

zSpace SDK 3.0.0 Migration Guide

The zSpace SDK 3.0.0 has undergone a complete transformation since version 2.11.0. We have simplified and rewritten all the APIs in C, very similar to the the version 2.11.0 SDK's CoreInterface. The new APIs can be organized in three logical groups: Display, Stereo, and Tracking. Additionally, we have moved to a single header and single static library:

- zSpace.h
- zSpaceApi.lib

The new zSpace SDK's runtime libraries are distributed with the new 3.0.0 System Software.

API Changes

The following sections map functions from version 2.11.0 to version 3.0.0. In some cases, an older function is replaced by a new function used with a specific enum as the parameter. The enum is listed in the Comments.

DisplayInfo

Version 2.11.0	Version 3.0.0	Comments
DisplayInfo::getNumDisplays()	zsGetNumDisplays()	
DisplayInfo::getDisplayIndex()	Removed	
DisplayInfo::getDisplay(int x, int y)	zsFindDisplay()	
DisplayInfo::getDisplay(int index)	zsFindDisplayByIndex()	
DisplayInfo::isZSpaceHardwarePresent()	zslsDisplayHardwarePresent()	
DisplayInfo::getViewPortOffset()	zsGetCoordinateSpaceTransform()	ZS_COORDINATE_SPACE_DISPLAY to ZS_COORDINATE_SPACE_VIEWPORT
DisplayInfo::getTrackerToCameraSpaceTransform()	zsGetCoordinateSpaceTransform()	ZS_COORDINATE_SPACE_TRACKER to ZS_COORDINATE_SPACE_CAMERA

Display

Version 2.11.0	Version 3.0.0	Comments
Display::isZSpaceDisplay	zsGetDisplayType()	
Display::displayNumber	zsGetDisplayNumber()	
Display::displayPosition	zsGetDisplayPosition()	

Display::displayResolution	zsGetDisplayNativeResolution()	
Display::displaySize	zsGetDisplaySize()	
Display::displayOffset	zsGetFrustumCameraOffset()	
Display::displayAngle	zsGetDisplayAngle()	
Display::displayVerticalRefreshRate	zsGetDisplayVerticalRefreshRate()	
Display::adapterIndex	zsGetDisplayAdapterIndex()	
Display::monitorIndex	zsGetDisplayMonitorIndex()	
Display::adapterName	zsGetDisplayAttribute()	ZS_DISPLAY_ATTRIBUTE_ADAPTER_NAME
Display::adapterString	zsGetDisplayAttribute()	ZS_DISPLAY_ATTRIBUTE_ADAPTER_STRING
Display::adapterId	zsGetDisplayAttribute()	ZS_DISPLAY_ATTRIBUTE_ADAPTER_ID
Display::adapterKey	zsGetDisplayAttribute()	ZS_DISPLAY_ATTRIBUTE_ADAPTER_KEY
Display::monitorName	zsGetDisplayAttribute()	ZS_DISPLAY_ATTRIBUTE_MONITOR_NAME
Display::monitorString	zsGetDisplayAttribute()	ZS_DISPLAY_ATTRIBUTE_MONITOR_STRING
Display::monitorId	zsGetDisplayAttribute()	ZS_DISPLAY_ATTRIBUTE_MONITOR_ID
Display::monitorKey	zsGetDisplayAttribute()	ZS_DISPLAY_ATTRIBUTE_MONITOR_KEY

StereoViewport

Version 2.11.0	Version 3.0.0	Comments
StereoViewport::StereoViewport()	zsCreateViewport()	
StereoViewport::~~StereoViewport()	zsDestroyViewport()	
StereoViewport::getStereoFrustum()	zsFindFrustum()	
StereoViewport::move() StereoViewport::setX() StereoViewport::setY()	zsSetViewportPosition()	
StereoViewport::getX() StereoViewport::getY()	zsGetViewportPosition()	
StereoViewport::resize() StereoViewport::setWidth() StereoViewport::setHeight()	zsSetViewportSize()	
StereoViewport::getWidth() StereoViewport::getHeight()	zsGetViewportSize()	

