

## Openview SDK 1.0 Release

### Notes

- **Important note:** InstallOpenViewSDK64.exe will include documentation and sample application. This executable should be used for development of your application. The InstallOpenView64.exe does not include the documentation and sample.
- **Important Installation note:** To update OpenView SDK from version 1.0R8 or earlier, Uninstall the older version then install the new version.

#### OpenView SDK 1.0R46

2019-11-13

##### Bug Fixed:

- Fixed Memory leaks with Instrumentation library.
- Fixed memory leaks with Storage library (ovs, .ovd and .odat files).
- Give correct value for LastElementOfLastLaw.
- Adjust cells quantity with .ovd data files.
- Give correct value for compression factor with .odat data files.
- Sample Quantity is computed correctly when saving .odat as an .ovd.
- Better exception management with .fpd data files.

#### OpenView SDK 1.0R45

2019-11-01

##### New Features

##### Change log

- Support open .odat data files while using debug with mixed mode.
- Support creating unlimited data files size. (see WriteDataFile snippet)

##### Bug Fixed:

- Fixed amplitude axes resolution.

#### OpenView SDK 1.0R44

2019-10-21

##### Bug Fixed:

- Amplitude resolution was wrong with .odat files
- Sampling amplitude was wrong with fpd files including unrectified data

## OpenView SDK 1.0R43

2019-10-13

---

Version no available

## OpenView SDK 1.0R42

2019-10-13

---

### Bug Fixed:

Bug with the data descriptors in a odat \_le with more than 2 contexts

## OpenView SDK 1.0R41

2019-10-08

---

### New Features

- Added DLA DMA support for Odat files

### Bug Fixed:

- Fixed the no data values problems that occurred with some Odat files

## OpenView SDK 1.0R40

2019-09-27

---

### New Features

- Storage: adjust the distinction between beam gain and beam set gain for FPD files
- Storage: Save inspection date and time automatically
- 2 new filters added in option (ETO 507) Note: This option needs to be unlocked by Olympus factory.

### Bug \_fixed:

- OVSDK-722: Storage: some capabilities were doubled
- OVSDK-558: Instrumentation: remove potential problem with IP address 192.168.0.0
- OVSDK-696: Configuration Tool: Show log when application is started and improve main buttons appearance

**New Features**

- Official release of the Unined Library File Format (Storage Library)
- Interpolate A-Scans with compression to retrieve maximums

**New Features**

- Add refracted angle parameter for conventional configuration
- Fix ascan amplitude limits for half-wave negative rectification in a FocusPC \_le
- Fix malformed openview registry key
- ODAT: Fix ascan and cscan data descriptor values
- ODAT: Add access to encoder parameters
- ODAT: Add device capabilities
- ODAT: Add exit points

**New Features**

- Installer correctly cleans installation folder when a version upgrade is done
- Revert to older version of Configuration Tool

**Note:** Important to uninstall "[InstallOpenViewSDK64](#)" and "[InstallOpenView64](#)" before install this version.

#### **New Features**

- Fixed: The AreEquals function on AcquisitionConfiguration with recurrence values.
- Fixed: The blocking ApplyConfiguration when using the encoder firing trigger mode.
- Data statuses when reading Ascan and Cscan data files.

#### **New Features**

- Storage project: Added Ultrasound and Amplitude for Ascan / Cscan data axis descriptors values with .ovd and .fpd files.
- Instrumentation: Fixed beam recurrence value is now in nanoseconds.
- Instrumentation / Storage: Logs improvement

Note:

Versions R31 to R3 (inclusively) are not compatible. ([Need to compile your code](#))

#### **Bug\_fixed:**

- Fix binary compatibility.
- Installer now correctly update all libraries.

#### **New Features**

- Storage project:
  - Scan and Index offset now available from FPD, OVD and for new files

**New Features**

- Instrumentation: Modified the Equals function to use shared pointer and renamed AreEquals
- Instrumentation: Added FocusPxConstants.h definitions
- Storage project:
  - Added the new data access interface
  - Removed Dal.lib from the public interface

**New Features**

- New Configuration Tool
- Labview sample
- Storage project:
  - New interface for the data access
  - Added an Equality comparison between an AcquisitionConfiguration (Instrumentation) and a DeviceConfiguration (Storage)

**New Features**

- Maximum PRF conditional to a configuration setting
- Interpolation of Gate I C-scan data

**New Features**

- Official release of the Unified File Format library

**Bug\_fixed:**

- Releasing all objects maintains a Device over a TCP disconnect

**New Features**

- Ability to store a device's calibration has been added to the Storage Pre-Release Project.

