detailed Description

Specifies a type and output destination for emitted post-build properties.

See Also

[optixAccelBuild\(\)](#)

#### 5.4.2 Member Data Documentation

##### 5.4.2.1 CUdeviceptr OptixAccelEmitDesc::result

Output buffer for the properties.

##### 5.4.2.2 OptixAccelPropertyType OptixAccelEmitDesc::type

Requested property.

## 5.5 OptixAccelRelocationInfo Struct Reference

### Public Attributes

- unsigned long long [info](#) [4]

#### 5.5.1 Detailed Description

Used to store information related to relocation of acceleration structures.

See Also

[optixAccelGetRelocationInfo\(\)](#), [optixAccelCheckRelocationCompatibility\(\)](#), [optixAccelRelocate\(\)](#)

#### 5.5.2 Member Data Documentation

##### 5.5.2.1 unsigned long long OptixAccelRelocationInfo::info[4]

Opaque data, used internally, should not be modified.

## 5.6 OptixBuildInput Struct Reference

### Public Attributes

- [OptixBuildInputType](#) type
- union {
  - [OptixBuildInputTriangleArray](#) triangleArray
  - [OptixBuildInputCurveArray](#) curveArray
  - [OptixBuildInputCustomPrimitiveArray](#) customPrimitiveArray
  - [OptixBuildInputInstanceArray](#) instanceArray
  - char [pad](#) [1024]
- };

### 5.6.1 Detailed Description

Build inputs.

All of them support motion and the size of the data arrays needs to match the number of motion steps

See Also

[optixAccelComputeMemoryUsage\(\)](#), [optixAccelBuild\(\)](#)

### 5.6.2 Member Data Documentation

#### 5.6.2.1 union { ... }

#### 5.6.2.2 OptixBuildInputCurveArray OptixBuildInput::curveArray

Curve inputs.

#### 5.6.2.3 OptixBuildInputCustomPrimitiveArray OptixBuildInput::customPrimitiveArray

Custom primitive inputs.

#### 5.6.2.4 OptixBuildInputInstanceArray OptixBuildInput::instanceArray

Instance and instance pointer inputs.

#### 5.6.2.5 char OptixBuildInput::pad[1024]

#### 5.6.2.6 OptixBuildInputTriangleArray OptixBuildInput::triangleArray

Triangle inputs.

#### 5.6.2.7 OptixBuildInputType OptixBuildInput::type

The type of the build input.

## 5.7 OptixBuildInputCurveArray Struct Reference

### Public Attributes

- [OptixPrimitiveType](#) curveType
- unsigned int numPrimitives
- const [CUdeviceptr](#) \* vertexBuffers
- unsigned int numVertices
- unsigned int vertexStrideInBytes
- const [CUdeviceptr](#) \* widthBuffers
- unsigned int widthStrideInBytes
- const [CUdeviceptr](#) \* normalBuffers
- unsigned int normalStrideInBytes
- [CUdeviceptr](#) indexBuffer
- unsigned int indexStrideInBytes
- unsigned int flag
- unsigned int primitiveIndexOffset
- unsigned int endcapFlags

### 5.7.1 Detailed Description

Curve inputs.

A curve is a swept surface defined by a 3D spline curve and a varying width (radius). A curve (or "strand") of degree  $d$  (3=cubic, 2=quadratic, 1=linear) is represented by  $N > d$  vertices and  $N$  width values, and comprises  $N - d$  segments. Each segment is defined by  $d+1$  consecutive vertices. Each curve may have a different number of vertices.

OptiX describes the curve array as a list of curve segments. The primitive id is the segment number. It is the user's responsibility to maintain a mapping between curves and curve segments. Each index buffer entry  $i = \text{indexBuffer}[\text{primid}]$  specifies the start of a curve segment, represented by  $d+1$  consecutive vertices in the vertex buffer, and  $d+1$  consecutive widths in the width buffer. Width is interpolated the same way vertices are interpolated, that is, using the curve basis.

Each curves build input has only one SBT record. To create curves with different materials in the same BVH, use multiple build inputs.

See Also

[OptixBuildInput::curveArray](#)

### 5.7.2 Member Data Documentation

#### 5.7.2.1 OptixPrimitiveType OptixBuildInputCurveArray::curveType

Curve degree and basis.

See Also

[OptixPrimitiveType](#)

### 5.7.2.2 unsigned int OptixBuildInputCurveArray::endcapFlags

End cap flags, see [OptixCurveEndcapFlags](#).

### 5.7.2.3 unsigned int OptixBuildInputCurveArray::flag

Combination of [OptixGeometryFlags](#) describing the primitive behavior.

### 5.7.2.4 CUdeviceptr OptixBuildInputCurveArray::indexBuffer

Device pointer to array of unsigned ints, one per curve segment. This buffer is required (unlike for [OptixBuildInputTriangleArray](#)). Each index is the start of degree+1 consecutive vertices in [vertexBuffers](#), and corresponding widths in [widthBuffers](#) and normals in [normalBuffers](#). These define a single segment. Size of array is [numPrimitives](#).

### 5.7.2.5 unsigned int OptixBuildInputCurveArray::indexStrideInBytes

Stride between indices. If set to zero, indices are assumed to be tightly packed and stride is `sizeof(unsigned int)`.

### 5.7.2.6 const CUdeviceptr\* OptixBuildInputCurveArray::normalBuffers

Reserved for future use.

### 5.7.2.7 unsigned int OptixBuildInputCurveArray::normalStrideInBytes

Reserved for future use.

### 5.7.2.8 unsigned int OptixBuildInputCurveArray::numPrimitives

Number of primitives. Each primitive is a polynomial curve segment.

### 5.7.2.9 unsigned int OptixBuildInputCurveArray::numVertices

Number of vertices in each buffer in [vertexBuffers](#).

### 5.7.2.10 unsigned int OptixBuildInputCurveArray::primitiveIndexOffset

Primitive index bias, applied in [optixGetPrimitiveIndex\(\)](#). Sum of [primitiveIndexOffset](#) and number of primitives must not overflow 32bits.

### 5.7.2.11 const CUdeviceptr\* OptixBuildInputCurveArray::vertexBuffers

Pointer to host array of device pointers, one per motion step. Host array size must match number of motion keys as set in [OptixMotionOptions](#) (or an array of size 1 if [OptixMotionOptions::numKeys](#) is set to 1). Each per-motion-key device pointer must point to an array of floats (the vertices of the curves).

**5.7.2.12 unsigned int OptixBuildInputCurveArray::vertexStrideInBytes**

Stride between vertices. If set to zero, vertices are assumed to be tightly packed and stride is sizeof( float3 ).

**5.7.2.13 const CUdeviceptr\* OptixBuildInputCurveArray::widthBuffers**

Parallel to vertexBuffers: a device pointer per motion step, each with numVertices float values, specifying the curve width (radius) corresponding to each vertex.

**5.7.2.14 unsigned int OptixBuildInputCurveArray::widthStrideInBytes**

Stride between widths. If set to zero, widths are assumed to be tightly packed and stride is sizeof( float ).

**5.8 OptixBuildInputCustomPrimitiveArray Struct Reference****Public Attributes**

- const CUdeviceptr \* aabbBuffers
- unsigned int numPrimitives
- unsigned int strideInBytes
- const unsigned int \* flags
- unsigned int numSbtRecords
- CUdeviceptr sbtIndexOffsetBuffer
- unsigned int sbtIndexOffsetSizeInBytes
- unsigned int sbtIndexOffsetStrideInBytes
- unsigned int primitiveIndexOffset

**5.8.1 Detailed Description**

Custom primitive inputs.

See Also

[OptixBuildInput::customPrimitiveArray](#)

**5.8.2 Member Data Documentation****5.8.2.1 const CUdeviceptr\* OptixBuildInputCustomPrimitiveArray::aabbBuffers**

Points to host array of device pointers to AABBs (type [OptixAabb](#)), one per motion step. Host array size must match number of motion keys as set in [OptixMotionOptions](#) (or an array of size 1 if [OptixMotionOptions::numKeys](#) is set to 1). Each device pointer must be a multiple of OPTIX\_AABB\_BUFFER\_BYTE\_ALIGNMENT.

### 5.8.2.2 `const unsigned int*` `OptixBuildInputCustomPrimitiveArray::flags`

Array of flags, to specify flags per sbt record, combinations of `OptixGeometryFlags` describing the primitive behavior, size must match `numSbtRecords`.

### 5.8.2.3 `unsigned int` `OptixBuildInputCustomPrimitiveArray::numPrimitives`

Number of primitives in each buffer (i.e., per motion step) in `OptixBuildInputCustomPrimitiveArray::aabbBuffers`.

### 5.8.2.4 `unsigned int` `OptixBuildInputCustomPrimitiveArray::numSbtRecords`

Number of sbt records available to the sbt index offset override.

### 5.8.2.5 `unsigned int` `OptixBuildInputCustomPrimitiveArray::primitiveIndexOffset`

Primitive index bias, applied in `optixGetPrimitiveIndex()`. Sum of `primitiveIndexOffset` and number of primitive must not overflow 32bits.

### 5.8.2.6 `CUdeviceptr` `OptixBuildInputCustomPrimitiveArray::sbtIndexOffsetBuffer`

Device pointer to per-primitive local sbt index offset buffer. May be NULL. Every entry must be in range `[0,numSbtRecords-1]`. Size needs to be the number of primitives.

### 5.8.2.7 `unsigned int` `OptixBuildInputCustomPrimitiveArray::sbtIndexOffsetSizeInBytes`

Size of type of the sbt index offset. Needs to be 0, 1, 2 or 4 (8, 16 or 32 bit).

### 5.8.2.8 `unsigned int` `OptixBuildInputCustomPrimitiveArray::sbtIndexOffsetStrideInBytes`

Stride between the index offsets. If set to zero, the offsets are assumed to be tightly packed and the stride matches the size of the type (`sbtIndexOffsetSizeInBytes`).

### 5.8.2.9 `unsigned int` `OptixBuildInputCustomPrimitiveArray::strideInBytes`

Stride between AABBs (per motion key). If set to zero, the aabbs are assumed to be tightly packed and the stride is assumed to be `sizeof( OptixAabb )`. If non-zero, the value must be a multiple of `OPTIX_AABB_BUFFER_BYTE_ALIGNMENT`.

## 5.9 OptixBuildInputInstanceArray Struct Reference

### Public Attributes

- `CUdeviceptr` `instances`
- `unsigned int` `numInstances`

### 5.9.1 Detailed Description

Instance and instance pointer inputs.

See Also

[OptixBuildInput::instanceArray](#)

## 5.9.2 Member Data Documentation

### 5.9.2.1 CUdeviceptr OptixBuildInputInstanceArray::instances

If [OptixBuildInput::type](#) is `OPTIX_BUILD_INPUT_TYPE_INSTANCE_POINTERS` instances and aabbs should be interpreted as arrays of pointers instead of arrays of structs.

This pointer must be a multiple of `OPTIX_INSTANCE_BYTE_ALIGNMENT` if [OptixBuildInput::type](#) is `OPTIX_BUILD_INPUT_TYPE_INSTANCES`. The array elements must be a multiple of `OPTIX_INSTANCE_BYTE_ALIGNMENT` if [OptixBuildInput::type](#) is `OPTIX_BUILD_INPUT_TYPE_INSTANCE_POINTERS`.

### 5.9.2.2 unsigned int OptixBuildInputInstanceArray::numInstances

Number of elements in [OptixBuildInputInstanceArray::instances](#).

## 5.10 OptixBuildInputTriangleArray Struct Reference

### Public Attributes

- const [CUdeviceptr](#) \* vertexBuffers
- unsigned int numVertices
- [OptixVertexFormat](#) vertexFormat
- unsigned int vertexStrideInBytes
- [CUdeviceptr](#) indexBuffer
- unsigned int numIndexTriplets
- [OptixIndicesFormat](#) indexFormat
- unsigned int indexStrideInBytes
- [CUdeviceptr](#) preTransform
- const unsigned int \* flags
- unsigned int numSbtRecords
- [CUdeviceptr](#) sbtIndexOffsetBuffer
- unsigned int sbtIndexOffsetSizeInBytes
- unsigned int sbtIndexOffsetStrideInBytes
- unsigned int primitiveIndexOffset
- [OptixTransformFormat](#) transformFormat

### 5.10.1 Detailed Description

Triangle inputs.

See Also

[OptixBuildInput::triangleArray](#)

## 5.10.2 Member Data Documentation

### 5.10.2.1 `const unsigned int*` `OptixBuildInputTriangleArray::flags`

Array of flags, to specify flags per sbt record, combinations of `OptixGeometryFlags` describing the primitive behavior, size must match `numSbtRecords`.

### 5.10.2.2 `CUdeviceptr` `OptixBuildInputTriangleArray::indexBuffer`

Optional pointer to array of 16 or 32-bit int triplets, one triplet per triangle. The minimum alignment must match the natural alignment of the type as specified in the `indexFormat`, i.e., for `OPTIX_INDICES_FORMAT_UNSIGNED_INT3` 4-byte and for `OPTIX_INDICES_FORMAT_UNSIGNED_SHORT3` a 2-byte alignment.

### 5.10.2.3 `OptixIndicesFormat` `OptixBuildInputTriangleArray::indexFormat`

See Also

[OptixIndicesFormat](#)

### 5.10.2.4 `unsigned int` `OptixBuildInputTriangleArray::indexStrideInBytes`

Stride between triplets of indices. If set to zero, indices are assumed to be tightly packed and stride is inferred from `indexFormat`.

### 5.10.2.5 `unsigned int` `OptixBuildInputTriangleArray::numIndexTriplets`

Size of array in [OptixBuildInputTriangleArray::indexBuffer](#). For build, needs to be zero if `indexBuffer` is `nullptr`.

### 5.10.2.6 `unsigned int` `OptixBuildInputTriangleArray::numSbtRecords`

Number of sbt records available to the sbt index offset override.

### 5.10.2.7 `unsigned int` `OptixBuildInputTriangleArray::numVertices`

Number of vertices in each of buffer in [OptixBuildInputTriangleArray::vertexBuffers](#).

### 5.10.2.8 `CUdeviceptr` `OptixBuildInputTriangleArray::preTransform`

Optional pointer to array of floats representing a 3x4 row major affine transformation matrix. This pointer must be a multiple of `OPTIX_GEOMETRY_TRANSFORM_BYTE_ALIGNMENT`.

### 5.10.2.9 `unsigned int` `OptixBuildInputTriangleArray::primitiveIndexOffset`

Primitive index bias, applied in [optixGetPrimitiveIndex\(\)](#). Sum of `primitiveIndexOffset` and number of triangles must not overflow 32bits.



**5.10.2.10 CUdeviceptr OptixBuildInputTriangleArray::sbtIndexOffsetBuffer**

Device pointer to per-primitive local sbt index offset buffer. May be NULL. Every entry must be in range [0,numSbtRecords-1]. Size needs to be the number of primitives.

**5.10.2.11 unsigned int OptixBuildInputTriangleArray::sbtIndexOffsetSizeInBytes**

Size of type of the sbt index offset. Needs to be 0, 1, 2 or 4 (8, 16 or 32 bit).

**5.10.2.12 unsigned int OptixBuildInputTriangleArray::sbtIndexOffsetStrideInBytes**

Stride between the index offsets. If set to zero, the offsets are assumed to be tightly packed and the stride matches the size of the type (sbtIndexOffsetSizeInBytes).

**5.10.2.13 OptixTransformFormat OptixBuildInputTriangleArray::transformFormat**

See Also

[OptixTransformFormat](#)

**5.10.2.14 const CUdeviceptr\* OptixBuildInputTriangleArray::vertexBuffers**

Points to host array of device pointers, one per motion step. Host array size must match the number of motion keys as set in [OptixMotionOptions](#) (or an array of size 1 if [OptixMotionOptions::numKeys](#) is set to 0 or 1). Each per motion key device pointer must point to an array of vertices of the triangles in the format as described by vertexFormat. The minimum alignment must match the natural alignment of the type as specified in the vertexFormat, i.e., for OPTIX\_VERTEX\_FORMAT\_FLOATX 4-byte, for all others a 2-byte alignment. However, an 16-byte stride (and buffer alignment) is recommended for vertices of format OPTIX\_VERTEX\_FORMAT\_FLOAT3 for GAS build performance.

**5.10.2.15 OptixVertexFormat OptixBuildInputTriangleArray::vertexFormat**

See Also

[OptixVertexFormat](#)

**5.10.2.16 unsigned int OptixBuildInputTriangleArray::vertexStrideInBytes**

Stride between vertices. If set to zero, vertices are assumed to be tightly packed and stride is inferred from vertexFormat.

**5.11 OptixBuiltinISOptions Struct Reference****Public Attributes**

- [OptixPrimitiveType](#) builtinISModuleType
- int [usesMotionBlur](#)
- unsigned int [buildFlags](#)
- unsigned int [curveEndcapFlags](#)

### 5.11.1 Detailed Description

Specifies the options for retrieving an intersection program for a built-in primitive type. The primitive type must not be `OPTIX_PRIMITIVE_TYPE_CUSTOM`.

See Also

[optxBuiltinISModuleGet\(\)](#)

### 5.11.2 Member Data Documentation

#### 5.11.2.1 `unsigned int OptixBuiltinISOptions::buildFlags`

Build flags, see `OptixBuildFlags`.

#### 5.11.2.2 `OptixPrimitiveType OptixBuiltinISOptions::builtinISModuleType`

#### 5.11.2.3 `unsigned int OptixBuiltinISOptions::curveEndcapFlags`

End cap properties of curves, see `OptixCurveEndcapFlags`, 0 for non-curve types.

#### 5.11.2.4 `int OptixBuiltinISOptions::usesMotionBlur`

Boolean value indicating whether vertex motion blur is used (but not motion transform blur).

## 5.12 OptixDenoiserGuideLayer Struct Reference

### Public Attributes

- [OptixImage2D](#) `albedo`
- [OptixImage2D](#) `normal`
- [OptixImage2D](#) `flow`

### 5.12.1 Detailed Description

Guide layer for the denoiser.

See Also

[optixDenoiserInvoke\(\)](#)

### 5.12.2 Member Data Documentation

#### 5.12.2.1 OptixImage2D OptixDenoiserGuideLayer::albedo

#### 5.12.2.2 OptixImage2D OptixDenoiserGuideLayer::flow

#### 5.12.2.3 OptixImage2D OptixDenoiserGuideLayer::normal

## 5.13 OptixDenoiserLayer Struct Reference

### Public Attributes

- [OptixImage2D](#) input
- [OptixImage2D](#) previousOutput
- [OptixImage2D](#) output

### 5.13.1 Detailed Description

Input/Output layers for the denoiser.

See Also

[optixDenoiserInvoke\(\)](#)

### 5.13.2 Member Data Documentation

#### 5.13.2.1 OptixImage2D OptixDenoiserLayer::input

#### 5.13.2.2 OptixImage2D OptixDenoiserLayer::output

#### 5.13.2.3 OptixImage2D OptixDenoiserLayer::previousOutput

## 5.14 OptixDenoiserOptions Struct Reference

### Public Attributes

- unsigned int [guideAlbedo](#)
- unsigned int [guideNormal](#)

### 5.14.1 Detailed Description

Options used by the denoiser.

See Also

[optixDenoiserCreate\(\)](#)

## 5.14.2 Member Data Documentation

### 5.14.2.1 unsigned int OptixDenoiserOptions::guideAlbedo

### 5.14.2.2 unsigned int OptixDenoiserOptions::guideNormal

## 5.15 OptixDenoiserParams Struct Reference

### Public Attributes

- unsigned int [denoiseAlpha](#)
- [CUdeviceptr](#) [hdrIntensity](#)
- float [blendFactor](#)
- [CUdeviceptr](#) [hdrAverageColor](#)

### 5.15.1 Detailed Description

Various parameters used by the denoiser.

See Also

[optixDenoiserInvoke\(\)](#)  
[optixDenoiserComputeIntensity\(\)](#)  
[optixDenoiserComputeAverageColor\(\)](#)

## 5.15.2 Member Data Documentation

### 5.15.2.1 float OptixDenoiserParams::blendFactor

blend factor. If set to 0 the output is 100% of the denoised input. If set to 1, the output is 100% of the unmodified input. Values between 0 and 1 will linearly interpolate between the denoised and unmodified input.

### 5.15.2.2 unsigned int OptixDenoiserParams::denoiseAlpha

alpha denoise mode: 0 Copy alpha (if present) from input layer, no denoising. 1 Denoise alpha separately. In AOV model kinds, treat alpha like an AOV. 2 In AOV model kinds, full denoise pass with alpha (slower than mode 1).

### 5.15.2.3 CUdeviceptr OptixDenoiserParams::hdrAverageColor

this parameter is used when the OPTIX\_DENOISER\_MODEL\_KIND\_AOV model kind is set. average log color of input image, separate for RGB channels (default null pointer). points to three floats. with the default (null pointer) denoised results will not be optimal.

#### 5.15.2.4 CUdeviceptr OptixDenoiserParams::hdrIntensity

average log intensity of input image (default null pointer). points to a single float. with the default (null pointer) denoised results will not be optimal for very dark or bright input images.

## 5.16 OptixDenoiserSizes Struct Reference

### Public Attributes

- [size\\_t stateSizeInBytes](#)
- [size\\_t withOverlapScratchSizeInBytes](#)
- [size\\_t withoutOverlapScratchSizeInBytes](#)
- [unsigned int overlapWindowSizeInPixels](#)

#### 5.16.1 Detailed Description

Various sizes related to the denoiser.

See Also

[optixDenoiserComputeMemoryResources\(\)](#)

#### 5.16.2 Member Data Documentation

##### 5.16.2.1 unsigned int OptixDenoiserSizes::overlapWindowSizeInPixels

##### 5.16.2.2 size\_t OptixDenoiserSizes::stateSizeInBytes

##### 5.16.2.3 size\_t OptixDenoiserSizes::withoutOverlapScratchSizeInBytes

##### 5.16.2.4 size\_t OptixDenoiserSizes::withOverlapScratchSizeInBytes

## 5.17 OptixDeviceContextOptions Struct Reference

### Public Attributes

- [OptixLogCallback logCallbackFunction](#)
- [void \\* logCallbackData](#)
- [int logCallbackLevel](#)
- [OptixDeviceContextValidationMode validationMode](#)

#### 5.17.1 Detailed Description

Parameters used for [optixDeviceContextCreate\(\)](#)

See Also

[optixDeviceContextCreate\(\)](#)

## 5.17.2 Member Data Documentation

### 5.17.2.1 void\* OptixDeviceContextOptions::logCallbackData

Pointer stored and passed to logCallbackFunction when a message is generated.

### 5.17.2.2 OptixLogCallback OptixDeviceContextOptions::logCallbackFunction

Function pointer used when OptiX wishes to generate messages.

### 5.17.2.3 int OptixDeviceContextOptions::logCallbackLevel

Maximum callback level to generate message for (see [OptixLogCallback](#))

### 5.17.2.4 OptixDeviceContextValidationMode OptixDeviceContextOptions::validationMode

Validation mode of context.

## 5.18 OptixFunctionTable Struct Reference

### Public Attributes

#### Error handling

- [const char \\*\(\\* optixGetErrorName \)\(OptixResult result\)](#)
- [const char \\*\(\\* optixGetErrorString \)\(OptixResult result\)](#)

#### Device context

- [OptixResult\(\\* optixDeviceContextCreate \)\(CUcontext fromContext, const OptixDeviceContextOptions \\*options, OptixDeviceContext \\*context\)](#)
- [OptixResult\(\\* optixDeviceContextDestroy \)\(OptixDeviceContext context\)](#)
- [OptixResult\(\\* optixDeviceContextGetProperty \)\(OptixDeviceContext context, OptixDeviceProperty property, void \\*value, size\\_t sizeInBytes\)](#)
- [OptixResult\(\\* optixDeviceContextSetLogCallback \)\(OptixDeviceContext context, OptixLogCallback callbackFunction, void \\*callbackData, unsigned int callbackLevel\)](#)
- [OptixResult\(\\* optixDeviceContextSetCacheEnabled \)\(OptixDeviceContext context, int enabled\)](#)
- [OptixResult\(\\* optixDeviceContextSetCacheLocation \)\(OptixDeviceContext context, const char \\*location\)](#)
- [OptixResult\(\\* optixDeviceContextSetCacheDatabaseSizes \)\(OptixDeviceContext context, size\\_t lowWaterMark, size\\_t highWaterMark\)](#)
- [OptixResult\(\\* optixDeviceContextGetCacheEnabled \)\(OptixDeviceContext context, int \\*enabled\)](#)
- [OptixResult\(\\* optixDeviceContextGetCacheLocation \)\(OptixDeviceContext context, char \\*location, size\\_t locationSize\)](#)

- `OptixResult(* optixDeviceContextGetCacheDatabaseSizes )(OptixDeviceContext context, size_t *lowWaterMark, size_t *highWaterMark)`

## Modules

- `OptixResult(* optixModuleCreateFromPTX )(OptixDeviceContext context, const OptixModuleCompileOptions *moduleCompileOptions, const OptixPipelineCompileOptions *pipelineCompileOptions, const char *PTX, size_t PTXsize, char *logString, size_t *logStringSize, OptixModule *module)`
- `OptixResult(* optixModuleCreateFromPTXWithTasks )(OptixDeviceContext context, const OptixModuleCompileOptions *moduleCompileOptions, const OptixPipelineCompileOptions *pipelineCompileOptions, const char *PTX, size_t PTXsize, char *logString, size_t *logStringSize, OptixModule *module, OptixTask *firstTask)`
- `OptixResult(* optixModuleGetCompilationState )(OptixModule module, OptixModuleCompileState *state)`
- `OptixResult(* optixModuleDestroy )(OptixModule module)`
- `OptixResult(* optixBuiltinISModuleGet )(OptixDeviceContext context, const OptixModuleCompileOptions *moduleCompileOptions, const OptixPipelineCompileOptions *pipelineCompileOptions, const OptixBuiltinISOptions *builtinISOptions, OptixModule *builtinModule)`

## Tasks

- `OptixResult(* optixTaskExecute )(OptixTask task, OptixTask *additionalTasks, unsigned int maxNumAdditionalTasks, unsigned int *numAdditionalTasksCreated)`

## Program groups

- `OptixResult(* optixProgramGroupCreate )(OptixDeviceContext context, const OptixProgramGroupDesc *programDescriptions, unsigned int numProgramGroups, const OptixProgramGroupOptions *options, char *logString, size_t *logStringSize, OptixProgramGroup *programGroups)`
- `OptixResult(* optixProgramGroupDestroy )(OptixProgramGroup programGroup)`
- `OptixResult(* optixProgramGroupGetStackSize )(OptixProgramGroup programGroup, OptixStackSizes *stackSizes)`

## Pipeline

- `OptixResult(* optixPipelineCreate )(OptixDeviceContext context, const OptixPipelineCompileOptions *pipelineCompileOptions, const OptixPipelineLinkOptions *pipelineLinkOptions, const OptixProgramGroup *programGroups, unsigned int numProgramGroups, char *logString, size_t *logStringSize, OptixPipeline *pipeline)`
- `OptixResult(* optixPipelineDestroy )(OptixPipeline pipeline)`
- `OptixResult(* optixPipelineSetStackSize )(OptixPipeline pipeline, unsigned int directCallableStackSizeFromTraversal, unsigned int directCallableStackSizeFromState, unsigned int continuationStackSize, unsigned int maxTraversableGraphDepth)`

## Acceleration structures

- `OptixResult(* optixAccelComputeMemoryUsage )(OptixDeviceContext context, const OptixAccelBuildOptions *accelOptions, const OptixBuildInput *buildInputs, unsigned int numBuildInputs, OptixAccelBufferSizes *bufferSizes)`

- `OptixResult(* optixAccelBuild )(OptixDeviceContext context, CUstream stream, const OptixAccelBuildOptions *accelOptions, const OptixBuildInput *buildInputs, unsigned int numBuildInputs, CUdeviceptr tempBuffer, size_t tempBufferSizeInBytes, CUdeviceptr outputBuffer, size_t outputBufferSizeInBytes, OptixTraversableHandle *outputHandle, const OptixAccelEmitDesc *emittedProperties, unsigned int numEmittedProperties)`
- `OptixResult(* optixAccelGetRelocationInfo )(OptixDeviceContext context, OptixTraversableHandle handle, OptixAccelRelocationInfo *info)`
- `OptixResult(* optixAccelCheckRelocationCompatibility )(OptixDeviceContext context, const OptixAccelRelocationInfo *info, int *compatible)`
- `OptixResult(* optixAccelRelocate )(OptixDeviceContext context, CUstream stream, const OptixAccelRelocationInfo *info, CUdeviceptr instanceTraversableHandles, size_t numInstanceTraversableHandles, CUdeviceptr targetAccel, size_t targetAccelSizeInBytes, OptixTraversableHandle *targetHandle)`
- `OptixResult(* optixAccelCompact )(OptixDeviceContext context, CUstream stream, OptixTraversableHandle inputHandle, CUdeviceptr outputBuffer, size_t outputBufferSizeInBytes, OptixTraversableHandle *outputHandle)`
- `OptixResult(* optixConvertPointerToTraversableHandle )(OptixDeviceContext onDevice, CUdeviceptr pointer, OptixTraversableType traversableType, OptixTraversableHandle *traversableHandle)`
- `void(* reserved1 )(void)`
- `void(* reserved2 )(void)`

### Launch

- `OptixResult(* optixSbtRecordPackHeader )(OptixProgramGroup programGroup, void *sbtRecordHeaderHostPointer)`
- `OptixResult(* optixLaunch )(OptixPipeline pipeline, CUstream stream, CUdeviceptr pipelineParams, size_t pipelineParamsSize, const OptixShaderBindingTable *sbt, unsigned int width, unsigned int height, unsigned int depth)`

### Denoiser

- `OptixResult(* optixDenoiserCreate )(OptixDeviceContext context, OptixDenoiserModelKind modelKind, const OptixDenoiserOptions *options, OptixDenoiser *returnHandle)`
- `OptixResult(* optixDenoiserDestroy )(OptixDenoiser handle)`
- `OptixResult(* optixDenoiserComputeMemoryResources )(const OptixDenoiser handle, unsigned int maximumInputWidth, unsigned int maximumInputHeight, OptixDenoiserSizes *returnSizes)`
- `OptixResult(* optixDenoiserSetup )(OptixDenoiser denoiser, CUstream stream, unsigned int inputWidth, unsigned int inputHeight, CUdeviceptr state, size_t stateSizeInBytes, CUdeviceptr scratch, size_t scratchSizeInBytes)`
- `OptixResult(* optixDenoiserInvoke )(OptixDenoiser denoiser, CUstream stream, const OptixDenoiserParams *params, CUdeviceptr denoiserState, size_t denoiserStateSizeInBytes, const OptixDenoiserGuideLayer *guideLayer, const OptixDenoiserLayer *layers, unsigned int numLayers, unsigned int inputOffsetX, unsigned int inputOffsetY, CUdeviceptr scratch, size_t scratchSizeInBytes)`
- `OptixResult(* optixDenoiserComputeIntensity )(OptixDenoiser handle, CUstream stream, const OptixImage2D *inputImage, CUdeviceptr outputIntensity, CUdeviceptr scratch, size_t scratchSizeInBytes)`



- **OptixResult**(\* **optixDenoiserComputeAverageColor** )(OptixDenoiser handle, CUstream stream, const **OptixImage2D** \*inputImage, CUdeviceptr outputAverageColor, CUdeviceptr scratch, size\_t scratchSizeInBytes)
- **OptixResult**(\* **optixDenoiserCreateWithUserModel** )(OptixDeviceContext context, const void \*data, size\_t dataSizeInBytes, **OptixDenoiser** \*returnHandle)

### 5.18.1 Detailed Description

The function table containing all API functions.

See [optixInit\(\)](#) and [optixInitWithHandle\(\)](#).

### 5.18.2 Member Data Documentation

**5.18.2.1** **OptixResult**( \* **OptixFunctionTable::optixAccelBuild**)(OptixDeviceContext context, CUstream stream, const **OptixAccelBuildOptions** \*accelOptions, const **OptixBuildInput** \*buildInputs, unsigned int numBuildInputs, CUdeviceptr tempBuffer, size\_t tempBufferSizeInBytes, CUdeviceptr outputBuffer, size\_t outputBufferSizeInBytes, **OptixTraversableHandle** \*outputHandle, const **OptixAccelEmitDesc** \*emittedProperties, unsigned int numEmittedProperties)

See [optixAccelBuild\(\)](#).

**5.18.2.2** **OptixResult**( \* **OptixFunctionTable::optixAccelCheckRelocationCompatibility**)(OptixDeviceContext context, const **OptixAccelRelocationInfo** \*info, int \*compatible)

See [optixAccelCheckRelocationCompatibility\(\)](#).

**5.18.2.3** **OptixResult**( \* **OptixFunctionTable::optixAccelCompact**)(OptixDeviceContext context, CUstream stream, **OptixTraversableHandle** inputHandle, CUdeviceptr outputBuffer, size\_t outputBufferSizeInBytes, **OptixTraversableHandle** \*outputHandle)

See [optixAccelCompact\(\)](#).

**5.18.2.4** **OptixResult**( \* **OptixFunctionTable::optixAccelComputeMemoryUsage**)(OptixDeviceContext context, const **OptixAccelBuildOptions** \*accelOptions, const **OptixBuildInput** \*buildInputs, unsigned int numBuildInputs, **OptixAccelBufferSizes** \*bufferSizes)

See [optixAccelComputeMemoryUsage\(\)](#).

**5.18.2.5** **OptixResult**( \* **OptixFunctionTable::optixAccelGetRelocationInfo**)(OptixDeviceContext context, **OptixTraversableHandle** handle, **OptixAccelRelocationInfo** \*info)

See [optixAccelGetRelocationInfo\(\)](#).

**5.18.2.6** **OptixResult( \* OptixFunctionTable::optixAccelRelocate)(OptixDeviceContext context, CUstream stream, const OptixAccelRelocationInfo \*info, CUdeviceptr instanceTraversableHandles, size\_t numInstanceTraversableHandles, CUdeviceptr targetAccel, size\_t targetAccelSizeInBytes, OptixTraversableHandle \*targetHandle)**

See [optixAccelRelocate\(\)](#).

**5.18.2.7** **OptixResult( \* OptixFunctionTable::optixBuiltinISModuleGet)(OptixDeviceContext context, const OptixModuleCompileOptions \*moduleCompileOptions, const OptixPipelineCompileOptions \*pipelineCompileOptions, const OptixBuiltinISOptions \*builtinISOptions, OptixModule \*builtinModule)**

See [optixBuiltinISModuleGet\(\)](#).

**5.18.2.8** **OptixResult( \* OptixFunctionTable::optixConvertPointerToTraversableHandle)(OptixDeviceContext onDevice, CUdeviceptr pointer, OptixTraversableType traversableType, OptixTraversableHandle \*traversableHandle)**

See [optixConvertPointerToTraversableHandle\(\)](#).

**5.18.2.9** **OptixResult( \* OptixFunctionTable::optixDenoiserComputeAverageColor)(OptixDenoiser handle, CUstream stream, const OptixImage2D \*inputImage, CUdeviceptr outputAverageColor, CUdeviceptr scratch, size\_t scratchSizeInBytes)**

See [optixDenoiserComputeAverageColor\(\)](#).

**5.18.2.10** **OptixResult( \* OptixFunctionTable::optixDenoiserComputeIntensity)(OptixDenoiser handle, CUstream stream, const OptixImage2D \*inputImage, CUdeviceptr outputIntensity, CUdeviceptr scratch, size\_t scratchSizeInBytes)**

See [optixDenoiserComputeIntensity\(\)](#).

**5.18.2.11** **OptixResult( \* OptixFunctionTable::optixDenoiserComputeMemoryResources)(const OptixDenoiser handle, unsigned int maximumInputWidth, unsigned int maximumInputHeight, OptixDenoiserSizes \*returnSizes)**

See [optixDenoiserComputeMemoryResources\(\)](#).

**5.18.2.12** **OptixResult( \* OptixFunctionTable::optixDenoiserCreate)(OptixDeviceContext context, OptixDenoiserModelKind modelKind, const OptixDenoiserOptions \*options, OptixDenoiser \*returnHandle)**

See [optixDenoiserCreate\(\)](#).

**5.18.2.13** **OptixResult( \* OptixFunctionTable::optixDenoiserCreateWithUserModel)(OptixDeviceContext context, const void \*data, size\_t dataSizeInBytes, OptixDenoiser \*returnHandle)**

See [optixDenoiserCreateWithUserModel\(\)](#).

**5.18.2.14** **OptixResult( \* OptixFunctionTable::optixDenoiserDestroy)(OptixDenoiser handle)**

See [optixDenoiserDestroy\(\)](#).

**5.18.2.15** **OptixResult( \* OptixFunctionTable::optixDenoiserInvoke)(OptixDenoiser denoiser, CUstream stream, const OptixDenoiserParams \*params, CUdeviceptr denoiserState, size\_t denoiserStateSizeInBytes, const OptixDenoiserGuideLayer \*guideLayer, const OptixDenoiserLayer \*layers, unsigned int numLayers, unsigned int inputOffsetX, unsigned int inputOffsetY, CUdeviceptr scratch, size\_t scratchSizeInBytes)**

See [optixDenoiserInvoke\(\)](#).

**5.18.2.16** **OptixResult( \* OptixFunctionTable::optixDenoiserSetup)(OptixDenoiser denoiser, CUstream stream, unsigned int inputWidth, unsigned int inputHeight, CUdeviceptr state, size\_t stateSizeInBytes, CUdeviceptr scratch, size\_t scratchSizeInBytes)**

See [optixDenoiserSetup\(\)](#).

**5.18.2.17** **OptixResult( \* OptixFunctionTable::optixDeviceContextCreate)(CUcontext fromContext, const OptixDeviceContextOptions \*options, OptixDeviceContext \*context)**

See [optixDeviceContextCreate\(\)](#).

**5.18.2.18** **OptixResult( \* OptixFunctionTable::optixDeviceContextDestroy)(OptixDeviceContext context)**

See [optixDeviceContextDestroy\(\)](#).

**5.18.2.19** **OptixResult( \* OptixFunctionTable::optixDeviceContextGetCacheDatabaseSizes)(OptixDeviceContext context, size\_t \*lowWaterMark, size\_t \*highWaterMark)**

See [optixDeviceContextGetCacheDatabaseSizes\(\)](#).

**5.18.2.20** **OptixResult( \* OptixFunctionTable::optixDeviceContextGetCacheEnabled)(OptixDeviceContext context, int \*enabled)**

See [optixDeviceContextGetCacheEnabled\(\)](#).

**5.18.2.21** **OptixResult( \* OptixFunctionTable::optixDeviceContextGetCacheLocation)(OptixDeviceContext context, char \*location, size\_t locationSize)**

See [optixDeviceContextGetCacheLocation\(\)](#).

**5.18.2.22** **OptixResult( \* OptixFunctionTable::optixDeviceContextGetProperty)(OptixDeviceContext context, OptixDeviceProperty property, void \*value, size\_t sizeInBytes)**

See [optixDeviceContextGetProperty\(\)](#).