StereoFrustum

Version 2.11.0	Version 3.0.0	Comments
StereoFrustum::setInterPupillaryDistance()	zsSetFrustumAttribute()	ZS_FRUSTUM_ATTRIBUTE_IPD
StereoFrustum::getInterPupillaryDistance()	zsGetFrustumAttribute()	ZS_FRUSTUM_ATTRIBUTE_IPD
StereoFrustum::setStereoLevel()	Removed	Directly modify IPD instead
StereoFrustum::getStereoLevel()	Removed	
StereoFrustum::setWorldScale()	zsSetFrustumAttribute()	ZS_FRUSTUM_ATTRIBUTE_VIEWER_SCALE
StereoFrustum::getWorldScale()	zsGetFrustumAttribute()	ZS_FRUSTUM_ATTRIBUTE_VIEWER_SCALE
StereoFrustum::setFieldOfViewScale()	zsSetFrustumAttribute()	ZS_FRUSTUM_ATTRIBUTE_FOV_SCALE
StereoFrustum::getFieldOfViewScale()	zsGetFrustumAttribute	ZS_FRUSTUM_ATTRIBUTE_FOV_SCALE
StereoFrustum::setZeroParallaxOffset()	Removed	
StereoFrustum::getZeroParallaxOffset()	Removed	
StereoFrustum::getZeroParallaxOffsetForPosition()	Removed	
StereoFrustum::setNearClip()	zsSetFrustumAttribute()	ZS_FRUSTUM_ATTRIBUTE_NEAR_CLIP
StereoFrustum::getNearClip()	zsGetFrustumAttribute()	ZS_FRUSTUM_ATTRIBUTE_NEAR_CLIP
StereoFrustum::setFarClip()	zsSetFrustumAttribute()	ZS_FRUSTUM_ATTRIBUTE_FAR_CLIP
StereoFrustum::getFarClip()	zsGetFrustumAttribute()	ZS_FRUSTUM_ATTRIBUTE_FAR_CLIP
StereoFrustum::setHeadPose()	zsSetFrustumHeadPose() OR zsUpdate()	
StereoFrustum::getHeadPose()	zsGetFrustumHeadPose()	
StereoFrustum::setHeadTrackingEnabled()	Removed	
StereoFrustum::isHeadTrackingEnabled()	Removed	
StereoFrustum::setHeadRotationEnabled()	Removed	
StereoFrustum::isHeadRotationEnabled()	Removed	
StereoFrustum::getViewMatrix()	zsGetFrustumViewMatrix()	
StereoFrustum::getProjectionMatrix()	zsGetFrustumProjectionMatrix()	

StereoFrustum::getEyePosition	zsGetFrustumEyePosition()	
StereoFrustum::getBounds()	zsGetFrustumBounds()	

StereoLeftRightDetect

Version 2.11.0	Version 3.0.0	Comments
StereoLeftRightDetect::initialize()	zsCreateStereoBuffer()	
StereoLeftRightDetect::update()	zsBeginStereoBufferFrame()	
StereoLeftRightDetect::sync()	zsSyncStereoBuffer()	
StereoLeftRightDetect::shutdown()	zsDestroyStereoBuffer()	

TrackerSystem

Version 2.11.0	Version 3.0.0	Comments
TrackerSystem::getTrackerDevice()	zsFindTrackerDeviceByName()	
TrackerSystem::getTrackerDeviceByIndex()	zsFindTrackerDevice()	
TrackerSystem::getNumTrackerDevices()	zsGetNumTrackerDevices()	
TrackerSystem::getDefaultTrackerTarget()	zsFindTargetByType()	
TrackerSystem::getTrackerMouseSimulator()	Removed	
TrackerSystem::captureTargets()	zsUpdate()	

TrackerDevice

Version 2.11.0	Version 3.0.0	Comments
TrackerDevice::getName()	zsGetTrackerDeviceName()	
TrackerDevice::getPluginName()	Removed	
TrackerDevice::addTarget()	Removed	
TrackerDevice::removeTarget()	Removed	
TrackerDevice::removeTargetByName()	Removed	
TrackerDevice::removeAllTargets()	Removed	
TrackerDevice::getNumTargets()	zsGetNumTargets()	