**Bug Fixes**

- Acquisition state machine must be separated in 2
  - IAcquisition::CreateEx
  - IAcquisition::WaitForDataEx

**New Features**

- Unified Library File Format
  - Indications were added
  - A File Container was added to facilitate the packing of indications with acquired data

**Bug Fixes**

- Unified Library File Format
  - Installer will not update the registry key

**New Features**

- Pre-release of the Storage project.
  - Added Data Access Layer to the storage library.

**New Features**

- Improved the ICSan class to simplify retrieval of the CScan. GetCrossingTime and GetBeamFiringOrder are the new functions added.
  - Note: If you upgraded your version from R19 to R22 or higher, recompilation is required.

**Bug Fixes**

- Binary compatibility with version R18 and previous restored. Version R19 to R21 not binary compatible.

**New Features**

- Pre-release of the Storage project.

**New Features**

- 36085 - Silent installer does not upgrade from a previous version.
  - Note: Before installing this new version, un-install the previous OpenView version. This will be valid for all previous version but will not necessary for future version.
- 35927 - Negative start time crash.

**Bug Fixes**

- Add the possibility to separate the A-Scan start and TCG start, allowing to have an A-Scan starting at the main bang while the TCG starts at the front wall.

## OpenView SDK 1.0R19

2018-12-11

---

### New Features

- 35949 - Add 25 ns pulse width option and extended BW option;
- 35949 - Add band pass Filter 11 MHz (1.1-20.5MHz) option

Note: These options need to be unlocked by Olympus factory.

---

## OpenView SDK 1.0R18

2018-11-29

- 
- Demo version

---

## OpenView SDK 1.0R17

2018-11-19

---

### Bug Fixes

- 35748 - When Focus PX hard drive D is full the Software cannot be Deploy.
- 36078 - Encoder does not reset.

---

## OpenView SDK 1.0R16

2018-10-31

- 
- Pre-release version for storage demo,

---

## OpenView SDK 1.0R15

2018-10-10

---

### Bug Fixes

- 36006: Exception on acquisition start while switching contexts.

---

## OpenView SDK 1.0R14

2018-09-24

---

### Bug Fixes

- 36001: Switching from UT to PA back to UT may restrict Pitch-Catch selection



- Removed 32 bits support

### Bug Fixes

- 35984: Beam with amplitude near to 0 give a gain problem.
- 35930: Installing the 1.0R13 will now correctly uninstall previous versions of OpenView
- Enforce a device to be reprogrammed when switching acquisitions to prevent undefined behavior
- Fix crash that happened while changing layouts too quickly.

### Bug Fixes

- 35838: DigitalOutput: SetState; Getting the state of digital output pin did not have always the good value.
- 35880: Change from Log to Lin Fix.

### Bug Fixes

- 35854: Law file element offset is not taken into account.

### New Features

- Sample application
- support private IP addresses other than 192.168.0.1
- Noticed we were not using the extra feature set from the .Net 4.5 framework so we lowered our references from 4.5 to 4.0 which means that if you wish to use the SDK in .Net it would work in Visual Studio 2010.

### Bug Fixes

- 35786 - Echo synchronisation mode with TCG

### New Features

- Configuration Tool (FocusPC version)
- Diagnostic tools
- Support 65535 flag in law file import (TTU)
- Need an installer containing only runtime libraries
- New installer has slightly changed as it comes in 4 versions
  - 32 and 64 bits
  - Programmer and End-User

\*To integrate them correctly refer to technical note in the

We deprecated some functions from the previous version. The functions can still be used but will generate a “warning: deprecated” indication. If this creates compiler errors, please verify the following in the project’s properties:

- *Treat Warnings As Errors* should be set to *No*
- *SDL Checks* should be set to *No*