**5.18.2.23** **OptixResult( \* OptixFunctionTable::optixDeviceContextSetCacheDatabaseSizes)(OptixDeviceContext context, size\_t lowWaterMark, size\_t highWaterMark)**

See [optixDeviceContextSetCacheDatabaseSizes\(\)](#).

**5.18.2.24** **OptixResult( \* OptixFunctionTable::optixDeviceContextSetCacheEnabled)(OptixDeviceContext context, int enabled)**

See [optixDeviceContextSetCacheEnabled\(\)](#).

**5.18.2.25** **OptixResult( \* OptixFunctionTable::optixDeviceContextSetCacheLocation)(OptixDeviceContext context, const char \*location)**

See [optixDeviceContextSetCacheLocation\(\)](#).

**5.18.2.26** **OptixResult( \* OptixFunctionTable::optixDeviceContextSetLogCallback)(OptixDeviceContext context, OptixLogCallback callbackFunction, void \*callbackData, unsigned int callbackLevel)**

See [optixDeviceContextSetLogCallback\(\)](#).

**5.18.2.27** **const char\*( \* OptixFunctionTable::optixGetErrorName)(OptixResult result)**

See [optixGetErrorName\(\)](#).

**5.18.2.28** **const char\*( \* OptixFunctionTable::optixGetErrorString)(OptixResult result)**

See [optixGetErrorString\(\)](#).

**5.18.2.29** **OptixResult( \* OptixFunctionTable::optixLaunch)(OptixPipeline pipeline, CUstream stream, CUdeviceptr pipelineParams, size\_t pipelineParamsSize, const OptixShaderBindingTable \*sbt, unsigned int width, unsigned int height, unsigned int depth)**

See [optixConvertPointerToTraversableHandle\(\)](#).

**5.18.2.30** **OptixResult( \* OptixFunctionTable::optixModuleCreateFromPTX)(OptixDeviceContext context, const OptixModuleCompileOptions \*moduleCompileOptions, const OptixPipelineCompileOptions \*pipelineCompileOptions, const char \*PTX, size\_t PTXsize, char \*logString, size\_t \*logStringSize, OptixModule \*module)**

See [optixModuleCreateFromPTX\(\)](#).

**5.18.2.31** **OptixResult( \* OptixFunctionTable::optixModuleCreateFromPTXWithTasks)(OptixDeviceContext context, const OptixModuleCompileOptions \*moduleCompileOptions, const OptixPipelineCompileOptions \*pipelineCompileOptions, const char \*PTX, size\_t PTXsize, char \*logString, size\_t \*logStringSize, OptixModule \*module, OptixTask \*firstTask)**

See [optixModuleCreateFromPTXWithTasks\(\)](#).

**5.18.2.32** **OptixResult( \* OptixFunctionTable::optixModuleDestroy)(OptixModule module)**

See [optixModuleDestroy\(\)](#).

**5.18.2.33** **OptixResult( \* OptixFunctionTable::optixModuleGetCompilationState)(OptixModule module, OptixModuleCompileState \*state)**

See [optixModuleGetCompilationState\(\)](#).

**5.18.2.34** **OptixResult( \* OptixFunctionTable::optixPipelineCreate)(OptixDeviceContext context, const OptixPipelineCompileOptions \*pipelineCompileOptions, const OptixPipelineLinkOptions \*pipelineLinkOptions, const OptixProgramGroup \*programGroups, unsigned int numProgramGroups, char \*logString, size\_t \*logStringSize, OptixPipeline \*pipeline)**

See [optixPipelineCreate\(\)](#).

**5.18.2.35** **OptixResult( \* OptixFunctionTable::optixPipelineDestroy)(OptixPipeline pipeline)**

See [optixPipelineDestroy\(\)](#).

**5.18.2.36** **OptixResult( \* OptixFunctionTable::optixPipelineSetStackSize)(OptixPipeline pipeline, unsigned int directCallableStackSizeFromTraversal, unsigned int directCallableStackSizeFromState, unsigned int continuationStackSize, unsigned int maxTraversableGraphDepth)**

See [optixPipelineSetStackSize\(\)](#).

**5.18.2.37** **OptixResult( \* OptixFunctionTable::optixProgramGroupCreate)(OptixDeviceContext context, const OptixProgramGroupDesc \*programDescriptions, unsigned int numProgramGroups, const OptixProgramGroupOptions \*options, char \*logString, size\_t \*logStringSize, OptixProgramGroup \*programGroups)**

See [optixProgramGroupCreate\(\)](#).

**5.18.2.38** **OptixResult( \* OptixFunctionTable::optixProgramGroupDestroy)(OptixProgramGroup programGroup)**

See [optixProgramGroupDestroy\(\)](#).

**5.18.2.39** **OptixResult( \* OptixFunctionTable::optixProgramGroupGetStackSize)(OptixProgramGroup programGroup, OptixStackSizes \*stackSizes)**

See [optixProgramGroupGetStackSize\(\)](#).

**5.18.2.40** **OptixResult( \* OptixFunctionTable::optixSbtRecordPackHeader)(OptixProgramGroup programGroup, void \*sbtRecordHeaderHostPointer)**

See [optixConvertPointerToTraversableHandle\(\)](#).

**5.18.2.41** **OptixResult( \* OptixFunctionTable::optixTaskExecute)(OptixTask task, OptixTask \*additionalTasks, unsigned int maxNumAdditionalTasks, unsigned int \*numAdditionalTasksCreated)**

See [optixTaskExecute\(\)](#).

**5.18.2.42** **void( \* OptixFunctionTable::reserved1)(void)**

See [optixAccelComputeMemoryUsage\(\)](#).

**5.18.2.43** **void( \* OptixFunctionTable::reserved2)(void)**

See [optixAccelComputeMemoryUsage\(\)](#).

## 5.19 OptixImage2D Struct Reference

### Public Attributes

- [CUdeviceptr](#) data
- unsigned int width
- unsigned int height
- unsigned int rowStrideInBytes
- unsigned int pixelStrideInBytes
- [OptixPixelFormat](#) format

### 5.19.1 Detailed Description

Image descriptor used by the denoiser.

See Also

[optixDenoiserInvoke\(\)](#), [optixDenoiserComputeIntensity\(\)](#)

### 5.19.2 Member Data Documentation

#### 5.19.2.1 [CUdeviceptr](#) `OptixImage2D::data`

Pointer to the actual pixel data.

#### 5.19.2.2 [OptixPixelFormat](#) `OptixImage2D::format`

Pixel format.

#### 5.19.2.3 unsigned int `OptixImage2D::height`

Height of the image (in pixels)

#### 5.19.2.4 unsigned int `OptixImage2D::pixelStrideInBytes`

Stride between subsequent pixels of the image (in bytes). For now, only 0 or the value that corresponds to a dense packing of pixels (no gaps) is supported.

#### 5.19.2.5 unsigned int `OptixImage2D::rowStrideInBytes`

Stride between subsequent rows of the image (in bytes).

#### 5.19.2.6 unsigned int `OptixImage2D::width`

Width of the image (in pixels)

## 5.20 OptixInstance Struct Reference

### Public Attributes

- float `transform` [12]
- unsigned int `instanceId`
- unsigned int `sbtOffset`
- unsigned int `visibilityMask`
- unsigned int `flags`
- `OptixTraversableHandle` `traversableHandle`
- unsigned int `pad` [2]

### 5.20.1 Detailed Description

Instances.

See Also

`OptixBuildInputInstanceArray::instances`

### 5.20.2 Member Data Documentation

#### 5.20.2.1 unsigned int `OptixInstance::flags`

Any combination of `OptixInstanceFlags` is allowed.

#### 5.20.2.2 unsigned int `OptixInstance::instanceId`

Application supplied ID. The maximal ID can be queried using `OPTIX_DEVICE_PROPERTY_LIMIT_MAX_INSTANCE_ID`.

#### 5.20.2.3 unsigned int `OptixInstance::pad`[2]

round up to 80-byte, to ensure 16-byte alignment

#### 5.20.2.4 unsigned int `OptixInstance::sbtOffset`

SBT record offset. Will only be used for instances of geometry acceleration structure (GAS) objects. Needs to be set to 0 for instances of instance acceleration structure (IAS) objects. The maximal SBT offset can be queried using `OPTIX_DEVICE_PROPERTY_LIMIT_MAX_INSTANCE_SBT_OFFSET`.

#### 5.20.2.5 float `OptixInstance::transform`[12]

affine object-to-world transformation as 3x4 matrix in row-major layout

#### 5.20.2.6 `OptixTraversableHandle` `OptixInstance::traversableHandle`

Set with an `OptixTraversableHandle`.



### 5.20.2.7 unsigned int OptixInstance::visibilityMask

Visibility mask. If rayMask & instanceMask == 0 the instance is culled. The number of available bits can be queried using OPTIX\_DEVICE\_PROPERTY\_LIMIT\_NUM\_BITS\_INSTANCE\_VISIBILITY\_MASK.

## 5.21 OptixMatrixMotionTransform Struct Reference

### Public Attributes

- [OptixTraversableHandle](#) child
- [OptixMotionOptions](#) motionOptions
- unsigned int pad [3]
- float transform [2][12]

### 5.21.1 Detailed Description

Represents a matrix motion transformation.

The device address of instances of this type must be a multiple of OPTIX\_TRANSFORM\_BYTE\_ALIGNMENT.

This struct, as defined here, handles only N=2 motion keys due to the fixed array length of its transform member. The following example shows how to create instances for an arbitrary number N of motion keys:

```
float matrixData[N][12];
... // setup matrixData

size_t transformSizeInBytes = sizeof( OptixMatrixMotionTransform ) + ( N-2 ) * 12
    * sizeof( float );
OptixMatrixMotionTransform* matrixMoptionTransform = (
    OptixMatrixMotionTransform*) malloc( transformSizeInBytes );
memset( matrixMoptionTransform, 0, transformSizeInBytes );

... // setup other members of matrixMoptionTransform
matrixMoptionTransform->motionOptions.numKeys
memcpy( matrixMoptionTransform->transform, matrixData, N * 12 * sizeof( float ) );

... // copy matrixMoptionTransform to device memory
free( matrixMoptionTransform )
```

See Also

[optixConvertPointerToTraversableHandle\(\)](#)

### 5.21.2 Member Data Documentation

#### 5.21.2.1 OptixTraversableHandle OptixMatrixMotionTransform::child

The traversable that is transformed by this transformation.

#### 5.21.2.2 OptixMotionOptions OptixMatrixMotionTransform::motionOptions

The motion options for this transformation.

#### 5.21.2.3 unsigned int OptixMatrixMotionTransform::pad[3]

Padding to make the transformation 16 byte aligned.

#### 5.21.2.4 float OptixMatrixMotionTransform::transform[2][12]

Affine object-to-world transformation as 3x4 matrix in row-major layout.

## 5.22 OptixModuleCompileBoundValueEntry Struct Reference

### Public Attributes

- size\_t [pipelineParamOffsetInBytes](#)
- size\_t [sizeInBytes](#)
- const void \* [boundValuePtr](#)
- const char \* [annotation](#)

### 5.22.1 Detailed Description

Struct for specifying specializations for pipelineParams as specified in [OptixPipelineCompileOptions::pipelineLaunchParamsVariableName](#).

The bound values are supposed to represent a constant value in the pipelineParams. OptiX will attempt to locate all loads from the pipelineParams and correlate them to the appropriate bound value, but there are cases where OptiX cannot safely or reliably do this. For example if the pointer to the pipelineParams is passed as an argument to a non-inline function or the offset of the load to the pipelineParams cannot be statically determined (e.g. accessed in a loop). No module should rely on the value being specialized in order to work correctly. The values in the pipelineParams specified on optixLaunch should match the bound value. If validation mode is enabled on the context, OptiX will verify that the bound values specified matches the values in pipelineParams specified to optixLaunch.

These values are compiled in to the module as constants. Once the constants are inserted into the code, an optimization pass will be run that will attempt to propagate the constants and remove unreachable code.

If caching is enabled, changes in these values will result in newly compiled modules.

The pipelineParamOffset and sizeInBytes must be within the bounds of the pipelineParams variable. OPTIX\_ERROR\_INVALID\_VALUE will be returned from optixModuleCreateFromPTX otherwise.

If more than one bound value overlaps or the size of a bound value is equal to 0, an `OPTIX_ERROR_INVALID_VALUE` will be returned from `optixModuleCreateFromPTX`.

The same set of bound values do not need to be used for all modules in a pipeline, but overlapping values between modules must have the same value. `OPTIX_ERROR_INVALID_VALUE` will be returned from `optixPipelineCreate` otherwise.

See Also

[OptixModuleCompileOptions](#)

## 5.22.2 Member Data Documentation

**5.22.2.1** `const char*` `OptixModuleCompileBoundValueEntry::annotation`

**5.22.2.2** `const void*` `OptixModuleCompileBoundValueEntry::boundValuePtr`

**5.22.2.3** `size_t` `OptixModuleCompileBoundValueEntry::pipelineParamOffsetInBytes`

**5.22.2.4** `size_t` `OptixModuleCompileBoundValueEntry::sizeInBytes`

## 5.23 OptixModuleCompileOptions Struct Reference

### Public Attributes

- `int` `maxRegisterCount`
- `OptixCompileOptimizationLevel` `optLevel`
- `OptixCompileDebugLevel` `debugLevel`
- `const`  
`OptixModuleCompileBoundValueEntry *` `boundValues`
- `unsigned int` `numBoundValues`
- `unsigned int` `numPayloadTypes`
- `OptixPayloadType *` `payloadTypes`

### 5.23.1 Detailed Description

Compilation options for module.

See Also

[optixModuleCreateFromPTX\(\)](#)

## 5.23.2 Member Data Documentation

**5.23.2.1** `const OptixModuleCompileBoundValueEntry*` `OptixModuleCompileOptions::boundValues`

Ingored if `numBoundValues` is set to 0.

### 5.23.2.2 OptixCompileDebugLevel OptixModuleCompileOptions::debugLevel

Generate debug information.

### 5.23.2.3 int OptixModuleCompileOptions::maxRegisterCount

Maximum number of registers allowed when compiling to SASS. Set to 0 for no explicit limit. May vary within a pipeline.

### 5.23.2.4 unsigned int OptixModuleCompileOptions::numBoundValues

set to 0 if unused

### 5.23.2.5 unsigned int OptixModuleCompileOptions::numPayloadTypes

The number of different payload types available for compilation. Must be zero if [OptixPipelineCompileOptions::numPayloadValues](#) is not zero.

### 5.23.2.6 OptixCompileOptimizationLevel OptixModuleCompileOptions::optLevel

Optimization level. May vary within a pipeline.

### 5.23.2.7 OptixPayloadType\* OptixModuleCompileOptions::payloadTypes

Points to host array of payload type definitions, size must match numPayloadTypes.

## 5.24 OptixMotionOptions Struct Reference

### Public Attributes

- unsigned short [numKeys](#)
- unsigned short [flags](#)
- float [timeBegin](#)
- float [timeEnd](#)

### 5.24.1 Detailed Description

Motion options.

See Also

[OptixAccelBuildOptions::motionOptions](#), [OptixMatrixMotionTransform::motionOptions](#),  
[OptixSRTMotionTransform::motionOptions](#)

### 5.24.2 Member Data Documentation

#### 5.24.2.1 unsigned short OptixMotionOptions::flags

Combinations of [OptixMotionFlags](#).

### 5.24.2.2 unsigned short OptixMotionOptions::numKeys

If numKeys > 1, motion is enabled. timeBegin, timeEnd and flags are all ignored when motion is disabled.

### 5.24.2.3 float OptixMotionOptions::timeBegin

Point in time where motion starts.

### 5.24.2.4 float OptixMotionOptions::timeEnd

Point in time where motion ends.

## 5.25 OptixPayloadType Struct Reference

### Public Attributes

- unsigned int numPayloadValues
- const unsigned int \* payloadSemantics

### 5.25.1 Detailed Description

Specifies a single payload type.

### 5.25.2 Member Data Documentation

#### 5.25.2.1 unsigned int OptixPayloadType::numPayloadValues

The number of 32b words the payload of this type holds.

#### 5.25.2.2 const unsigned int\* OptixPayloadType::payloadSemantics

Points to host array of payload word semantics, size must match numPayloadValues.

## 5.26 OptixPipelineCompileOptions Struct Reference

### Public Attributes

- int usesMotionBlur
- unsigned int traversableGraphFlags
- int numPayloadValues
- int numAttributeValues
- unsigned int exceptionFlags
- const char \* pipelineLaunchParamsVariableName
- unsigned int usesPrimitiveTypeFlags

### 5.26.1 Detailed Description

Compilation options for all modules of a pipeline.

Similar to [OptixModuleCompileOptions](#), but these options here need to be equal for all modules of a pipeline.

See Also

[optixModuleCreateFromPTX\(\)](#), [optixPipelineCreate\(\)](#)

### 5.26.2 Member Data Documentation

#### 5.26.2.1 unsigned int OptixPipelineCompileOptions::exceptionFlags

A bitmask of OptixExceptionFlags indicating which exceptions are enabled.

#### 5.26.2.2 int OptixPipelineCompileOptions::numAttributeValues

How much storage, in 32b words, to make available for the attributes. The minimum number is 2. Values below that will automatically be changed to 2. [2..8].

#### 5.26.2.3 int OptixPipelineCompileOptions::numPayloadValues

How much storage, in 32b words, to make available for the payload, [0..32] Must be zero if numPayloadTypes is not zero.

#### 5.26.2.4 const char\* OptixPipelineCompileOptions::pipelineLaunchParamsVariableName

The name of the pipeline parameter variable. If 0, no pipeline parameter will be available. This will be ignored if the launch param variable was optimized out or was not found in the modules linked to the pipeline.

#### 5.26.2.5 unsigned int OptixPipelineCompileOptions::traversableGraphFlags

Traversable graph bitfield. See [OptixTraversableGraphFlags](#).

#### 5.26.2.6 int OptixPipelineCompileOptions::usesMotionBlur

Boolean value indicating whether motion blur could be used.

#### 5.26.2.7 unsigned int OptixPipelineCompileOptions::usesPrimitiveTypeFlags

Bit field enabling primitive types. See [OptixPrimitiveTypeFlags](#). Setting to zero corresponds to enabling OPTIX\_PRIMITIVE\_TYPE\_FLAGS\_CUSTOM and OPTIX\_PRIMITIVE\_TYPE\_FLAGS\_TRIANGLE.

## 5.27 OptixPipelineLinkOptions Struct Reference

### Public Attributes

- unsigned int [maxTraceDepth](#)

- [OptixCompileDebugLevel debugLevel](#)

### 5.27.1 Detailed Description

Link options for a pipeline.

See Also

[optixPipelineCreate\(\)](#)

### 5.27.2 Member Data Documentation

#### 5.27.2.1 OptixCompileDebugLevel OptixPipelineLinkOptions::debugLevel

Generate debug information.

#### 5.27.2.2 unsigned int OptixPipelineLinkOptions::maxTraceDepth

Maximum trace recursion depth. 0 means a ray generation program can be launched, but can't trace any rays. The maximum allowed value is 31.

## 5.28 OptixProgramGroupCallables Struct Reference

### Public Attributes

- [OptixModule moduleDC](#)
- `const char * entryFunctionNameDC`
- [OptixModule moduleCC](#)
- `const char * entryFunctionNameCC`

### 5.28.1 Detailed Description

Program group representing callables.

Module and entry function name need to be valid for at least one of the two callables.

See Also

[#OptixProgramGroupDesc::callables](#)

### 5.28.2 Member Data Documentation

#### 5.28.2.1 const char\* OptixProgramGroupCallables::entryFunctionNameCC

Entry function name of the continuation callable (CC) program.

#### 5.28.2.2 const char\* OptixProgramGroupCallables::entryFunctionNameDC

Entry function name of the direct callable (DC) program.

### 5.28.2.3 OptixModule OptixProgramGroupCallables::moduleCC

Module holding the continuation callable (CC) program.

### 5.28.2.4 OptixModule OptixProgramGroupCallables::moduleDC

Module holding the direct callable (DC) program.

## 5.29 OptixProgramGroupDesc Struct Reference

### Public Attributes

- [OptixProgramGroupKind](#) kind
- unsigned int flags
- union {
  - [OptixProgramGroupSingleModule](#) raygen
  - [OptixProgramGroupSingleModule](#) miss
  - [OptixProgramGroupSingleModule](#) exception
  - [OptixProgramGroupCallables](#) callables
  - [OptixProgramGroupHitgroup](#) hitgroup
- };

### 5.29.1 Detailed Description

Descriptor for program groups.

### 5.29.2 Member Data Documentation

#### 5.29.2.1 union { ... }

#### 5.29.2.2 OptixProgramGroupCallables OptixProgramGroupDesc::callables

See Also

[OPTIX\\_PROGRAM\\_GROUP\\_KIND\\_CALLABLES](#)

#### 5.29.2.3 OptixProgramGroupSingleModule OptixProgramGroupDesc::exception

See Also

[OPTIX\\_PROGRAM\\_GROUP\\_KIND\\_EXCEPTION](#)

#### 5.29.2.4 unsigned int OptixProgramGroupDesc::flags

See [OptixProgramGroupFlags](#).



**5.29.2.5 OptixProgramGroupHitgroup OptixProgramGroupDesc::hitgroup**

See Also

[OPTIX\\_PROGRAM\\_GROUP\\_KIND\\_HITGROUP](#)

**5.29.2.6 OptixProgramGroupKind OptixProgramGroupDesc::kind**

The kind of program group.

**5.29.2.7 OptixProgramGroupSingleModule OptixProgramGroupDesc::miss**

See Also

[OPTIX\\_PROGRAM\\_GROUP\\_KIND\\_MISS](#)

**5.29.2.8 OptixProgramGroupSingleModule OptixProgramGroupDesc::raygen**

See Also

[OPTIX\\_PROGRAM\\_GROUP\\_KIND\\_RAYGEN](#)

**5.30 OptixProgramGroupHitgroup Struct Reference****Public Attributes**

- [OptixModule moduleCH](#)
- [const char \\* entryFunctionNameCH](#)
- [OptixModule moduleAH](#)
- [const char \\* entryFunctionNameAH](#)
- [OptixModule moduleIS](#)
- [const char \\* entryFunctionNameIS](#)

**5.30.1 Detailed Description**

Program group representing the hitgroup.

For each of the three program types, module and entry function name might both be nullptr.

See Also

[OptixProgramGroupDesc::hitgroup](#)

**5.30.2 Member Data Documentation****5.30.2.1 const char\* OptixProgramGroupHitgroup::entryFunctionNameAH**

Entry function name of the any hit (AH) program.

### 5.30.2.2 `const char* OptixProgramGroupHitgroup::entryFunctionNameCH`

Entry function name of the closest hit (CH) program.

### 5.30.2.3 `const char* OptixProgramGroupHitgroup::entryFunctionNameIS`

Entry function name of the intersection (IS) program.

### 5.30.2.4 `OptixModule OptixProgramGroupHitgroup::moduleAH`

Module holding the any hit (AH) program.

### 5.30.2.5 `OptixModule OptixProgramGroupHitgroup::moduleCH`

Module holding the closest hit (CH) program.

### 5.30.2.6 `OptixModule OptixProgramGroupHitgroup::moduleIS`

Module holding the intersection (IS) program.

## 5.31 OptixProgramGroupOptions Struct Reference

### Public Attributes

- `OptixPayloadType * payloadType`

### 5.31.1 Detailed Description

Program group options.

See Also

[`optixProgramGroupCreate\(\)`](#)

### 5.31.2 Member Data Documentation

#### 5.31.2.1 `OptixPayloadType* OptixProgramGroupOptions::payloadType`

Specifies the payload type of this program group. All programs in the group must support the payload type (Program support for a type is specified by calling.

See Also

[`optixSetPayloadTypes`](#) or otherwise all types specified in [`OptixModuleCompileOptions`](#) are supported). If a program is not available for the requested payload type, [`optixProgramGroupCreate`](#) returns `OPTIX_ERROR_PAYLOAD_TYPE_MISMATCH`. If the `payloadType` is left zero, a unique type is deduced. The payload type can be uniquely deduced if there is exactly one payload type for which all programs in the group are available. If the payload type could not be deduced uniquely [`optixProgramGroupCreate`](#) returns `OPTIX_ERROR_PAYLOAD_TYPE_RESOLUTION_FAILED`.

## 5.32 OptixProgramGroupSingleModule Struct Reference

### Public Attributes

- [OptixModule](#) module
- const char \* [entryFunctionName](#)

#### 5.32.1 Detailed Description

Program group representing a single module.

Used for raygen, miss, and exception programs. In case of raygen and exception programs, module and entry function name need to be valid. For miss programs, module and entry function name might both be nullptr.

See Also

[OptixProgramGroupDesc::raygen](#), [OptixProgramGroupDesc::miss](#),  
[OptixProgramGroupDesc::exception](#)

#### 5.32.2 Member Data Documentation

##### 5.32.2.1 const char\* [OptixProgramGroupSingleModule::entryFunctionName](#)

Entry function name of the single program.

##### 5.32.2.2 [OptixModule](#) [OptixProgramGroupSingleModule::module](#)

Module holding single program.

## 5.33 OptixShaderBindingTable Struct Reference

### Public Attributes

- CUdeviceptr [raygenRecord](#)
- CUdeviceptr [exceptionRecord](#)
- CUdeviceptr [missRecordBase](#)
- unsigned int [missRecordStrideInBytes](#)
- unsigned int [missRecordCount](#)
- CUdeviceptr [hitgroupRecordBase](#)
- unsigned int [hitgroupRecordStrideInBytes](#)
- unsigned int [hitgroupRecordCount](#)
- CUdeviceptr [callablesRecordBase](#)
- unsigned int [callablesRecordStrideInBytes](#)
- unsigned int [callablesRecordCount](#)

### 5.33.1 Detailed Description

Describes the shader binding table (SBT)

See Also

[optixLaunch\(\)](#)

### 5.33.2 Member Data Documentation

#### 5.33.2.1 CUdeviceptr OptixShaderBindingTable::callablesRecordBase

Arrays of SBT records for callable programs. If the base address is not null, the stride and count must not be zero. If the base address is null, then the count needs to zero. The base address and the stride must be a multiple of OPTIX\_SBT\_RECORD\_ALIGNMENT.

#### 5.33.2.2 unsigned int OptixShaderBindingTable::callablesRecordCount

Arrays of SBT records for callable programs. If the base address is not null, the stride and count must not be zero. If the base address is null, then the count needs to zero. The base address and the stride must be a multiple of OPTIX\_SBT\_RECORD\_ALIGNMENT.

#### 5.33.2.3 unsigned int OptixShaderBindingTable::callablesRecordStrideInBytes

Arrays of SBT records for callable programs. If the base address is not null, the stride and count must not be zero. If the base address is null, then the count needs to zero. The base address and the stride must be a multiple of OPTIX\_SBT\_RECORD\_ALIGNMENT.

#### 5.33.2.4 CUdeviceptr OptixShaderBindingTable::exceptionRecord

Device address of the SBT record of the exception program. The address must be a multiple of OPTIX\_SBT\_RECORD\_ALIGNMENT.

#### 5.33.2.5 CUdeviceptr OptixShaderBindingTable::hitgroupRecordBase

Arrays of SBT records for hit groups. The base address and the stride must be a multiple of OPTIX\_SBT\_RECORD\_ALIGNMENT.

#### 5.33.2.6 unsigned int OptixShaderBindingTable::hitgroupRecordCount

Arrays of SBT records for hit groups. The base address and the stride must be a multiple of OPTIX\_SBT\_RECORD\_ALIGNMENT.

#### 5.33.2.7 unsigned int OptixShaderBindingTable::hitgroupRecordStrideInBytes

Arrays of SBT records for hit groups. The base address and the stride must be a multiple of OPTIX\_SBT\_RECORD\_ALIGNMENT.

**5.33.2.8 CUdeviceptr OptixShaderBindingTable::missRecordBase**

Arrays of SBT records for miss programs. The base address and the stride must be a multiple of OPTIX\_SBT\_RECORD\_ALIGNMENT.

**5.33.2.9 unsigned int OptixShaderBindingTable::missRecordCount**

Arrays of SBT records for miss programs. The base address and the stride must be a multiple of OPTIX\_SBT\_RECORD\_ALIGNMENT.

**5.33.2.10 unsigned int OptixShaderBindingTable::missRecordStrideInBytes**

Arrays of SBT records for miss programs. The base address and the stride must be a multiple of OPTIX\_SBT\_RECORD\_ALIGNMENT.

**5.33.2.11 CUdeviceptr OptixShaderBindingTable::raygenRecord**

Device address of the SBT record of the ray gen program to start launch at. The address must be a multiple of OPTIX\_SBT\_RECORD\_ALIGNMENT.

**5.34 OptixSRTData Struct Reference****Public Attributes****Parameters describing the SRT transformation**

- float `sx`
- float `a`
- float `b`
- float `pvx`
- float `sy`
- float `c`
- float `pvy`
- float `sz`
- float `pvz`
- float `qx`
- float `qy`
- float `qz`
- float `qw`
- float `tx`
- float `ty`
- float `tz`

**5.34.1 Detailed Description**

Represents an SRT transformation.

An SRT transformation can represent a smooth rotation with fewer motion keys than a matrix transformation. Each motion key is constructed from elements taken from a matrix S, a quaternion R, and a translation T.

The scaling matrix  $S = \begin{bmatrix} sx & a & b & pvx \\ 0 & sy & c & pvy \\ 0 & 0 & sz & pvz \end{bmatrix}$  defines an affine transformation that can include scale,

shear, and a translation. The translation allows to define the pivot point for the subsequent rotation.

The quaternion  $R = [qx, qy, qz, qw]$  describes a rotation with angular component  $qw = \cos(\theta/2)$  and other components  $[qx, qy, qz] = \sin(\theta/2) * [ax, ay, az]$  where the axis  $[ax, ay, az]$  is normalized.

The translation matrix  $T = \begin{bmatrix} 1 & 0 & 0 & tx \\ 0 & 1 & 0 & ty \\ 0 & 0 & 1 & tz \end{bmatrix}$  defines another translation that is applied after the rotation.

Typically, this translation includes the inverse translation from the matrix  $S$  to reverse the translation for the pivot point for  $R$ .

To obtain the effective transformation at time  $t$ , the elements of the components of  $S$ ,  $R$ , and  $T$  will be interpolated linearly. The components are then multiplied to obtain the combined transformation  $C = T * R * S$ . The transformation  $C$  is the effective object-to-world transformations at time  $t$ , and  $C^{-1}$  is the effective world-to-object transformation at time  $t$ .

See Also

[OptixSRTMotionTransform::srtData](#), [optixConvertPointerToTraversableHandle\(\)](#)

### 5.34.2 Member Data Documentation

5.34.2.1 float OptixSRTData::a

5.34.2.2 float OptixSRTData::b

5.34.2.3 float OptixSRTData::c

5.34.2.4 float OptixSRTData::pvx

5.34.2.5 float OptixSRTData::pvy

5.34.2.6 float OptixSRTData::pvz

5.34.2.7 float OptixSRTData::qw

5.34.2.8 float OptixSRTData::qx

5.34.2.9 float OptixSRTData::qy

5.34.2.10 float OptixSRTData::qz

5.34.2.11 float OptixSRTData::sx

5.34.2.12 float OptixSRTData::sy

5.34.2.13 float OptixSRTData::sz

5.34.2.14 float OptixSRTData::tx

5.34.2.15 float OptixSRTData::ty

5.34.2.16 float OptixSRTData::tz

## 5.35 OptixSRTMotionTransform Struct Reference

### Public Attributes

- [OptixTraversableHandle](#) child
- [OptixMotionOptions](#) motionOptions
- unsigned int pad [3]
- [OptixSRTData](#) srtData [2]

### 5.35.1 Detailed Description

Represents an SRT motion transformation.

The device address of instances of this type must be a multiple of OPTIX\_TRANSFORM\_BYTE\_ALIGNMENT.

This struct, as defined here, handles only N=2 motion keys due to the fixed array length of its `srtData` member. The following example shows how to create instances for an arbitrary number N of motion keys:

```
OptixSRTData srtData[N];
... // setup srtData

size_t transformSizeInBytes = sizeof( OptixSRTMotionTransform ) + ( N-2 ) * sizeof(
    OptixSRTData );
OptixSRTMotionTransform* srtMotionTransform = (
    OptixSRTMotionTransform*) malloc( transformSizeInBytes );
memset( srtMotionTransform, 0, transformSizeInBytes );

... // setup other members of srtMotionTransform
srtMotionTransform->motionOptions.numKeys = N;
memcpy( srtMotionTransform->srtData, srtData, N * sizeof( OptixSRTData ) );

... // copy srtMotionTransform to device memory
free( srtMotionTransform )
```

See Also

[optixConvertPointerToTraversableHandle\(\)](#)

## 5.35.2 Member Data Documentation

### 5.35.2.1 OptixTraversableHandle OptixSRTMotionTransform::child

The traversable transformed by this transformation.

### 5.35.2.2 OptixMotionOptions OptixSRTMotionTransform::motionOptions

The motion options for this transformation.

### 5.35.2.3 unsigned int OptixSRTMotionTransform::pad[3]

Padding to make the SRT data 16 byte aligned.

### 5.35.2.4 OptixSRTData OptixSRTMotionTransform::srtData[2]

The actual SRT data describing the transformation.

## 5.36 OptixStackSize Struct Reference

### Public Attributes

- unsigned int [cssRG](#)



- unsigned int `cssMS`
- unsigned int `cssCH`
- unsigned int `cssAH`
- unsigned int `cssIS`
- unsigned int `cssCC`
- unsigned int `dssDC`

### 5.36.1 Detailed Description

Describes the stack size requirements of a program group.

See Also

`optixProgramGroupGetStackSize()`

### 5.36.2 Member Data Documentation

#### 5.36.2.1 unsigned int `OptixStackSizes::cssAH`

Continuation stack size of AH programs in bytes.

#### 5.36.2.2 unsigned int `OptixStackSizes::cssCC`

Continuation stack size of CC programs in bytes.

#### 5.36.2.3 unsigned int `OptixStackSizes::cssCH`

Continuation stack size of CH programs in bytes.

#### 5.36.2.4 unsigned int `OptixStackSizes::cssIS`

Continuation stack size of IS programs in bytes.

#### 5.36.2.5 unsigned int `OptixStackSizes::cssMS`

Continuation stack size of MS programs in bytes.

#### 5.36.2.6 unsigned int `OptixStackSizes::cssRG`

Continuation stack size of RG programs in bytes.

#### 5.36.2.7 unsigned int `OptixStackSizes::dssDC`

Direct stack size of DC programs in bytes.

## 5.37 OptixStaticTransform Struct Reference

### Public Attributes

- [OptixTraversableHandle](#) child
- unsigned int [pad](#) [2]
- float [transform](#) [12]
- float [invTransform](#) [12]

### 5.37.1 Detailed Description

Static transform.

The device address of instances of this type must be a multiple of OPTIX\_TRANSFORM\_BYTE\_ALIGNMENT.

See Also

[optixConvertPointerToTraversableHandle\(\)](#)

### 5.37.2 Member Data Documentation

#### 5.37.2.1 [OptixTraversableHandle](#) [OptixStaticTransform::child](#)

The traversable transformed by this transformation.

#### 5.37.2.2 [float](#) [OptixStaticTransform::invTransform](#)[12]

Affine world-to-object transformation as 3x4 matrix in row-major layout Must be the inverse of the transform matrix.

#### 5.37.2.3 [unsigned int](#) [OptixStaticTransform::pad](#)[2]

Padding to make the transformations 16 byte aligned.

#### 5.37.2.4 [float](#) [OptixStaticTransform::transform](#)[12]

Affine object-to-world transformation as 3x4 matrix in row-major layout.

## 5.38 OptixUtilDenoiserImageTile Struct Reference

### Public Attributes

- [OptixImage2D](#) input
- [OptixImage2D](#) output
- unsigned int [inputOffsetX](#)
- unsigned int [inputOffsetY](#)

### 5.38.1 Detailed Description

Tile definition.

see [optixUtilDenoiserSplitImage](#)

### 5.38.2 Member Data Documentation

#### 5.38.2.1 OptixImage2D OptixUtilDenoiserImageTile::input

#### 5.38.2.2 unsigned int OptixUtilDenoiserImageTile::inputOffsetX

#### 5.38.2.3 unsigned int OptixUtilDenoiserImageTile::inputOffsetY

#### 5.38.2.4 OptixImage2D OptixUtilDenoiserImageTile::output

## 6 File Documentation

### 6.1 optix.h File Reference

#### Macros

- [#define OPTIX\\_VERSION](#)

#### 6.1.1 Detailed Description

OptiX public API header.