TrackerDevice::getTarget()	zsFindTargetByName()	
TrackerDevice::getTargetByIndex()	zsFindTarget()	
TrackerDevice::captureTargets()	Removed	
TrackerDevice::setBaseTransform()	Removed	
TrackerDevice::getBaseTransform()	Removed	
TrackerDevice::setEnabled()	zsSetTrackerDeviceEnabled()	
TrackerDevice::isEnabled()	zslsTrackerDeviceEnabled()	

TrackerTarget

Version 2.11.0	Version 3.0.0	Comments
TrackerTarget::getName()	zsGetTargetName()	
TrackerTarget::getId()	Removed	
TrackerTarget::getType()	Removed	
TrackerTarget::getTrackerDevice()	Removed	
TrackerTarget::setTrackerFilter()	Removed	
TrackerTarget::getTrackerFilter()	Removed	
TrackerTarget::setBaseTransform()	Removed	
TrackerTarget::getBaseTransform()	Removed	
TrackerTarget::setInitialPose()	Removed	
TrackerTarget::getLocalPose()	Removed	
TrackerTarget::getPose()	zsGetTargetPose()	
TrackerTarget::setPoseBufferingEnabled()	zsSetTargetPoseBufferingEnabled()	
TrackerTarget::isPoseBufferingEnabled()	zslsTargetPoseBufferingEnabled()	
TrackerTarget::resizePoseBuffer()	zsResizeTargetPoseBuffer()	
TrackerTarget::getPoseBuffer()	Removed	
TrackerTarget::getPoseBufferRange()	zsGetTargetPoseBuffer()	
TrackerTarget::getBufferedPose()	Removed	
TrackerTarget::isVisible()	zslsTargetVisible()	

TrackerTarget::isEnabled()	zsisTargetEnabled()	
TrackerTarget::getCapability()	Removed	
TrackerTarget::addMoveHandler()	zsAddTrackerEventHandler()	ZS_TRACKER_EVENT_MOVE and ZSTrackerMoveEventData
TrackerTarget::removeMoveHandler()	zsRemoveTrackerEventHandler()	
TrackerTarget::setMoveEventTimeThreshold() TrackerTarget::setMoveEventDistanceThreshold() TrackerTarget::setMoveEventRotationThreshold()	zsSetTargetMoveEventThresholds()	
TrackerTarget::getMoveEventTimeThreshold() TrackerTarget::getMoveEventDistanceThreshold() TrackerTarget::getMoveEventRotationThreshold()	zsGetTargetMoveEventThresholds()	

ITrackerButtonCapability

Version 2.11.0	Version 3.0.0	Comments
ITrackerButtonCapability::getNumButtons()	zsGetNumTargetButtons()	
ITrackerButtonCapability::isButtonPressed()	zsisTargetButtonPressed()	
ITrackerButtonCapability::addButtonPressHandler()	zsAddTrackerEventHandler()	ZS_TRACKER_EVENT_BUTTON_PRESS & ZSTrackerButtonEventData
ITrackerButtonCapability::removeButtonPressHandler()	zsRemoveTrackerEventHandler()	
ITrackerButtonCapability::addButtonReleaseHandler()	zsAddTrackerEventHandler()	ZS_TRACKER_EVENT_BUTTON_RELEASE & ZSTrackerButtonEventData
ITrackerButtonCapability::removeButtonReleaseHandler()	zsRemoveTrackerEventHandler()	
ITrackerButtonCapability::addButtonClickHandler()	Removed	
ITrackerButtonCapability::removeButtonClickHandler()	Removed	
ITrackerButtonCapability::addButtonDoubleClickHandler()	Removed	
ITrackerButtonCapability::removeButtonDoubleClickHandler()	Removed	

ITrackerLedCapability

Version 2.11.0	Version 3.0.0	Comments
----------------	---------------	----------