#### Bug Fixes

- None

**New Features**

- High Dynamic range
- Multi-device support

**Bug Fixes**

- 35617 - No beams have been applied error (manual recurrence)
- 35406 - Code snippets fix
- 35682 - [UT Settings] Pulse Width value causes ApplyConfig error

**New Features**

- Library integration compilation specificity
- Multi-user accounts environment

**Bug Fixes**

- 35413 Recurrence in Manual Mode
- 35538 Installation shows error 1152
- 35539 Encoders going backward issue
- 35615 Improvement of the deployment process stability

**General information:**

The OpenView SDK (SDK: Software Development Kit) is designed for the integration of FocusPX in custom software environments. OpenView SDK is mainly used for the development of application-dedicated software interface and system automation.

**Compatible Programming Languages (64-bit)**

- C++
- C#
- LabVIEW

**Supported Acquisition Units**

- UT4
- FOCUS PX 16:64PR
- FOCUS PX 16:128PR
- FOCUS PX 32:128PR

**Supported Operating Systems**

- Microsoft Windows 10 (64-bit)
- Microsoft Windows 7 (32-bit and 64-bit)

**Note:** You must have administrator privileges to install Openview SDK and configure the computer for data acquisition.

**Computer Requirements**

- **CPU:** Intel Core i7 or Xeon E3
- **RAM:** 16 GB (DDR3 or better)
- **Data storage drive:** SSD
- **Network adaptor:** Gigabit Ethernet card (dedicated for FOCUS PX)
- **Display:** 1280 × 1024 or higher
- One USB port for the HASP security hardware key
- A keyboard and a pointing device

**Available in standard Languages**

## Appendix for OpenView SDK 1.0R8 and newest version:

### End-user requirements for software using OpenView SDK

This section promotes the best practices recommended while integrating OpenView SDK to your software.

#### 1. Required Firewall Rules

The following table lists all ports to enable during the installation of your software. The second column shows the command to execute when adding a firewall rule for the corresponding port.

Port	Command
21	netsh advfirewall firewall add rule name="Olympus OpenView SDK" dir=in action=allow protocol=TCP localport=21
67	netsh advfirewall firewall add rule name="Olympus OpenView SDK" dir=in action=allow protocol=UDP localport=67
68	netsh advfirewall firewall add rule name="Olympus OpenView SDK" dir=out action=allow protocol=UDP remoteport=68
9994	netsh advfirewall firewall add rule name="Olympus OpenView SDK" dir=out action=allow protocol=TCP remoteport=9994,10994,12000,27015
10994	
12000	
27015	

#### 2. End-user Installer

There are 2 installers available for OpenView.

Installer	Description
InstallOpenViewSDK-1.0RXXX.exe	To be installed on the programmer's computer. Includes libraries, documentation and code snippets. This installs on C:\OlympusNDT.
InstallOpenView-1.0RXXX.exe	To be installed on the end-user's computer. Includes libraries only. This installs on C:\Program Files.

Integrators must integrate the end-user's installer into their own installer. Locating and accessing the libraries at runtime requires the use of one of those keys:

Installer	Key
InstallOpenViewSDK-1.0RXXX.exe	HKEY_LOCAL_MACHINE\SOFTWARE\OlympusNDT\OpenView SDK\1.0\VersionPath
InstallOpenView-1.0RXXX.exe	HKEY_LOCAL_MACHINE\SOFTWARE\OlympusNDT\OpenView\1.0\VersionPath

Each installer automatically configures the value of those keys.

To make an installer silent you need to add the command line arguments: /s /v/qn

### 3. Firmware Package

SDK versions and software packages used by the FocusPX are bundled together. Since the SDK always uses the package used on the computer, the application code should search for the latest version installed. Here is a sample code showing good practice.

```
// Select the latest version of firmware packages.
shared_ptr<IFirmwarePackage> package;
auto packages = IFirmwarePackageScanner::GetFirmwarePackageCollection();
if (!packages.empty() )
    package = packages->GetFirmwarePackage(0);

if (package == nullptr)
    throw std::exception("Could not find the firmware package.");

// Start the package on the device.
if (!device->HasPackage(package))
    device->Download(package);

device->Start(package);
```

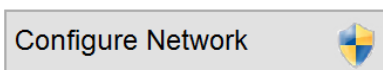
### 4. Config Tool

The configuration tools is installed by the end-user's installer and the root directory is located at the content of registry key :

Installer	Key
InstallOpenViewSDK-1.0RXXX.exe	HKEY_LOCAL_MACHINE\SOFTWARE\OlympusNDT\OpenView SDK\1.0\VersionPath
InstallOpenView-1.0RXXX.exe	HKEY_LOCAL_MACHINE\SOFTWARE\OlympusNDT\OpenView\1.0\VersionPath

The folders “\Tools\Configuration” must be added to this root directory. The program name is ConfigurationTool.exe.