Author

NVIDIA Corporation Includes the host api if compiling host code, includes the cuda api if compiling device code. For the math library routines include `optix_math.h`

#### 6.1.2 Macro Definition Documentation

##### 6.1.2.1 [#define OPTIX\\_VERSION](#)

Value:

```
60801  /* major =  OPTIX_VERSION/10000,          *
        * minor = (OPTIX_VERSION%10000)/100,    *
        * micro =  OPTIX_VERSION%100           */
```

## 6.2 optix\_7\_device.h File Reference

### Functions

- static `__forceinline__`  
`__device__ void optixTrace (OptixTraversableHandle handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, OptixVisibilityMask visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex)`
- static `__forceinline__`  
`__device__ void optixTrace (OptixTraversableHandle handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, OptixVisibilityMask visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex, unsigned int &p0)`
- static `__forceinline__`  
`__device__ void optixTrace (OptixTraversableHandle handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, OptixVisibilityMask visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex, unsigned int &p0, unsigned int &p1)`
- static `__forceinline__`  
`__device__ void optixTrace (OptixTraversableHandle handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, OptixVisibilityMask visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex, unsigned int &p0, unsigned int &p1, unsigned int &p2)`
- static `__forceinline__`  
`__device__ void optixTrace (OptixTraversableHandle handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, OptixVisibilityMask visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex, unsigned int &p0, unsigned int &p1, unsigned int &p2, unsigned int &p3)`
- static `__forceinline__`  
`__device__ void optixTrace (OptixTraversableHandle handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, OptixVisibilityMask visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex, unsigned int &p0, unsigned int &p1, unsigned int &p2, unsigned int &p3, unsigned int &p4)`
- static `__forceinline__`  
`__device__ void optixTrace (OptixTraversableHandle handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, OptixVisibilityMask visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex, unsigned int &p0, unsigned int &p1, unsigned int &p2, unsigned int &p3, unsigned int &p4, unsigned int &p5)`
- static `__forceinline__`  
`__device__ void optixTrace (OptixTraversableHandle handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, OptixVisibilityMask visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex, unsigned int &p0, unsigned int &p1, unsigned int &p2, unsigned int &p3, unsigned int &p4, unsigned int &p5, unsigned int &p6)`
- static `__forceinline__`  
`__device__ void optixTrace (OptixTraversableHandle handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, OptixVisibilityMask visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex, unsigned int &p0,`

```
unsigned int &p1, unsigned int &p2, unsigned int &p3, unsigned int &p4, unsigned int &p5,  
unsigned int &p6, unsigned int &p7)
```

- [illegible]











- static `__forceinline__`  
`__device__ void optixTrace (OptixTraversableHandle handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, OptixVisibilityMask visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex, unsigned int &p0, unsigned int &p1, unsigned int &p2, unsigned int &p3, unsigned int &p4, unsigned int &p5, unsigned int &p6, unsigned int &p7, unsigned int &p8, unsigned int &p9, unsigned int &p10, unsigned int &p11, unsigned int &p12, unsigned int &p13, unsigned int &p14, unsigned int &p15, unsigned int &p16, unsigned int &p17, unsigned int &p18, unsigned int &p19, unsigned int &p20, unsigned int &p21, unsigned int &p22, unsigned int &p23, unsigned int &p24, unsigned int &p25, unsigned int &p26, unsigned int &p27, unsigned int &p28, unsigned int &p29, unsigned int &p30, unsigned int &p31)`
- static `__forceinline__`  
`__device__ void optixTrace (OptixPayloadTypeID type, OptixTraversableHandle handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, OptixVisibilityMask visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex)`
- static `__forceinline__`  
`__device__ void optixTrace (OptixPayloadTypeID type, OptixTraversableHandle handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, OptixVisibilityMask visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex, unsigned int &p0)`
- static `__forceinline__`  
`__device__ void optixTrace (OptixPayloadTypeID type, OptixTraversableHandle handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, OptixVisibilityMask visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex, unsigned int &p0, unsigned int &p1)`
- static `__forceinline__`  
`__device__ void optixTrace (OptixPayloadTypeID type, OptixTraversableHandle handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, OptixVisibilityMask visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex, unsigned int &p0, unsigned int &p1, unsigned int &p2)`
- static `__forceinline__`  
`__device__ void optixTrace (OptixPayloadTypeID type, OptixTraversableHandle handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, OptixVisibilityMask visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex, unsigned int &p0, unsigned int &p1, unsigned int &p2, unsigned int &p3)`
- static `__forceinline__`  
`__device__ void optixTrace (OptixPayloadTypeID type, OptixTraversableHandle handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, OptixVisibilityMask visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex, unsigned int &p0, unsigned int &p1, unsigned int &p2, unsigned int &p3, unsigned int &p4)`
- static `__forceinline__`  
`__device__ void optixTrace (OptixPayloadTypeID type, OptixTraversableHandle handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, OptixVisibilityMask visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex, unsigned int &p0, unsigned int &p1, unsigned int &p2, unsigned int &p3, unsigned int &p4, unsigned int &p5)`



rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, [OptixVisibilityMask](#)  
visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTStride, unsigned int  
missSBTIndex, unsigned int &p0, unsigned int &p1, unsigned int &p2, unsigned int &p3, unsigned  
int &p4, unsigned int &p5, unsigned int &p6, unsigned int &p7, unsigned int &p8, unsigned int  
&p9, unsigned int &p10, unsigned int &p11, unsigned int &p12, unsigned int &p13)

- [illegible]



```
&p9, unsigned int &p10, unsigned int &p11, unsigned int &p12, unsigned int &p13, unsigned int
&p14, unsigned int &p15, unsigned int &p16, unsigned int &p17, unsigned int &p18, unsigned int
&p19, unsigned int &p20, unsigned int &p21, unsigned int &p22, unsigned int &p23, unsigned int
&p24)
```

- [illegible]

missSBTIndex, unsigned int &p0, unsigned int &p1, unsigned int &p2, unsigned int &p3, unsigned int &p4, unsigned int &p5, unsigned int &p6, unsigned int &p7, unsigned int &p8, unsigned int &p9, unsigned int &p10, unsigned int &p11, unsigned int &p12, unsigned int &p13, unsigned int &p14, unsigned int &p15, unsigned int &p16, unsigned int &p17, unsigned int &p18, unsigned int &p19, unsigned int &p20, unsigned int &p21, unsigned int &p22, unsigned int &p23, unsigned int &p24, unsigned int &p25, unsigned int &p26, unsigned int &p27, unsigned int &p28, unsigned int &p29)

- static `__forceinline__`  
`__device__ void optixTrace (OptixPayloadTypeID type, OptixTraversableHandle handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, OptixVisibilityMask visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex, unsigned int &p0, unsigned int &p1, unsigned int &p2, unsigned int &p3, unsigned int &p4, unsigned int &p5, unsigned int &p6, unsigned int &p7, unsigned int &p8, unsigned int &p9, unsigned int &p10, unsigned int &p11, unsigned int &p12, unsigned int &p13, unsigned int &p14, unsigned int &p15, unsigned int &p16, unsigned int &p17, unsigned int &p18, unsigned int &p19, unsigned int &p20, unsigned int &p21, unsigned int &p22, unsigned int &p23, unsigned int &p24, unsigned int &p25, unsigned int &p26, unsigned int &p27, unsigned int &p28, unsigned int &p29, unsigned int &p30)`
- static `__forceinline__`  
`__device__ void optixTrace (OptixPayloadTypeID type, OptixTraversableHandle handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, OptixVisibilityMask visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex, unsigned int &p0, unsigned int &p1, unsigned int &p2, unsigned int &p3, unsigned int &p4, unsigned int &p5, unsigned int &p6, unsigned int &p7, unsigned int &p8, unsigned int &p9, unsigned int &p10, unsigned int &p11, unsigned int &p12, unsigned int &p13, unsigned int &p14, unsigned int &p15, unsigned int &p16, unsigned int &p17, unsigned int &p18, unsigned int &p19, unsigned int &p20, unsigned int &p21, unsigned int &p22, unsigned int &p23, unsigned int &p24, unsigned int &p25, unsigned int &p26, unsigned int &p27, unsigned int &p28, unsigned int &p29, unsigned int &p30, unsigned int &p31)`
- static `__forceinline__`  
`__device__ void optixSetPayload_0 (unsigned int p)`
- static `__forceinline__`  
`__device__ void optixSetPayload_1 (unsigned int p)`
- static `__forceinline__`  
`__device__ void optixSetPayload_2 (unsigned int p)`
- static `__forceinline__`  
`__device__ void optixSetPayload_3 (unsigned int p)`
- static `__forceinline__`  
`__device__ void optixSetPayload_4 (unsigned int p)`
- static `__forceinline__`  
`__device__ void optixSetPayload_5 (unsigned int p)`
- static `__forceinline__`  
`__device__ void optixSetPayload_6 (unsigned int p)`
- static `__forceinline__`  
`__device__ void optixSetPayload_7 (unsigned int p)`
- static `__forceinline__`  
`__device__ void optixSetPayload_8 (unsigned int p)`



- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_9](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_10](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_11](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_12](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_13](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_14](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_15](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_16](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_17](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_18](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_19](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_20](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_21](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_22](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_23](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_24](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_25](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_26](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_27](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_28](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_29](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_30](#) (unsigned int p)

- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_31](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_0](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_1](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_2](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_3](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_4](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_5](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_6](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_7](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_8](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_9](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_10](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_11](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_12](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_13](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_14](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_15](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_16](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_17](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_18](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_19](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_20](#) ()



- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_21](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_22](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_23](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_24](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_25](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_26](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_27](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_28](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_29](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_30](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_31](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayloadTypes](#) (unsigned int typeMask)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixUndefinedValue](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 [optixGetWorldRayOrigin](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 [optixGetWorldRayDirection](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 [optixGetObjectRayOrigin](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 [optixGetObjectRayDirection](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float [optixGetRayTmin](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float [optixGetRayTmax](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float [optixGetRayTime](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetRayFlags](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetRayVisibilityMask](#) ()

- static \_\_forceinline\_\_  
\_\_device\_\_  
OptixTraversableHandle optixGetInstanceTraversableFromIAS (OptixTraversableHandle ias,  
unsigned int instIdx)
- static \_\_forceinline\_\_  
\_\_device\_\_ void optixGetTriangleVertexData (OptixTraversableHandle gas, unsigned int primIdx,  
unsigned int sbtGASIndex, float time, float3 data[3])
- static \_\_forceinline\_\_  
\_\_device\_\_ void optixGetLinearCurveVertexData (OptixTraversableHandle gas, unsigned int  
primIdx, unsigned int sbtGASIndex, float time, float4 data[2])
- static \_\_forceinline\_\_  
\_\_device\_\_ void optixGetQuadraticBSplineVertexData (OptixTraversableHandle gas, unsigned int  
primIdx, unsigned int sbtGASIndex, float time, float4 data[3])
- static \_\_forceinline\_\_  
\_\_device\_\_ void optixGetCubicBSplineVertexData (OptixTraversableHandle gas, unsigned int  
primIdx, unsigned int sbtGASIndex, float time, float4 data[4])
- static \_\_forceinline\_\_  
\_\_device\_\_ void optixGetCatmullRomVertexData (OptixTraversableHandle gas, unsigned int  
primIdx, unsigned int sbtGASIndex, float time, float4 data[4])
- static \_\_forceinline\_\_  
\_\_device\_\_  
OptixTraversableHandle optixGetGASTraversableHandle ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float optixGetGASMotionTimeBegin (OptixTraversableHandle gas)
- static \_\_forceinline\_\_  
\_\_device\_\_ float optixGetGASMotionTimeEnd (OptixTraversableHandle gas)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int optixGetGASMotionStepCount (OptixTraversableHandle gas)
- static \_\_forceinline\_\_  
\_\_device\_\_ void optixGetWorldToObjectTransformMatrix (float m[12])
- static \_\_forceinline\_\_  
\_\_device\_\_ void optixGetObjectToWorldTransformMatrix (float m[12])
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 optixTransformPointFromWorldToObjectSpace (float3 point)
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 optixTransformVectorFromWorldToObjectSpace (float3 vec)
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 optixTransformNormalFromWorldToObjectSpace (float3 normal)
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 optixTransformPointFromObjectToWorldSpace (float3 point)
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 optixTransformVectorFromObjectToWorldSpace (float3 vec)
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 optixTransformNormalFromObjectToWorldSpace (float3 normal)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int optixGetTransformListSize ()

- static \_\_forceinline\_\_  
\_\_device\_\_  
[OptixTraversableHandle](#) [optixGetTransformListHandle](#) (unsigned int index)
- static \_\_forceinline\_\_  
\_\_device\_\_ [OptixTransformType](#) [optixGetTransformTypeFromHandle](#) ([OptixTraversableHandle](#) handle)
- static \_\_forceinline\_\_  
\_\_device\_\_ const  
[OptixStaticTransform](#) \* [optixGetStaticTransformFromHandle](#) ([OptixTraversableHandle](#) handle)
- static \_\_forceinline\_\_  
\_\_device\_\_ const  
[OptixSRTMotionTransform](#) \* [optixGetSRTMotionTransformFromHandle](#) ([OptixTraversableHandle](#) handle)
- static \_\_forceinline\_\_  
\_\_device\_\_ const  
[OptixMatrixMotionTransform](#) \* [optixGetMatrixMotionTransformFromHandle](#) ([OptixTraversableHandle](#) handle)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetInstanceIdFromHandle](#) ([OptixTraversableHandle](#) handle)
- static \_\_forceinline\_\_  
\_\_device\_\_  
[OptixTraversableHandle](#) [optixGetInstanceChildFromHandle](#) ([OptixTraversableHandle](#) handle)
- static \_\_forceinline\_\_  
\_\_device\_\_ const float4 \* [optixGetInstanceTransformFromHandle](#) ([OptixTraversableHandle](#) handle)
- static \_\_forceinline\_\_  
\_\_device\_\_ const float4 \* [optixGetInstanceInverseTransformFromHandle](#) ([OptixTraversableHandle](#) handle)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixReportIntersection](#) (float hitT, unsigned int hitKind)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixReportIntersection](#) (float hitT, unsigned int hitKind, unsigned int a0)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixReportIntersection](#) (float hitT, unsigned int hitKind, unsigned int a0, unsigned int a1)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixReportIntersection](#) (float hitT, unsigned int hitKind, unsigned int a0, unsigned int a1, unsigned int a2)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixReportIntersection](#) (float hitT, unsigned int hitKind, unsigned int a0, unsigned int a1, unsigned int a2, unsigned int a3)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixReportIntersection](#) (float hitT, unsigned int hitKind, unsigned int a0, unsigned int a1, unsigned int a2, unsigned int a3, unsigned int a4)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixReportIntersection](#) (float hitT, unsigned int hitKind, unsigned int a0, unsigned int a1, unsigned int a2, unsigned int a3, unsigned int a4, unsigned int a5)

- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixReportIntersection](#) (float hitT, unsigned int hitKind, unsigned int a0, unsigned int a1, unsigned int a2, unsigned int a3, unsigned int a4, unsigned int a5, unsigned int a6)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixReportIntersection](#) (float hitT, unsigned int hitKind, unsigned int a0, unsigned int a1, unsigned int a2, unsigned int a3, unsigned int a4, unsigned int a5, unsigned int a6, unsigned int a7)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetAttribute\\_0](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetAttribute\\_1](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetAttribute\\_2](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetAttribute\\_3](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetAttribute\\_4](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetAttribute\\_5](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetAttribute\\_6](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetAttribute\\_7](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixTerminateRay](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixIgnoreIntersection](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPrimitiveIndex](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetSbtGASIndex](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetInstanceId](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetInstanceIndex](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetHitKind](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ [OptixPrimitiveType](#) [optixGetPrimitiveType](#) (unsigned int hitKind)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixIsFrontFaceHit](#) (unsigned int hitKind)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixIsBackFaceHit](#) (unsigned int hitKind)

- static \_\_forceinline\_\_  
\_\_device\_\_ [OptixPrimitiveType](#) [optixGetPrimitiveType](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixIsFrontFaceHit](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixIsBackFaceHit](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixIsTriangleHit](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixIsTriangleFrontFaceHit](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixIsTriangleBackFaceHit](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float2 [optixGetTriangleBarycentrics](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float [optixGetCurveParameter](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ uint3 [optixGetLaunchIndex](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ uint3 [optixGetLaunchDimensions](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ CUdeviceptr [optixGetSbtDataPointer](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode, unsigned int exceptionDetail0)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode, unsigned int exceptionDetail0,  
unsigned int exceptionDetail1)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode, unsigned int exceptionDetail0,  
unsigned int exceptionDetail1, unsigned int exceptionDetail2)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode, unsigned int exceptionDetail0,  
unsigned int exceptionDetail1, unsigned int exceptionDetail2, unsigned int exceptionDetail3)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode, unsigned int exceptionDetail0,  
unsigned int exceptionDetail1, unsigned int exceptionDetail2, unsigned int exceptionDetail3,  
unsigned int exceptionDetail4)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode, unsigned int exceptionDetail0,  
unsigned int exceptionDetail1, unsigned int exceptionDetail2, unsigned int exceptionDetail3,  
unsigned int exceptionDetail4, unsigned int exceptionDetail5)

- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode, unsigned int exceptionDetail0, unsigned int exceptionDetail1, unsigned int exceptionDetail2, unsigned int exceptionDetail3, unsigned int exceptionDetail4, unsigned int exceptionDetail5, unsigned int exceptionDetail6)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode, unsigned int exceptionDetail0, unsigned int exceptionDetail1, unsigned int exceptionDetail2, unsigned int exceptionDetail3, unsigned int exceptionDetail4, unsigned int exceptionDetail5, unsigned int exceptionDetail6, unsigned int exceptionDetail7)
- static \_\_forceinline\_\_  
\_\_device\_\_ int [optixGetExceptionCode](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetExceptionDetail\\_0](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetExceptionDetail\\_1](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetExceptionDetail\\_2](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetExceptionDetail\\_3](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetExceptionDetail\\_4](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetExceptionDetail\\_5](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetExceptionDetail\\_6](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetExceptionDetail\\_7](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_  
[OptixTraversableHandle](#) [optixGetExceptionInvalidTraversable](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ int [optixGetExceptionInvalidSbtOffset](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_  
[OptixInvalidRayExceptionDetails](#) [optixGetExceptionInvalidRay](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_  
[OptixParameterMismatchExceptionDetails](#) [optixGetExceptionParameterMismatch](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ char \* [optixGetExceptionLineInfo](#) ()
- template<typename ReturnT , typename... ArgTypes>  
static \_\_forceinline\_\_  
\_\_device\_\_ ReturnT [optixDirectCall](#) (unsigned int sbtIndex, ArgTypes...args)
- template<typename ReturnT , typename... ArgTypes>  
static \_\_forceinline\_\_  
\_\_device\_\_ ReturnT [optixContinuationCall](#) (unsigned int sbtIndex, ArgTypes...args)

- static `__forceinline__`  
`__device__ uint4 optixTexFootprint2D` (unsigned long long tex, unsigned int texInfo, float x, float y, unsigned int \*singleMipLevel)
- static `__forceinline__`  
`__device__ uint4 optixTexFootprint2DLod` (unsigned long long tex, unsigned int texInfo, float x, float y, float level, bool coarse, unsigned int \*singleMipLevel)
- static `__forceinline__`  
`__device__ uint4 optixTexFootprint2DGrad` (unsigned long long tex, unsigned int texInfo, float x, float y, float dPdx\_x, float dPdx\_y, float dPdy\_x, float dPdy\_y, bool coarse, unsigned int \*singleMipLevel)

### 6.2.1 Detailed Description

OptiX public API header.

Author

NVIDIA Corporation OptiX public API Reference - Device API declarations

## 6.3 optix\_7\_device\_impl.h File Reference

### Macros

- `#define OPTIX_DEFINE_optixGetAttribute_BODY(which)`
- `#define OPTIX_DEFINE_optixGetExceptionDetail_BODY(which)`

### Functions

- static `__forceinline__`  
`__device__ void optixTrace` (`OptixTraversableHandle` handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, `OptixVisibilityMask` visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex)
- static `__forceinline__`  
`__device__ void optixTrace` (`OptixTraversableHandle` handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, `OptixVisibilityMask` visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex, unsigned int &p0)
- static `__forceinline__`  
`__device__ void optixTrace` (`OptixTraversableHandle` handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, `OptixVisibilityMask` visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex, unsigned int &p0, unsigned int &p1)
- static `__forceinline__`  
`__device__ void optixTrace` (`OptixTraversableHandle` handle, float3 rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, `OptixVisibilityMask` visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex, unsigned int &p0, unsigned int &p1, unsigned int &p2)





```
float tmin, float tmax, float rayTime, OptixVisibilityMask visibilityMask, unsigned int rayFlags,
unsigned int SBTOffset, unsigned int SBTstride, unsigned int missSBTIndex, unsigned int &p0,
unsigned int &p1, unsigned int &p2, unsigned int &p3, unsigned int &p4, unsigned int &p5,
unsigned int &p6, unsigned int &p7, unsigned int &p8, unsigned int &p9, unsigned int &p10,
unsigned int &p11)
```

- [illegible]



```
unsigned int &p1, unsigned int &p2, unsigned int &p3, unsigned int &p4, unsigned int &p5,  
unsigned int &p6, unsigned int &p7, unsigned int &p8, unsigned int &p9, unsigned int &p10,  
unsigned int &p11, unsigned int &p12, unsigned int &p13, unsigned int &p14, unsigned int &p15,  
unsigned int &p16, unsigned int &p17, unsigned int &p18, unsigned int &p19, unsigned int &p20,  
unsigned int &p21, unsigned int &p22, unsigned int &p23)
```

- [illegible]



rayOrigin, float3 rayDirection, float tmin, float tmax, float rayTime, [OptixVisibilityMask](#)  
visibilityMask, unsigned int rayFlags, unsigned int SBTOffset, unsigned int SBTstride, unsigned int  
missSBTIndex, unsigned int &p0, unsigned int &p1)

- [illegible]





- [illegible]





```
&p14, unsigned int &p15, unsigned int &p16, unsigned int &p17, unsigned int &p18, unsigned int
&p19, unsigned int &p20, unsigned int &p21, unsigned int &p22, unsigned int &p23, unsigned int
&p24, unsigned int &p25, unsigned int &p26)
```

- [illegible]

visibilityMask, unsigned int rayFlags, unsigned int SBToffset, unsigned int SBTstride, unsigned int missSBTIndex, unsigned int &p0, unsigned int &p1, unsigned int &p2, unsigned int &p3, unsigned int &p4, unsigned int &p5, unsigned int &p6, unsigned int &p7, unsigned int &p8, unsigned int &p9, unsigned int &p10, unsigned int &p11, unsigned int &p12, unsigned int &p13, unsigned int &p14, unsigned int &p15, unsigned int &p16, unsigned int &p17, unsigned int &p18, unsigned int &p19, unsigned int &p20, unsigned int &p21, unsigned int &p22, unsigned int &p23, unsigned int &p24, unsigned int &p25, unsigned int &p26, unsigned int &p27, unsigned int &p28, unsigned int &p29, unsigned int &p30, unsigned int &p31)

- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_0](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_1](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_2](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_3](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_4](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_5](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_6](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_7](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_8](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_9](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_10](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_11](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_12](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_13](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_14](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_15](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_16](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_17](#) (unsigned int p)

- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_18](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_19](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_20](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_21](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_22](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_23](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_24](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_25](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_26](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_27](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_28](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_29](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_30](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayload\\_31](#) (unsigned int p)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_0](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_1](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_2](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_3](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_4](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_5](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_6](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_7](#) ()

- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_8 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_9 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_10 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_11 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_12 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_13 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_14 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_15 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_16 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_17 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_18 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_19 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_20 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_21 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_22 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_23 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_24 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_25 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_26 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_27 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_28 \(\)](#)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_29 \(\)](#)

- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_30](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPayload\\_31](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixSetPayloadTypes](#) (unsigned int types)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixUndefinedValue](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 [optixGetWorldRayOrigin](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 [optixGetWorldRayDirection](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 [optixGetObjectRayOrigin](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 [optixGetObjectRayDirection](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float [optixGetRayTmin](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float [optixGetRayTmax](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float [optixGetRayTime](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetRayFlags](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetRayVisibilityMask](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_  
[OptixTraversableHandle](#) [optixGetInstanceTraversableFromIAS](#) ([OptixTraversableHandle](#) ias,  
unsigned int instIdx)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixGetTriangleVertexData](#) ([OptixTraversableHandle](#) gas, unsigned int primIdx,  
unsigned int sbtGASIndex, float time, float3 data[3])
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixGetLinearCurveVertexData](#) ([OptixTraversableHandle](#) gas, unsigned int  
primIdx, unsigned int sbtGASIndex, float time, float4 data[2])
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixGetQuadraticBSplineVertexData](#) ([OptixTraversableHandle](#) gas, unsigned int  
primIdx, unsigned int sbtGASIndex, float time, float4 data[3])
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixGetCubicBSplineVertexData](#) ([OptixTraversableHandle](#) gas, unsigned int  
primIdx, unsigned int sbtGASIndex, float time, float4 data[4])
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixGetCatmullRomVertexData](#) ([OptixTraversableHandle](#) gas, unsigned int  
primIdx, unsigned int sbtGASIndex, float time, float4 data[4])

- static \_\_forceinline\_\_  
\_\_device\_\_  
OptixTraversableHandle optixGetGASTraversableHandle ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float optixGetGASMotionTimeBegin (OptixTraversableHandle handle)
- static \_\_forceinline\_\_  
\_\_device\_\_ float optixGetGASMotionTimeEnd (OptixTraversableHandle handle)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int optixGetGASMotionStepCount (OptixTraversableHandle handle)
- static \_\_forceinline\_\_  
\_\_device\_\_ void optixGetWorldToObjectTransformMatrix (float m[12])
- static \_\_forceinline\_\_  
\_\_device\_\_ void optixGetObjectToWorldTransformMatrix (float m[12])
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 optixTransformPointFromWorldToObjectSpace (float3 point)
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 optixTransformVectorFromWorldToObjectSpace (float3 vec)
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 optixTransformNormalFromWorldToObjectSpace (float3 normal)
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 optixTransformPointFromObjectToWorldSpace (float3 point)
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 optixTransformVectorFromObjectToWorldSpace (float3 vec)
- static \_\_forceinline\_\_  
\_\_device\_\_ float3 optixTransformNormalFromObjectToWorldSpace (float3 normal)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int optixGetTransformListSize ()
- static \_\_forceinline\_\_  
\_\_device\_\_  
OptixTraversableHandle optixGetTransformListHandle (unsigned int index)
- static \_\_forceinline\_\_  
\_\_device\_\_ OptixTransformType optixGetTransformTypeFromHandle (OptixTraversableHandle handle)
- static \_\_forceinline\_\_  
\_\_device\_\_ const  
OptixStaticTransform \* optixGetStaticTransformFromHandle (OptixTraversableHandle handle)
- static \_\_forceinline\_\_  
\_\_device\_\_ const  
OptixSRTMotionTransform \* optixGetSRTMotionTransformFromHandle (OptixTraversableHandle handle)
- static \_\_forceinline\_\_  
\_\_device\_\_ const  
OptixMatrixMotionTransform \* optixGetMatrixMotionTransformFromHandle (OptixTraversableHandle handle)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int optixGetInstanceIdFromHandle (OptixTraversableHandle handle)

- static \_\_forceinline\_\_  
\_\_device\_\_  
[OptixTraversableHandle](#) [optixGetInstanceChildFromHandle](#) ([OptixTraversableHandle](#) handle)
- static \_\_forceinline\_\_  
\_\_device\_\_ const float4 \* [optixGetInstanceTransformFromHandle](#) ([OptixTraversableHandle](#) handle)
- static \_\_forceinline\_\_  
\_\_device\_\_ const float4 \* [optixGetInstanceInverseTransformFromHandle](#) ([OptixTraversableHandle](#) handle)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixReportIntersection](#) (float hitT, unsigned int hitKind)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixReportIntersection](#) (float hitT, unsigned int hitKind, unsigned int a0)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixReportIntersection](#) (float hitT, unsigned int hitKind, unsigned int a0, unsigned int a1)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixReportIntersection](#) (float hitT, unsigned int hitKind, unsigned int a0, unsigned int a1, unsigned int a2)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixReportIntersection](#) (float hitT, unsigned int hitKind, unsigned int a0, unsigned int a1, unsigned int a2, unsigned int a3)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixReportIntersection](#) (float hitT, unsigned int hitKind, unsigned int a0, unsigned int a1, unsigned int a2, unsigned int a3, unsigned int a4)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixReportIntersection](#) (float hitT, unsigned int hitKind, unsigned int a0, unsigned int a1, unsigned int a2, unsigned int a3, unsigned int a4, unsigned int a5)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixReportIntersection](#) (float hitT, unsigned int hitKind, unsigned int a0, unsigned int a1, unsigned int a2, unsigned int a3, unsigned int a4, unsigned int a5, unsigned int a6)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixReportIntersection](#) (float hitT, unsigned int hitKind, unsigned int a0, unsigned int a1, unsigned int a2, unsigned int a3, unsigned int a4, unsigned int a5, unsigned int a6, unsigned int a7)
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetAttribute\\_0](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetAttribute\\_1](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetAttribute\\_2](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetAttribute\\_3](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetAttribute\\_4](#) ()

- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetAttribute\\_5](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetAttribute\\_6](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetAttribute\\_7](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixTerminateRay](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixIgnoreIntersection](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetPrimitiveIndex](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetSbtGASIndex](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetInstanceId](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetInstanceIndex](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetHitKind](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ [OptixPrimitiveType](#) [optixGetPrimitiveType](#) (unsigned int hitKind)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixIsBackFaceHit](#) (unsigned int hitKind)
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixIsFrontFaceHit](#) (unsigned int hitKind)
- static \_\_forceinline\_\_  
\_\_device\_\_ [OptixPrimitiveType](#) [optixGetPrimitiveType](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixIsBackFaceHit](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixIsFrontFaceHit](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixIsTriangleHit](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixIsTriangleFrontFaceHit](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ bool [optixIsTriangleBackFaceHit](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float [optixGetCurveParameter](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ float2 [optixGetTriangleBarycentrics](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ uint3 [optixGetLaunchIndex](#) ()



- static \_\_forceinline\_\_  
\_\_device\_\_ uint3 [optixGetLaunchDimensions](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ CUdeviceptr [optixGetSbtDataPointer](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode, unsigned int exceptionDetail0)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode, unsigned int exceptionDetail0,  
unsigned int exceptionDetail1)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode, unsigned int exceptionDetail0,  
unsigned int exceptionDetail1, unsigned int exceptionDetail2)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode, unsigned int exceptionDetail0,  
unsigned int exceptionDetail1, unsigned int exceptionDetail2, unsigned int exceptionDetail3)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode, unsigned int exceptionDetail0,  
unsigned int exceptionDetail1, unsigned int exceptionDetail2, unsigned int exceptionDetail3,  
unsigned int exceptionDetail4)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode, unsigned int exceptionDetail0,  
unsigned int exceptionDetail1, unsigned int exceptionDetail2, unsigned int exceptionDetail3,  
unsigned int exceptionDetail4, unsigned int exceptionDetail5)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode, unsigned int exceptionDetail0,  
unsigned int exceptionDetail1, unsigned int exceptionDetail2, unsigned int exceptionDetail3,  
unsigned int exceptionDetail4, unsigned int exceptionDetail5, unsigned int exceptionDetail6)
- static \_\_forceinline\_\_  
\_\_device\_\_ void [optixThrowException](#) (int exceptionCode, unsigned int exceptionDetail0,  
unsigned int exceptionDetail1, unsigned int exceptionDetail2, unsigned int exceptionDetail3,  
unsigned int exceptionDetail4, unsigned int exceptionDetail5, unsigned int exceptionDetail6,  
unsigned int exceptionDetail7)
- static \_\_forceinline\_\_  
\_\_device\_\_ int [optixGetExceptionCode](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetExceptionDetail\\_0](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetExceptionDetail\\_1](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetExceptionDetail\\_2](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetExceptionDetail\\_3](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetExceptionDetail\\_4](#) ()

- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetExceptionDetail\\_5](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetExceptionDetail\\_6](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ unsigned int [optixGetExceptionDetail\\_7](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_  
[OptixTraversableHandle](#) [optixGetExceptionInvalidTraversable](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ int [optixGetExceptionInvalidSbtOffset](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_  
[OptixInvalidRayExceptionDetails](#) [optixGetExceptionInvalidRay](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_  
[OptixParameterMismatchExceptionDetails](#) [optixGetExceptionParameterMismatch](#) ()
- static \_\_forceinline\_\_  
\_\_device\_\_ char \* [optixGetExceptionLineInfo](#) ()
- template<typename ReturnT, typename... ArgTypes>  
static \_\_forceinline\_\_  
\_\_device\_\_ ReturnT [optixDirectCall](#) (unsigned int sbtIndex, ArgTypes...args)
- template<typename ReturnT, typename... ArgTypes>  
static \_\_forceinline\_\_  
\_\_device\_\_ ReturnT [optixContinuationCall](#) (unsigned int sbtIndex, ArgTypes...args)
- static \_\_forceinline\_\_  
\_\_device\_\_ uint4 [optixTexFootprint2D](#) (unsigned long long tex, unsigned int texInfo, float x, float y,  
unsigned int \*singleMipLevel)
- static \_\_forceinline\_\_  
\_\_device\_\_ uint4 [optixTexFootprint2DGrad](#) (unsigned long long tex, unsigned int texInfo, float x,  
float y, float dPdx\_x, float dPdx\_y, float dPdy\_x, float dPdy\_y, bool coarse, unsigned int  
\*singleMipLevel)
- static \_\_forceinline\_\_  
\_\_device\_\_ uint4 [optixTexFootprint2DLod](#) (unsigned long long tex, unsigned int texInfo, float x,  
float y, float level, bool coarse, unsigned int \*singleMipLevel)

### 6.3.1 Detailed Description

OptiX public API.

Author

NVIDIA Corporation OptiX public API Reference - Device side implementation

### 6.3.2 Macro Definition Documentation

#### 6.3.2.1 #define OPTIX\_DEFINE\_optixGetAttribute\_BODY(

***which* )**

**Value:**

```
unsigned int ret;
    \
    asm( "call (%0), _optix_get_attribute_" #which ", ();" : "=r"( ret ) : );
    \
    return ret;
```

### 6.3.2.2 #define OPTIX\_DEFINE\_optixGetExceptionDetail\_BODY( ***which* )**

**Value:**

```
unsigned int ret;
    \
    asm( "call (%0), _optix_get_exception_detail_" #which ", ();" : "=r"( ret ) : );
    \
    return ret;
```

## 6.3.3 Function Documentation

**6.3.3.1** `template<typename ReturnT , typename... ArgTypes> static __forceinline__  
__device__ ReturnT optixContinuationCall (  
    unsigned int sbtIndex,  
    ArgTypes... args ) [static]`

**6.3.3.2** `template<typename ReturnT , typename... ArgTypes> static __forceinline__  
__device__ ReturnT optixDirectCall (  
    unsigned int sbtIndex,`

**ArgTypes... *args* ) [static]**

**6.3.3.3 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetAttribute\_0 ( ) [static]**

**6.3.3.4 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetAttribute\_1 ( ) [static]**

**6.3.3.5 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetAttribute\_2 ( ) [static]**

**6.3.3.6 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetAttribute\_3 ( ) [static]**

**6.3.3.7 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetAttribute\_4 ( ) [static]**

**6.3.3.8 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetAttribute\_5 ( ) [static]**

**6.3.3.9 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetAttribute\_6 ( ) [static]**

**6.3.3.10 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetAttribute\_7 ( ) [static]**

**6.3.3.11 static \_\_forceinline\_\_ \_\_device\_\_ void optixGetCatmullRomVertexData (**  
     **OptixTraversableHandle *gas*,**  
     **unsigned int *primIdx*,**  
     **unsigned int *sbtGASIndex*,**  
     **float *time*,**  
     **float4 *data*[4] ) [static]**

**6.3.3.12 static \_\_forceinline\_\_ \_\_device\_\_ void optixGetCubicBSplineVertexData (**  
     **OptixTraversableHandle *gas*,**  
     **unsigned int *primIdx*,**  
     **unsigned int *sbtGASIndex*,**  
     **float *time*,**

**float4 *data*[4] ) [static]**

**6.3.3.13 static \_\_forceinline\_\_ \_\_device\_\_ float optixGetCurveParameter ( ) [static]**

**6.3.3.14 static \_\_forceinline\_\_ \_\_device\_\_ int optixGetExceptionCode ( ) [static]**

**6.3.3.15 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetExceptionDetail\_0 ( ) [static]**

**6.3.3.16 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetExceptionDetail\_1 ( ) [static]**

**6.3.3.17 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetExceptionDetail\_2 ( ) [static]**

**6.3.3.18 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetExceptionDetail\_3 ( ) [static]**

**6.3.3.19 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetExceptionDetail\_4 ( ) [static]**

**6.3.3.20 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetExceptionDetail\_5 ( ) [static]**

**6.3.3.21 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetExceptionDetail\_6 ( ) [static]**

**6.3.3.22 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetExceptionDetail\_7 ( ) [static]**

**6.3.3.23 static \_\_forceinline\_\_ \_\_device\_\_ OptixInvalidRayExceptionDetails  
optixGetExceptionInvalidRay ( ) [static]**

**6.3.3.24 static \_\_forceinline\_\_ \_\_device\_\_ int optixGetExceptionInvalidSbtOffset ( ) [static]**

**6.3.3.25 static \_\_forceinline\_\_ \_\_device\_\_ OptixTraversableHandle  
optixGetExceptionInvalidTraversable ( ) [static]**

**6.3.3.26 static \_\_forceinline\_\_ \_\_device\_\_ char\* optixGetExceptionLineInfo ( ) [static]**

**6.3.3.27 static \_\_forceinline\_\_ \_\_device\_\_ OptixParameterMismatchExceptionDetails  
optixGetExceptionParameterMismatch ( ) [static]**

**6.3.3.28 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetGASMotionStepCount (   
OptixTraversableHandle *handle* ) [static]**

**6.3.3.29 static \_\_forceinline\_\_ \_\_device\_\_ float optixGetGASMotionTimeBegin (**

- OptixTraversableHandle *handle* ) [static]**
- 6.3.3.30 static \_\_forceinline\_\_ \_\_device\_\_ float optixGetGASMotionTimeEnd (**  
**OptixTraversableHandle *handle* ) [static]**
- 6.3.3.31 static \_\_forceinline\_\_ \_\_device\_\_ OptixTraversableHandle**  
**optixGetGASTraversableHandle ( ) [static]**
- 6.3.3.32 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetHitKind ( ) [static]**
- 6.3.3.33 static \_\_forceinline\_\_ \_\_device\_\_ OptixTraversableHandle**  
**optixGetInstanceChildFromHandle (**  
**OptixTraversableHandle *handle* ) [static]**
- 6.3.3.34 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetInstanceId ( ) [static]**
- 6.3.3.35 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetInstanceIdFromHandle (**  
**OptixTraversableHandle *handle* ) [static]**
- 6.3.3.36 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetInstanceIndex ( ) [static]**
- 6.3.3.37 static \_\_forceinline\_\_ \_\_device\_\_ const float4\* optixGetInstanceInverseTransform-**  
**FromHandle (**  
**OptixTraversableHandle *handle* ) [static]**
- 6.3.3.38 static \_\_forceinline\_\_ \_\_device\_\_ const float4\* optixGetInstanceTransformFromHandle**  
**(**  
**OptixTraversableHandle *handle* ) [static]**
- 6.3.3.39 static \_\_forceinline\_\_ \_\_device\_\_ OptixTraversableHandle**  
**optixGetInstanceTraversableFromIAS (**  
**OptixTraversableHandle *ias*,**  
**unsigned int *instIdx* ) [static]**
- 6.3.3.40 static \_\_forceinline\_\_ \_\_device\_\_ uint3 optixGetLaunchDimensions ( ) [static]**
- 6.3.3.41 static \_\_forceinline\_\_ \_\_device\_\_ uint3 optixGetLaunchIndex ( ) [static]**
- 6.3.3.42 static \_\_forceinline\_\_ \_\_device\_\_ void optixGetLinearCurveVertexData (**  
**OptixTraversableHandle *gas*,**  
**unsigned int *primIdx*,**  
**unsigned int *sbtGASIndex*,**  
**float *time*,**  
**float4 *data*[2] ) [static]**
- 6.3.3.43 static \_\_forceinline\_\_ \_\_device\_\_ const OptixMatrixMotionTransform\***  
**optixGetMatrixMotionTransformFromHandle (**