ITrackerLedCapability::setEnabled()	zsSetTargetLedEnabled()	
ITrackerLedCapability::isEnabled()	zsIsTargetLedEnabled()	
ITrackerLedCapability::setColor()	zsSetTargetLedColor()	
ITrackerLedCapability::getColor()	zsGetTargetLedColor()	
ITrackerLedCapability::isOn()	zsIsTargetLedOn()	

ITrackerTapCapability

Version 2.11.0	Version 3.0.0	Comments
ITrackerTapCapability::isTapPressed()	zsIsTargetTapPressed()	
ITrackerTapCapability::getTapHoldThreshold()	zsSetTargetTapHoldThreshold()	
ITrackerTapCapability::setTapHoldThreshold()	zsGetTargetTapHoldThreshold()	
ITrackerTapCapability::addTapHandler()	zsAddTrackerEventHandler()	ZS_TRACKER_EVENT_TAP_SINGLE
ITrackerTapCapability::removeTapHandler()	zsRemoveTrackerEventHandler()	
ITrackerTapCapability::addDoubleTapHandler()	zsAddTrackerEventHandler()	ZS_TRACKER_EVENT_TAP_DOUBLE
ITrackerTapCapability::removeDoubleTapHandler()	zsRemoveTrackerEventHandler()	
ITrackerTapCapability::addPressHandler()	zsAddTrackerEventHandler()	ZS_TRACKER_EVENT_TAP_PRESS
ITrackerTapCapability::removePressHandler()	zsRemoveTrackerEventHandler()	
ITrackerTapCapability::addReleaseHandler()	zsAddTrackerEventHandler()	ZS_TRACKER_EVENT_TAP_RELEASE
ITrackerTapCapability::removeReleaseHandler()	zsRemoveTrackerEventHandler()	
ITrackerTapCapability::addHoldHandler()	zsAddTrackerEventHandler()	ZS_TRACKER_EVENT_TAP_HOLD
ITrackerTapCapability::removeHoldHandler()	zsRemoveTrackerEventHandler()	

ITrackerVibrateCapability

Version 2.11.0	Version 3.0.0	Comments
ITrackerVibrateCapability::setEnabled()	zsSetTargetVibrationEnabled()	
ITrackerVibrateCapability::isEnabled()	zsIsTargetVibrationEnabled()	
ITrackerVibrateCapability::setOnPeriod()	Removed	Now a parameter of zsStartTargetVibration()
ITrackerVibrateCapability::getOnPeriod()	Removed	

ITrackerVibrateCapability::setOffPeriod()	Removed	Now a parameter of zsStartTargetVibration()
ITrackerVibrateCapability::getOffPeriod()	Removed	
ITrackerVibrateCapability::setRepeatCount()	Removed	Now a parameter of zsStartTargetVibration()
ITrackerVibrateCapability::getRepeatCount()	Removed	
ITrackerVibrateCapability::startVibration()	zsStartTargetVibration()	
ITrackerVibrateCapability::stopVibration()	zsStopTargetVibration()	
ITrackerVibrateCapability::isVibrating()	zsIsTargetVibrating()	

TrackerMouseSimulator

Version 2.11.0	Version 3.0.0	Comments
TrackerMouseSimulator::setTrackerTarget()	zsSetMouseEmulationTarget()	
TrackerMouseSimulator::getTrackerTarget()	zsGetMouseEmulationTarget()	
TrackerMouseSimulator::setEnabled()	zsSetMouseEmulationEnabled()	
TrackerMouseSimulator::isEnabled()	zsIsMouseEmulationEnabled()	
TrackerMouseSimulator::setMaxSimulationDistance()	zsSetMouseEmulationMaxDistance()	
TrackerMouseSimulator::getMaxSimulationDistance()	zsGetMouseEmulationMaxDistance()	
TrackerMouseSimulator::setMovementMode()	zsSetMouseEmulationMovementMode()	
TrackerMouseSimulator::getMovementMode()	zsGetMouseEmulationMovementMode()	
TrackerMouseSimulator::setButtonMapping()	zsSetMouseEmulationButtonMapping()	
TrackerMouseSimulator::getButtonMapping()	zsGetMouseEmulationButtonMapping()	
TrackerMouseSimulator::setPoseLagTime()	Removed	
TrackerMouseSimulator::getPoseLagTime()	Removed	
TrackerMouseSimulator::getCurrentIntersectionInfo()	Removed	