We recommend adding a link in your software to call this configuration tool. The link should show Microsoft shield icon indicating the administrative rights may be required.



**IMPORTANT:** The tool should be called prior to any other SDK calls since it may change some network settings and test connection with devices.

## Appendix A – Matlab

The Matlab code below has been validated. However during our limited tests, we were not able to reach data throughputs of more than 0.1 Mb/s, which is very limitative. We believe this problem could be resolved with more investigation.

### % Import OpenView

```
asmInfo = NET.addAssembly('C:\Program Files\Olympus  
NDT\OpenView\1.0\Bin\v141\OlympusNDT.Instrumentation.NET.dll');  
OlympusNDT.Instrumentation.NET.Utilities.ResolveDependenciesPath()
```

### % Discover device

```
discovery = OlympusNDT.Instrumentation.NET.IDeviceDiscovery.Create('192.168.0.1');  
result = discovery.DiscoverFor(1000);  
device = result.device;
```

### % Setup

```
firmwarePackage =  
OlympusNDT.Instrumentation.NET.IFirmwarePackageScanner.GetFirmwarePackageCollection().GetFirmwarePac  
kage(0)  
device.Start(firmwarePackage)  
utConfig = device.GetConfiguration().GetUltrasoundConfiguration()  
firingBeamSets = utConfig.GetFiringBeamSetCollection()  
convTechno =  
utConfig.GetDigitizerTechnology(OlympusNDT.Instrumentation.NET.UltrasoundTechnology.Conventional)  
beamSetFactory = convTechno.GetBeamSetFactory()  
connectorCollection = convTechno.GetConnectorCollection()  
connector = connectorCollection.GetConnector(4)  
beamSet = beamSetFactory.CreateBeamSetConventional('BS')  
firingBeamSets.Add(beamSet, connector, 0, 0)
```

### % Start acquisition

```
acquisition = OlympusNDT.Instrumentation.NET.IAcquisition.Create(device)  
acquisition.SetRate(1);  
acquisition.ApplyConfiguration()  
acquisition.Start();
```

### % Acquire cycles

```
for iCycle = 0:5  
    waitForResult = acquisition.WaitForData();  
    cycleData = waitForResult.cycleData;  
    aScanCollection = cycleData.GetAScanCollection();  
  
    for iAScan = 0:aScanCollection.GetCount() - 1  
        aScan = aScanCollection.GetAScan(iAScan);  
        ascanSampleQty = int32(aScan.GetSampleQuantity()); % Marshal from uint32 to int32  
        aScanPtr = aScan.GetData();
```

```
        % Matlab indexes start at 1.
```

```
        % OpenView indexes start at 0
```

```
        matlabArray = zeros(1, ascanSampleQty);
```

```
        for iAScanSample = 1:ascanSampleQty
```

```
            % We read the IntPtr by offsetting it by sampleIndex * 4 bytes
```

```
            matlabArray(iAScanSample) = System.Runtime.InteropServices.Marshal.ReadInt32(aScanPtr,
```

```
(iAScanSample - 1)*4);
```

```
        end
```



```
        aScan.Dispose();
    end

    aScanCollection.Dispose();
    cycleData.Dispose();
    waitForDataResult.Dispose(); % This is VERY important, or memory will overflow
end
acquisition.Stop()
```

## **Appendix B – Network Diagnostic**

The following tools can be used for advanced diagnostic on network errors:

- C:\OlympusNDT\OpenView SDK\1.0\Tools\Configuration\InstrumentConfig.exe will verify the general parameters of the network
- If the previous tool identifies a port cannot be acquired, start a command prompt as admin and run:  
> netstat -a -b