**OptixTraversableHandle *handle* ) [static]**

**6.3.3.44 static \_\_forceinline\_\_ \_\_device\_\_ float3 optixGetObjectRayDirection ( ) [static]**

**6.3.3.45 static \_\_forceinline\_\_ \_\_device\_\_ float3 optixGetObjectRayOrigin ( ) [static]**

**6.3.3.46 static \_\_forceinline\_\_ \_\_device\_\_ void optixGetObjectToWorldTransformMatrix (**

**float *m*[12] ) [static]**

- 6.3.3.47 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_0 ( ) [static]**
- 6.3.3.48 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_1 ( ) [static]**
- 6.3.3.49 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_10 ( ) [static]**
- 6.3.3.50 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_11 ( ) [static]**
- 6.3.3.51 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_12 ( ) [static]**
- 6.3.3.52 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_13 ( ) [static]**
- 6.3.3.53 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_14 ( ) [static]**
- 6.3.3.54 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_15 ( ) [static]**
- 6.3.3.55 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_16 ( ) [static]**
- 6.3.3.56 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_17 ( ) [static]**
- 6.3.3.57 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_18 ( ) [static]**
- 6.3.3.58 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_19 ( ) [static]**
- 6.3.3.59 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_2 ( ) [static]**
- 6.3.3.60 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_20 ( ) [static]**
- 6.3.3.61 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_21 ( ) [static]**
- 6.3.3.62 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_22 ( ) [static]**
- 6.3.3.63 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_23 ( ) [static]**
- 6.3.3.64 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_24 ( ) [static]**
- 6.3.3.65 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_25 ( ) [static]**
- 6.3.3.66 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_26 ( ) [static]**
- 6.3.3.67 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_27 ( ) [static]**
- 6.3.3.68 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_28 ( ) [static]**
- 6.3.3.69 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_29 ( ) [static]**
- 6.3.3.70 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_3 ( ) [static]**
- 6.3.3.71 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_30 ( ) [static]**
- 6.3.3.72 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_31 ( ) [static]**
- 6.3.3.73 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetPayload\_4 ( ) [static]**



**unsigned int *hitKind* ) [static]**

**6.3.3.81 static \_\_forceinline\_\_ \_\_device\_\_ OptixPrimitiveType optixGetPrimitiveType ( ) [static]**

**6.3.3.82 static \_\_forceinline\_\_ \_\_device\_\_ void optixGetQuadraticBSplineVertexData (  
OptixTraversableHandle *gas*,  
unsigned int *primIdx*,  
unsigned int *sbtGASIndex*,  
float *time*,  
float4 *data*[3] ) [static]**

**6.3.3.83 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetRayFlags ( ) [static]**

**6.3.3.84 static \_\_forceinline\_\_ \_\_device\_\_ float optixGetRayTime ( ) [static]**

**6.3.3.85 static \_\_forceinline\_\_ \_\_device\_\_ float optixGetRayTmax ( ) [static]**

**6.3.3.86 static \_\_forceinline\_\_ \_\_device\_\_ float optixGetRayTmin ( ) [static]**

**6.3.3.87 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetRayVisibilityMask ( ) [static]**

**6.3.3.88 static \_\_forceinline\_\_ \_\_device\_\_ CUdeviceptr optixGetSbtDataPointer ( ) [static]**

**6.3.3.89 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetSbtGASIndex ( ) [static]**

**6.3.3.90 static \_\_forceinline\_\_ \_\_device\_\_ const OptixSRTMotionTransform\*  
optixGetSRTMotionTransformFromHandle (  
OptixTraversableHandle *handle* ) [static]**

**6.3.3.91 static \_\_forceinline\_\_ \_\_device\_\_ const OptixStaticTransform\*  
optixGetStaticTransformFromHandle (  
OptixTraversableHandle *handle* ) [static]**

**6.3.3.92 static \_\_forceinline\_\_ \_\_device\_\_ OptixTraversableHandle  
optixGetTransformListHandle (  
unsigned int *index* ) [static]**

**6.3.3.93 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixGetTransformListSize ( ) [static]**

**6.3.3.94 static \_\_forceinline\_\_ \_\_device\_\_ OptixTransformType optixGetTransformType-  
FromHandle (**

**OptixTraversableHandle *handle* ) [static]**

**6.3.3.95 static \_\_forceinline\_\_ \_\_device\_\_ float2 optixGetTriangleBarycentrics ( ) [static]**

**6.3.3.96 static \_\_forceinline\_\_ \_\_device\_\_ void optixGetTriangleVertexData (**  
**OptixTraversableHandle *gas*,**  
**unsigned int *primIdx*,**  
**unsigned int *sbtGASIndex*,**  
**float *time*,**  
**float3 *data*[3] ) [static]**

**6.3.3.97 static \_\_forceinline\_\_ \_\_device\_\_ float3 optixGetWorldRayDirection ( ) [static]**

**6.3.3.98 static \_\_forceinline\_\_ \_\_device\_\_ float3 optixGetWorldRayOrigin ( ) [static]**

**6.3.3.99 static \_\_forceinline\_\_ \_\_device\_\_ void optixGetWorldToObjectTransformMatrix (**  
**float *m*[12] ) [static]**

**6.3.3.100 static \_\_forceinline\_\_ \_\_device\_\_ void optixIgnoreIntersection ( ) [static]**

**6.3.3.101 static \_\_forceinline\_\_ \_\_device\_\_ bool optixIsBackFaceHit (**  
**unsigned int *hitKind* ) [static]**

**6.3.3.102 static \_\_forceinline\_\_ \_\_device\_\_ bool optixIsBackFaceHit ( ) [static]**

**6.3.3.103 static \_\_forceinline\_\_ \_\_device\_\_ bool optixIsFrontFaceHit (**  
**unsigned int *hitKind* ) [static]**

**6.3.3.104 static \_\_forceinline\_\_ \_\_device\_\_ bool optixIsFrontFaceHit ( ) [static]**

**6.3.3.105 static \_\_forceinline\_\_ \_\_device\_\_ bool optixIsTriangleBackFaceHit ( ) [static]**

**6.3.3.106 static \_\_forceinline\_\_ \_\_device\_\_ bool optixIsTriangleFrontFaceHit ( ) [static]**

**6.3.3.107 static \_\_forceinline\_\_ \_\_device\_\_ bool optixIsTriangleHit ( ) [static]**

**6.3.3.108 static \_\_forceinline\_\_ \_\_device\_\_ bool optixReportIntersection (**  
**float *hitT*,**  
**unsigned int *hitKind* ) [static]**

**6.3.3.109 static \_\_forceinline\_\_ \_\_device\_\_ bool optixReportIntersection (**  
**float *hitT*,**  
**unsigned int *hitKind*,**  
**unsigned int *a0* ) [static]**

**6.3.3.110 static \_\_forceinline\_\_ \_\_device\_\_ bool optixReportIntersection (**  
**float *hitT*,**

```
    unsigned int hitKind,  
    unsigned int a0,  
    unsigned int a1 ) [static]
```

```
6.3.3.111 static __forceinline__ __device__ bool optixReportIntersection (  
    float hitT,  
    unsigned int hitKind,  
    unsigned int a0,  
    unsigned int a1,  
    unsigned int a2 ) [static]
```

```
6.3.3.112 static __forceinline__ __device__ bool optixReportIntersection (  
    float hitT,  
    unsigned int hitKind,  
    unsigned int a0,  
    unsigned int a1,  
    unsigned int a2,  
    unsigned int a3 ) [static]
```

```
6.3.3.113 static __forceinline__ __device__ bool optixReportIntersection (  
    float hitT,  
    unsigned int hitKind,  
    unsigned int a0,  
    unsigned int a1,  
    unsigned int a2,  
    unsigned int a3,  
    unsigned int a4 ) [static]
```

```
6.3.3.114 static __forceinline__ __device__ bool optixReportIntersection (  
    float hitT,  
    unsigned int hitKind,  
    unsigned int a0,  
    unsigned int a1,  
    unsigned int a2,  
    unsigned int a3,  
    unsigned int a4,  
    unsigned int a5 ) [static]
```

```
6.3.3.115 static __forceinline__ __device__ bool optixReportIntersection (  
    float hitT,  
    unsigned int hitKind,  
    unsigned int a0,  
    unsigned int a1,
```

```

    unsigned int a2,
    unsigned int a3,
    unsigned int a4,
    unsigned int a5,
    unsigned int a6 ) [static]

```

```

6.3.3.116 static __forceinline__ __device__ bool optixReportIntersection (
    float hitT,
    unsigned int hitKind,
    unsigned int a0,
    unsigned int a1,
    unsigned int a2,
    unsigned int a3,
    unsigned int a4,
    unsigned int a5,
    unsigned int a6,
    unsigned int a7 ) [static]

```

```

6.3.3.117 static __forceinline__ __device__ void optixSetPayload_0 (
    unsigned int p ) [static]

```

```

6.3.3.118 static __forceinline__ __device__ void optixSetPayload_1 (
    unsigned int p ) [static]

```

```

6.3.3.119 static __forceinline__ __device__ void optixSetPayload_10 (
    unsigned int p ) [static]

```

```

6.3.3.120 static __forceinline__ __device__ void optixSetPayload_11 (
    unsigned int p ) [static]

```

```

6.3.3.121 static __forceinline__ __device__ void optixSetPayload_12 (
    unsigned int p ) [static]

```

```

6.3.3.122 static __forceinline__ __device__ void optixSetPayload_13 (
    unsigned int p ) [static]

```

```

6.3.3.123 static __forceinline__ __device__ void optixSetPayload_14 (
    unsigned int p ) [static]

```

```

6.3.3.124 static __forceinline__ __device__ void optixSetPayload_15 (
    unsigned int p ) [static]

```

```

6.3.3.125 static __forceinline__ __device__ void optixSetPayload_16 (
    unsigned int p ) [static]

```

```

6.3.3.126 static __forceinline__ __device__ void optixSetPayload_17 (

```

**unsigned int *p* ) [static]**

**6.3.3.127 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_18 (**  
**unsigned int *p* ) [static]**

**6.3.3.128 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_19 (**  
**unsigned int *p* ) [static]**

**6.3.3.129 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_2 (**  
**unsigned int *p* ) [static]**

**6.3.3.130 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_20 (**  
**unsigned int *p* ) [static]**

**6.3.3.131 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_21 (**  
**unsigned int *p* ) [static]**

**6.3.3.132 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_22 (**  
**unsigned int *p* ) [static]**

**6.3.3.133 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_23 (**  
**unsigned int *p* ) [static]**

**6.3.3.134 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_24 (**  
**unsigned int *p* ) [static]**

**6.3.3.135 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_25 (**  
**unsigned int *p* ) [static]**

**6.3.3.136 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_26 (**  
**unsigned int *p* ) [static]**

**6.3.3.137 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_27 (**  
**unsigned int *p* ) [static]**

**6.3.3.138 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_28 (**  
**unsigned int *p* ) [static]**

**6.3.3.139 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_29 (**  
**unsigned int *p* ) [static]**

**6.3.3.140 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_3 (**  
**unsigned int *p* ) [static]**

**6.3.3.141 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_30 (**

- unsigned int *p* ) [static]
- 6.3.3.142 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_31 ( unsigned int *p* ) [static]
- 6.3.3.143 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_4 ( unsigned int *p* ) [static]
- 6.3.3.144 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_5 ( unsigned int *p* ) [static]
- 6.3.3.145 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_6 ( unsigned int *p* ) [static]
- 6.3.3.146 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_7 ( unsigned int *p* ) [static]
- 6.3.3.147 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_8 ( unsigned int *p* ) [static]
- 6.3.3.148 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayload\_9 ( unsigned int *p* ) [static]
- 6.3.3.149 static \_\_forceinline\_\_ \_\_device\_\_ void optixSetPayloadTypes ( unsigned int *types* ) [static]
- 6.3.3.150 static \_\_forceinline\_\_ \_\_device\_\_ void optixTerminateRay ( ) [static]
- 6.3.3.151 static \_\_forceinline\_\_ \_\_device\_\_ uint4 optixTexFootprint2D ( unsigned long long *tex*, unsigned int *texInfo*, float *x*, float *y*, unsigned int \* *singleMipLevel* ) [static]
- 6.3.3.152 static \_\_forceinline\_\_ \_\_device\_\_ uint4 optixTexFootprint2DGrad ( unsigned long long *tex*, unsigned int *texInfo*, float *x*, float *y*, float *dPdx\_x*, float *dPdx\_y*, float *dPdy\_x*, float *dPdy\_y*, bool *coarse*,

unsigned int \* *singleMipLevel* ) [static]

6.3.3.153 static \_\_forceinline\_\_ \_\_device\_\_ uint4 optixTexFootprint2DLod (   
 unsigned long long *tex*,   
 unsigned int *texInfo*,   
 float *x*,   
 float *y*,   
 float *level*,   
 bool *coarse*,   
 unsigned int \* *singleMipLevel* ) [static]

6.3.3.154 static \_\_forceinline\_\_ \_\_device\_\_ void optixThrowException (   
 int *exceptionCode* ) [static]

6.3.3.155 static \_\_forceinline\_\_ \_\_device\_\_ void optixThrowException (   
 int *exceptionCode*,   
 unsigned int *exceptionDetail0* ) [static]

6.3.3.156 static \_\_forceinline\_\_ \_\_device\_\_ void optixThrowException (   
 int *exceptionCode*,   
 unsigned int *exceptionDetail0*,   
 unsigned int *exceptionDetail1* ) [static]

6.3.3.157 static \_\_forceinline\_\_ \_\_device\_\_ void optixThrowException (   
 int *exceptionCode*,   
 unsigned int *exceptionDetail0*,   
 unsigned int *exceptionDetail1*,   
 unsigned int *exceptionDetail2* ) [static]

6.3.3.158 static \_\_forceinline\_\_ \_\_device\_\_ void optixThrowException (   
 int *exceptionCode*,   
 unsigned int *exceptionDetail0*,   
 unsigned int *exceptionDetail1*,   
 unsigned int *exceptionDetail2*,   
 unsigned int *exceptionDetail3* ) [static]

6.3.3.159 static \_\_forceinline\_\_ \_\_device\_\_ void optixThrowException (   
 int *exceptionCode*,   
 unsigned int *exceptionDetail0*,   
 unsigned int *exceptionDetail1*,   
 unsigned int *exceptionDetail2*,   
 unsigned int *exceptionDetail3*,   
 unsigned int *exceptionDetail4* ) [static]

6.3.3.160 static \_\_forceinline\_\_ \_\_device\_\_ void optixThrowException (

```

    int exceptionCode,
    unsigned int exceptionDetail0,
    unsigned int exceptionDetail1,
    unsigned int exceptionDetail2,
    unsigned int exceptionDetail3,
    unsigned int exceptionDetail4,
    unsigned int exceptionDetail5 ) [static]

```

```

6.3.3.161 static __forceinline__ __device__ void optixThrowException (
    int exceptionCode,
    unsigned int exceptionDetail0,
    unsigned int exceptionDetail1,
    unsigned int exceptionDetail2,
    unsigned int exceptionDetail3,
    unsigned int exceptionDetail4,
    unsigned int exceptionDetail5,
    unsigned int exceptionDetail6 ) [static]

```

```

6.3.3.162 static __forceinline__ __device__ void optixThrowException (
    int exceptionCode,
    unsigned int exceptionDetail0,
    unsigned int exceptionDetail1,
    unsigned int exceptionDetail2,
    unsigned int exceptionDetail3,
    unsigned int exceptionDetail4,
    unsigned int exceptionDetail5,
    unsigned int exceptionDetail6,
    unsigned int exceptionDetail7 ) [static]

```

```

6.3.3.163 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex ) [static]

```

```

6.3.3.164 static __forceinline__ __device__ void optixTrace (

```



```

    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0 ) [static]

```

```

6.3.3.165 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1 ) [static]

```

```

6.3.3.166 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,

```

unsigned int & *p2* ) [static]

**6.3.3.167** static \_\_forceinline\_\_ \_\_device\_\_ void optixTrace (  
 OptixTraversableHandle *handle*,  
 float3 *rayOrigin*,  
 float3 *rayDirection*,  
 float *tmin*,  
 float *tmax*,  
 float *rayTime*,  
 OptixVisibilityMask *visibilityMask*,  
 unsigned int *rayFlags*,  
 unsigned int *SBTOffset*,  
 unsigned int *SBTstride*,  
 unsigned int *missSBTIndex*,  
 unsigned int & *p0*,  
 unsigned int & *p1*,  
 unsigned int & *p2*,  
 unsigned int & *p3* ) [static]

**6.3.3.168** static \_\_forceinline\_\_ \_\_device\_\_ void optixTrace (  
 OptixTraversableHandle *handle*,  
 float3 *rayOrigin*,  
 float3 *rayDirection*,  
 float *tmin*,  
 float *tmax*,  
 float *rayTime*,  
 OptixVisibilityMask *visibilityMask*,  
 unsigned int *rayFlags*,  
 unsigned int *SBTOffset*,  
 unsigned int *SBTstride*,  
 unsigned int *missSBTIndex*,  
 unsigned int & *p0*,  
 unsigned int & *p1*,  
 unsigned int & *p2*,  
 unsigned int & *p3*,  
 unsigned int & *p4* ) [static]

**6.3.3.169** static \_\_forceinline\_\_ \_\_device\_\_ void optixTrace (  
 OptixTraversableHandle *handle*,  
 float3 *rayOrigin*,  
 float3 *rayDirection*,  
 float *tmin*,  
 float *tmax*,

```

float rayTime,
OptixVisibilityMask visibilityMask,
unsigned int rayFlags,
unsigned int SBToffset,
unsigned int SBTstride,
unsigned int missSBTIndex,
unsigned int & p0,
unsigned int & p1,
unsigned int & p2,
unsigned int & p3,
unsigned int & p4,
unsigned int & p5 ) [static]

```

```

6.3.3.170 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBToffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6 ) [static]

```

```

6.3.3.171 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,

```

```

    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7 ) [static]

```

```

6.3.3.172 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8 ) [static]

```

```

6.3.3.173 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,

```

```

    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9 ) [static]

```

```

6.3.3.174 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10 ) [static]

```

```

6.3.3.175 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,

```

```

float3 rayDirection,
float tmin,
float tmax,
float rayTime,
OptixVisibilityMask visibilityMask,
unsigned int rayFlags,
unsigned int SBTOffset,
unsigned int SBTstride,
unsigned int missSBTIndex,
unsigned int & p0,
unsigned int & p1,
unsigned int & p2,
unsigned int & p3,
unsigned int & p4,
unsigned int & p5,
unsigned int & p6,
unsigned int & p7,
unsigned int & p8,
unsigned int & p9,
unsigned int & p10,
unsigned int & p11 ) [static]

```

```

6.3.3.176 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,

```

```

    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12 ) [static]

```

```

6.3.3.177 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13 ) [static]

```

```

6.3.3.178 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,

```

```

    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14 ) [static]

```

```

6.3.3.179 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,

```



```

    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15 ) [static]

```

```

6.3.3.180 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16 ) [static]

```

```

6.3.3.181 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,

```

```

float rayTime,
OptixVisibilityMask visibilityMask,
unsigned int rayFlags,
unsigned int SBToffset,
unsigned int SBTstride,
unsigned int missSBTIndex,
unsigned int & p0,
unsigned int & p1,
unsigned int & p2,
unsigned int & p3,
unsigned int & p4,
unsigned int & p5,
unsigned int & p6,
unsigned int & p7,
unsigned int & p8,
unsigned int & p9,
unsigned int & p10,
unsigned int & p11,
unsigned int & p12,
unsigned int & p13,
unsigned int & p14,
unsigned int & p15,
unsigned int & p16,
unsigned int & p17 ) [static]

```

```

6.3.3.182 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBToffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,

```

```

    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18 ) [static]

```

```

6.3.3.183 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,

```

```

    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19 ) [static]

```

```

6.3.3.184 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20 ) [static]

```

```

6.3.3.185 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,

```

```

float3 rayOrigin,
float3 rayDirection,
float tmin,
float tmax,
float rayTime,
OptixVisibilityMask visibilityMask,
unsigned int rayFlags,
unsigned int SBTOffset,
unsigned int SBTstride,
unsigned int missSBTIndex,
unsigned int & p0,
unsigned int & p1,
unsigned int & p2,
unsigned int & p3,
unsigned int & p4,
unsigned int & p5,
unsigned int & p6,
unsigned int & p7,
unsigned int & p8,
unsigned int & p9,
unsigned int & p10,
unsigned int & p11,
unsigned int & p12,
unsigned int & p13,
unsigned int & p14,
unsigned int & p15,
unsigned int & p16,
unsigned int & p17,
unsigned int & p18,
unsigned int & p19,
unsigned int & p20,
unsigned int & p21 ) [static]

```

```

6.3.3.186 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,

```

```

    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22 ) [static]

```

```

6.3.3.187 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,

```

```

    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23 ) [static]

```

```

6.3.3.188 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,

```

```

    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23,
    unsigned int & p24 ) [static]

```

```

6.3.3.189 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,

```



```

    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23,
    unsigned int & p24,
    unsigned int & p25 ) [static]

```

```

6.3.3.190 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,

```

```

    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23,
    unsigned int & p24,
    unsigned int & p25,
    unsigned int & p26 ) [static]

```

```

6.3.3.191 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,

```

```

    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23,
    unsigned int & p24,
    unsigned int & p25,
    unsigned int & p26,
    unsigned int & p27 ) [static]

```

```

6.3.3.192 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,

```

```

    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23,
    unsigned int & p24,
    unsigned int & p25,
    unsigned int & p26,
    unsigned int & p27,
    unsigned int & p28 ) [static]

```

```

6.3.3.193 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,

```

```

    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23,
    unsigned int & p24,
    unsigned int & p25,
    unsigned int & p26,
    unsigned int & p27,
    unsigned int & p28,
    unsigned int & p29 ) [static]

```

```

6.3.3.194 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,

```

```

    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23,
    unsigned int & p24,
    unsigned int & p25,
    unsigned int & p26,
    unsigned int & p27,
    unsigned int & p28,
    unsigned int & p29,
    unsigned int & p30 ) [static]

```

```

6.3.3.195 static __forceinline__ __device__ void optixTrace (
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,

```

```

    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23,
    unsigned int & p24,
    unsigned int & p25,
    unsigned int & p26,
    unsigned int & p27,
    unsigned int & p28,
    unsigned int & p29,
    unsigned int & p30,
    unsigned int & p31 ) [static]

```

```

6.3.3.196 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex ) [static]

```

```

6.3.3.197 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,

```

```

    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0 ) [static]

```

```

6.3.3.198 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1 ) [static]

```

```

6.3.3.199 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2 ) [static]

```

```

6.3.3.200 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,

```



```

float tmin,
float tmax,
float rayTime,
OptixVisibilityMask visibilityMask,
unsigned int rayFlags,
unsigned int SBTOffset,
unsigned int SBTstride,
unsigned int missSBTIndex,
unsigned int & p0,
unsigned int & p1,
unsigned int & p2,
unsigned int & p3 ) [static]

```

6.3.3.201 static \_\_forceinline\_\_ \_\_device\_\_ void optixTrace (

```

    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4 ) [static]

```

6.3.3.202 static \_\_forceinline\_\_ \_\_device\_\_ void optixTrace (

```

    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,

```

```

    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5 ) [static]

```

```

6.3.3.203 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6 ) [static]

```

```

6.3.3.204 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,

```

```

    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7 ) [static]

```

```

6.3.3.205 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBToffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8 ) [static]

```

```

6.3.3.206 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,

```

```

OptixVisibilityMask visibilityMask,
unsigned int rayFlags,
unsigned int SBTOffset,
unsigned int SBTstride,
unsigned int missSBTIndex,
unsigned int & p0,
unsigned int & p1,
unsigned int & p2,
unsigned int & p3,
unsigned int & p4,
unsigned int & p5,
unsigned int & p6,
unsigned int & p7,
unsigned int & p8,
unsigned int & p9 ) [static]

```

```

6.3.3.207 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10 ) [static]

```

```

6.3.3.208 static __forceinline__ __device__ void optixTrace (

```

**OptixPayloadTypeID** *type*,  
**OptixTraversableHandle** *handle*,  
**float3** *rayOrigin*,  
**float3** *rayDirection*,  
**float** *tmin*,  
**float** *tmax*,  
**float** *rayTime*,  
**OptixVisibilityMask** *visibilityMask*,  
**unsigned int** *rayFlags*,  
**unsigned int** *SBTOffset*,  
**unsigned int** *SBTstride*,  
**unsigned int** *missSBTIndex*,  
**unsigned int** & *p0*,  
**unsigned int** & *p1*,  
**unsigned int** & *p2*,  
**unsigned int** & *p3*,  
**unsigned int** & *p4*,  
**unsigned int** & *p5*,  
**unsigned int** & *p6*,  
**unsigned int** & *p7*,  
**unsigned int** & *p8*,  
**unsigned int** & *p9*,  
**unsigned int** & *p10*,  
**unsigned int** & *p11* ) [static]

**6.3.3.209 static \_\_forceinline\_\_ \_\_device\_\_ void optixTrace (**  
**OptixPayloadTypeID** *type*,  
**OptixTraversableHandle** *handle*,  
**float3** *rayOrigin*,  
**float3** *rayDirection*,  
**float** *tmin*,  
**float** *tmax*,  
**float** *rayTime*,  
**OptixVisibilityMask** *visibilityMask*,  
**unsigned int** *rayFlags*,  
**unsigned int** *SBTOffset*,  
**unsigned int** *SBTstride*,  
**unsigned int** *missSBTIndex*,  
**unsigned int** & *p0*,  
**unsigned int** & *p1*,  
**unsigned int** & *p2*,  
**unsigned int** & *p3*,

```

    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12 ) [static]

```

```

6.3.3.210 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13 ) [static]

```

```

6.3.3.211 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,

```

```

float3 rayDirection,
float tmin,
float tmax,
float rayTime,
OptixVisibilityMask visibilityMask,
unsigned int rayFlags,
unsigned int SBToffset,
unsigned int SBTstride,
unsigned int missSBTIndex,
unsigned int & p0,
unsigned int & p1,
unsigned int & p2,
unsigned int & p3,
unsigned int & p4,
unsigned int & p5,
unsigned int & p6,
unsigned int & p7,
unsigned int & p8,
unsigned int & p9,
unsigned int & p10,
unsigned int & p11,
unsigned int & p12,
unsigned int & p13,
unsigned int & p14 ) [static]

```

```

6.3.3.212 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBToffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,

```

```

    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15 ) [static]

```

```

6.3.3.213 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,

```



unsigned int & *p16* ) [static]

6.3.3.214 static \_\_forceinline\_\_ \_\_device\_\_ void optixTrace (  
 OptixPayloadTypeID *type*,  
 OptixTraversableHandle *handle*,  
 float3 *rayOrigin*,  
 float3 *rayDirection*,  
 float *tmin*,  
 float *tmax*,  
 float *rayTime*,  
 OptixVisibilityMask *visibilityMask*,  
 unsigned int *rayFlags*,  
 unsigned int *SBTOffset*,  
 unsigned int *SBTstride*,  
 unsigned int *missSBTIndex*,  
 unsigned int & *p0*,  
 unsigned int & *p1*,  
 unsigned int & *p2*,  
 unsigned int & *p3*,  
 unsigned int & *p4*,  
 unsigned int & *p5*,  
 unsigned int & *p6*,  
 unsigned int & *p7*,  
 unsigned int & *p8*,  
 unsigned int & *p9*,  
 unsigned int & *p10*,  
 unsigned int & *p11*,  
 unsigned int & *p12*,  
 unsigned int & *p13*,  
 unsigned int & *p14*,  
 unsigned int & *p15*,  
 unsigned int & *p16*,  
 unsigned int & *p17* ) [static]

6.3.3.215 static \_\_forceinline\_\_ \_\_device\_\_ void optixTrace (  
 OptixPayloadTypeID *type*,  
 OptixTraversableHandle *handle*,  
 float3 *rayOrigin*,  
 float3 *rayDirection*,  
 float *tmin*,  
 float *tmax*,  
 float *rayTime*,

```

OptixVisibilityMask visibilityMask,
unsigned int rayFlags,
unsigned int SBTOffset,
unsigned int SBTstride,
unsigned int missSBTIndex,
unsigned int & p0,
unsigned int & p1,
unsigned int & p2,
unsigned int & p3,
unsigned int & p4,
unsigned int & p5,
unsigned int & p6,
unsigned int & p7,
unsigned int & p8,
unsigned int & p9,
unsigned int & p10,
unsigned int & p11,
unsigned int & p12,
unsigned int & p13,
unsigned int & p14,
unsigned int & p15,
unsigned int & p16,
unsigned int & p17,
unsigned int & p18 ) [static]

```

```

6.3.3.216 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,

```

```

    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19 ) [static]

```

```

6.3.3.217 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,

```

```

    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20 ) [static]

```

```

6.3.3.218 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,

```

```

    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21 ) [static]

```

```

6.3.3.219 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22 ) [static]

```

```

6.3.3.220 static __forceinline__ __device__ void optixTrace (

```

**OptixPayloadTypeID** *type*,  
**OptixTraversableHandle** *handle*,  
**float3** *rayOrigin*,  
**float3** *rayDirection*,  
**float** *tmin*,  
**float** *tmax*,  
**float** *rayTime*,  
**OptixVisibilityMask** *visibilityMask*,  
**unsigned int** *rayFlags*,  
**unsigned int** *SBTOffset*,  
**unsigned int** *SBTstride*,  
**unsigned int** *missSBTIndex*,  
**unsigned int** & *p0*,  
**unsigned int** & *p1*,  
**unsigned int** & *p2*,  
**unsigned int** & *p3*,  
**unsigned int** & *p4*,  
**unsigned int** & *p5*,  
**unsigned int** & *p6*,  
**unsigned int** & *p7*,  
**unsigned int** & *p8*,  
**unsigned int** & *p9*,  
**unsigned int** & *p10*,  
**unsigned int** & *p11*,  
**unsigned int** & *p12*,  
**unsigned int** & *p13*,  
**unsigned int** & *p14*,  
**unsigned int** & *p15*,  
**unsigned int** & *p16*,  
**unsigned int** & *p17*,  
**unsigned int** & *p18*,  
**unsigned int** & *p19*,  
**unsigned int** & *p20*,  
**unsigned int** & *p21*,  
**unsigned int** & *p22*,  
**unsigned int** & *p23* ) [static]

**6.3.3.221** **static \_\_forceinline\_\_ \_\_device\_\_ void** **optixTrace** (  
**OptixPayloadTypeID** *type*,  
**OptixTraversableHandle** *handle*,  
**float3** *rayOrigin*,  
**float3** *rayDirection*,

```

float tmin,
float tmax,
float rayTime,
OptixVisibilityMask visibilityMask,
unsigned int rayFlags,
unsigned int SBTOffset,
unsigned int SBTstride,
unsigned int missSBTIndex,
unsigned int & p0,
unsigned int & p1,
unsigned int & p2,
unsigned int & p3,
unsigned int & p4,
unsigned int & p5,
unsigned int & p6,
unsigned int & p7,
unsigned int & p8,
unsigned int & p9,
unsigned int & p10,
unsigned int & p11,
unsigned int & p12,
unsigned int & p13,
unsigned int & p14,
unsigned int & p15,
unsigned int & p16,
unsigned int & p17,
unsigned int & p18,
unsigned int & p19,
unsigned int & p20,
unsigned int & p21,
unsigned int & p22,
unsigned int & p23,
unsigned int & p24 ) [static]

```

```

6.3.3.222 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,

```

```

OptixVisibilityMask visibilityMask,
unsigned int rayFlags,
unsigned int SBTOffset,
unsigned int SBTstride,
unsigned int missSBTIndex,
unsigned int & p0,
unsigned int & p1,
unsigned int & p2,
unsigned int & p3,
unsigned int & p4,
unsigned int & p5,
unsigned int & p6,
unsigned int & p7,
unsigned int & p8,
unsigned int & p9,
unsigned int & p10,
unsigned int & p11,
unsigned int & p12,
unsigned int & p13,
unsigned int & p14,
unsigned int & p15,
unsigned int & p16,
unsigned int & p17,
unsigned int & p18,
unsigned int & p19,
unsigned int & p20,
unsigned int & p21,
unsigned int & p22,
unsigned int & p23,
unsigned int & p24,
unsigned int & p25 ) [static]

```

```

6.3.3.223 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,

```



```

    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23,
    unsigned int & p24,
    unsigned int & p25,
    unsigned int & p26 ) [static]

```

```

6.3.3.224 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,

```

```

    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23,
    unsigned int & p24,
    unsigned int & p25,
    unsigned int & p26,
    unsigned int & p27 ) [static]

```

```

6.3.3.225 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,

```

```

    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23,
    unsigned int & p24,
    unsigned int & p25,
    unsigned int & p26,
    unsigned int & p27,
    unsigned int & p28 ) [static]

```

```

6.3.3.226 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,
    OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,

```

```

    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23,
    unsigned int & p24,
    unsigned int & p25,
    unsigned int & p26,
    unsigned int & p27,
    unsigned int & p28,
    unsigned int & p29 ) [static]

```

```

6.3.3.227 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,
    float tmin,
    float tmax,
    float rayTime,

```

```

OptixVisibilityMask visibilityMask,
    unsigned int rayFlags,
    unsigned int SBTOffset,
    unsigned int SBTstride,
    unsigned int missSBTIndex,
    unsigned int & p0,
    unsigned int & p1,
    unsigned int & p2,
    unsigned int & p3,
    unsigned int & p4,
    unsigned int & p5,
    unsigned int & p6,
    unsigned int & p7,
    unsigned int & p8,
    unsigned int & p9,
    unsigned int & p10,
    unsigned int & p11,
    unsigned int & p12,
    unsigned int & p13,
    unsigned int & p14,
    unsigned int & p15,
    unsigned int & p16,
    unsigned int & p17,
    unsigned int & p18,
    unsigned int & p19,
    unsigned int & p20,
    unsigned int & p21,
    unsigned int & p22,
    unsigned int & p23,
    unsigned int & p24,
    unsigned int & p25,
    unsigned int & p26,
    unsigned int & p27,
    unsigned int & p28,
    unsigned int & p29,
    unsigned int & p30 ) [static]

```

```

6.3.3.228 static __forceinline__ __device__ void optixTrace (
    OptixPayloadTypeID type,
    OptixTraversableHandle handle,
    float3 rayOrigin,
    float3 rayDirection,

```

**float *tmin*,**  
**float *tmax*,**  
**float *rayTime*,**  
**OptixVisibilityMask *visibilityMask*,**  
**unsigned int *rayFlags*,**  
**unsigned int *SBTOffset*,**  
**unsigned int *SBTstride*,**  
**unsigned int *missSBTIndex*,**  
**unsigned int & *p0*,**  
**unsigned int & *p1*,**  
**unsigned int & *p2*,**  
**unsigned int & *p3*,**  
**unsigned int & *p4*,**  
**unsigned int & *p5*,**  
**unsigned int & *p6*,**  
**unsigned int & *p7*,**  
**unsigned int & *p8*,**  
**unsigned int & *p9*,**  
**unsigned int & *p10*,**  
**unsigned int & *p11*,**  
**unsigned int & *p12*,**  
**unsigned int & *p13*,**  
**unsigned int & *p14*,**  
**unsigned int & *p15*,**  
**unsigned int & *p16*,**  
**unsigned int & *p17*,**  
**unsigned int & *p18*,**  
**unsigned int & *p19*,**  
**unsigned int & *p20*,**  
**unsigned int & *p21*,**  
**unsigned int & *p22*,**  
**unsigned int & *p23*,**  
**unsigned int & *p24*,**  
**unsigned int & *p25*,**  
**unsigned int & *p26*,**  
**unsigned int & *p27*,**  
**unsigned int & *p28*,**  
**unsigned int & *p29*,**  
**unsigned int & *p30*,**

**unsigned int & *p31* ) [static]**

**6.3.3.229 static \_\_forceinline\_\_ \_\_device\_\_ float3 optixTransformNormalFromObjectToWorldSpace ( float3 *normal* ) [static]**

**6.3.3.230 static \_\_forceinline\_\_ \_\_device\_\_ float3 optixTransformNormalFromWorldToObjectSpace ( float3 *normal* ) [static]**

**6.3.3.231 static \_\_forceinline\_\_ \_\_device\_\_ float3 optixTransformPointFromObjectToWorldSpace ( float3 *point* ) [static]**

**6.3.3.232 static \_\_forceinline\_\_ \_\_device\_\_ float3 optixTransformPointFromWorldToObjectSpace ( float3 *point* ) [static]**

**6.3.3.233 static \_\_forceinline\_\_ \_\_device\_\_ float3 optixTransformVectorFromObjectToWorldSpace ( float3 *vec* ) [static]**

**6.3.3.234 static \_\_forceinline\_\_ \_\_device\_\_ float3 optixTransformVectorFromWorldToObjectSpace ( float3 *vec* ) [static]**

**6.3.3.235 static \_\_forceinline\_\_ \_\_device\_\_ unsigned int optixUndefinedValue ( ) [static]**

## 6.4 optix\_7\_device\_impl\_exception.h File Reference

### Namespaces

- [optix\\_impl](#)

### Constant Groups

- [optix\\_impl](#)

### Functions

- static \_\_forceinline\_\_ \_\_device\_\_ void [optix\\_impl::optixDumpStaticTransformFromHandle](#) (OptixTraversableHandle handle)
- static \_\_forceinline\_\_ \_\_device\_\_ void [optix\\_impl::optixDumpMotionMatrixTransformFromHandle](#) (OptixTraversableHandle handle)

- static `__forceinline__`  
`__device__ void optix_impl::optixDumpSrtMatrixTransformFromHandle (OptixTraversableHandle handle)`
- static `__forceinline__`  
`__device__ void optix_impl::optixDumpInstanceFromHandle (OptixTraversableHandle handle)`
- static `__forceinline__`  
`__device__ void optix_impl::optixDumpTransform (OptixTraversableHandle handle)`
- static `__forceinline__`  
`__device__ void optix_impl::optixDumpTransformList ()`
- static `__forceinline__`  
`__device__ void optix_impl::optixDumpExceptionDetails ()`

### 6.4.1 Detailed Description

OptiX public API.

Author

NVIDIA Corporation OptiX public API Reference - Device side implementation for exception helper function.

## 6.5 optix\_7\_device\_impl\_transformations.h File Reference

### Namespaces

- `optix_impl`

### Constant Groups

- `optix_impl`

### Functions

- static `__forceinline__`  
`__device__ float4 optix_impl::optixAddFloat4 (const float4 &a, const float4 &b)`
- static `__forceinline__`  
`__device__ float4 optix_impl::optixMulFloat4 (const float4 &a, float b)`
- static `__forceinline__`  
`__device__ uint4 optix_impl::optixLdg (unsigned long long addr)`
- template<class T >  
static `__forceinline__ __device__ T optix_impl::optixLoadReadOnlyAlign16 (const T *ptr)`
- static `__forceinline__`  
`__device__ float4 optix_impl::optixMultiplyRowMatrix (const float4 vec, const float4 m0, const float4 m1, const float4 m2)`



- static `__forceinline__`  
`__device__ void optix_impl::optixGetMatrixFromSrt (float4 &m0, float4 &m1, float4 &m2, const OptixSRTData &srt)`
- static `__forceinline__`  
`__device__ void optix_impl::optixInvertMatrix (float4 &m0, float4 &m1, float4 &m2)`
- static `__forceinline__`  
`__device__ void optix_impl::optixLoadInterpolatedMatrixKey (float4 &m0, float4 &m1, float4 &m2, const float4 *matrix, const float t1)`
- static `__forceinline__`  
`__device__ void optix_impl::optixLoadInterpolatedSrtKey (float4 &srt0, float4 &srt1, float4 &srt2, float4 &srt3, const float4 *srt, const float t1)`
- static `__forceinline__`  
`__device__ void optix_impl::optixResolveMotionKey (float &localt, int &key, const OptixMotionOptions &options, const float globalt)`
- static `__forceinline__`  
`__device__ void optix_impl::optixGetInterpolatedTransformation (float4 &trf0, float4 &trf1, float4 &trf2, const OptixMatrixMotionTransform *transformData, const float time)`
- static `__forceinline__`  
`__device__ void optix_impl::optixGetInterpolatedTransformation (float4 &trf0, float4 &trf1, float4 &trf2, const OptixSRTMotionTransform *transformData, const float time)`
- static `__forceinline__`  
`__device__ void optix_impl::optixGetInterpolatedTransformationFromHandle (float4 &trf0, float4 &trf1, float4 &trf2, const OptixTraversableHandle handle, const float time, const bool objectToWorld)`
- static `__forceinline__`  
`__device__ void optix_impl::optixGetWorldToObjectTransformMatrix (float4 &m0, float4 &m1, float4 &m2)`
- static `__forceinline__`  
`__device__ void optix_impl::optixGetObjectToWorldTransformMatrix (float4 &m0, float4 &m1, float4 &m2)`
- static `__forceinline__`  
`__device__ float3 optix_impl::optixTransformPoint (const float4 &m0, const float4 &m1, const float4 &m2, const float3 &p)`
- static `__forceinline__`  
`__device__ float3 optix_impl::optixTransformVector (const float4 &m0, const float4 &m1, const float4 &m2, const float3 &v)`
- static `__forceinline__`  
`__device__ float3 optix_impl::optixTransformNormal (const float4 &m0, const float4 &m1, const float4 &m2, const float3 &n)`

### 6.5.1 Detailed Description

OptiX public API.