TrackerDisplayIntersection

Version 2.11.0	Version 3.0.0	Comments
TrackerDisplayIntersection::intersect()	zsIntersectDisplay()	
TrackerDisplayIntersection::setDisplayIntersectable()	Removed	

TrackerDisplayIntersection::isDisplayIntersectable()	Removed	
--	---------	--

Changes In General Usage

Finding the zSpace Display

```
// OLD IMPLEMENTATION:
zspace::common::DisplayInfo displayInfo;

int index = 0;
int numDisplays = displayInfo.getNumDisplays();
while (index < numDisplays && !displayInfo.getDisplay(index)->isZSpaceDisplay)
    index++;

if (index < numDisplays)
    const zspace::common::DisplayInfo::Display* display = displayInfo.getDisplay(index);
```

```
// NEW IMPLEMENTATION:
ZSContext zSpaceContext = ...; // Assume this has been initialized by zsInitialize(...)

ZSHandle displayHandle = NULL;
ZSError error = zsFindDisplayByType(zSpaceContext, ZS_DISPLAY_TYPE_ZSPACE, 0, &displayHandle);

if (error != ZS_ERROR_OKAY)
    // The zSpace display was not found.
```

Converting a Target Pose from Tracker to Camera Space

```
// OLD IMPLEMENTATION:
zspace::tracker::TrackerSystem* trackerSystem = ...;
zspace::stereo::StereoWindow* stereoWindow = ...;

// Get the primary tracker target.
zspace::tracker::TrackerTarget* primaryTarget =
    trackerSystem->getDefaultTrackerTarget(zspace::tracker::TrackerTarget::TYPE_PRIMARY);

// Check to make sure the primary target is not NULL. If the primary target is NULL,
// this signifies that the default primary target cannot be found.
if (primaryTarget)
{
    // Get the primary target's tracker space pose.
    zspace::common::Matrix4 primaryPose = zspace::common::Matrix4::IDENTITY();
    primaryTarget->getPose(primaryPose);
```

```

// Get the tracker to camera space transform and viewport offset.
zspace::common::Matrix4 trackerToCameraSpace = zspace::common::Matrix4::IDENTITY();
zspace::common::Vector3 viewportOffset      = zspace::common::Vector3::ZERO();

const zspace::common::DisplayInfo::Display* display = stereoWindow->getCurrentDisplay();
if (display)
{
    trackerToCameraSpace = zspace::common::DisplayInfo::getTrackerToCameraSpaceTransform(display);
    viewportOffset      = zspace::common::DisplayInfo::getViewportOffset(display,
                                                                    stereoWindow->getX(),
                                                                    stereoWindow->getY(),
                                                                    stereoWindow->getWidth(),
                                                                    stereoWindow->getHeight());
}

// Transform the primary target's pose from tracker space to camera space.
zspace::common::Matrix4 primaryPoseCameraSpace = trackerToCameraSpace *
                                                zspace::common::Matrix4::getTrans(-viewportOffset) *
                                                primaryPose;
}

```

```

// NEW IMPLEMENTATION:
ZSContext zSpaceContext = ...; // Assume this has been initialized by zsInitialize(...)
ZSHandle viewportHandle = ...; // Assume this has been initialized by zsCreateViewport(...)

// Get a handle to the primary tracker target.
ZSHandle primaryTargetHandle = NULL;
ZSError error = zsFindTargetByType(zSpaceContext, ZS_TARGET_TYPE_PRIMARY, 0, &primaryTargetHandle);

if (error == ZS_ERROR_OKAY)
{
    // Get the primary target's tracker space pose.
    ZSTrackerPose primaryPose;
    zsGetTargetPose(primaryTargetHandle, &primaryPose);

    // Transform the primary target's pose from tracker to camera space.
    zsTransformMatrix(viewportHandle, ZS_COORDINATE_SPACE_TRACKER, ZS_COORDINATE_SPACE_CAMERA,
                      &primaryPose.matrix);
}

```