## Author

NVIDIA Corporation OptiX public API Reference - Device side implementation for transformation helper functions.

## 6.6 optix\_7\_host.h File Reference

### Functions

- `const char * optixGetErrorName (OptixResult result)`
- `const char * optixGetErrorString (OptixResult result)`
- `OptixResult optixDeviceContextCreate (CUcontext fromContext, const OptixDeviceContextOptions *options, OptixDeviceContext *context)`
- `OptixResult optixDeviceContextDestroy (OptixDeviceContext context)`
- `OptixResult optixDeviceContextGetProperty (OptixDeviceContext context, OptixDeviceProperty property, void *value, size_t sizeInBytes)`
- `OptixResult optixDeviceContextSetLogCallback (OptixDeviceContext context, OptixLogCallback callbackFunction, void *callbackData, unsigned int callbackLevel)`
- `OptixResult optixDeviceContextSetCacheEnabled (OptixDeviceContext context, int enabled)`
- `OptixResult optixDeviceContextSetCacheLocation (OptixDeviceContext context, const char *location)`
- `OptixResult optixDeviceContextSetCacheDatabaseSizes (OptixDeviceContext context, size_t lowWaterMark, size_t highWaterMark)`
- `OptixResult optixDeviceContextGetCacheEnabled (OptixDeviceContext context, int *enabled)`
- `OptixResult optixDeviceContextGetCacheLocation (OptixDeviceContext context, char *location, size_t locationSize)`
- `OptixResult optixDeviceContextGetCacheDatabaseSizes (OptixDeviceContext context, size_t *lowWaterMark, size_t *highWaterMark)`
- `OptixResult optixPipelineCreate (OptixDeviceContext context, const OptixPipelineCompileOptions *pipelineCompileOptions, const OptixPipelineLinkOptions *pipelineLinkOptions, const OptixProgramGroup *programGroups, unsigned int numProgramGroups, char *logString, size_t *logStringSize, OptixPipeline *pipeline)`
- `OptixResult optixPipelineDestroy (OptixPipeline pipeline)`
- `OptixResult optixPipelineSetStackSize (OptixPipeline pipeline, unsigned int directCallableStackSizeFromTraversal, unsigned int directCallableStackSizeFromState, unsigned int continuationStackSize, unsigned int maxTraversableGraphDepth)`
- `OptixResult optixModuleCreateFromPTX (OptixDeviceContext context, const OptixModuleCompileOptions *moduleCompileOptions, const OptixPipelineCompileOptions *pipelineCompileOptions, const char *PTX, size_t PTXsize, char *logString, size_t *logStringSize, OptixModule *module)`
- `OptixResult optixModuleCreateFromPTXWithTasks (OptixDeviceContext context, const OptixModuleCompileOptions *moduleCompileOptions, const OptixPipelineCompileOptions *pipelineCompileOptions, const char *PTX, size_t PTXsize, char *logString, size_t *logStringSize, OptixModule *module, OptixTask *firstTask)`

- [OptixResult optixModuleGetCompilationState](#) ([OptixModule](#) module, [OptixModuleCompileState](#) \*state)
- [OptixResult optixModuleDestroy](#) ([OptixModule](#) module)
- [OptixResult optixBuiltinISModuleGet](#) ([OptixDeviceContext](#) context, const [OptixModuleCompileOptions](#) \*moduleCompileOptions, const [OptixPipelineCompileOptions](#) \*pipelineCompileOptions, const [OptixBuiltinISOptions](#) \*builtinISOptions, [OptixModule](#) \*builtinModule)
- [OptixResult optixTaskExecute](#) ([OptixTask](#) task, [OptixTask](#) \*additionalTasks, unsigned int maxNumAdditionalTasks, unsigned int \*numAdditionalTasksCreated)
- [OptixResult optixProgramGroupGetStackSize](#) ([OptixProgramGroup](#) programGroup, [OptixStackSizes](#) \*stackSizes)
- [OptixResult optixProgramGroupCreate](#) ([OptixDeviceContext](#) context, const [OptixProgramGroupDesc](#) \*programDescriptions, unsigned int numProgramGroups, const [OptixProgramGroupOptions](#) \*options, char \*logString, size\_t \*logStringSize, [OptixProgramGroup](#) \*programGroups)
- [OptixResult optixProgramGroupDestroy](#) ([OptixProgramGroup](#) programGroup)
- [OptixResult optixLaunch](#) ([OptixPipeline](#) pipeline, [CUstream](#) stream, [CUdeviceptr](#) pipelineParams, size\_t pipelineParamsSize, const [OptixShaderBindingTable](#) \*sbt, unsigned int width, unsigned int height, unsigned int depth)
- [OptixResult optixSbtRecordPackHeader](#) ([OptixProgramGroup](#) programGroup, void \*sbtRecordHeaderHostPointer)
- [OptixResult optixAccelComputeMemoryUsage](#) ([OptixDeviceContext](#) context, const [OptixAccelBuildOptions](#) \*accelOptions, const [OptixBuildInput](#) \*buildInputs, unsigned int numBuildInputs, [OptixAccelBufferSizes](#) \*bufferSizes)
- [OptixResult optixAccelBuild](#) ([OptixDeviceContext](#) context, [CUstream](#) stream, const [OptixAccelBuildOptions](#) \*accelOptions, const [OptixBuildInput](#) \*buildInputs, unsigned int numBuildInputs, [CUdeviceptr](#) tempBuffer, size\_t tempBufferSizeInBytes, [CUdeviceptr](#) outputBuffer, size\_t outputBufferSizeInBytes, [OptixTraversableHandle](#) \*outputHandle, const [OptixAccelEmitDesc](#) \*emittedProperties, unsigned int numEmittedProperties)
- [OptixResult optixAccelGetRelocationInfo](#) ([OptixDeviceContext](#) context, [OptixTraversableHandle](#) handle, [OptixAccelRelocationInfo](#) \*info)
- [OptixResult optixAccelCheckRelocationCompatibility](#) ([OptixDeviceContext](#) context, const [OptixAccelRelocationInfo](#) \*info, int \*compatible)
- [OptixResult optixAccelRelocate](#) ([OptixDeviceContext](#) context, [CUstream](#) stream, const [OptixAccelRelocationInfo](#) \*info, [CUdeviceptr](#) instanceTraversableHandles, size\_t numInstanceTraversableHandles, [CUdeviceptr](#) targetAccel, size\_t targetAccelSizeInBytes, [OptixTraversableHandle](#) \*targetHandle)
- [OptixResult optixAccelCompact](#) ([OptixDeviceContext](#) context, [CUstream](#) stream, [OptixTraversableHandle](#) inputHandle, [CUdeviceptr](#) outputBuffer, size\_t outputBufferSizeInBytes, [OptixTraversableHandle](#) \*outputHandle)
- [OptixResult optixConvertPointerToTraversableHandle](#) ([OptixDeviceContext](#) onDevice, [CUdeviceptr](#) pointer, [OptixTraversableType](#) traversableType, [OptixTraversableHandle](#) \*traversableHandle)
- [OptixResult optixDenoiserCreate](#) ([OptixDeviceContext](#) context, [OptixDenoiserModelKind](#) modelKind, const [OptixDenoiserOptions](#) \*options, [OptixDenoiser](#) \*denoiser)

- `OptixResult optixDenoiserCreateWithUserModel` (`OptixDeviceContext` context, const void \*userData, size\_t userDataSizeInBytes, `OptixDenoiser` \*denoiser)
- `OptixResult optixDenoiserDestroy` (`OptixDenoiser` denoiser)
- `OptixResult optixDenoiserComputeMemoryResources` (const `OptixDenoiser` denoiser, unsigned int outputWidth, unsigned int outputHeight, `OptixDenoiserSizes` \*returnSizes)
- `OptixResult optixDenoiserSetup` (`OptixDenoiser` denoiser, `CUstream` stream, unsigned int inputWidth, unsigned int inputHeight, `CUdeviceptr` denoiserState, size\_t denoiserStateSizeInBytes, `CUdeviceptr` scratch, size\_t scratchSizeInBytes)
- `OptixResult optixDenoiserInvoke` (`OptixDenoiser` denoiser, `CUstream` stream, const `OptixDenoiserParams` \*params, `CUdeviceptr` denoiserState, size\_t denoiserStateSizeInBytes, const `OptixDenoiserGuideLayer` \*guideLayer, const `OptixDenoiserLayer` \*layers, unsigned int numLayers, unsigned int inputOffsetX, unsigned int inputOffsetY, `CUdeviceptr` scratch, size\_t scratchSizeInBytes)
- `OptixResult optixDenoiserComputeIntensity` (`OptixDenoiser` denoiser, `CUstream` stream, const `OptixImage2D` \*inputImage, `CUdeviceptr` outputIntensity, `CUdeviceptr` scratch, size\_t scratchSizeInBytes)
- `OptixResult optixDenoiserComputeAverageColor` (`OptixDenoiser` denoiser, `CUstream` stream, const `OptixImage2D` \*inputImage, `CUdeviceptr` outputAverageColor, `CUdeviceptr` scratch, size\_t scratchSizeInBytes)

### 6.6.1 Detailed Description

OptiX public API header.

Author

NVIDIA Corporation OptiX host include file – includes the host api if compiling host code. For the math library routines include `optix_math.h`

## 6.7 optix\_7\_types.h File Reference

### Classes

- struct `OptixDeviceContextOptions`
- struct `OptixBuildInputTriangleArray`
- struct `OptixBuildInputCurveArray`
- struct `OptixAabb`
- struct `OptixBuildInputCustomPrimitiveArray`
- struct `OptixBuildInputInstanceArray`
- struct `OptixBuildInput`
- struct `OptixInstance`
- struct `OptixMotionOptions`
- struct `OptixAccelBuildOptions`
- struct `OptixAccelBufferSizes`

- struct [OptixAccelEmitDesc](#)
- struct [OptixAccelRelocationInfo](#)
- struct [OptixStaticTransform](#)
- struct [OptixMatrixMotionTransform](#)
- struct [OptixSRTData](#)
- struct [OptixSRTMotionTransform](#)
- struct [OptixImage2D](#)
- struct [OptixDenoiserOptions](#)
- struct [OptixDenoiserGuideLayer](#)
- struct [OptixDenoiserLayer](#)
- struct [OptixDenoiserParams](#)
- struct [OptixDenoiserSizes](#)
- struct [OptixModuleCompileBoundValueEntry](#)
- struct [OptixPayloadType](#)
- struct [OptixModuleCompileOptions](#)
- struct [OptixProgramGroupSingleModule](#)
- struct [OptixProgramGroupHitgroup](#)
- struct [OptixProgramGroupCallables](#)
- struct [OptixProgramGroupDesc](#)
- struct [OptixProgramGroupOptions](#)
- struct [OptixPipelineCompileOptions](#)
- struct [OptixPipelineLinkOptions](#)
- struct [OptixShaderBindingTable](#)
- struct [OptixStackSizes](#)
- struct [OptixBuiltinISOOptions](#)

## Macros

- `#define OPTIX_SBT_RECORD_HEADER_SIZE ( (size_t)32 )`
- `#define OPTIX_SBT_RECORD_ALIGNMENT 16ull`
- `#define OPTIX_ACCEL_BUFFER_BYTE_ALIGNMENT 128ull`
- `#define OPTIX_INSTANCE_BYTE_ALIGNMENT 16ull`
- `#define OPTIX_AABB_BUFFER_BYTE_ALIGNMENT 8ull`
- `#define OPTIX_GEOMETRY_TRANSFORM_BYTE_ALIGNMENT 16ull`
- `#define OPTIX_TRANSFORM_BYTE_ALIGNMENT 64ull`
- `#define OPTIX_COMPILE_DEFAULT_MAX_REGISTER_COUNT 0`
- `#define OPTIX_COMPILE_DEFAULT_MAX_PAYLOAD_TYPE_COUNT 8`
- `#define OPTIX_COMPILE_DEFAULT_MAX_PAYLOAD_VALUE_COUNT 32`

## Typedefs

- typedef unsigned long long CUdeviceptr
- typedef struct  
OptixDeviceContext\_t \* OptixDeviceContext
- typedef struct OptixModule\_t \* OptixModule
- typedef struct  
OptixProgramGroup\_t \* OptixProgramGroup
- typedef struct OptixPipeline\_t \* OptixPipeline
- typedef struct OptixDenoiser\_t \* OptixDenoiser
- typedef struct OptixTask\_t \* OptixTask
- typedef unsigned long long OptixTraversableHandle
- typedef unsigned int OptixVisibilityMask
- typedef enum OptixResult OptixResult
- typedef enum OptixDeviceProperty OptixDeviceProperty
- typedef void(\* OptixLogCallback )(unsigned int level, const char \*tag, const char \*message, void \*cbdata)
- typedef enum  
OptixDeviceContextValidationMode OptixDeviceContextValidationMode
- typedef struct  
OptixDeviceContextOptions OptixDeviceContextOptions
- typedef enum OptixGeometryFlags OptixGeometryFlags
- typedef enum OptixHitKind OptixHitKind
- typedef enum OptixIndicesFormat OptixIndicesFormat
- typedef enum OptixVertexFormat OptixVertexFormat
- typedef enum OptixTransformFormat OptixTransformFormat
- typedef struct  
OptixBuildInputTriangleArray OptixBuildInputTriangleArray
- typedef enum OptixPrimitiveType OptixPrimitiveType
- typedef enum  
OptixPrimitiveTypeFlags OptixPrimitiveTypeFlags
- typedef enum OptixCurveEndcapFlags OptixCurveEndcapFlags
- typedef struct  
OptixBuildInputCurveArray OptixBuildInputCurveArray
- typedef struct OptixAabb OptixAabb
- typedef struct  
OptixBuildInputCustomPrimitiveArray OptixBuildInputCustomPrimitiveArray
- typedef struct  
OptixBuildInputInstanceArray OptixBuildInputInstanceArray
- typedef enum OptixBuildInputType OptixBuildInputType
- typedef struct OptixBuildInput OptixBuildInput
- typedef enum OptixInstanceFlags OptixInstanceFlags
- typedef struct OptixInstance OptixInstance

- typedef enum [OptixBuildFlags](#) [OptixBuildFlags](#)
- typedef enum [OptixBuildOperation](#) [OptixBuildOperation](#)
- typedef enum [OptixMotionFlags](#) [OptixMotionFlags](#)
- typedef struct [OptixMotionOptions](#) [OptixMotionOptions](#)
- typedef struct [OptixAccelBuildOptions](#) [OptixAccelBuildOptions](#)
- typedef struct [OptixAccelBufferSizes](#) [OptixAccelBufferSizes](#)
- typedef enum [OptixAccelPropertyType](#) [OptixAccelPropertyType](#)
- typedef struct [OptixAccelEmitDesc](#) [OptixAccelEmitDesc](#)
- typedef struct [OptixAccelRelocationInfo](#) [OptixAccelRelocationInfo](#)
- typedef struct [OptixStaticTransform](#) [OptixStaticTransform](#)
- typedef struct [OptixMatrixMotionTransform](#) [OptixMatrixMotionTransform](#)
- typedef struct [OptixSRTData](#) [OptixSRTData](#)
- typedef struct [OptixSRTMotionTransform](#) [OptixSRTMotionTransform](#)
- typedef enum [OptixTraversableType](#) [OptixTraversableType](#)
- typedef enum [OptixPixelFormat](#) [OptixPixelFormat](#)
- typedef struct [OptixImage2D](#) [OptixImage2D](#)
- typedef enum [OptixDenoiserModelKind](#) [OptixDenoiserModelKind](#)
- typedef struct [OptixDenoiserOptions](#) [OptixDenoiserOptions](#)
- typedef struct [OptixDenoiserGuideLayer](#) [OptixDenoiserGuideLayer](#)
- typedef struct [OptixDenoiserLayer](#) [OptixDenoiserLayer](#)
- typedef struct [OptixDenoiserParams](#) [OptixDenoiserParams](#)
- typedef struct [OptixDenoiserSizes](#) [OptixDenoiserSizes](#)
- typedef enum [OptixRayFlags](#) [OptixRayFlags](#)
- typedef enum [OptixTransformType](#) [OptixTransformType](#)
- typedef enum [OptixTraversableGraphFlags](#) [OptixTraversableGraphFlags](#)
- typedef enum [OptixCompileOptimizationLevel](#) [OptixCompileOptimizationLevel](#)
- typedef enum [OptixCompileDebugLevel](#) [OptixCompileDebugLevel](#)
- typedef enum [OptixModuleCompileState](#) [OptixModuleCompileState](#)
- typedef struct [OptixModuleCompileBoundValueEntry](#) [OptixModuleCompileBoundValueEntry](#)
- typedef enum [OptixPayloadTypeID](#) [OptixPayloadTypeID](#)
- typedef enum [OptixPayloadSemantics](#) [OptixPayloadSemantics](#)
- typedef struct [OptixPayloadType](#) [OptixPayloadType](#)

- typedef struct  
OptixModuleCompileOptions OptixModuleCompileOptions
- typedef enum OptixProgramGroupKind OptixProgramGroupKind
- typedef enum OptixProgramGroupFlags OptixProgramGroupFlags
- typedef struct  
OptixProgramGroupSingleModule OptixProgramGroupSingleModule
- typedef struct  
OptixProgramGroupHitgroup OptixProgramGroupHitgroup
- typedef struct  
OptixProgramGroupCallables OptixProgramGroupCallables
- typedef struct  
OptixProgramGroupDesc OptixProgramGroupDesc
- typedef struct  
OptixProgramGroupOptions OptixProgramGroupOptions
- typedef enum OptixExceptionCodes OptixExceptionCodes
- typedef enum OptixExceptionFlags OptixExceptionFlags
- typedef struct  
OptixPipelineCompileOptions OptixPipelineCompileOptions
- typedef struct  
OptixPipelineLinkOptions OptixPipelineLinkOptions
- typedef struct  
OptixShaderBindingTable OptixShaderBindingTable
- typedef struct OptixStackSizes OptixStackSizes
- typedef enum  
OptixQueryFunctionTableOptions OptixQueryFunctionTableOptions
- typedef OptixResult( OptixQueryFunctionTable\_t )(int abiId, unsigned int numOptions,  
OptixQueryFunctionTableOptions \*, const void \*\*, void \*functionTable, size\_t sizeOfTable)
- typedef struct  
OptixBuiltinISOOptions OptixBuiltinISOOptions



## Enumerations

- enum OptixResult {  
OPTIX\_SUCCESS = 0,  
OPTIX\_ERROR\_INVALID\_VALUE = 7001,  
OPTIX\_ERROR\_HOST\_OUT\_OF\_MEMORY = 7002,  
OPTIX\_ERROR\_INVALID\_OPERATION = 7003,  
OPTIX\_ERROR\_FILE\_IO\_ERROR = 7004,  
OPTIX\_ERROR\_INVALID\_FILE\_FORMAT = 7005,  
OPTIX\_ERROR\_DISK\_CACHE\_INVALID\_PATH = 7010,  
OPTIX\_ERROR\_DISK\_CACHE\_PERMISSION\_ERROR = 7011,  
OPTIX\_ERROR\_DISK\_CACHE\_DATABASE\_ERROR = 7012,  
OPTIX\_ERROR\_DISK\_CACHE\_INVALID\_DATA = 7013,  
OPTIX\_ERROR\_LAUNCH\_FAILURE = 7050,  
OPTIX\_ERROR\_INVALID\_DEVICE\_CONTEXT = 7051,  
OPTIX\_ERROR\_CUDA\_NOT\_INITIALIZED = 7052,  
OPTIX\_ERROR\_VALIDATION\_FAILURE = 7053,  
OPTIX\_ERROR\_INVALID\_PTX = 7200,  
OPTIX\_ERROR\_INVALID\_LAUNCH\_PARAMETER = 7201,  
OPTIX\_ERROR\_INVALID\_PAYLOAD\_ACCESS = 7202,  
OPTIX\_ERROR\_INVALID\_ATTRIBUTE\_ACCESS = 7203,  
OPTIX\_ERROR\_INVALID\_FUNCTION\_USE = 7204,  
OPTIX\_ERROR\_INVALID\_FUNCTION\_ARGUMENTS = 7205,  
OPTIX\_ERROR\_PIPELINE\_OUT\_OF\_CONSTANT\_MEMORY = 7250,  
OPTIX\_ERROR\_PIPELINE\_LINK\_ERROR = 7251,  
OPTIX\_ERROR\_ILLEGAL\_DURING\_TASK\_EXECUTE = 7270,  
OPTIX\_ERROR\_INTERNAL\_COMPILER\_ERROR = 7299,  
OPTIX\_ERROR\_DENOISER\_MODEL\_NOT\_SET = 7300,  
OPTIX\_ERROR\_DENOISER\_NOT\_INITIALIZED = 7301,  
OPTIX\_ERROR\_ACCEL\_NOT\_COMPATIBLE = 7400,  
OPTIX\_ERROR\_PAYLOAD\_TYPE\_MISMATCH = 7500,  
OPTIX\_ERROR\_PAYLOAD\_TYPE\_RESOLUTION\_FAILED = 7501,  
OPTIX\_ERROR\_PAYLOAD\_TYPE\_ID\_INVALID = 7502,  
OPTIX\_ERROR\_NOT\_SUPPORTED = 7800,  
OPTIX\_ERROR\_UNSUPPORTED\_ABI\_VERSION = 7801,  
OPTIX\_ERROR\_FUNCTION\_TABLE\_SIZE\_MISMATCH = 7802,  
OPTIX\_ERROR\_INVALID\_ENTRY\_FUNCTION\_OPTIONS = 7803,  
OPTIX\_ERROR\_LIBRARY\_NOT\_FOUND = 7804,  
OPTIX\_ERROR\_ENTRY\_SYMBOL\_NOT\_FOUND = 7805,  
OPTIX\_ERROR\_LIBRARY\_UNLOAD\_FAILURE = 7806,  
OPTIX\_ERROR\_CUDA\_ERROR = 7900,  
OPTIX\_ERROR\_INTERNAL\_ERROR = 7990,  
OPTIX\_ERROR\_UNKNOWN = 7999 }

- enum OptixDeviceProperty {  
OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_TRACE\_DEPTH = 0x2001,  
OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_TRAVERSABLE\_GRAPH\_DEPTH = 0x2002,  
OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_PRIMITIVES\_PER\_GAS = 0x2003,  
OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_INSTANCES\_PER\_IAS = 0x2004,  
OPTIX\_DEVICE\_PROPERTY\_RTCORE\_VERSION = 0x2005,  
OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_INSTANCE\_ID = 0x2006,  
OPTIX\_DEVICE\_PROPERTY\_LIMIT\_NUM\_BITS\_INSTANCE\_VISIBILITY\_MASK = 0x2007,  
OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_SBT\_RECORDS\_PER\_GAS = 0x2008,  
OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_SBT\_OFFSET = 0x2009 }
- enum OptixDeviceContextValidationMode {  
OPTIX\_DEVICE\_CONTEXT\_VALIDATION\_MODE\_OFF = 0,  
OPTIX\_DEVICE\_CONTEXT\_VALIDATION\_MODE\_ALL = 0xFFFFFFFF }
- enum OptixGeometryFlags {  
OPTIX\_GEOMETRY\_FLAG\_NONE = 0,  
OPTIX\_GEOMETRY\_FLAG\_DISABLE\_ANYHIT = 1u << 0,  
OPTIX\_GEOMETRY\_FLAG\_REQUIRE\_SINGLE\_ANYHIT\_CALL = 1u << 1 }
- enum OptixHitKind {  
OPTIX\_HIT\_KIND\_TRIANGLE\_FRONT\_FACE = 0xFE,  
OPTIX\_HIT\_KIND\_TRIANGLE\_BACK\_FACE = 0xFF }
- enum OptixIndicesFormat {  
OPTIX\_INDICES\_FORMAT\_NONE = 0,  
OPTIX\_INDICES\_FORMAT\_UNSIGNED\_SHORT3 = 0x2102,  
OPTIX\_INDICES\_FORMAT\_UNSIGNED\_INT3 = 0x2103 }
- enum OptixVertexFormat {  
OPTIX\_VERTEX\_FORMAT\_NONE = 0,  
OPTIX\_VERTEX\_FORMAT\_FLOAT3 = 0x2121,  
OPTIX\_VERTEX\_FORMAT\_FLOAT2 = 0x2122,  
OPTIX\_VERTEX\_FORMAT\_HALF3 = 0x2123,  
OPTIX\_VERTEX\_FORMAT\_HALF2 = 0x2124,  
OPTIX\_VERTEX\_FORMAT\_SNORM16\_3 = 0x2125,  
OPTIX\_VERTEX\_FORMAT\_SNORM16\_2 = 0x2126 }
- enum OptixTransformFormat {  
OPTIX\_TRANSFORM\_FORMAT\_NONE = 0,  
OPTIX\_TRANSFORM\_FORMAT\_MATRIX\_FLOAT12 = 0x21E1 }
- enum OptixPrimitiveType {  
OPTIX\_PRIMITIVE\_TYPE\_CUSTOM = 0x2500,  
OPTIX\_PRIMITIVE\_TYPE\_ROUND\_QUADRATIC\_BSPLINE = 0x2501,  
OPTIX\_PRIMITIVE\_TYPE\_ROUND\_CUBIC\_BSPLINE = 0x2502,  
OPTIX\_PRIMITIVE\_TYPE\_ROUND\_LINEAR = 0x2503,  
OPTIX\_PRIMITIVE\_TYPE\_ROUND\_CATMULLROM = 0x2504,  
OPTIX\_PRIMITIVE\_TYPE\_TRIANGLE = 0x2531 }

- enum OptixPrimitiveTypeFlags {  
OPTIX\_PRIMITIVE\_TYPE\_FLAGS\_CUSTOM = 1 << 0,  
OPTIX\_PRIMITIVE\_TYPE\_FLAGS\_ROUND\_QUADRATIC\_BSPLINE = 1 << 1,  
OPTIX\_PRIMITIVE\_TYPE\_FLAGS\_ROUND\_CUBIC\_BSPLINE = 1 << 2,  
OPTIX\_PRIMITIVE\_TYPE\_FLAGS\_ROUND\_LINEAR = 1 << 3,  
OPTIX\_PRIMITIVE\_TYPE\_FLAGS\_ROUND\_CATMULLROM = 1 << 4,  
OPTIX\_PRIMITIVE\_TYPE\_FLAGS\_TRIANGLE = 1 << 31 }
- enum OptixCurveEndcapFlags {  
OPTIX\_CURVE\_ENDCAP\_DEFAULT = 0,  
OPTIX\_CURVE\_ENDCAP\_ON = 1 << 0 }
- enum OptixBuildInputType {  
OPTIX\_BUILD\_INPUT\_TYPE\_TRIANGLES = 0x2141,  
OPTIX\_BUILD\_INPUT\_TYPE\_CUSTOM\_PRIMITIVES = 0x2142,  
OPTIX\_BUILD\_INPUT\_TYPE\_INSTANCES = 0x2143,  
OPTIX\_BUILD\_INPUT\_TYPE\_INSTANCE\_POINTERS = 0x2144,  
OPTIX\_BUILD\_INPUT\_TYPE\_CURVES = 0x2145 }
- enum OptixInstanceFlags {  
OPTIX\_INSTANCE\_FLAG\_NONE = 0,  
OPTIX\_INSTANCE\_FLAG\_DISABLE\_TRIANGLE\_FACE\_CULLING = 1u << 0,  
OPTIX\_INSTANCE\_FLAG\_FLIP\_TRIANGLE\_FACING = 1u << 1,  
OPTIX\_INSTANCE\_FLAG\_DISABLE\_ANYHIT = 1u << 2,  
OPTIX\_INSTANCE\_FLAG\_ENFORCE\_ANYHIT = 1u << 3 }
- enum OptixBuildFlags {  
OPTIX\_BUILD\_FLAG\_NONE = 0,  
OPTIX\_BUILD\_FLAG\_ALLOW\_UPDATE = 1u << 0,  
OPTIX\_BUILD\_FLAG\_ALLOW\_COMPACTION = 1u << 1,  
OPTIX\_BUILD\_FLAG\_PREFER\_FAST\_TRACE = 1u << 2,  
OPTIX\_BUILD\_FLAG\_PREFER\_FAST\_BUILD = 1u << 3,  
OPTIX\_BUILD\_FLAG\_ALLOW\_RANDOM\_VERTEX\_ACCESS = 1u << 4,  
OPTIX\_BUILD\_FLAG\_ALLOW\_RANDOM\_INSTANCE\_ACCESS = 1u << 5 }
- enum OptixBuildOperation {  
OPTIX\_BUILD\_OPERATION\_BUILD = 0x2161,  
OPTIX\_BUILD\_OPERATION\_UPDATE = 0x2162 }
- enum OptixMotionFlags {  
OPTIX\_MOTION\_FLAG\_NONE = 0,  
OPTIX\_MOTION\_FLAG\_START\_VANISH = 1u << 0,  
OPTIX\_MOTION\_FLAG\_END\_VANISH = 1u << 1 }
- enum OptixAccelPropertyType {  
OPTIX\_PROPERTY\_TYPE\_COMPACTED\_SIZE = 0x2181,  
OPTIX\_PROPERTY\_TYPE\_AABBS = 0x2182 }
- enum OptixTraversableType {  
OPTIX\_TRAVERSABLE\_TYPE\_STATIC\_TRANSFORM = 0x21C1,  
OPTIX\_TRAVERSABLE\_TYPE\_MATRIX\_MOTION\_TRANSFORM = 0x21C2,  
OPTIX\_TRAVERSABLE\_TYPE\_SRT\_MOTION\_TRANSFORM = 0x21C3 }

- enum OptixPixelFormat {  
OPTIX\_PIXEL\_FORMAT\_HALF2 = 0x2207,  
OPTIX\_PIXEL\_FORMAT\_HALF3 = 0x2201,  
OPTIX\_PIXEL\_FORMAT\_HALF4 = 0x2202,  
OPTIX\_PIXEL\_FORMAT\_FLOAT2 = 0x2208,  
OPTIX\_PIXEL\_FORMAT\_FLOAT3 = 0x2203,  
OPTIX\_PIXEL\_FORMAT\_FLOAT4 = 0x2204,  
OPTIX\_PIXEL\_FORMAT\_UCHAR3 = 0x2205,  
OPTIX\_PIXEL\_FORMAT\_UCHAR4 = 0x2206 }
- enum OptixDenoiserModelKind {  
OPTIX\_DENOISER\_MODEL\_KIND\_LDR = 0x2322,  
OPTIX\_DENOISER\_MODEL\_KIND\_HDR = 0x2323,  
OPTIX\_DENOISER\_MODEL\_KIND\_AOV = 0x2324,  
OPTIX\_DENOISER\_MODEL\_KIND\_TEMPORAL = 0x2325,  
OPTIX\_DENOISER\_MODEL\_KIND\_TEMPORAL\_AOV = 0x2326 }
- enum OptixRayFlags {  
OPTIX\_RAY\_FLAG\_NONE = 0u,  
OPTIX\_RAY\_FLAG\_DISABLE\_ANYHIT = 1u << 0,  
OPTIX\_RAY\_FLAG\_ENFORCE\_ANYHIT = 1u << 1,  
OPTIX\_RAY\_FLAG\_TERMINATE\_ON\_FIRST\_HIT = 1u << 2,  
OPTIX\_RAY\_FLAG\_DISABLE\_CLOSESTHIT = 1u << 3,  
OPTIX\_RAY\_FLAG\_CULL\_BACK\_FACING\_TRIANGLES = 1u << 4,  
OPTIX\_RAY\_FLAG\_CULL\_FRONT\_FACING\_TRIANGLES = 1u << 5,  
OPTIX\_RAY\_FLAG\_CULL\_DISABLED\_ANYHIT = 1u << 6,  
OPTIX\_RAY\_FLAG\_CULL\_ENFORCED\_ANYHIT = 1u << 7 }
- enum OptixTransformType {  
OPTIX\_TRANSFORM\_TYPE\_NONE = 0,  
OPTIX\_TRANSFORM\_TYPE\_STATIC\_TRANSFORM = 1,  
OPTIX\_TRANSFORM\_TYPE\_MATRIX\_MOTION\_TRANSFORM = 2,  
OPTIX\_TRANSFORM\_TYPE\_SRT\_MOTION\_TRANSFORM = 3,  
OPTIX\_TRANSFORM\_TYPE\_INSTANCE = 4 }
- enum OptixTraversableGraphFlags {  
OPTIX\_TRAVERSABLE\_GRAPH\_FLAG\_ALLOW\_ANY = 0,  
OPTIX\_TRAVERSABLE\_GRAPH\_FLAG\_ALLOW\_SINGLE\_GAS = 1u << 0,  
OPTIX\_TRAVERSABLE\_GRAPH\_FLAG\_ALLOW\_SINGLE\_LEVEL\_INSTANCING = 1u << 1 }
- enum OptixCompileOptimizationLevel {  
OPTIX\_COMPILE\_OPTIMIZATION\_DEFAULT = 0,  
OPTIX\_COMPILE\_OPTIMIZATION\_LEVEL\_0 = 0x2340,  
OPTIX\_COMPILE\_OPTIMIZATION\_LEVEL\_1 = 0x2341,  
OPTIX\_COMPILE\_OPTIMIZATION\_LEVEL\_2 = 0x2342,  
OPTIX\_COMPILE\_OPTIMIZATION\_LEVEL\_3 = 0x2343 }
- enum OptixCompileDebugLevel {  
OPTIX\_COMPILE\_DEBUG\_LEVEL\_DEFAULT = 0,  
OPTIX\_COMPILE\_DEBUG\_LEVEL\_NONE = 0x2350,  
OPTIX\_COMPILE\_DEBUG\_LEVEL\_MINIMAL = 0x2351,  
OPTIX\_COMPILE\_DEBUG\_LEVEL\_MODERATE = 0x2353,  
OPTIX\_COMPILE\_DEBUG\_LEVEL\_FULL = 0x2352 }

- enum OptixModuleCompileState {  
OPTIX\_MODULE\_COMPILE\_STATE\_NOT\_STARTED = 0x2360,  
OPTIX\_MODULE\_COMPILE\_STATE\_STARTED = 0x2361,  
OPTIX\_MODULE\_COMPILE\_STATE\_IMPENDING\_FAILURE = 0x2362,  
OPTIX\_MODULE\_COMPILE\_STATE\_FAILED = 0x2363,  
OPTIX\_MODULE\_COMPILE\_STATE\_COMPLETED = 0x2364 }
- enum OptixPayloadTypeID {  
OPTIX\_PAYLOAD\_TYPE\_DEFAULT = 0,  
OPTIX\_PAYLOAD\_TYPE\_ID\_0 = (1 << 0u),  
OPTIX\_PAYLOAD\_TYPE\_ID\_1 = (1 << 1u),  
OPTIX\_PAYLOAD\_TYPE\_ID\_2 = (1 << 2u),  
OPTIX\_PAYLOAD\_TYPE\_ID\_3 = (1 << 3u),  
OPTIX\_PAYLOAD\_TYPE\_ID\_4 = (1 << 4u),  
OPTIX\_PAYLOAD\_TYPE\_ID\_5 = (1 << 5u),  
OPTIX\_PAYLOAD\_TYPE\_ID\_6 = (1 << 6u),  
OPTIX\_PAYLOAD\_TYPE\_ID\_7 = (1 << 7u) }
- enum OptixPayloadSemantics {  
OPTIX\_PAYLOAD\_SEMANTICS\_TRACE\_CALLER\_NONE = 0,  
OPTIX\_PAYLOAD\_SEMANTICS\_TRACE\_CALLER\_READ = 1u << 0,  
OPTIX\_PAYLOAD\_SEMANTICS\_TRACE\_CALLER\_WRITE = 2u << 0,  
OPTIX\_PAYLOAD\_SEMANTICS\_TRACE\_CALLER\_READ\_WRITE = 3u << 0,  
OPTIX\_PAYLOAD\_SEMANTICS\_CH\_NONE = 0,  
OPTIX\_PAYLOAD\_SEMANTICS\_CH\_READ = 1u << 2,  
OPTIX\_PAYLOAD\_SEMANTICS\_CH\_WRITE = 2u << 2,  
OPTIX\_PAYLOAD\_SEMANTICS\_CH\_READ\_WRITE = 3u << 2,  
OPTIX\_PAYLOAD\_SEMANTICS\_MS\_NONE = 0,  
OPTIX\_PAYLOAD\_SEMANTICS\_MS\_READ = 1u << 4,  
OPTIX\_PAYLOAD\_SEMANTICS\_MS\_WRITE = 2u << 4,  
OPTIX\_PAYLOAD\_SEMANTICS\_MS\_READ\_WRITE = 3u << 4,  
OPTIX\_PAYLOAD\_SEMANTICS\_AH\_NONE = 0,  
OPTIX\_PAYLOAD\_SEMANTICS\_AH\_READ = 1u << 6,  
OPTIX\_PAYLOAD\_SEMANTICS\_AH\_WRITE = 2u << 6,  
OPTIX\_PAYLOAD\_SEMANTICS\_AH\_READ\_WRITE = 3u << 6,  
OPTIX\_PAYLOAD\_SEMANTICS\_IS\_NONE = 0,  
OPTIX\_PAYLOAD\_SEMANTICS\_IS\_READ = 1u << 8,  
OPTIX\_PAYLOAD\_SEMANTICS\_IS\_WRITE = 2u << 8,  
OPTIX\_PAYLOAD\_SEMANTICS\_IS\_READ\_WRITE = 3u << 8 }
- enum OptixProgramGroupKind {  
OPTIX\_PROGRAM\_GROUP\_KIND\_RAYGEN = 0x2421,  
OPTIX\_PROGRAM\_GROUP\_KIND\_MISS = 0x2422,  
OPTIX\_PROGRAM\_GROUP\_KIND\_EXCEPTION = 0x2423,  
OPTIX\_PROGRAM\_GROUP\_KIND\_HITGROUP = 0x2424,  
OPTIX\_PROGRAM\_GROUP\_KIND\_CALLABLES = 0x2425 }
- enum OptixProgramGroupFlags { OPTIX\_PROGRAM\_GROUP\_FLAGS\_NONE = 0 }

- enum OptixExceptionCodes {
  - OPTIX\_EXCEPTION\_CODE\_STACK\_OVERFLOW = -1,
  - OPTIX\_EXCEPTION\_CODE\_TRACE\_DEPTH\_EXCEEDED = -2,
  - OPTIX\_EXCEPTION\_CODE\_TRAVERSAL\_DEPTH\_EXCEEDED = -3,
  - OPTIX\_EXCEPTION\_CODE\_TRAVERSAL\_INVALID\_TRAVERSABLE = -5,
  - OPTIX\_EXCEPTION\_CODE\_TRAVERSAL\_INVALID\_MISS\_SBT = -6,
  - OPTIX\_EXCEPTION\_CODE\_TRAVERSAL\_INVALID\_HIT\_SBT = -7,
  - OPTIX\_EXCEPTION\_CODE\_UNSUPPORTED\_PRIMITIVE\_TYPE = -8,
  - OPTIX\_EXCEPTION\_CODE\_INVALID\_RAY = -9,
  - OPTIX\_EXCEPTION\_CODE\_CALLABLE\_PARAMETER\_MISMATCH = -10,
  - OPTIX\_EXCEPTION\_CODE\_BUILTIN\_IS\_MISMATCH = -11,
  - OPTIX\_EXCEPTION\_CODE\_CALLABLE\_INVALID\_SBT = -12,
  - OPTIX\_EXCEPTION\_CODE\_CALLABLE\_NO\_DC\_SBT\_RECORD = -13,
  - OPTIX\_EXCEPTION\_CODE\_CALLABLE\_NO\_CC\_SBT\_RECORD = -14,
  - OPTIX\_EXCEPTION\_CODE\_UNSUPPORTED\_SINGLE\_LEVEL\_GAS = -15,
  - OPTIX\_EXCEPTION\_CODE\_INVALID\_VALUE\_ARGUMENT\_0 = -16,
  - OPTIX\_EXCEPTION\_CODE\_INVALID\_VALUE\_ARGUMENT\_1 = -17,
  - OPTIX\_EXCEPTION\_CODE\_INVALID\_VALUE\_ARGUMENT\_2 = -18,
  - OPTIX\_EXCEPTION\_CODE\_UNSUPPORTED\_DATA\_ACCESS = -32,
  - OPTIX\_EXCEPTION\_CODE\_PAYLOAD\_TYPE\_MISMATCH = -33 }
- enum OptixExceptionFlags {
  - OPTIX\_EXCEPTION\_FLAG\_NONE = 0,
  - OPTIX\_EXCEPTION\_FLAG\_STACK\_OVERFLOW = 1u << 0,
  - OPTIX\_EXCEPTION\_FLAG\_TRACE\_DEPTH = 1u << 1,
  - OPTIX\_EXCEPTION\_FLAG\_USER = 1u << 2,
  - OPTIX\_EXCEPTION\_FLAG\_DEBUG = 1u << 3 }
- enum OptixQueryFunctionTableOptions {
  - OPTIX\_QUERY\_FUNCTION\_TABLE\_OPTION\_DUMMY = 0 }

### 6.7.1 Detailed Description

OptiX public API header.

Author

NVIDIA Corporation OptiX types include file – defines types and enums used by the API. For the math library routines include `optix_math.h`

## 6.8 optix\_denoiser\_tiling.h File Reference

### Classes

- struct `OptixUtilDenoiserImageTile`

### Functions

- unsigned int `optixUtilGetPixelStride` (const `OptixImage2D` &image)

- [OptixResult optixUtilDenoiserSplitImage](#) (const [OptixImage2D](#) &input, const [OptixImage2D](#) &output, unsigned int overlapWindowSizeInPixels, unsigned int tileWidth, unsigned int tileHeight, std::vector< [OptixUtilDenoiserImageTile](#) > &tiles)
- [OptixResult optixUtilDenoiserInvokeTiled](#) ([OptixDenoiser](#) denoiser, CUstream stream, const [OptixDenoiserParams](#) \*params, CUdeviceptr denoiserState, size\_t denoiserStateSizeInBytes, const [OptixDenoiserGuideLayer](#) \*guideLayer, const [OptixDenoiserLayer](#) \*layers, unsigned int numLayers, CUdeviceptr scratch, size\_t scratchSizeInBytes, unsigned int overlapWindowSizeInPixels, unsigned int tileWidth, unsigned int tileHeight)

### 6.8.1 Detailed Description

OptiX public API header.

Author

NVIDIA Corporation

## 6.9 optix\_device.h File Reference

### Macros

- [#define \\_\\_UNDEF\\_OPTIX\\_INCLUDE\\_INTERNAL\\_HEADERS\\_OPTIX\\_DEVICE\\_H\\_\\_](#)

### 6.9.1 Detailed Description

OptiX public API.

Author

NVIDIA Corporation OptiX public API Reference - Host/Device side

### 6.9.2 Macro Definition Documentation

#### 6.9.2.1 [#define \\_\\_UNDEF\\_OPTIX\\_INCLUDE\\_INTERNAL\\_HEADERS\\_OPTIX\\_DEVICE\\_H\\_\\_](#)

## 6.10 optix\_function\_table.h File Reference

### Classes

- struct [OptixFunctionTable](#)

### Macros

- [#define OPTIX\\_ABI\\_VERSION](#) 55

### Typedefs

- typedef struct [OptixFunctionTable](#) [OptixFunctionTable](#)

### 6.10.1 Detailed Description

OptiX public API header.

Author

NVIDIA Corporation

### 6.10.2 Macro Definition Documentation

#### 6.10.2.1 `#define OPTIX_ABI_VERSION 55`

The OptiX ABI version.

## 6.11 `optix_function_table_definition.h` File Reference

### Variables

- [OptixFunctionTable g\\_optixFunctionTable](#)

### 6.11.1 Detailed Description

OptiX public API header.

Author

NVIDIA Corporation

## 6.12 `optix_host.h` File Reference

### Macros

- `#define __UNDEF_OPTIX_INCLUDE_INTERNAL_HEADERS_OPTIX_HOST_H__`

### 6.12.1 Detailed Description

OptiX public API.



Author

NVIDIA Corporation OptiX public API Reference - Host side

## 6.12.2 Macro Definition Documentation

### 6.12.2.1 #define \_\_UNDEF\_OPTIX\_INCLUDE\_INTERNAL\_HEADERS\_OPTIX\_HOST\_H\_\_

## 6.13 optix\_stack\_size.h File Reference

### Functions

- [OptixResult optixUtilAccumulateStackSizes](#) ([OptixProgramGroup](#) programGroup, [OptixStackSizes](#) \*stackSizes)
- [OptixResult optixUtilComputeStackSizes](#) (const [OptixStackSizes](#) \*stackSizes, unsigned int maxTraceDepth, unsigned int maxCCDepth, unsigned int maxDCDepth, unsigned int \*directCallableStackSizeFromTraversal, unsigned int \*directCallableStackSizeFromState, unsigned int \*continuationStackSize)
- [OptixResult optixUtilComputeStackSizesDCSplit](#) (const [OptixStackSizes](#) \*stackSizes, unsigned int dssDCFromTraversal, unsigned int dssDCFromState, unsigned int maxTraceDepth, unsigned int maxCCDepth, unsigned int maxDCDepthFromTraversal, unsigned int maxDCDepthFromState, unsigned int \*directCallableStackSizeFromTraversal, unsigned int \*directCallableStackSizeFromState, unsigned int \*continuationStackSize)
- [OptixResult optixUtilComputeStackSizesCssCCTree](#) (const [OptixStackSizes](#) \*stackSizes, unsigned int cssCCTree, unsigned int maxTraceDepth, unsigned int maxDCDepth, unsigned int \*directCallableStackSizeFromTraversal, unsigned int \*directCallableStackSizeFromState, unsigned int \*continuationStackSize)
- [OptixResult optixUtilComputeStackSizesSimplePathTracer](#) ([OptixProgramGroup](#) programGroupRG, [OptixProgramGroup](#) programGroupMS1, const [OptixProgramGroup](#) \*programGroupCH1, unsigned int programGroupCH1Count, [OptixProgramGroup](#) programGroupMS2, const [OptixProgramGroup](#) \*programGroupCH2, unsigned int programGroupCH2Count, unsigned int \*directCallableStackSizeFromTraversal, unsigned int \*directCallableStackSizeFromState, unsigned int \*continuationStackSize)

### 6.13.1 Detailed Description

OptiX public API header.

Author

NVIDIA Corporation

## 6.14 optix\_stubs.h File Reference

### Macros

- #define [WIN32\\_LEAN\\_AND\\_MEAN](#) 1

## Functions

- static void \* [optixLoadWindowsDllFromName](#) (const char \*optixDllName)
- static void \* [optixLoadWindowsDll](#) ()
- [OptixResult](#) [optixInitWithHandle](#) (void \*\*handlePtr)
- [OptixResult](#) [optixInit](#) (void)
- [OptixResult](#) [optixUninitWithHandle](#) (void \*handle)

## Variables

- [OptixFunctionTable](#) [g\\_optixFunctionTable](#)

### 6.14.1 Detailed Description

OptiX public API header.

Author

NVIDIA Corporation

### 6.14.2 Macro Definition Documentation

#### 6.14.2.1 `#define WIN32_LEAN_AND_MEAN 1`

### 6.14.3 Function Documentation

#### 6.14.3.1 `static void* optixLoadWindowsDll ( ) [static]`

#### 6.14.3.2 `static void* optixLoadWindowsDllFromName ( const char * optixDllName ) [static]`

## 6.15 optix\_types.h File Reference

### Macros

- `#define __OPTIX_INCLUDE_INTERNAL_HEADERS__`
- `#define __UNDEF_OPTIX_INCLUDE_INTERNAL_HEADERS_OPTIX_TYPES_H__`

### 6.15.1 Detailed Description

OptiX public API header.

Author

NVIDIA Corporation

## 6.15.2 Macro Definition Documentation

6.15.2.1 `#define __OPTIX_INCLUDE_INTERNAL_HEADERS__`

6.15.2.2 `#define __UNDEF_OPTIX_INCLUDE_INTERNAL_HEADERS_OPTIX_TYPES_H__`

# Index

- `__OPTIX_INCLUDE_INTERNAL_HEADERS__`
- `optix_types.h`, 351
- `__UNDEF_OPTIX_INCLUDE_INTERNAL_HEADERS__`
- `optix_device.h`, 347
- `__UNDEF_OPTIX_INCLUDE_INTERNAL_HEADERS__`
- `optix_host.h`, 349
- `__UNDEF_OPTIX_INCLUDE_INTERNAL_HEADERS__`
- `optix_types.h`, 351
- a
  - `OptixSRTData`, 227
- `aabbBuffers`
  - `OptixBuildInputCustomPrimitiveArray`, 193
- Acceleration structures, 123
  - `optixAccelBuild`, 123
  - `optixAccelCheckRelocationCompatibility`, 124
  - `optixAccelCompact`, 124
  - `optixAccelComputeMemoryUsage`, 125
  - `optixAccelGetRelocationInfo`, 125
  - `optixAccelRelocate`, 126
  - `optixConvertPointerToTraversableHandle`, 127
- albedo
  - `OptixDenoiserGuideLayer`, 199
- annotation
  - `OptixModuleCompileBoundValueEntry`, 215
- b
  - `OptixSRTData`, 227
- `blendFactor`
  - `OptixDenoiserParams`, 200
- `boundValuePtr`
  - `OptixModuleCompileBoundValueEntry`, 215
- `boundValues`
  - `OptixModuleCompileOptions`, 215
- `buildFlags`
  - `OptixAccelBuildOptions`, 188
  - `OptixBuiltinISOOptions`, 198
- `builtinISModuleType`
  - `OptixBuiltinISOOptions`, 198
- c
  - `OptixSRTData`, 227
- callables
  - `OptixProgramGroupDesc`, 220
- `callablesRecordBase`
  - `OptixShaderBindingTable`, 224
- `callablesRecordCount`
  - `OptixShaderBindingTable`, 224
- `callablesRecordStrideInBytes`
  - `OptixShaderBindingTable`, 224
- `child`
- `OptixMotionTransform`, 214
- `OptixSRTMotionTransform`, 228
- `OptixSRTTransform`, 230
- `cssAH`
  - `OptixStackSizes`, 229
- `cssCC`
  - `OptixStackSizes`, 229
- `cssCH`
  - `OptixStackSizes`, 229
- `cssIS`
  - `OptixStackSizes`, 229
- `cssMS`
  - `OptixStackSizes`, 229
- `cssRG`
  - `OptixStackSizes`, 229
- `CUdeviceptr`
  - Types, 144
- `curveArray`
  - `OptixBuildInput`, 190
- `curveEndcapFlags`
  - `OptixBuiltinISOOptions`, 198
- `curveType`
  - `OptixBuildInputCurveArray`, 191
- `customPrimitiveArray`
  - `OptixBuildInput`, 190
- data
  - `OptixImage2D`, 211
- `debugLevel`
  - `OptixModuleCompileOptions`, 215
  - `OptixPipelineLinkOptions`, 219
- `denoiseAlpha`
  - `OptixDenoiserParams`, 200
- `Denoiser`, 128
  - `optixDenoiserComputeAverageColor`, 128
  - `optixDenoiserComputeIntensity`, 129
  - `optixDenoiserComputeMemoryResources`, 129
  - `optixDenoiserCreate`, 130
  - `optixDenoiserCreateWithUserModel`, 130
  - `optixDenoiserDestroy`, 131
  - `optixDenoiserInvoke`, 131

- [optixDenoiserSetup](#), 133
- Device API, 3
  - [optixContinuationCall](#), 23
  - [optixDirectCall](#), 23
  - [optixGetAttribute\\_0](#), 24
  - [optixGetAttribute\\_1](#), 24
  - [optixGetAttribute\\_2](#), 24
  - [optixGetAttribute\\_3](#), 24
  - [optixGetAttribute\\_4](#), 24
  - [optixGetAttribute\\_5](#), 24
  - [optixGetAttribute\\_6](#), 24
  - [optixGetAttribute\\_7](#), 24
  - [optixGetCatmullRomVertexData](#), 24
  - [optixGetCubicBSplineVertexData](#), 25
  - [optixGetCurveParameter](#), 25
  - [optixGetExceptionCode](#), 25
  - [optixGetExceptionDetail\\_0](#), 25
  - [optixGetExceptionDetail\\_1](#), 25
  - [optixGetExceptionDetail\\_2](#), 26
  - [optixGetExceptionDetail\\_3](#), 26
  - [optixGetExceptionDetail\\_4](#), 26
  - [optixGetExceptionDetail\\_5](#), 26
  - [optixGetExceptionDetail\\_6](#), 26
  - [optixGetExceptionDetail\\_7](#), 26
  - [optixGetExceptionInvalidRay](#), 27
  - [optixGetExceptionInvalidSbtOffset](#), 27
  - [optixGetExceptionInvalidTraversable](#), 27
  - [optixGetExceptionLineInfo](#), 27
  - [optixGetExceptionParameterMismatch](#), 27
  - [optixGetGASMotionStepCount](#), 28
  - [optixGetGASMotionTimeBegin](#), 28
  - [optixGetGASMotionTimeEnd](#), 28
  - [optixGetGASTraversableHandle](#), 28
  - [optixGetHitKind](#), 28
  - [optixGetInstanceChildFromHandle](#), 28
  - [optixGetInstanceId](#), 29
  - [optixGetInstanceIdFromHandle](#), 29
  - [optixGetInstanceIndex](#), 29
  - [optixGetInstanceInverseTransformFromHandle](#), 29
  - [optixGetInstanceTransformFromHandle](#), 29
  - [optixGetInstanceTraversableFromIAS](#), 29
  - [optixGetLaunchDimensions](#), 30
  - [optixGetLaunchIndex](#), 30
  - [optixGetLinearCurveVertexData](#), 30
  - [optixGetMatrixMotionTransformFromHandle](#), 30
  - [optixGetObjectRayDirection](#), 30
  - [optixGetObjectRayOrigin](#), 30
  - [optixGetObjectToWorldTransformMatrix](#), 30
  - [optixGetPayload\\_0](#), 31
  - [optixGetPayload\\_1](#), 31
  - [optixGetPayload\\_10](#), 31
  - [optixGetPayload\\_11](#), 31
  - [optixGetPayload\\_12](#), 31
  - [optixGetPayload\\_13](#), 31
  - [optixGetPayload\\_14](#), 31
  - [optixGetPayload\\_15](#), 31
  - [optixGetPayload\\_16](#), 31
  - [optixGetPayload\\_17](#), 31
  - [optixGetPayload\\_18](#), 31
  - [optixGetPayload\\_19](#), 31
  - [optixGetPayload\\_2](#), 31
  - [optixGetPayload\\_20](#), 32
  - [optixGetPayload\\_21](#), 32
  - [optixGetPayload\\_22](#), 32
  - [optixGetPayload\\_23](#), 32
  - [optixGetPayload\\_24](#), 32
  - [optixGetPayload\\_25](#), 32
  - [optixGetPayload\\_26](#), 32
  - [optixGetPayload\\_27](#), 32
  - [optixGetPayload\\_28](#), 32
  - [optixGetPayload\\_29](#), 32
  - [optixGetPayload\\_3](#), 32
  - [optixGetPayload\\_30](#), 32
  - [optixGetPayload\\_31](#), 32
  - [optixGetPayload\\_4](#), 33
  - [optixGetPayload\\_5](#), 33
  - [optixGetPayload\\_6](#), 33
  - [optixGetPayload\\_7](#), 33
  - [optixGetPayload\\_8](#), 33
  - [optixGetPayload\\_9](#), 33
  - [optixGetPrimitiveIndex](#), 33
  - [optixGetPrimitiveType](#), 33
  - [optixGetQuadraticBSplineVertexData](#), 33
  - [optixGetRayFlags](#), 34
  - [optixGetRayTime](#), 34
  - [optixGetRayTmax](#), 34
  - [optixGetRayTmin](#), 34
  - [optixGetRayVisibilityMask](#), 34
  - [optixGetSbtDataPointer](#), 34
  - [optixGetSbtGASIndex](#), 34
  - [optixGetSRTMotionTransformFromHandle](#), 35
  - [optixGetStaticTransformFromHandle](#), 35
  - [optixGetTransformListHandle](#), 35
  - [optixGetTransformListSize](#), 35
  - [optixGetTransformTypeFromHandle](#), 35

- optixGetTriangleBarycentrics, 35
- optixGetTriangleVertexData, 35
- optixGetWorldRayDirection, 36
- optixGetWorldRayOrigin, 36
- optixGetWorldToObjectTransformMatrix, 36
- optixIgnoreIntersection, 36
- optixIsBackFaceHit, 36
- optixIsFrontFaceHit, 36
- optixIsTriangleBackFaceHit, 36
- optixIsTriangleFrontFaceHit, 37
- optixIsTriangleHit, 37
- optixReportIntersection, 37–39
- optixSetPayload\_0, 40
- optixSetPayload\_1, 40
- optixSetPayload\_10, 40
- optixSetPayload\_11, 40
- optixSetPayload\_12, 40
- optixSetPayload\_13, 40
- optixSetPayload\_14, 40
- optixSetPayload\_15, 40
- optixSetPayload\_16, 41
- optixSetPayload\_17, 41
- optixSetPayload\_18, 41
- optixSetPayload\_19, 41
- optixSetPayload\_2, 41
- optixSetPayload\_20, 41
- optixSetPayload\_21, 41
- optixSetPayload\_22, 41
- optixSetPayload\_23, 41
- optixSetPayload\_24, 41
- optixSetPayload\_25, 42
- optixSetPayload\_26, 42
- optixSetPayload\_27, 42
- optixSetPayload\_28, 42
- optixSetPayload\_29, 42
- optixSetPayload\_3, 42
- optixSetPayload\_30, 42
- optixSetPayload\_31, 42
- optixSetPayload\_4, 42
- optixSetPayload\_5, 42
- optixSetPayload\_6, 43
- optixSetPayload\_7, 43
- optixSetPayload\_8, 43
- optixSetPayload\_9, 43
- optixSetPayloadTypes, 43
- optixTerminateRay, 43
- optixTexFootprint2D, 43
- optixTexFootprint2DGrad, 44
- optixTexFootprint2DLod, 45
- optixThrowException, 46–48
- optixTrace, 48–100, 102, 103
- optixTransformNormalFromObject-  
ToWorldSpace, 104
- optixTransformNormalFromWorldToOb-  
jectSpace, 104
- optixTransformPointFromObject-  
ToWorldSpace, 104
- optixTransformPointFromWorldToOb-  
jectSpace, 105
- optixTransformVectorFromObject-  
ToWorldSpace, 105
- optixTransformVectorFromWorldToOb-  
jectSpace, 105
- optixUndefinedValue, 105
- Device context, 108
  - optixDeviceContextCreate, 108
  - optixDeviceContextDestroy, 109
  - optixDeviceContextGetCacheDatabaseSizes, 109
  - optixDeviceContextGetCacheEnabled, 109
  - optixDeviceContextGetCacheLocation, 109
  - optixDeviceContextGetProperty, 110
  - optixDeviceContextSetCacheDatabaseSizes, 110
  - optixDeviceContextSetCacheEnabled, 111
  - optixDeviceContextSetCacheLocation, 111
  - optixDeviceContextSetLogCallback, 112
- dssDC
  - OptixStackSizes, 229
- endcapFlags
  - OptixBuildInputCurveArray, 192
- entryFunctionName
  - OptixProgramGroupSingleModule, 223
- entryFunctionNameAH
  - OptixProgramGroupHitgroup, 221
- entryFunctionNameCC
  - OptixProgramGroupCallables, 219
- entryFunctionNameCH
  - OptixProgramGroupHitgroup, 221
- entryFunctionNameDC
  - OptixProgramGroupCallables, 219

- entryFunctionNameIs
  - OptixProgramGroupHitgroup, 222
- Error handling, 107
  - optixGetErrorName, 107
  - optixGetErrorString, 107
- exception
  - OptixProgramGroupDesc, 220
- exceptionFlags
  - OptixPipelineCompileOptions, 218
- exceptionRecord
  - OptixShaderBindingTable, 224
- flag
  - OptixBuildInputCurveArray, 192
- flags
  - OptixBuildInputCustomPrimitiveArray, 193
  - OptixBuildInputTriangleArray, 196
  - OptixInstance, 212
  - OptixMotionOptions, 216
  - OptixProgramGroupDesc, 220
- flow
  - OptixDenoiserGuideLayer, 199
- format
  - OptixImage2D, 211
- Function Table, 174
  - g\_optixFunctionTable, 174
  - OptixFunctionTable, 174
- g\_optixFunctionTable
  - Function Table, 174
- guideAlbedo
  - OptixDenoiserOptions, 200
- guideNormal
  - OptixDenoiserOptions, 200
- hdrAverageColor
  - OptixDenoiserParams, 200
- hdrIntensity
  - OptixDenoiserParams, 200
- height
  - OptixImage2D, 211
- hitgroup
  - OptixProgramGroupDesc, 220
- hitgroupRecordBase
  - OptixShaderBindingTable, 224
- hitgroupRecordCount
  - OptixShaderBindingTable, 224
- hitgroupRecordStrideInBytes
  - OptixShaderBindingTable, 224
- Host API, 106
- indexBuffer
  - OptixBuildInputCurveArray, 192
  - OptixBuildInputTriangleArray, 196
- indexFormat
  - OptixBuildInputTriangleArray, 196
- indexStrideInBytes
  - OptixBuildInputCurveArray, 192
  - OptixBuildInputTriangleArray, 196
- info
  - OptixAccelRelocationInfo, 189
- input
  - OptixDenoiserLayer, 199
  - OptixUtilDenoiserImageTile, 231
- inputOffsetX
  - OptixUtilDenoiserImageTile, 231
- inputOffsetY
  - OptixUtilDenoiserImageTile, 231
- instanceArray
  - OptixBuildInput, 190
- instanceId
  - OptixInstance, 212
- instances
  - OptixBuildInputInstanceArray, 195
- invTransform
  - OptixStaticTransform, 230
- kind
  - OptixProgramGroupDesc, 221
- Launches, 121
  - optixLaunch, 121
  - optixSbtRecordPackHeader, 122
- logCallbackData
  - OptixDeviceContextOptions, 202
- logCallbackFunction
  - OptixDeviceContextOptions, 202
- logCallbackLevel
  - OptixDeviceContextOptions, 202
- maxRegisterCount
  - OptixModuleCompileOptions, 216
- maxTraceDepth
  - OptixPipelineLinkOptions, 219
- maxX
  - OptixAabb, 187
- maxY
  - OptixAabb, 187
- maxZ
  - OptixAabb, 187
- minX

- OptixAabb, 187
- minY
  - OptixAabb, 187
- minZ
  - OptixAabb, 187
- miss
  - OptixProgramGroupDesc, 221
- missRecordBase
  - OptixShaderBindingTable, 224
- missRecordCount
  - OptixShaderBindingTable, 225
- missRecordStrideInBytes
  - OptixShaderBindingTable, 225
- module
  - OptixProgramGroupSingleModule, 223
- moduleAH
  - OptixProgramGroupHitgroup, 222
- moduleCC
  - OptixProgramGroupCallables, 219
- moduleCH
  - OptixProgramGroupHitgroup, 222
- moduleDC
  - OptixProgramGroupCallables, 220
- moduleIS
  - OptixProgramGroupHitgroup, 222
- Modules, 115
  - optixBuiltinISModuleGet, 115
  - optixModuleCreateFromPTX, 115
  - optixModuleCreateFromPTXWithTasks, 116
  - optixModuleDestroy, 117
  - optixModuleGetCompilationState, 117
- motionOptions
  - OptixAccelBuildOptions, 188
  - OptixMatrixMotionTransform, 214
  - OptixSRTMotionTransform, 228
- normal
  - OptixDenoiserGuideLayer, 199
- normalBuffers
  - OptixBuildInputCurveArray, 192
- normalStrideInBytes
  - OptixBuildInputCurveArray, 192
- numAttributeValues
  - OptixPipelineCompileOptions, 218
- numBoundValues
  - OptixModuleCompileOptions, 216
- numIndexTriplets
  - OptixBuildInputTriangleArray, 196
- numInstances
  - OptixBuildInputInstanceArray, 195
- numKeys
  - OptixMotionOptions, 216
- numPayloadTypes
  - OptixModuleCompileOptions, 216
- numPayloadValues
  - OptixPayloadType, 217
  - OptixPipelineCompileOptions, 218
- numPrimitives
  - OptixBuildInputCurveArray, 192
  - OptixBuildInputCustomPrimitiveArray, 194
- numSbtRecords
  - OptixBuildInputCustomPrimitiveArray, 194
  - OptixBuildInputTriangleArray, 196
- numVertices
  - OptixBuildInputCurveArray, 192
  - OptixBuildInputTriangleArray, 196
- operation
  - OptixAccelBuildOptions, 188
- optix.h, 231
  - OPTIX\_VERSION, 231
- optix\_7\_device.h, 232
- optix\_7\_device\_impl.h, 251
  - OPTIX\_DEFINE\_optixGetAttribute\_BODY, 270
  - OP-
    - TIX\_DEFINE\_optixGetExceptionDetail\_BODY, 271
- optixContinuationCall, 271
- optixDirectCall, 271
- optixGetAttribute\_0, 272
- optixGetAttribute\_1, 272
- optixGetAttribute\_2, 272
- optixGetAttribute\_3, 272
- optixGetAttribute\_4, 272
- optixGetAttribute\_5, 272
- optixGetAttribute\_6, 272
- optixGetAttribute\_7, 272
- optixGetCatmullRomVertexData, 272
- optixGetCubicBSplineVertexData, 272
- optixGetCurveParameter, 273
- optixGetExceptionCode, 273
- optixGetExceptionDetail\_0, 273
- optixGetExceptionDetail\_1, 273
- optixGetExceptionDetail\_2, 273
- optixGetExceptionDetail\_3, 273
- optixGetExceptionDetail\_4, 273
- optixGetExceptionDetail\_5, 273



- [optixGetExceptionDetail\\_6](#), 273
- [optixGetExceptionDetail\\_7](#), 273
- [optixGetExceptionInvalidRay](#), 273
- [optixGetExceptionInvalidSbtOffset](#), 273
- [optixGetExceptionInvalidTraversable](#), 273
- [optixGetExceptionLineInfo](#), 273
- [optixGetExceptionParameterMismatch](#), 273
- [optixGetGASMotionStepCount](#), 273
- [optixGetGASMotionTimeBegin](#), 273
- [optixGetGASMotionTimeEnd](#), 274
- [optixGetGASTraversableHandle](#), 274
- [optixGetHitKind](#), 274
- [optixGetInstanceChildFromHandle](#), 274
- [optixGetInstanceId](#), 274
- [optixGetInstanceIdFromHandle](#), 274
- [optixGetInstanceIndex](#), 274
- [optixGetInstanceInverseTransformFromHandle](#),  
    274
- [optixGetInstanceTransformFromHandle](#), 274
- [optixGetInstanceTraversableFromIAS](#), 274
- [optixGetLaunchDimensions](#), 274
- [optixGetLaunchIndex](#), 274
- [optixGetLinearCurveVertexData](#), 274
- [optixGetMatrixMotionTransformFromHandle](#),  
    274
- [optixGetObjectRayDirection](#), 275
- [optixGetObjectRayOrigin](#), 275
- [optixGetObjectToWorldTransformMatrix](#), 275
- [optixGetPayload\\_0](#), 276
- [optixGetPayload\\_1](#), 276
- [optixGetPayload\\_10](#), 276
- [optixGetPayload\\_11](#), 276
- [optixGetPayload\\_12](#), 276
- [optixGetPayload\\_13](#), 276
- [optixGetPayload\\_14](#), 276
- [optixGetPayload\\_15](#), 276
- [optixGetPayload\\_16](#), 276
- [optixGetPayload\\_17](#), 276
- [optixGetPayload\\_18](#), 276
- [optixGetPayload\\_19](#), 276
- [optixGetPayload\\_2](#), 276
- [optixGetPayload\\_20](#), 276
- [optixGetPayload\\_21](#), 276
- [optixGetPayload\\_22](#), 276
- [optixGetPayload\\_23](#), 276
- [optixGetPayload\\_24](#), 276
- [optixGetPayload\\_25](#), 276
- [optixGetPayload\\_26](#), 276
- [optixGetPayload\\_27](#), 276
- [optixGetPayload\\_28](#), 276
- [optixGetPayload\\_29](#), 276
- [optixGetPayload\\_3](#), 276
- [optixGetPayload\\_30](#), 276
- [optixGetPayload\\_31](#), 276
- [optixGetPayload\\_4](#), 276
- [optixGetPayload\\_5](#), 276
- [optixGetPayload\\_6](#), 276
- [optixGetPayload\\_7](#), 276
- [optixGetPayload\\_8](#), 276
- [optixGetPayload\\_9](#), 276
- [optixGetPrimitiveIndex](#), 276
- [optixGetPrimitiveType](#), 276, 277
- [optixGetQuadraticBSplineVertexData](#), 277
- [optixGetRayFlags](#), 277
- [optixGetRayTime](#), 277
- [optixGetRayTmax](#), 277
- [optixGetRayTmin](#), 277
- [optixGetRayVisibilityMask](#), 277
- [optixGetSbtDataPointer](#), 277
- [optixGetSbtGASIndex](#), 277
- [optixGetSRTMotionTransformFromHandle](#),  
    277
- [optixGetStaticTransformFromHandle](#), 277
- [optixGetTransformListHandle](#), 277
- [optixGetTransformListSize](#), 277
- [optixGetTransformTypeFromHandle](#), 277
- [optixGetTriangleBarycentrics](#), 278
- [optixGetTriangleVertexData](#), 278
- [optixGetWorldRayDirection](#), 278
- [optixGetWorldRayOrigin](#), 278
- [optixGetWorldToObjectTransformMatrix](#), 278
- [optixIgnoreIntersection](#), 278
- [optixIsBackFaceHit](#), 278
- [optixIsFrontFaceHit](#), 278
- [optixIsTriangleBackFaceHit](#), 278
- [optixIsTriangleFrontFaceHit](#), 278
- [optixIsTriangleHit](#), 278
- [optixReportIntersection](#), 278–280
- [optixSetPayload\\_0](#), 280
- [optixSetPayload\\_1](#), 280
- [optixSetPayload\\_10](#), 280
- [optixSetPayload\\_11](#), 280
- [optixSetPayload\\_12](#), 280
- [optixSetPayload\\_13](#), 280
- [optixSetPayload\\_14](#), 280
- [optixSetPayload\\_15](#), 280
- [optixSetPayload\\_16](#), 280

- optixSetPayload\_17, 280
- optixSetPayload\_18, 281
- optixSetPayload\_19, 281
- optixSetPayload\_2, 281
- optixSetPayload\_20, 281
- optixSetPayload\_21, 281
- optixSetPayload\_22, 281
- optixSetPayload\_23, 281
- optixSetPayload\_24, 281
- optixSetPayload\_25, 281
- optixSetPayload\_26, 281
- optixSetPayload\_27, 281
- optixSetPayload\_28, 281
- optixSetPayload\_29, 281
- optixSetPayload\_3, 281
- optixSetPayload\_30, 281
- optixSetPayload\_31, 282
- optixSetPayload\_4, 282
- optixSetPayload\_5, 282
- optixSetPayload\_6, 282
- optixSetPayload\_7, 282
- optixSetPayload\_8, 282
- optixSetPayload\_9, 282
- optixSetPayloadTypes, 282
- optixTerminateRay, 282
- optixTexFootprint2D, 282
- optixTexFootprint2DGrad, 282
- optixTexFootprint2DLod, 283
- optixThrowException, 283, 284
- optixTrace, 284–329
- optixTransformNormalFromObject-  
ToWorldSpace,  
331
- optixTransformNormalFromWorldToOb-  
jectSpace,  
331
- optixTransformPointFromObject-  
ToWorldSpace,  
331
- optixTransformPointFromWorldToOb-  
jectSpace,  
331
- optixTransformVectorFromObject-  
ToWorldSpace,  
331
- optixTransformVectorFromWorldToOb-  
jectSpace,  
331
- optixUndefinedValue, 331
- optix\_7\_device\_impl\_exception.h, 331
- optix\_7\_device\_impl\_transformations.h, 332
- optix\_7\_host.h, 334
- optix\_7\_types.h, 336
- OPTIX\_AABB\_BUFFER\_BYTE\_ALIGNMENT  
Types, 143
- OPTIX\_ABI\_VERSION  
optix\_function\_table.h, 348
- OPTIX\_ACCEL\_BUFFER\_BYTE\_ALIGNMENT  
Types, 143
- OPTIX\_BUILD\_FLAG\_ALLOW\_COMPACTION  
Types, 158
- OPTIX\_BUILD\_FLAG\_ALLOW\_RANDOM\_INSTANCE\_ACCESS  
Types, 158
- OPTIX\_BUILD\_FLAG\_ALLOW\_RANDOM\_VERTEX\_ACCESS  
Types, 158
- OPTIX\_BUILD\_FLAG\_ALLOW\_UPDATE  
Types, 158
- OPTIX\_BUILD\_FLAG\_NONE  
Types, 158
- OPTIX\_BUILD\_FLAG\_PREFER\_FAST\_BUILD  
Types, 158
- OPTIX\_BUILD\_FLAG\_PREFER\_FAST\_TRACE  
Types, 158
- OPTIX\_BUILD\_INPUT\_TYPE\_CURVES  
Types, 159
- OPTIX\_BUILD\_INPUT\_TYPE\_CUSTOM\_PRIMITIVES  
Types, 159
- OPTIX\_BUILD\_INPUT\_TYPE\_INSTANCE\_POINTERS  
Types, 159
- OPTIX\_BUILD\_INPUT\_TYPE\_INSTANCES  
Types, 159
- OPTIX\_BUILD\_INPUT\_TYPE\_TRIANGLES  
Types, 159
- OPTIX\_BUILD\_OPERATION\_BUILD  
Types, 159
- OPTIX\_BUILD\_OPERATION\_UPDATE  
Types, 159
- OPTIX\_COMPILE\_DEBUG\_LEVEL\_DEFAULT  
Types, 160
- OPTIX\_COMPILE\_DEBUG\_LEVEL\_FULL  
Types, 160
- OPTIX\_COMPILE\_DEBUG\_LEVEL\_MINIMAL  
Types, 160
- OPTIX\_COMPILE\_DEBUG\_LEVEL\_MODERATE  
Types, 160
- OPTIX\_COMPILE\_DEBUG\_LEVEL\_NONE  
Types, 160
- OPTIX\_COMPILE\_DEFAULT\_MAX\_PAYLOAD\_TYPE\_COUNT

- Types, 143
- OPTIX\_COMPILE\_DEFAULT\_MAX\_PAYLOAD\_VALUE
  - Types, 143
- OPTIX\_COMPILE\_DEFAULT\_MAX\_REGISTER\_COUNT
  - Types, 144
- OPTIX\_COMPILE\_OPTIMIZATION\_DEFAULT
  - Types, 160
- OPTIX\_COMPILE\_OPTIMIZATION\_LEVEL\_0
  - Types, 160
- OPTIX\_COMPILE\_OPTIMIZATION\_LEVEL\_1
  - Types, 160
- OPTIX\_COMPILE\_OPTIMIZATION\_LEVEL\_2
  - Types, 160
- OPTIX\_COMPILE\_OPTIMIZATION\_LEVEL\_3
  - Types, 160
- OPTIX\_CURVE\_ENDCAP\_DEFAULT
  - Types, 160
- OPTIX\_CURVE\_ENDCAP\_ON
  - Types, 160
- OPTIX\_DEFINE\_optixGetAttribute\_BODY
  - optix\_7\_device\_impl.h, 270
- OPTIX\_DEFINE\_optixGetExceptionDetail\_BODY
  - optix\_7\_device\_impl.h, 271
- OPTIX\_DENOISER\_MODEL\_KIND\_AOV
  - Types, 161
- OPTIX\_DENOISER\_MODEL\_KIND\_HDR
  - Types, 161
- OPTIX\_DENOISER\_MODEL\_KIND\_LDR
  - Types, 161
- OPTIX\_DENOISER\_MODEL\_KIND\_TEMPORAL
  - Types, 161
- OPTIX\_DENOISER\_MODEL\_KIND\_TEMPORAL\_AOV
  - Types, 161
- optix\_denoiser\_tiling.h, 346
- optix\_device.h, 347
  - \_\_UN-
    - DEF\_OPTIX\_INCLUDE\_INTERNAL\_HEADERSTOP, 347
- OPTIX\_DEVICE\_CONTEXT\_VALIDATION\_MODE\_ALL
  - Types, 161
- OPTIX\_DEVICE\_CONTEXT\_VALIDATION\_MODE\_OFF
  - Types, 161
- OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_INSTANCE\_ID
  - Types, 162
- OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_INSTANCE\_SIZE
  - Types, 161
- OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_PRIMITIVE\_SIZE
  - Types, 161
- OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_SBT\_OFFSET
  - Types, 162
- OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_SBT\_RECORDS\_PER\_CONTEXT
  - Types, 162
- OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_TRACE\_DEPTH
  - Types, 161
- OPTIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_TRAVERSABLE\_GRAPH\_SIZE
  - Types, 161
- OPTIX\_DEVICE\_PROPERTY\_LIMIT\_NUM\_BITS\_INSTANCE\_VISIBILITY
  - Types, 162
- OPTIX\_DEVICE\_PROPERTY\_RTCORE\_VERSION
  - Types, 162
- OPTIX\_ERROR\_ACCEL\_NOT\_COMPATIBLE
  - Types, 171
- OPTIX\_ERROR\_CUDA\_ERROR
  - Types, 171
- OPTIX\_ERROR\_CUDA\_NOT\_INITIALIZED
  - Types, 170
- OPTIX\_ERROR\_DENOISER\_MODEL\_NOT\_SET
  - Types, 171
- OPTIX\_ERROR\_DENOISER\_NOT\_INITIALIZED
  - Types, 171
- OPTIX\_ERROR\_DISK\_CACHE\_DATABASE\_ERROR
  - Types, 170
- OPTIX\_ERROR\_DISK\_CACHE\_INVALID\_DATA
  - Types, 170
- OPTIX\_ERROR\_DISK\_CACHE\_INVALID\_PATH
  - Types, 170
- OPTIX\_ERROR\_DISK\_CACHE\_PERMISSION\_ERROR
  - Types, 170
- OPTIX\_ERROR\_ENTRY\_SYMBOL\_NOT\_FOUND
  - Types, 171
- OPTIX\_ERROR\_FILE\_IO\_ERROR
  - Types, 170
- OPTIX\_ERROR\_FUNCTION\_TABLE\_SIZE\_MISMATCH
  - Types, 171
- OPTIX\_ERROR\_HOST\_OUT\_OF\_MEMORY
  - Types, 171
- OPTIX\_ERROR\_ILLEGAL\_DURING\_TASK\_EXECUTE
  - Types, 171
- OPTIX\_ERROR\_INTERNAL\_COMPILER\_ERROR
  - Types, 171
- OPTIX\_ERROR\_INTERNAL\_ERROR
  - Types, 171
- OPTIX\_ERROR\_INVALID\_ATTRIBUTE\_ACCESS
  - Types, 171
- OPTIX\_ERROR\_INVALID\_DEVICE\_CONTEXT
  - Types, 171
- OPTIX\_ERROR\_INVALID\_ENTRY\_FUNCTION\_OPTIONS
  - Types, 171

- OPTIX\_ERROR\_INVALID\_FILE\_FORMAT
  - Types, 170
- OPTIX\_ERROR\_INVALID\_FUNCTION\_ARGUMENTS
  - Types, 170
- OPTIX\_ERROR\_INVALID\_FUNCTION\_USE
  - Types, 170
- OPTIX\_ERROR\_INVALID\_LAUNCH\_PARAMETER
  - Types, 170
- OPTIX\_ERROR\_INVALID\_OPERATION
  - Types, 170
- OPTIX\_ERROR\_INVALID\_PAYLOAD\_ACCESS
  - Types, 170
- OPTIX\_ERROR\_INVALID\_PTX
  - Types, 170
- OPTIX\_ERROR\_INVALID\_VALUE
  - Types, 170
- OPTIX\_ERROR\_LAUNCH\_FAILURE
  - Types, 170
- OPTIX\_ERROR\_LIBRARY\_NOT\_FOUND
  - Types, 171
- OPTIX\_ERROR\_LIBRARY\_UNLOAD\_FAILURE
  - Types, 171
- OPTIX\_ERROR\_NOT\_SUPPORTED
  - Types, 171
- OPTIX\_ERROR\_PAYLOAD\_TYPE\_ID\_INVALID
  - Types, 171
- OPTIX\_ERROR\_PAYLOAD\_TYPE\_MISMATCH
  - Types, 171
- OPTIX\_ERROR\_PAYLOAD\_TYPE\_RESOLUTION\_FAILURE
  - Types, 171
- OPTIX\_ERROR\_PIPELINE\_LINK\_ERROR
  - Types, 171
- OPTIX\_ERROR\_PIPELINE\_OUT\_OF\_CONSTANT\_MEMORY
  - Types, 171
- OPTIX\_ERROR\_UNKNOWN
  - Types, 171
- OPTIX\_ERROR\_UNSUPPORTED\_ABI\_VERSION
  - Types, 171
- OPTIX\_ERROR\_VALIDATION\_FAILURE
  - Types, 170
- OPTIX\_EXCEPTION\_CODE\_BUILTIN\_IS\_MISMATCH
  - Types, 163
- OPTIX\_EXCEPTION\_CODE\_CALLABLE\_INVALID\_ID
  - Types, 163
- OPTIX\_EXCEPTION\_CODE\_CALLABLE\_NO\_CC\_SUPPORT
  - Types, 163
- OPTIX\_EXCEPTION\_CODE\_CALLABLE\_NO\_DC\_SUPPORT
  - Types, 163
- OPTIX\_EXCEPTION\_CODE\_CALLABLE\_PARAMETER\_MISMATCH
  - Types, 163
- Types, 163
- OPTIX\_EXCEPTION\_CODE\_INVALID\_RAY
  - Types, 162
- OPTIX\_EXCEPTION\_CODE\_INVALID\_VALUE\_ARGUMENT\_0
  - Types, 163
- OPTIX\_EXCEPTION\_CODE\_INVALID\_VALUE\_ARGUMENT\_1
  - Types, 163
- OPTIX\_EXCEPTION\_CODE\_INVALID\_VALUE\_ARGUMENT\_2
  - Types, 163
- OPTIX\_EXCEPTION\_CODE\_PAYLOAD\_TYPE\_MISMATCH
  - Types, 163
- OPTIX\_EXCEPTION\_CODE\_STACK\_OVERFLOW
  - Types, 162
- OPTIX\_EXCEPTION\_CODE\_TRACE\_DEPTH\_EXCEEDED
  - Types, 162
- OPTIX\_EXCEPTION\_CODE\_TRAVERSAL\_DEPTH\_EXCEEDED
  - Types, 162
- OPTIX\_EXCEPTION\_CODE\_TRAVERSAL\_INVALID\_HIT\_SBT
  - Types, 162
- OPTIX\_EXCEPTION\_CODE\_TRAVERSAL\_INVALID\_MISS\_SBT
  - Types, 162
- OPTIX\_EXCEPTION\_CODE\_TRAVERSAL\_INVALID\_TRAVERSAL
  - Types, 162
- OPTIX\_EXCEPTION\_CODE\_UNSUPPORTED\_DATA\_ACCESS
  - Types, 163
- OPTIX\_EXCEPTION\_CODE\_UNSUPPORTED\_PRIMITIVE\_TYPE
  - Types, 162
- OPTIX\_EXCEPTION\_CODE\_UNSUPPORTED\_SINGLE\_LEVEL
  - Types, 163
- OPTIX\_EXCEPTION\_FLAG\_DEBUG
  - Types, 164
- OPTIX\_EXCEPTION\_FLAG\_NONE
  - Types, 163
- OPTIX\_EXCEPTION\_FLAG\_STACK\_OVERFLOW
  - Types, 163
- OPTIX\_EXCEPTION\_FLAG\_TRACE\_DEPTH
  - Types, 163
- OPTIX\_EXCEPTION\_FLAG\_USER
  - Types, 163
- optix\_function\_table.h, 347
- OPTIX\_ABI\_VERSION, 348
- optix\_function\_table\_definition.h, 348
- OPTIX\_GEOMETRY\_FLAG\_DISABLE\_ANYHIT
  - Types, 164
- OPTIX\_GEOMETRY\_FLAG\_NONE
  - Types, 164
- OPTIX\_GEOMETRY\_FLAG\_REQUIRE\_SINGLE\_ANYHIT\_CALL
  - Types, 164
- OPTIMIZATION
- OPTIMIZATION\_GEOMETRY\_TRANSFORM\_BYTE\_ALIGNMENT

- Types, 144
- OPTIX\_HIT\_KIND\_TRIANGLE\_BACK\_FACE
  - Types, 164
- OPTIX\_HIT\_KIND\_TRIANGLE\_FRONT\_FACE
  - Types, 164
- optix\_host.h, 348
- \_\_UN-
  - DEF\_OPTIX\_INCLUDE\_INTERNAL\_HEADERS, 349
- optix\_impl, 182
  - optixAddFloat4, 183
  - optixDumpExceptionDetails, 183
  - optixDumpInstanceFromHandle, 183
  - optixDumpMotionMatrixTransformFromHandle, 183
  - optixDumpSrtMatrixTransformFromHandle, 183
  - optixDumpStaticTransformFromHandle, 183
  - optixDumpTransform, 184
  - optixDumpTransformList, 184
  - optixGetInterpolatedTransformation, 184
  - optixGetInterpolatedTransformationFromHandle, 184
  - optixGetMatrixFromSrt, 184
  - optixGetObjectToWorldTransformMatrix, 184
  - optixGetWorldToObjectTransformMatrix, 185
  - optixInvertMatrix, 185
  - optixLdg, 185
  - optixLoadInterpolatedMatrixKey, 185
  - optixLoadInterpolatedSrtKey, 185
  - optixLoadReadOnlyAlign16, 185
  - optixMulFloat4, 185
  - optixMultiplyRowMatrix, 185
  - optixResolveMotionKey, 186
  - optixTransformNormal, 186
  - optixTransformPoint, 186
  - optixTransformVector, 186
- OPTIX\_INDICES\_FORMAT\_NONE
  - Types, 164
- OPTIX\_INDICES\_FORMAT\_UNSIGNED\_INT3
  - Types, 164
- OPTIX\_INDICES\_FORMAT\_UNSIGNED\_SHORT3
  - Types, 164
- OPTIX\_INSTANCE\_BYTE\_ALIGNMENT
  - Types, 144
- OPTIX\_INSTANCE\_FLAG\_DISABLE\_ANYHIT
  - Types, 165
- OPTIX\_INSTANCE\_FLAG\_DISABLE\_TRIANGLE\_FACE\_CULLING
  - Types, 165
- OPTIX\_INSTANCE\_FLAG\_ENFORCE\_ANYHIT
  - Types, 165
- OPTIX\_INSTANCE\_FLAG\_FLIP\_TRIANGLE\_FACING
  - Types, 165
- OPTIX\_INSTANCE\_FLAG\_NONE
  - Types, 165
- OPTIX\_MODULE\_COMPILE\_STATE\_COMPLETED
  - Types, 165
- OPTIX\_MODULE\_COMPILE\_STATE\_FAILED
  - Types, 165
- OPTIX\_MODULE\_COMPILE\_STATE\_IMPENDING\_FAILURE
  - Types, 165
- OPTIX\_MODULE\_COMPILE\_STATE\_NOT\_STARTED
  - Types, 165
- OPTIX\_MODULE\_COMPILE\_STATE\_STARTED
  - Types, 165
- OPTIX\_MOTION\_FLAG\_END\_VANISH
  - Types, 165
- OPTIX\_MOTION\_FLAG\_NONE
  - Types, 165
- OPTIX\_MOTION\_FLAG\_START\_VANISH
  - Types, 165
- OPTIX\_PAYLOAD\_SEMANTICS\_AH\_NONE
  - Types, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_AH\_READ
  - Types, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_AH\_READ\_WRITE
  - Types, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_AH\_WRITE
  - Types, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_CH\_NONE
  - Types, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_CH\_READ
  - Types, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_CH\_READ\_WRITE
  - Types, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_CH\_WRITE
  - Types, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_IS\_NONE
  - Types, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_IS\_READ
  - Types, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_IS\_READ\_WRITE
  - Types, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_IS\_WRITE
  - Types, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_MS\_NONE

- Types, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_MS\_READ
  - Types, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_MS\_READ\_WRITE
  - Types, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_MS\_WRITE
  - Types, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_TRACE\_CALLER\_NONE
  - Types, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_TRACE\_CALLER\_READ
  - Types, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_TRACE\_CALLER\_READ\_WRITE
  - Types, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_TRACE\_CALLER\_WRITE
  - Types, 166
- OPTIX\_PAYLOAD\_TYPE\_DEFAULT
  - Types, 166
- OPTIX\_PAYLOAD\_TYPE\_ID\_0
  - Types, 166
- OPTIX\_PAYLOAD\_TYPE\_ID\_1
  - Types, 166
- OPTIX\_PAYLOAD\_TYPE\_ID\_2
  - Types, 166
- OPTIX\_PAYLOAD\_TYPE\_ID\_3
  - Types, 167
- OPTIX\_PAYLOAD\_TYPE\_ID\_4
  - Types, 167
- OPTIX\_PAYLOAD\_TYPE\_ID\_5
  - Types, 167
- OPTIX\_PAYLOAD\_TYPE\_ID\_6
  - Types, 167
- OPTIX\_PAYLOAD\_TYPE\_ID\_7
  - Types, 167
- OPTIX\_PIXEL\_FORMAT\_FLOAT2
  - Types, 167
- OPTIX\_PIXEL\_FORMAT\_FLOAT3
  - Types, 167
- OPTIX\_PIXEL\_FORMAT\_FLOAT4
  - Types, 167
- OPTIX\_PIXEL\_FORMAT\_HALF2
  - Types, 167
- OPTIX\_PIXEL\_FORMAT\_HALF3
  - Types, 167
- OPTIX\_PIXEL\_FORMAT\_HALF4
  - Types, 167
- OPTIX\_PIXEL\_FORMAT\_UCHAR3
  - Types, 167
- OPTIX\_PIXEL\_FORMAT\_UCHAR4
  - Types, 167
- OPTIX\_PRIMITIVE\_TYPE\_CUSTOM
  - Types, 167
- OPTIX\_PRIMITIVE\_TYPE\_FLAGS\_CUSTOM
  - Types, 168
- OPTIX\_PRIMITIVE\_TYPE\_FLAGS\_ROUND\_CATMULLROM
  - Types, 168
- OPTIX\_PRIMITIVE\_TYPE\_FLAGS\_ROUND\_CUBIC\_BSPLINE
  - Types, 168
- OPTIX\_PRIMITIVE\_TYPE\_FLAGS\_ROUND\_LINEAR
  - Types, 168
- OPTIX\_PRIMITIVE\_TYPE\_FLAGS\_ROUND\_QUADRATIC\_BSPLINE
  - Types, 168
- OPTIX\_PRIMITIVE\_TYPE\_FLAGS\_TRIANGLE
  - Types, 168
- OPTIX\_PRIMITIVE\_TYPE\_ROUND\_CATMULLROM
  - Types, 167
- OPTIX\_PRIMITIVE\_TYPE\_ROUND\_CUBIC\_BSPLINE
  - Types, 167
- OPTIX\_PRIMITIVE\_TYPE\_ROUND\_LINEAR
  - Types, 167
- OPTIX\_PRIMITIVE\_TYPE\_ROUND\_QUADRATIC\_BSPLINE
  - Types, 167
- OPTIX\_PRIMITIVE\_TYPE\_TRIANGLE
  - Types, 167
- OPTIX\_PROGRAM\_GROUP\_FLAGS\_NONE
  - Types, 168
- OPTIX\_PROGRAM\_GROUP\_KIND\_CALLABLES
  - Types, 169
- OPTIX\_PROGRAM\_GROUP\_KIND\_EXCEPTION
  - Types, 168
- OPTIX\_PROGRAM\_GROUP\_KIND\_HITGROUP
  - Types, 168
- OPTIX\_PROGRAM\_GROUP\_KIND\_MISS
  - Types, 168
- OPTIX\_PROGRAM\_GROUP\_KIND\_RAYGEN
  - Types, 168
- OPTIX\_PROPERTY\_TYPE\_AABBS
  - Types, 158
- OPTIX\_PROPERTY\_TYPE\_COMPACTED\_SIZE
  - Types, 158
- OPTIX\_QUERY\_FUNCTION\_TABLE\_OPTION\_DUMMY
  - Types, 169
- OPTIX\_RAY\_FLAG\_CULL\_BACK\_FACING\_TRIANGLES
  - Types, 169
- OPTIX\_RAY\_FLAG\_CULL\_DISABLED\_ANYHIT
  - Types, 169
- OPTIX\_RAY\_FLAG\_CULL\_ENFORCED\_ANYHIT
  - Types, 170
- OPTIX\_RAY\_FLAG\_CULL\_FRONT\_FACING\_TRIANGLES



- Types, 169
- OPTIX\_RAY\_FLAG\_DISABLE\_ANYHIT
  - Types, 169
- OPTIX\_RAY\_FLAG\_DISABLE\_CLOSESTHIT
  - Types, 169
- OPTIX\_RAY\_FLAG\_ENFORCE\_ANYHIT
  - Types, 169
- OPTIX\_RAY\_FLAG\_NONE
  - Types, 169
- OPTIX\_RAY\_FLAG\_TERMINATE\_ON\_FIRST\_HIT
  - Types, 169
- OPTIX\_SBT\_RECORD\_ALIGNMENT
  - Types, 144
- OPTIX\_SBT\_RECORD\_HEADER\_SIZE
  - Types, 144
- optix\_stack\_size.h, 349
- optix\_stubs.h, 349
  - optixLoadWindowsDll, 350
  - optixLoadWindowsDllFromName, 350
  - WIN32\_LEAN\_AND\_MEAN, 350
- OPTIX\_SUCCESS
  - Types, 170
- OPTIX\_TRANSFORM\_BYTE\_ALIGNMENT
  - Types, 144
- OPTIX\_TRANSFORM\_FORMAT\_MATRIX\_FLOAT12
  - Types, 171
- OPTIX\_TRANSFORM\_FORMAT\_NONE
  - Types, 171
- OPTIX\_TRANSFORM\_TYPE\_INSTANCE
  - Types, 172
- OPTIX\_TRANSFORM\_TYPE\_MATRIX\_MOTION\_TRANSFORM
  - Types, 171
- OPTIX\_TRANSFORM\_TYPE\_NONE
  - Types, 171
- OPTIX\_TRANSFORM\_TYPE\_SRT\_MOTION\_TRANSFORM
  - Types, 172
- OPTIX\_TRANSFORM\_TYPE\_STATIC\_TRANSFORM
  - Types, 171
- OPTIX\_TRAVERSABLE\_GRAPH\_FLAG\_ALLOW\_ANY
  - Types, 172
- OPTIX\_TRAVERSABLE\_GRAPH\_FLAG\_ALLOW\_SINGLE\_PASS
  - Types, 172
- OPTIX\_TRAVERSABLE\_GRAPH\_FLAG\_ALLOW\_SINGLE\_PASS\_INSTANCE
  - Types, 172
- OPTIX\_TRAVERSABLE\_TYPE\_MATRIX\_MOTION\_TRANSFORM
  - Types, 172
- OPTIX\_TRAVERSABLE\_TYPE\_SRT\_MOTION\_TRANSFORM
  - Types, 172
- OPTIX\_TRAVERSABLE\_TYPE\_STATIC\_TRANSFORM
  - Types, 172
- Types, 172
- optix\_types.h, 350
  - \_\_OP-
    - TIX\_INCLUDE\_INTERNAL\_HEADERS\_\_, 351
  - \_\_UN-
    - DEF\_OPTIX\_INCLUDE\_INTERNAL\_HEADERS\_OPTIX\_\_, 351
- OPTIX\_VERSION
  - optix.h, 231
- OPTIX\_VERTEX\_FORMAT\_FLOAT2
  - Types, 173
- OPTIX\_VERTEX\_FORMAT\_FLOAT3
  - Types, 173
- OPTIX\_VERTEX\_FORMAT\_HALF2
  - Types, 173
- OPTIX\_VERTEX\_FORMAT\_HALF3
  - Types, 173
- OPTIX\_VERTEX\_FORMAT\_NONE
  - Types, 173
- OPTIX\_VERTEX\_FORMAT\_SNORM16\_2
  - Types, 173
- OPTIX\_VERTEX\_FORMAT\_SNORM16\_3
  - Types, 173
- OptixAabb, 186
  - maxX, 187
  - maxY, 187
  - maxZ, 187
  - minX, 187
  - minY, 187
  - minZ, 187
  - Types, 144
- OptixAccelBufferSizes, 187
  - outputSizeInBytes, 187
  - tempSizeInBytes, 187
  - tempUpdateSizeInBytes, 188
  - Types, 144
- optixAccelBuild
  - Acceleration structures, 123
  - OptixFunctionTable, 205
  - OptixAccelBuildOptions, 188
  - buildFlags, 188
  - FunctionTable, 205
  - motionOptions, 188
  - operation, 188
  - Types, 144
- optixAccelCheckRelocationCompatibility
  - Acceleration structures, 124
  - OptixFunctionTable, 205
- OptixAccelCompact
  - Acceleration structures, 124

- Acceleration structures, 124
- OptixFunctionTable, 205
- optixAccelComputeMemoryUsage
  - Acceleration structures, 125
  - OptixFunctionTable, 205
- OptixAccelEmitDesc, 189
  - result, 189
  - type, 189
  - Types, 145
- optixAccelGetRelocationInfo
  - Acceleration structures, 125
  - OptixFunctionTable, 205
- OptixAccelPropertyType
  - Types, 145, 158
- optixAccelRelocate
  - Acceleration structures, 126
  - OptixFunctionTable, 205
- OptixAccelRelocationInfo, 189
  - info, 189
  - Types, 145
- optixAddFloat4
  - optix\_impl, 183
- OptixBuildFlags
  - Types, 145, 158
- OptixBuildInput, 190
  - curveArray, 190
  - customPrimitiveArray, 190
  - instanceArray, 190
  - pad, 190
  - triangleArray, 190
  - type, 190
  - Types, 145
- OptixBuildInputCurveArray, 191
  - curveType, 191
  - endcapFlags, 192
  - flag, 192
  - indexBuffer, 192
  - indexStrideInBytes, 192
  - normalBuffers, 192
  - normalStrideInBytes, 192
  - numPrimitives, 192
  - numVertices, 192
  - primitiveIndexOffset, 192
  - Types, 145
  - vertexBuffers, 192
  - vertexStrideInBytes, 192
  - widthBuffers, 193
  - widthStrideInBytes, 193
- OptixBuildInputCustomPrimitiveArray, 193
  - aabbBuffers, 193
  - flags, 193
  - numPrimitives, 194
  - numSbtRecords, 194
  - primitiveIndexOffset, 194
  - sbtIndexOffsetBuffer, 194
  - sbtIndexOffsetSizeInBytes, 194
  - sbtIndexOffsetStrideInBytes, 194
  - strideInBytes, 194
  - Types, 146
- OptixBuildInputInstanceArray, 194
  - instances, 195
  - numInstances, 195
  - Types, 146
- OptixBuildInputTriangleArray, 195
  - flags, 196
  - indexBuffer, 196
  - indexFormat, 196
  - indexStrideInBytes, 196
  - numIndexTriplets, 196
  - numSbtRecords, 196
  - numVertices, 196
  - preTransform, 196
  - primitiveIndexOffset, 196
  - sbtIndexOffsetBuffer, 196
  - sbtIndexOffsetSizeInBytes, 197
  - sbtIndexOffsetStrideInBytes, 197
  - transformFormat, 197
  - Types, 146
  - vertexBuffers, 197
  - vertexFormat, 197
  - vertexStrideInBytes, 197
- OptixBuildInputType
  - Types, 146, 158
- OptixBuildOperation
  - Types, 146, 159
- optixBuiltinISModuleGet
  - Modules, 115
  - OptixFunctionTable, 206
- OptixBuiltinISOptions, 197
  - buildFlags, 198
  - builtinISModuleType, 198
  - curveEndcapFlags, 198
  - Types, 147
  - usesMotionBlur, 198
- OptixCompileDebugLevel
  - Types, 147, 159
- OptixCompileOptimizationLevel
  - Types, 147, 160



- optixContinuationCall
  - Device API, 23
  - optix\_7\_device\_impl.h, 271
- optixConvertPointerToTraversableHandle
  - Acceleration structures, 127
  - OptixFunctionTable, 206
- OptixCurveEndcapFlags
  - Types, 147, 160
- OptixDenoiser
  - Types, 147
- optixDenoiserComputeAverageColor
  - Denoiser, 128
  - OptixFunctionTable, 206
- optixDenoiserComputeIntensity
  - Denoiser, 129
  - OptixFunctionTable, 206
- optixDenoiserComputeMemoryResources
  - Denoiser, 129
  - OptixFunctionTable, 206
- optixDenoiserCreate
  - Denoiser, 130
  - OptixFunctionTable, 206
- optixDenoiserCreateWithUserModel
  - Denoiser, 130
  - OptixFunctionTable, 206
- optixDenoiserDestroy
  - Denoiser, 131
  - OptixFunctionTable, 207
- OptixDenoiserGuideLayer, 198
  - albedo, 199
  - flow, 199
  - normal, 199
  - Types, 147
- optixDenoiserInvoke
  - Denoiser, 131
  - OptixFunctionTable, 207
- OptixDenoiserLayer, 199
  - input, 199
  - output, 199
  - previousOutput, 199
  - Types, 147
- OptixDenoiserModelKind
  - Types, 148, 160
- OptixDenoiserOptions, 199
  - guideAlbedo, 200
  - guideNormal, 200
  - Types, 148
- OptixDenoiserParams, 200
  - blendFactor, 200
  - denoiseAlpha, 200
  - hdrAverageColor, 200
  - hdrIntensity, 200
  - Types, 148
- optixDenoiserSetup
  - Denoiser, 133
  - OptixFunctionTable, 207
- OptixDenoiserSizes, 201
  - overlapWindowSizeInPixels, 201
  - stateSizeInBytes, 201
  - Types, 148
  - withoutOverlapScratchSizeInBytes, 201
  - withOverlapScratchSizeInBytes, 201
- OptixDeviceContext
  - Types, 148
- optixDeviceContextCreate
  - Device context, 108
  - OptixFunctionTable, 207
- optixDeviceContextDestroy
  - Device context, 109
  - OptixFunctionTable, 207
- optixDeviceContextGetCacheDatabaseSizes
  - Device context, 109
  - OptixFunctionTable, 207
- optixDeviceContextGetCacheEnabled
  - Device context, 109
  - OptixFunctionTable, 207
- optixDeviceContextGetCacheLocation
  - Device context, 109
  - OptixFunctionTable, 207
- optixDeviceContextGetProperty
  - Device context, 110
  - OptixFunctionTable, 208
- OptixDeviceContextOptions, 201
  - logCallbackData, 202
  - logCallbackFunction, 202
  - logCallbackLevel, 202
  - Types, 148
  - validationMode, 202
- optixDeviceContextSetCacheDatabaseSizes
  - Device context, 110
  - OptixFunctionTable, 208
- optixDeviceContextSetCacheEnabled
  - Device context, 111
  - OptixFunctionTable, 208
- optixDeviceContextSetCacheLocation
  - Device context, 111
  - OptixFunctionTable, 208
- optixDeviceContextSetLogCallback

- Device context, 112
- OptixFunctionTable, 208
- OptixDeviceContextValidationMode
  - Types, 148, 161
- OptixDeviceProperty
  - Types, 149, 161
- optixDirectCall
  - Device API, 23
  - optix\_7\_device\_impl.h, 271
- optixDumpExceptionDetails
  - optix\_impl, 183
- optixDumpInstanceFromHandle
  - optix\_impl, 183
- optixDumpMotionMatrixTransformFromHandle
  - optix\_impl, 183
- optixDumpSrtMatrixTransformFromHandle
  - optix\_impl, 183
- optixDumpStaticTransformFromHandle
  - optix\_impl, 183
- optixDumpTransform
  - optix\_impl, 184
- optixDumpTransformList
  - optix\_impl, 184
- OptixExceptionCodes
  - Types, 149, 162
- OptixExceptionFlags
  - Types, 149, 163
- OptixFunctionTable, 202
  - Function Table, 174
  - optixAccelBuild, 205
  - optixAccelCheckRelocationCompatibility, 205
  - optixAccelCompact, 205
  - optixAccelComputeMemoryUsage, 205
  - optixAccelGetRelocationInfo, 205
  - optixAccelRelocate, 205
  - optixBuiltinISModuleGet, 206
  - optixConvertPointerToTraversableHandle, 206
  - optixDenoiserComputeAverageColor, 206
  - optixDenoiserComputeIntensity, 206
  - optixDenoiserComputeMemoryResources, 206
  - optixDenoiserCreate, 206
  - optixDenoiserCreateWithUserModel, 206
  - optixDenoiserDestroy, 207
  - optixDenoiserInvoke, 207
  - optixDenoiserSetup, 207
  - optixDeviceContextCreate, 207
  - optixDeviceContextDestroy, 207
  - optixDeviceContextGetCacheDatabaseSizes, 207
  - optixDeviceContextGetCacheEnabled, 207
  - optixDeviceContextGetCacheLocation, 207
  - optixDeviceContextGetProperty, 208
  - optixDeviceContextSetCacheDatabaseSizes, 208
  - optixDeviceContextSetCacheEnabled, 208
  - optixDeviceContextSetCacheLocation, 208
  - optixDeviceContextSetLogCallback, 208
  - optixGetErrorName, 208
  - optixGetErrorString, 208
  - optixLaunch, 208
  - optixModuleCreateFromPTX, 209
  - optixModuleCreateFromPTXWithTasks, 209
  - optixModuleDestroy, 209
  - optixModuleGetCompilationState, 209
  - optixPipelineCreate, 209
  - optixPipelineDestroy, 209
  - optixPipelineSetStackSize, 209
  - optixProgramGroupCreate, 210
  - optixProgramGroupDestroy, 210
  - optixProgramGroupGetStackSize, 210
  - optixSbtRecordPackHeader, 210
  - optixTaskExecute, 210
  - reserved1, 210
  - reserved2, 210
- OptixGeometryFlags
  - Types, 149, 164
- optixGetAttribute\_0
  - Device API, 24
  - optix\_7\_device\_impl.h, 272
- optixGetAttribute\_1
  - Device API, 24
  - optix\_7\_device\_impl.h, 272
- optixGetAttribute\_2
  - Device API, 24
  - optix\_7\_device\_impl.h, 272
- optixGetAttribute\_3
  - Device API, 24
  - optix\_7\_device\_impl.h, 272
- optixGetAttribute\_4
  - Device API, 24
  - optix\_7\_device\_impl.h, 272
- optixGetAttribute\_5
  - Device API, 24
  - optix\_7\_device\_impl.h, 272
- optixGetAttribute\_6
  - Device API, 24

- [optix\\_7\\_device\\_impl.h, 272](#)
- [optixGetAttribute\\_7](#)
  - [Device API, 24](#)
  - [optix\\_7\\_device\\_impl.h, 272](#)
- [optixGetCatmullRomVertexData](#)
  - [Device API, 24](#)
  - [optix\\_7\\_device\\_impl.h, 272](#)
- [optixGetCubicBSplineVertexData](#)
  - [Device API, 25](#)
  - [optix\\_7\\_device\\_impl.h, 272](#)
- [optixGetCurveParameter](#)
  - [Device API, 25](#)
  - [optix\\_7\\_device\\_impl.h, 273](#)
- [optixGetErrorName](#)
  - [Error handling, 107](#)
  - [OptixFunctionTable, 208](#)
- [optixGetErrorString](#)
  - [Error handling, 107](#)
  - [OptixFunctionTable, 208](#)
- [optixGetExceptionCode](#)
  - [Device API, 25](#)
  - [optix\\_7\\_device\\_impl.h, 273](#)
- [optixGetExceptionDetail\\_0](#)
  - [Device API, 25](#)
  - [optix\\_7\\_device\\_impl.h, 273](#)
- [optixGetExceptionDetail\\_1](#)
  - [Device API, 25](#)
  - [optix\\_7\\_device\\_impl.h, 273](#)
- [optixGetExceptionDetail\\_2](#)
  - [Device API, 26](#)
  - [optix\\_7\\_device\\_impl.h, 273](#)
- [optixGetExceptionDetail\\_3](#)
  - [Device API, 26](#)
  - [optix\\_7\\_device\\_impl.h, 273](#)
- [optixGetExceptionDetail\\_4](#)
  - [Device API, 26](#)
  - [optix\\_7\\_device\\_impl.h, 273](#)
- [optixGetExceptionDetail\\_5](#)
  - [Device API, 26](#)
  - [optix\\_7\\_device\\_impl.h, 273](#)
- [optixGetExceptionDetail\\_6](#)
  - [Device API, 26](#)
  - [optix\\_7\\_device\\_impl.h, 273](#)
- [optixGetExceptionDetail\\_7](#)
  - [Device API, 26](#)
  - [optix\\_7\\_device\\_impl.h, 273](#)
- [optixGetExceptionInvalidRay](#)
  - [Device API, 27](#)
  - [optix\\_7\\_device\\_impl.h, 273](#)
- [optixGetExceptionInvalidSbtOffset](#)
  - [Device API, 27](#)
  - [optix\\_7\\_device\\_impl.h, 273](#)
- [optixGetExceptionInvalidTraversable](#)
  - [Device API, 27](#)
  - [optix\\_7\\_device\\_impl.h, 273](#)
- [optixGetExceptionLineInfo](#)
  - [Device API, 27](#)
  - [optix\\_7\\_device\\_impl.h, 273](#)
- [optixGetExceptionParameterMismatch](#)
  - [Device API, 27](#)
  - [optix\\_7\\_device\\_impl.h, 273](#)
- [optixGetGASMotionStepCount](#)
  - [Device API, 28](#)
  - [optix\\_7\\_device\\_impl.h, 273](#)
- [optixGetGASMotionTimeBegin](#)
  - [Device API, 28](#)
  - [optix\\_7\\_device\\_impl.h, 273](#)
- [optixGetGASMotionTimeEnd](#)
  - [Device API, 28](#)
  - [optix\\_7\\_device\\_impl.h, 274](#)
- [optixGetGASTraversableHandle](#)
  - [Device API, 28](#)
  - [optix\\_7\\_device\\_impl.h, 274](#)
- [optixGetHitKind](#)
  - [Device API, 28](#)
  - [optix\\_7\\_device\\_impl.h, 274](#)
- [optixGetInstanceChildFromHandle](#)
  - [Device API, 28](#)
  - [optix\\_7\\_device\\_impl.h, 274](#)
- [optixGetInstanceId](#)
  - [Device API, 29](#)
  - [optix\\_7\\_device\\_impl.h, 274](#)
- [optixGetInstanceIdFromHandle](#)
  - [Device API, 29](#)
  - [optix\\_7\\_device\\_impl.h, 274](#)
- [optixGetInstanceIndex](#)
  - [Device API, 29](#)
  - [optix\\_7\\_device\\_impl.h, 274](#)
- [optixGetInstanceInverseTransformFromHandle](#)
  - [Device API, 29](#)
  - [optix\\_7\\_device\\_impl.h, 274](#)
- [optixGetInstanceTransformFromHandle](#)
  - [Device API, 29](#)
  - [optix\\_7\\_device\\_impl.h, 274](#)
- [optixGetInstanceTraversableFromIAS](#)
  - [Device API, 29](#)
  - [optix\\_7\\_device\\_impl.h, 274](#)
- [optixGetInterpolatedTransformation](#)

- optix\_impl, 184
- optixGetInterpolatedTransformationFromHandle
  - optix\_impl, 184
- optixGetLaunchDimensions
  - Device API, 30
  - optix\_7\_device\_impl.h, 274
- optixGetLaunchIndex
  - Device API, 30
  - optix\_7\_device\_impl.h, 274
- optixGetLinearCurveVertexData
  - Device API, 30
  - optix\_7\_device\_impl.h, 274
- optixGetMatrixFromSrt
  - optix\_impl, 184
- optixGetMatrixMotionTransformFromHandle
  - Device API, 30
  - optix\_7\_device\_impl.h, 274
- optixGetObjectRayDirection
  - Device API, 30
  - optix\_7\_device\_impl.h, 275
- optixGetObjectRayOrigin
  - Device API, 30
  - optix\_7\_device\_impl.h, 275
- optixGetObjectToWorldTransformMatrix
  - Device API, 30
  - optix\_7\_device\_impl.h, 275
  - optix\_impl, 184
- optixGetPayload\_0
  - Device API, 31
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_1
  - Device API, 31
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_10
  - Device API, 31
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_11
  - Device API, 31
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_12
  - Device API, 31
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_13
  - Device API, 31
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_14
  - Device API, 31
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_15
  - Device API, 31
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_16
  - Device API, 31
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_17
  - Device API, 31
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_18
  - Device API, 31
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_19
  - Device API, 31
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_2
  - Device API, 31
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_20
  - Device API, 32
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_21
  - Device API, 32
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_22
  - Device API, 32
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_23
  - Device API, 32
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_24
  - Device API, 32
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_25
  - Device API, 32
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_26
  - Device API, 32
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_27
  - Device API, 32
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_28
  - Device API, 32
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_29
  - Device API, 32
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_3
  - Device API, 32

- optix\_7\_device\_impl.h, 276
- optixGetPayload\_30
  - Device API, 32
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_31
  - Device API, 32
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_4
  - Device API, 33
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_5
  - Device API, 33
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_6
  - Device API, 33
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_7
  - Device API, 33
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_8
  - Device API, 33
  - optix\_7\_device\_impl.h, 276
- optixGetPayload\_9
  - Device API, 33
  - optix\_7\_device\_impl.h, 276
- optixGetPrimitiveIndex
  - Device API, 33
  - optix\_7\_device\_impl.h, 276
- optixGetPrimitiveType
  - Device API, 33
  - optix\_7\_device\_impl.h, 276, 277
- optixGetQuadraticBSplineVertexData
  - Device API, 33
  - optix\_7\_device\_impl.h, 277
- optixGetRayFlags
  - Device API, 34
  - optix\_7\_device\_impl.h, 277
- optixGetRayTime
  - Device API, 34
  - optix\_7\_device\_impl.h, 277
- optixGetRayTmax
  - Device API, 34
  - optix\_7\_device\_impl.h, 277
- optixGetRayTmin
  - Device API, 34
  - optix\_7\_device\_impl.h, 277
- optixGetRayVisibilityMask
  - Device API, 34
  - optix\_7\_device\_impl.h, 277
- optixGetSbtDataPointer
  - Device API, 34
  - optix\_7\_device\_impl.h, 277
- optixGetSbtGASIndex
  - Device API, 34
  - optix\_7\_device\_impl.h, 277
- optixGetSRTMotionTransformFromHandle
  - Device API, 35
  - optix\_7\_device\_impl.h, 277
- optixGetStaticTransformFromHandle
  - Device API, 35
  - optix\_7\_device\_impl.h, 277
- optixGetTransformListHandle
  - Device API, 35
  - optix\_7\_device\_impl.h, 277
- optixGetTransformListSize
  - Device API, 35
  - optix\_7\_device\_impl.h, 277
- optixGetTransformTypeFromHandle
  - Device API, 35
  - optix\_7\_device\_impl.h, 277
- optixGetTriangleBarycentrics
  - Device API, 35
  - optix\_7\_device\_impl.h, 278
- optixGetTriangleVertexData
  - Device API, 35
  - optix\_7\_device\_impl.h, 278
- optixGetWorldRayDirection
  - Device API, 36
  - optix\_7\_device\_impl.h, 278
- optixGetWorldRayOrigin
  - Device API, 36
  - optix\_7\_device\_impl.h, 278
- optixGetWorldToObjectTransformMatrix
  - Device API, 36
  - optix\_7\_device\_impl.h, 278
  - optix\_impl, 185
- OptixHitKind
  - Types, 149, 164
- optixIgnoreIntersection
  - Device API, 36
  - optix\_7\_device\_impl.h, 278
- OptixImage2D, 211
  - data, 211
  - format, 211
  - height, 211
  - pixelStrideInBytes, 211
  - rowStrideInBytes, 211
  - Types, 149

- width, 211
- OptixIndicesFormat
  - Types, 149, 164
- optixInit
  - Utilities, 176
- optixInitWithHandle
  - Utilities, 176
- OptixInstance, 212
  - flags, 212
  - instanceId, 212
  - pad, 212
  - sbtOffset, 212
  - transform, 212
  - traversableHandle, 212
  - Types, 150
  - visibilityMask, 212
- OptixInstanceFlags
  - Types, 150, 164
- optixInvertMatrix
  - optix\_impl, 185
- optixIsBackFaceHit
  - Device API, 36
  - optix\_7\_device\_impl.h, 278
- optixIsFrontFaceHit
  - Device API, 36
  - optix\_7\_device\_impl.h, 278
- optixIsTriangleBackFaceHit
  - Device API, 36
  - optix\_7\_device\_impl.h, 278
- optixIsTriangleFrontFaceHit
  - Device API, 37
  - optix\_7\_device\_impl.h, 278
- optixIsTriangleHit
  - Device API, 37
  - optix\_7\_device\_impl.h, 278
- optixLaunch
  - Launches, 121
  - OptixFunctionTable, 208
- optixLdg
  - optix\_impl, 185
- optixLoadInterpolatedMatrixKey
  - optix\_impl, 185
- optixLoadInterpolatedSrtKey
  - optix\_impl, 185
- optixLoadReadOnlyAlign16
  - optix\_impl, 185
- optixLoadWindowsDll
  - optix\_stubs.h, 350
- optixLoadWindowsDllFromName
  - optix\_stubs.h, 350
- OptixLogCallback
  - Types, 150
- OptixMatrixMotionTransform, 213
  - child, 214
  - motionOptions, 214
  - pad, 214
  - transform, 214
  - Types, 150
- OptixModule
  - Types, 151
- OptixModuleCompileBoundValueEntry, 214
  - annotation, 215
  - boundValuePtr, 215
  - pipelineParamOffsetInBytes, 215
  - sizeInBytes, 215
  - Types, 151
- OptixModuleCompileOptions, 215
  - boundValues, 215
  - debugLevel, 215
  - maxRegisterCount, 216
  - numBoundValues, 216
  - numPayloadTypes, 216
  - optLevel, 216
  - payloadTypes, 216
  - Types, 152
- OptixModuleCompileState
  - Types, 152, 165
- optixModuleCreateFromPTX
  - Modules, 115
  - OptixFunctionTable, 209
- optixModuleCreateFromPTXWithTasks
  - Modules, 116
  - OptixFunctionTable, 209
- optixModuleDestroy
  - Modules, 117
  - OptixFunctionTable, 209
- optixModuleGetCompilationState
  - Modules, 117
  - OptixFunctionTable, 209
- OptixMotionFlags
  - Types, 152, 165
- OptixMotionOptions, 216
  - flags, 216
  - numKeys, 216
  - timeBegin, 217
  - timeEnd, 217
  - Types, 152
- optixMulFloat4

- optix\_impl, 185
- optixMultiplyRowMatrix
  - optix\_impl, 185
- OptixPayloadSemantics
  - Types, 152, 165
- OptixPayloadType, 217
  - numPayloadValues, 217
  - payloadSemantics, 217
  - Types, 153
- OptixPayloadTypeID
  - Types, 153, 166
- OptixPipeline
  - Types, 153
- OptixPipelineCompileOptions, 217
  - exceptionFlags, 218
  - numAttributeValues, 218
  - numPayloadValues, 218
  - pipelineLaunchParamsVariableName, 218
  - traversableGraphFlags, 218
  - Types, 153
  - usesMotionBlur, 218
  - usesPrimitiveTypeFlags, 218
- optixPipelineCreate
  - OptixFunctionTable, 209
  - Pipelines, 113
- optixPipelineDestroy
  - OptixFunctionTable, 209
  - Pipelines, 114
- OptixPipelineLinkOptions, 218
  - debugLevel, 219
  - maxTraceDepth, 219
  - Types, 153
- optixPipelineSetStackSize
  - OptixFunctionTable, 209
  - Pipelines, 114
- OptixPixelFormat
  - Types, 153, 167
- OptixPrimitiveType
  - Types, 153, 167
- OptixPrimitiveTypeFlags
  - Types, 153, 167
- OptixProgramGroup
  - Types, 154
- OptixProgramGroupCallables, 219
  - entryFunctionNameCC, 219
  - entryFunctionNameDC, 219
  - moduleCC, 219
  - moduleDC, 220
  - Types, 154
- optixProgramGroupCreate
  - OptixFunctionTable, 210
  - Program groups, 119
- OptixProgramGroupDesc, 220
  - callables, 220
  - exception, 220
  - flags, 220
  - hitgroup, 220
  - kind, 221
  - miss, 221
  - raygen, 221
  - Types, 154
- optixProgramGroupDestroy
  - OptixFunctionTable, 210
  - Program groups, 120
- OptixProgramGroupFlags
  - Types, 154, 168
- optixProgramGroupGetStackSize
  - OptixFunctionTable, 210
  - Program groups, 120
- OptixProgramGroupHitgroup, 221
  - entryFunctionNameAH, 221
  - entryFunctionNameCH, 221
  - entryFunctionNameIS, 222
  - moduleAH, 222
  - moduleCH, 222
  - moduleIS, 222
  - Types, 154
- OptixProgramGroupKind
  - Types, 154, 168
- OptixProgramGroupOptions, 222
  - payloadType, 222
  - Types, 154
- OptixProgramGroupSingleModule, 223
  - entryFunctionName, 223
  - module, 223
  - Types, 154
- OptixQueryFunctionTable\_t
  - Types, 155
- OptixQueryFunctionTableOptions
  - Types, 155, 169
- OptixRayFlags
  - Types, 155, 169
- optixReportIntersection
  - Device API, 37–39
  - optix\_7\_device\_impl.h, 278–280
- optixResolveMotionKey
  - optix\_impl, 186
- OptixResult



- Types, 155, 170
- optixSbtRecordPackHeader
  - Launches, 122
  - OptixFunctionTable, 210
- optixSetPayload\_0
  - Device API, 40
  - optix\_7\_device\_impl.h, 280
- optixSetPayload\_1
  - Device API, 40
  - optix\_7\_device\_impl.h, 280
- optixSetPayload\_10
  - Device API, 40
  - optix\_7\_device\_impl.h, 280
- optixSetPayload\_11
  - Device API, 40
  - optix\_7\_device\_impl.h, 280
- optixSetPayload\_12
  - Device API, 40
  - optix\_7\_device\_impl.h, 280
- optixSetPayload\_13
  - Device API, 40
  - optix\_7\_device\_impl.h, 280
- optixSetPayload\_14
  - Device API, 40
  - optix\_7\_device\_impl.h, 280
- optixSetPayload\_15
  - Device API, 40
  - optix\_7\_device\_impl.h, 280
- optixSetPayload\_16
  - Device API, 41
  - optix\_7\_device\_impl.h, 280
- optixSetPayload\_17
  - Device API, 41
  - optix\_7\_device\_impl.h, 280
- optixSetPayload\_18
  - Device API, 41
  - optix\_7\_device\_impl.h, 281
- optixSetPayload\_19
  - Device API, 41
  - optix\_7\_device\_impl.h, 281
- optixSetPayload\_2
  - Device API, 41
  - optix\_7\_device\_impl.h, 281
- optixSetPayload\_20
  - Device API, 41
  - optix\_7\_device\_impl.h, 281
- optixSetPayload\_21
  - Device API, 41
  - optix\_7\_device\_impl.h, 281
- optixSetPayload\_22
  - Device API, 41
  - optix\_7\_device\_impl.h, 281
- optixSetPayload\_23
  - Device API, 41
  - optix\_7\_device\_impl.h, 281
- optixSetPayload\_24
  - Device API, 41
  - optix\_7\_device\_impl.h, 281
- optixSetPayload\_25
  - Device API, 42
  - optix\_7\_device\_impl.h, 281
- optixSetPayload\_26
  - Device API, 42
  - optix\_7\_device\_impl.h, 281
- optixSetPayload\_27
  - Device API, 42
  - optix\_7\_device\_impl.h, 281
- optixSetPayload\_28
  - Device API, 42
  - optix\_7\_device\_impl.h, 281
- optixSetPayload\_29
  - Device API, 42
  - optix\_7\_device\_impl.h, 281
- optixSetPayload\_3
  - Device API, 42
  - optix\_7\_device\_impl.h, 281
- optixSetPayload\_30
  - Device API, 42
  - optix\_7\_device\_impl.h, 281
- optixSetPayload\_31
  - Device API, 42
  - optix\_7\_device\_impl.h, 282
- optixSetPayload\_4
  - Device API, 42
  - optix\_7\_device\_impl.h, 282
- optixSetPayload\_5
  - Device API, 42
  - optix\_7\_device\_impl.h, 282
- optixSetPayload\_6
  - Device API, 43
  - optix\_7\_device\_impl.h, 282
- optixSetPayload\_7
  - Device API, 43
  - optix\_7\_device\_impl.h, 282
- optixSetPayload\_8
  - Device API, 43
  - optix\_7\_device\_impl.h, 282
- optixSetPayload\_9



- Device API, 43
- optix\_7\_device\_impl.h, 282
- optixSetPayloadTypes
  - Device API, 43
  - optix\_7\_device\_impl.h, 282
- OptixShaderBindingTable, 223
  - callablesRecordBase, 224
  - callablesRecordCount, 224
  - callablesRecordStrideInBytes, 224
  - exceptionRecord, 224
  - hitgroupRecordBase, 224
  - hitgroupRecordCount, 224
  - hitgroupRecordStrideInBytes, 224
  - missRecordBase, 224
  - missRecordCount, 225
  - missRecordStrideInBytes, 225
  - raygenRecord, 225
  - Types, 155
- OptixSRTData, 225
  - a, 227
  - b, 227
  - c, 227
  - pvx, 227
  - pvy, 227
  - pvz, 227
  - qw, 227
  - qx, 227
  - qy, 227
  - qz, 227
  - sx, 227
  - sy, 227
  - sz, 227
  - tx, 227
  - ty, 227
  - Types, 155
  - tz, 227
- OptixSRTMotionTransform, 227
  - child, 228
  - motionOptions, 228
  - pad, 228
  - srtData, 228
  - Types, 156
- OptixStackSizes, 228
  - cssAH, 229
  - cssCC, 229
  - cssCH, 229
  - cssIS, 229
  - cssMS, 229
  - cssRG, 229
  - dssDC, 229
  - Types, 157
- OptixStaticTransform, 230
  - child, 230
  - invTransform, 230
  - pad, 230
  - transform, 230
  - Types, 157
- OptixTask
  - Types, 157
- optixTaskExecute
  - OptixFunctionTable, 210
  - Tasks, 118
- optixTerminateRay
  - Device API, 43
  - optix\_7\_device\_impl.h, 282
- optixTexFootprint2D
  - Device API, 43
  - optix\_7\_device\_impl.h, 282
- optixTexFootprint2DGrad
  - Device API, 44
  - optix\_7\_device\_impl.h, 282
- optixTexFootprint2DLod
  - Device API, 45
  - optix\_7\_device\_impl.h, 283
- optixThrowException
  - Device API, 46–48
  - optix\_7\_device\_impl.h, 283, 284
- optixTrace
  - Device API, 48–100, 102, 103
  - optix\_7\_device\_impl.h, 284–329
- OptixTransformFormat
  - Types, 157, 171
- optixTransformNormal
  - optix\_impl, 186
- optixTransformNormalFromObjectToWorldSpace
  - Device API, 104
  - optix\_7\_device\_impl.h, 331
- optixTransformNormalFromWorldToObjectSpace
  - Device API, 104
  - optix\_7\_device\_impl.h, 331
- optixTransformPoint
  - optix\_impl, 186
- optixTransformPointFromObjectToWorldSpace
  - Device API, 104
  - optix\_7\_device\_impl.h, 331
- optixTransformPointFromWorldToObjectSpace
  - Device API, 105
  - optix\_7\_device\_impl.h, 331

- OptixTransformType
  - Types, 157, 171
- optixTransformVector
  - optix\_impl, 186
- optixTransformVectorFromObjectToWorldSpace
  - Device API, 105
  - optix\_7\_device\_impl.h, 331
- optixTransformVectorFromWorldToObjectSpace
  - Device API, 105
  - optix\_7\_device\_impl.h, 331
- OptixTraversableGraphFlags
  - Types, 157, 172
- OptixTraversableHandle
  - Types, 157
- OptixTraversableType
  - Types, 157, 172
- optixUndefinedValue
  - Device API, 105
  - optix\_7\_device\_impl.h, 331
- optixUninitWithHandle
  - Utilities, 176
- optixUtilAccumulateStackSizes
  - Utilities, 176
- optixUtilComputeStackSizes
  - Utilities, 176
- optixUtilComputeStackSizesCssCCTree
  - Utilities, 177
- optixUtilComputeStackSizesDCSplit
  - Utilities, 177
- optixUtilComputeStackSizesSimplePathTracer
  - Utilities, 178
- OptixUtilDenoiserImageTile, 230
  - input, 231
  - inputOffsetX, 231
  - inputOffsetY, 231
  - output, 231
- optixUtilDenoiserInvokeTiled
  - Utilities, 179
- optixUtilDenoiserSplitImage
  - Utilities, 180
- optixUtilGetPixelStride
  - Utilities, 180
- OptixVertexFormat
  - Types, 157, 172
- OptixVisibilityMask
  - Types, 158
- optLevel
  - OptixModuleCompileOptions, 216
- output
  - OptixDenoiserLayer, 199
  - OptixUtilDenoiserImageTile, 231
- outputSizeInBytes
  - OptixAccelBufferSizes, 187
- overlapWindowSizeInPixels
  - OptixDenoiserSizes, 201
- pad
  - OptixBuildInput, 190
  - OptixInstance, 212
  - OptixMatrixMotionTransform, 214
  - OptixSRTMotionTransform, 228
  - OptixStaticTransform, 230
- payloadSemantics
  - OptixPayloadType, 217
- payloadType
  - OptixProgramGroupOptions, 222
- payloadTypes
  - OptixModuleCompileOptions, 216
- pipelineLaunchParamsVariableName
  - OptixPipelineCompileOptions, 218
- pipelineParamOffsetInBytes
  - OptixModuleCompileBoundValueEntry, 215
- Pipelines, 113
  - optixPipelineCreate, 113
  - optixPipelineDestroy, 114
  - optixPipelineSetStackSize, 114
- pixelStrideInBytes
  - OptixImage2D, 211
- preTransform
  - OptixBuildInputTriangleArray, 196
- previousOutput
  - OptixDenoiserLayer, 199
- primitiveIndexOffset
  - OptixBuildInputCurveArray, 192
  - OptixBuildInputCustomPrimitiveArray, 194
  - OptixBuildInputTriangleArray, 196
- Program groups, 119
  - optixProgramGroupCreate, 119
  - optixProgramGroupDestroy, 120
  - optixProgramGroupGetStackSize, 120
- pvx
  - OptixSRTData, 227
- pvy
  - OptixSRTData, 227
- pvz
  - OptixSRTData, 227
- qw

- OptixSRTData, [227](#)
- qx
  - OptixSRTData, [227](#)
- qy
  - OptixSRTData, [227](#)
- qz
  - OptixSRTData, [227](#)
- raygen
  - OptixProgramGroupDesc, [221](#)
- raygenRecord
  - OptixShaderBindingTable, [225](#)
- reserved1
  - OptixFunctionTable, [210](#)
- reserved2
  - OptixFunctionTable, [210](#)
- result
  - OptixAccelEmitDesc, [189](#)
- rowStrideInBytes
  - OptixImage2D, [211](#)
- sbtIndexOffsetBuffer
  - OptixBuildInputCustomPrimitiveArray, [194](#)
  - OptixBuildInputTriangleArray, [196](#)
- sbtIndexOffsetSizeInBytes
  - OptixBuildInputCustomPrimitiveArray, [194](#)
  - OptixBuildInputTriangleArray, [197](#)
- sbtIndexOffsetStrideInBytes
  - OptixBuildInputCustomPrimitiveArray, [194](#)
  - OptixBuildInputTriangleArray, [197](#)
- sbtOffset
  - OptixInstance, [212](#)
- sizeInBytes
  - OptixModuleCompileBoundValueEntry, [215](#)
- srtData
  - OptixSRTMotionTransform, [228](#)
- stateSizeInBytes
  - OptixDenoiserSizes, [201](#)
- strideInBytes
  - OptixBuildInputCustomPrimitiveArray, [194](#)
- sx
  - OptixSRTData, [227](#)
- sy
  - OptixSRTData, [227](#)
- sz
  - OptixSRTData, [227](#)
- Tasks, [118](#)
  - optixTaskExecute, [118](#)
- tempSizeInBytes

- OptixAccelBufferSizes, [187](#)
- tempUpdateSizeInBytes
  - OptixAccelBufferSizes, [188](#)
- timeBegin
  - OptixMotionOptions, [217](#)
- timeEnd
  - OptixMotionOptions, [217](#)
- transform
  - OptixInstance, [212](#)
  - OptixMatrixMotionTransform, [214](#)
  - OptixStaticTransform, [230](#)
- transformFormat
  - OptixBuildInputTriangleArray, [197](#)
- traversableGraphFlags
  - OptixPipelineCompileOptions, [218](#)
- traversableHandle
  - OptixInstance, [212](#)
- triangleArray
  - OptixBuildInput, [190](#)
- tx
  - OptixSRTData, [227](#)
- ty
  - OptixSRTData, [227](#)
- type
  - OptixAccelEmitDesc, [189](#)
  - OptixBuildInput, [190](#)
- Types, [134](#)
  - CUdeviceptr, [144](#)
  - OP-
    - TIX\_AABB\_BUFFER\_BYTE\_ALIGNMENT, [143](#)
  - OP-
    - TIX\_ACCEL\_BUFFER\_BYTE\_ALIGNMENT, [143](#)
  - OP-
    - TIX\_BUILD\_FLAG\_ALLOW\_COMPACTION, [158](#)
  - OP-
    - TIX\_BUILD\_FLAG\_ALLOW\_RANDOM\_INSTANCE\_ACC, [158](#)
  - OP-
    - TIX\_BUILD\_FLAG\_ALLOW\_RANDOM\_VERTEX\_ACCES, [158](#)
  - OPTIX\_BUILD\_FLAG\_ALLOW\_UPDATE, [158](#)
  - OPTIX\_BUILD\_FLAG\_NONE, [158](#)
  - OP-
    - TIX\_BUILD\_FLAG\_PREFER\_FAST\_BUILD, [158](#)

- OP- 160
- TIX\_BUILD\_FLAG\_PREFER\_FAST\_TRACE, OP- 158
- OPTIX\_BUILD\_INPUT\_TYPE\_CURVES, 159
- OP- OPTIX\_CURVE\_ENDCAP\_DEFAULT, 160
- TIX\_BUILD\_INPUT\_TYPE\_CUSTOM\_PRIMITIVE, OPTIX\_CURVE\_ENDCAP\_ON, 160
- 159 OPTIX\_DENOISER\_MODEL\_KIND\_AOV, 161
- OP- TIX\_BUILD\_INPUT\_TYPE\_INSTANCE\_POINTCLOUD, OPTIX\_DENOISER\_MODEL\_KIND\_HDR, 161
- 159 OPTIX\_DENOISER\_MODEL\_KIND\_LDR, 161
- OPTIX\_BUILD\_INPUT\_TYPE\_INSTANCES, OP- 159
- OPTIX\_BUILD\_INPUT\_TYPE\_TRIANGLES, TIX\_DENOISER\_MODEL\_KIND\_TEMPORAL, 161
- 159
- OPTIX\_BUILD\_OPERATION\_BUILD, 159
- OPTIX\_BUILD\_OPERATION\_UPDATE, 159
- OP- TIX\_DENOISER\_MODEL\_KIND\_TEMPORAL\_AOV, 161
- TIX\_COMPILE\_DEBUG\_LEVEL\_DEFAULT, OP- 160
- OPTIX\_COMPILE\_DEBUG\_LEVEL\_FULL, TIX\_DEVICE\_CONTEXT\_VALIDATION\_MODE\_ALL, 161
- 160
- OP- OP- 161
- TIX\_COMPILE\_DEBUG\_LEVEL\_MINIMAL, TIX\_DEVICE\_CONTEXT\_VALIDATION\_MODE\_OFF, 161
- 160
- OP- OP- 162
- TIX\_COMPILE\_DEBUG\_LEVEL\_MODERATE, TIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_INSTANCE\_ID, 162
- 160
- OPTIX\_COMPILE\_DEBUG\_LEVEL\_NONE, OP- 160
- 160
- OP- TIX\_COMPILE\_DEFAULT\_MAX\_PAYLOAD\_TYPE\_COUNT, 161
- 143 TIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_PRIMITIVES\_PER\_INSTANCE, 161
- OP- 161
- TIX\_COMPILE\_DEFAULT\_MAX\_PAYLOAD\_VALIDATION\_COUNT, 162
- 143
- OP- TIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_SBT\_OFFSET, 162
- 144
- OP- TIX\_COMPILE\_DEFAULT\_MAX\_REGISTER\_COUNT, 162
- 144
- OP- TIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_SBT\_RECORDS, 162
- 160
- OP- TIX\_COMPILE\_OPTIMIZATION\_DEFAULT, OP- 161
- 160
- OP- TIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_TRACE\_DEPTH, 161
- 160
- OP- TIX\_DEVICE\_PROPERTY\_LIMIT\_MAX\_TRAVERSABLE\_NODES, 161
- 160
- OP- TIX\_COMPILE\_OPTIMIZATION\_LEVEL\_1, OP- 162
- 160
- OP- TIX\_DEVICE\_PROPERTY\_LIMIT\_NUM\_BITS\_INSTANCE\_ID, 162
- 160
- OP- TIX\_COMPILE\_OPTIMIZATION\_LEVEL\_2, OP- 162

TIX_DEVICE_PROPERTY_RTCORE_VERSION, 162	OP- TIX_ERROR_INVALID_ENTRY_FUNCTION_OPTIONS, 171
OP- TIX_ERROR_ACCEL_NOT_COMPATIBLE, 171	OPTIX_ERROR_INVALID_FILE_FORMAT, 170
OPTIX_ERROR_CUDA_ERROR, 171	OP- TIX_ERROR_INVALID_FUNCTION_ARGUMENTS, 170
OPTIX_ERROR_CUDA_NOT_INITIALIZED, 170	OPTIX_ERROR_INVALID_FUNCTION_USE, 170
OP- TIX_ERROR_DENOISER_MODEL_NOT_SET, 171	OP- TIX_ERROR_INVALID_LAUNCH_PARAMETER, 170
OP- TIX_ERROR_DENOISER_NOT_INITIALIZED, 171	OPTIX_ERROR_INVALID_OPERATION, 170
OP- TIX_ERROR_DISK_CACHE_DATABASE_ERROR, 170	OP- TIX_ERROR_INVALID_PAYLOAD_ACCESS, 170
OP- TIX_ERROR_DISK_CACHE_INVALID_DATA, 170	OPTIX_ERROR_INVALID_PTX, 170
OP- TIX_ERROR_DISK_CACHE_INVALID_PATH, 170	OPTIX_ERROR_INVALID_VALUE, 170
OP- TIX_ERROR_DISK_CACHE_PERMISSION_ERROR, 170	OPTIX_ERROR_LAUNCH_FAILURE, 170
OP- TIX_ERROR_ENTRY_SYMBOL_NOT_FOUND, 171	OPTIX_ERROR_LIBRARY_NOT_FOUND, 171
OPTIX_ERROR_FILE_IO_ERROR, 170	OP- TIX_ERROR_LIBRARY_UNLOAD_FAILURE, 171
OP- TIX_ERROR_FUNCTION_TABLE_SIZE_MISMATCH, 171	OPTIX_ERROR_NOT_SUPPORTED, 171
OP- TIX_ERROR_HOST_OUT_OF_MEMORY, 170	OP- TIX_ERROR_PAYLOAD_TYPE_ID_INVALID, 171
OP- TIX_ERROR_ILLEGAL_DURING_TASK_EXECUTION, 171	OP- TIX_ERROR_PAYLOAD_TYPE_MISMATCH, 171
OP- TIX_ERROR_INTERNAL_COMPILER_ERROR, 171	OP- TIX_ERROR_PAYLOAD_TYPE_RESOLUTION_FAILED, 171
OPTIX_ERROR_INTERNAL_ERROR, 171	OPTIX_ERROR_PIPELINE_LINK_ERROR, 171
OP- TIX_ERROR_INVALID_ATTRIBUTE_ACCESS, 170	OP- TIX_ERROR_PIPELINE_OUT_OF_CONSTANT_MEMORY, 171
OP- TIX_ERROR_INVALID_DEVICE_CONTEXT, 170	OPTIX_ERROR_UNKNOWN, 171
	OP- TIX_ERROR_UNSUPPORTED_ABI_VERSION, 171
	OPTIX_ERROR_VALIDATION_FAILURE, 170
	OP- TIX_EXCEPTION_CODE_BUILTIN_IS_MISMATCH, 163



- OPTIX\_INSTANCE\_FLAG\_NONE, 165
- OP-  
TIX\_MODULE\_COMPILE\_STATE\_COMPLETED, 165
- OP-  
TIX\_MODULE\_COMPILE\_STATE\_FAILED, 165
- OP-  
TIX\_MODULE\_COMPILE\_STATE\_IMPENDING\_FAILURE, 165
- OP-  
TIX\_MODULE\_COMPILE\_STATE\_NOT\_STARTED, 165
- OP-  
TIX\_MODULE\_COMPILE\_STATE\_STARTED, 165
- OPTIX\_MOTION\_FLAG\_END\_VANISH, 165
- OPTIX\_MOTION\_FLAG\_NONE, 165
- OPTIX\_MOTION\_FLAG\_START\_VANISH, 165
- OPTIX\_PAYLOAD\_SEMANTICS\_AH\_NONE, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_AH\_READ, 166
- OP-  
TIX\_PAYLOAD\_SEMANTICS\_AH\_READ\_WRITE, 166
- OP-  
TIX\_PAYLOAD\_SEMANTICS\_AH\_WRITE, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_CH\_NONE, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_CH\_READ, 166
- OP-  
TIX\_PAYLOAD\_SEMANTICS\_CH\_READ\_WRITE, 166
- OP-  
TIX\_PAYLOAD\_SEMANTICS\_CH\_WRITE, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_IS\_NONE, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_IS\_READ, 166
- OP-  
TIX\_PAYLOAD\_SEMANTICS\_IS\_READ\_WRITE, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_IS\_WRITE, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_MS\_NONE, 166
- OPTIX\_PAYLOAD\_SEMANTICS\_MS\_READ, 166
- OP-  
TIX\_PAYLOAD\_SEMANTICS\_MS\_READ\_WRITE, 166
- OP-  
TIX\_PAYLOAD\_SEMANTICS\_MS\_WRITE, 166
- OP-  
TIX\_PAYLOAD\_SEMANTICS\_TRACE\_CALLER\_NONE, 166
- OP-  
TIX\_PAYLOAD\_SEMANTICS\_TRACE\_CALLER\_READ, 166
- OP-  
TIX\_PAYLOAD\_SEMANTICS\_TRACE\_CALLER\_READ\_WRITE, 166
- OP-  
TIX\_PAYLOAD\_SEMANTICS\_TRACE\_CALLER\_WRITE, 166
- OPTIX\_PAYLOAD\_TYPE\_DEFAULT, 166
- OPTIX\_PAYLOAD\_TYPE\_ID\_0, 166
- OPTIX\_PAYLOAD\_TYPE\_ID\_1, 166
- OPTIX\_PAYLOAD\_TYPE\_ID\_2, 166
- OPTIX\_PAYLOAD\_TYPE\_ID\_3, 167
- OPTIX\_PAYLOAD\_TYPE\_ID\_4, 167
- OPTIX\_PAYLOAD\_TYPE\_ID\_5, 167
- OPTIX\_PAYLOAD\_TYPE\_ID\_6, 167
- OPTIX\_PAYLOAD\_TYPE\_ID\_7, 167
- OPTIX\_PIXEL\_FORMAT\_FLOAT2, 167
- OPTIX\_PIXEL\_FORMAT\_FLOAT3, 167
- OPTIX\_PIXEL\_FORMAT\_FLOAT4, 167
- OPTIX\_PIXEL\_FORMAT\_HALF2, 167
- OPTIX\_PIXEL\_FORMAT\_HALF3, 167
- OPTIX\_PIXEL\_FORMAT\_HALF4, 167
- OPTIX\_PIXEL\_FORMAT\_UCHAR3, 167
- OPTIX\_PIXEL\_FORMAT\_UCHAR4, 167
- OPTIX\_PRIMITIVE\_TYPE\_CUSTOM, 167
- OP-  
TIX\_PRIMITIVE\_TYPE\_FLAGS\_CUSTOM, 168
- OP-  
TIX\_PRIMITIVE\_TYPE\_FLAGS\_ROUND\_CATMULLROM, 168
- OP-  
TIX\_PRIMITIVE\_TYPE\_FLAGS\_ROUND\_CUBIC\_BSPLINE, 168

- OP-  
TIX\_PRIMITIVE\_TYPE\_FLAGS\_ROUND\_LINEAR, 168
- OP-  
TIX\_PRIMITIVE\_TYPE\_FLAGS\_ROUND\_QUADRATIC\_BSPLINE, 168
- OP-  
TIX\_PRIMITIVE\_TYPE\_FLAGS\_TRIANGLE, 168
- OP-  
TIX\_PRIMITIVE\_TYPE\_ROUND\_CATMULLROM, 167
- OP-  
TIX\_PRIMITIVE\_TYPE\_ROUND\_CUBIC\_BSPLINE, 167
- OP-  
TIX\_PRIMITIVE\_TYPE\_ROUND\_LINEAR, 167
- OP-  
TIX\_PRIMITIVE\_TYPE\_ROUND\_QUADRATIC\_BSPLINE, 167
- OPTIX\_PRIMITIVE\_TYPE\_TRIANGLE, 167
- OP-  
TIX\_PROGRAM\_GROUP\_FLAGS\_NONE, 168
- OP-  
TIX\_PROGRAM\_GROUP\_KIND\_CALLABLES, 169
- OP-  
TIX\_PROGRAM\_GROUP\_KIND\_EXCEPTION, 168
- OP-  
TIX\_PROGRAM\_GROUP\_KIND\_HITGROUP, 168
- OPTIX\_PROGRAM\_GROUP\_KIND\_MISS, 168
- OP-  
TIX\_PROGRAM\_GROUP\_KIND\_RAYGEN, 168
- OPTIX\_PROPERTY\_TYPE\_AABBS, 158
- OP-  
TIX\_PROPERTY\_TYPE\_COMPACTED\_SIZE, 158
- OP-  
TIX\_QUERY\_FUNCTION\_TABLE\_OPTION\_DUMMY, 169
- OP-  
TIX\_RAY\_FLAG\_CULL\_BACK\_FACING\_TRIANGLES, 169
- OP-  
TIX\_RAY\_FLAG\_CULL\_DISABLED\_ANYHIT, 169
- OP-  
TIX\_RAY\_FLAG\_CULL\_ENFORCED\_ANYHIT, 170
- OP-  
TIX\_RAY\_FLAG\_CULL\_FRONT\_FACING\_TRIANGLES, 169
- OPTIX\_RAY\_FLAG\_DISABLE\_ANYHIT, 169
- OP-  
TIX\_RAY\_FLAG\_DISABLE\_CLOSESTHIT, 169
- OPTIX\_RAY\_FLAG\_ENFORCE\_ANYHIT, 169
- OPTIX\_RAY\_FLAG\_NONE, 169
- OP-  
TIX\_RAY\_FLAG\_TERMINATE\_ON\_FIRST\_HIT, 169
- OPTIX\_SBT\_RECORD\_ALIGNMENT, 144
- OPTIX\_SBT\_RECORD\_HEADER\_SIZE, 144
- OPTIX\_SUCCESS, 170
- OPTIX\_TRANSFORM\_BYTE\_ALIGNMENT, 144
- OP-  
TIX\_TRANSFORM\_FORMAT\_MATRIX\_FLOAT12, 171
- OPTIX\_TRANSFORM\_FORMAT\_NONE, 171
- OPTIX\_TRANSFORM\_TYPE\_INSTANCE, 172
- OP-  
TIX\_TRANSFORM\_TYPE\_MATRIX\_MOTION\_TRANSFORM, 171
- OPTIX\_TRANSFORM\_TYPE\_NONE, 171
- OP-  
TIX\_TRANSFORM\_TYPE\_SRT\_MOTION\_TRANSFORM, 172
- OP-  
TIX\_TRANSFORM\_TYPE\_STATIC\_TRANSFORM, 171
- OP-  
TIX\_TRAVERSABLE\_GRAPH\_FLAG\_ALLOW\_ANY, 172
- OP-  
TIX\_TRAVERSABLE\_GRAPH\_FLAG\_ALLOW\_SINGLE, 172
- OP-  
TIX\_TRAVERSABLE\_GRAPH\_FLAG\_ALLOW\_SINGLE, 172



- OP-
  - TIX\_TRAVERSABLE\_TYPE\_MATRIX\_MOTION\_TRANSFORM, 172
- OP-
  - TIX\_TRAVERSABLE\_TYPE\_SRT\_MOTION\_TRANSFORM, 172
- OP-
  - TIX\_TRAVERSABLE\_TYPE\_STATIC\_TRANSFORM, 172
- OPTIX\_VERTEX\_FORMAT\_FLOAT2, 173
- OPTIX\_VERTEX\_FORMAT\_FLOAT3, 173
- OPTIX\_VERTEX\_FORMAT\_HALF2, 173
- OPTIX\_VERTEX\_FORMAT\_HALF3, 173
- OPTIX\_VERTEX\_FORMAT\_NONE, 173
- OPTIX\_VERTEX\_FORMAT\_SNORM16\_2, 173
- OPTIX\_VERTEX\_FORMAT\_SNORM16\_3, 173
- OptixAabb, 144
- OptixAccelBufferSizes, 144
- OptixAccelBuildOptions, 144
- OptixAccelEmitDesc, 145
- OptixAccelPropertyType, 145, 158
- OptixAccelRelocationInfo, 145
- OptixBuildFlags, 145, 158
- OptixBuildInput, 145
- OptixBuildInputCurveArray, 145
- OptixBuildInputCustomPrimitiveArray, 146
- OptixBuildInputInstanceArray, 146
- OptixBuildInputTriangleArray, 146
- OptixBuildInputType, 146, 158
- OptixBuildOperation, 146, 159
- OptixBuiltInOptions, 147
- OptixCompileDebugLevel, 147, 159
- OptixCompileOptimizationLevel, 147, 160
- OptixCurveEndcapFlags, 147, 160
- OptixDenoiser, 147
- OptixDenoiserGuideLayer, 147
- OptixDenoiserLayer, 147
- OptixDenoiserModelKind, 148, 160
- OptixDenoiserOptions, 148
- OptixDenoiserParams, 148
- OptixDenoiserSizes, 148
- OptixDeviceContext, 148
- OptixDeviceContextOptions, 148
- OptixDeviceContextValidationMode, 148, 161
- OptixDeviceProperty, 149, 161
- OptixExceptionCodes, 149, 162
- OptixExceptionFlags, 149, 163
- OptixGeometryFlags, 149, 164
- OptixInstanceFormat, 149, 164
- OptixImage2D, 149
- OptixIndicesFormat, 149, 164
- OptixInstance, 150
- OptixInstanceFlags, 150, 164
- OptixLogCallback, 150
- OptixMatrixMotionTransform, 150
- OptixModule, 151
- OptixModuleCompileBoundValueEntry, 151
- OptixModuleCompileOptions, 152
- OptixModuleCompileState, 152, 165
- OptixMotionFlags, 152, 165
- OptixMotionOptions, 152
- OptixPayloadSemantics, 152, 165
- OptixPayloadType, 153
- OptixPayloadTypeID, 153, 166
- OptixPipeline, 153
- OptixPipelineCompileOptions, 153
- OptixPipelineLinkOptions, 153
- OptixPixelFormat, 153, 167
- OptixPrimitiveType, 153, 167
- OptixPrimitiveTypeFlags, 153, 167
- OptixProgramGroup, 154
- OptixProgramGroupCallables, 154
- OptixProgramGroupDesc, 154
- OptixProgramGroupFlags, 154, 168
- OptixProgramGroupHitgroup, 154
- OptixProgramGroupKind, 154, 168
- OptixProgramGroupOptions, 154
- OptixProgramGroupSingleModule, 154
- OptixQueryFunctionTable\_t, 155
- OptixQueryFunctionTableOptions, 155, 169
- OptixRayFlags, 155, 169
- OptixResult, 155, 170
- OptixShaderBindingTable, 155
- OptixSRTData, 155
- OptixSRTMotionTransform, 156
- OptixStackSizes, 157
- OptixStaticTransform, 157
- OptixTask, 157
- OptixTransformFormat, 157, 171
- OptixTransformType, 157, 171
- OptixTraversableGraphFlags, 157, 172
- OptixTraversableHandle, 157
- OptixTraversableType, 157, 172
- OptixVertexFormat, 157, 172
- OptixVisibilityMask, 158

tz

- OptixSRTData, 227
- usesMotionBlur
  - OptixBuiltInISOOptions, 198
  - OptixPipelineCompileOptions, 218
- usesPrimitiveTypeFlags
  - OptixPipelineCompileOptions, 218
- Utilities, 175
  - optixInit, 176
  - optixInitWithHandle, 176
  - optixUninitWithHandle, 176
  - optixUtilAccumulateStackSizes, 176
  - optixUtilComputeStackSizes, 176
  - optixUtilComputeStackSizesCssCCTree, 177
  - optixUtilComputeStackSizesDCSplit, 177
  - optixUtilComputeStackSizesSimplePath-
    - Tracer, 178
  - optixUtilDenoiserInvokeTiled, 179
  - optixUtilDenoiserSplitImage, 180
  - optixUtilGetPixelStride, 180
- validationMode
  - OptixDeviceContextOptions, 202
- vertexBuffers
  - OptixBuildInputCurveArray, 192
  - OptixBuildInputTriangleArray, 197
- vertexFormat
  - OptixBuildInputTriangleArray, 197
- vertexStrideInBytes
  - OptixBuildInputCurveArray, 192
  - OptixBuildInputTriangleArray, 197
- visibilityMask
  - OptixInstance, 212
- width
  - OptixImage2D, 211
- widthBuffers
  - OptixBuildInputCurveArray, 193
- widthStrideInBytes
  - OptixBuildInputCurveArray, 193
- WIN32\_LEAN\_AND\_MEAN
  - optix\_stubs.h, 350
- withoutOverlapScratchSizeInBytes
  - OptixDenoiserSizes, 201
- withOverlapScratchSizeInBytes
  - OptixDenoiserSizes, 201