NetSDK_JAVA Development

FAQ Manual

V2.0.0

Foreword

Symbol Instructions

The following signal words might appear in the manual.

Signal Words	Meaning
⚠ **DANGER**	Indicates a high potential hazard which, if not avoided, will result in death or serious injury.
⚠ **WARNING**	Indicates a medium or low potential hazard which, if not avoided, could result in slight or moderate injury.
⚠ **CAUTION**	Indicates a potential risk which, if not avoided, could result in property damage, data loss, lower performance, or unpredictable result.
🔑 TIPS	Provides methods to help you solve a problem or save you time.
📖 NOTE	Provides additional information as the emphasis and supplement to the text.

Revision History

Version number	Revisions	Release Date
V2.0.0	Add questions	July 2025
V1.0.0	First release	April 2021

Table of contents

1 Structure Encapsulation

1.1 When Pointer type is used as an input parameter, a null pointer is prompted

Phenomenon

When .getPointer() converts a structure into a pointer type and transmits it into the interface as an input parameter, a null pointer is reported.

Possible cause

You use the structure object as a parameter for calling the interface.

Solutions

- Method 1: Execute the write() operation on the structure, and then the getPointer() operation. You can then write the data of the structure into the pointer and call the interface normally.
- Method 2: Use SetStructDataToPointer (single structure) and SetStructArrToPointerData (structure array) in the SDK ToolKits.java.

1.2 When Pointer type is used as an output parameter, a null pointer is prompted

Phenomenon

When the Pointer type is parsed into the corresponding structure as a method parameter, a null pointer exception is reported.

Possible cause

After calling the interface successfully, you use the structure object directly for parsing the structure.

Solutions

- Method 1: Read the data in the pointer memory into the corresponding structure through the read() method, and then parse the data into required fields.
- Method 2: Use GetStructDataToPointer (single structure) and

GetStructArrToPointerData (structure array) in the ToolKits.java provided by the SDK.

1.3 Unaligned JAVA-encapsulated structure memory causes garbled fields

Phenomenon

When modifying the SDK encapsulation layer or replacing a new dynamic library, the fields for parsing the JAVA-encapsulated structure object are partly or even completely garbled.

Possible cause

When reading memory data, invalid memory is read.

Solution

Step 1 Troubleshoot encapsulation issues.

Step 2 Align the encapsulated class with the C structure byte and keep their field orders consistent.

1.4 The program reports unmatched structure fields

Phenomenon

The error is reported as follows:

```
java.lang.reflect.InvocationTargetException
	at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
	at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
	at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
	at java.lang.reflect.Method.invoke(Method.java:497)
	at com.netsdk.demo.util.CaseMenu.run(CaseMenu.java:78)
	at com.netsdk.demo.customize.querySystemState.QuerySystemStateDemo.runTest(QuerySystemStateDemo.java:230)
	at com.netsdk.demo.customize.querySystemState.QuerySystemStateDemo.main(QuerySystemStateDemo.java:251)
```

```
Caused by: java.lang.Error: Structure.getFieldOrder() on class
com.netsdk.lib.structure.NET_SYSTEM_STATUS does not provide enough names [0] ([])
to match declared fields.
```

Possible cause

The java class uses Structure.

Solution

Use SdkStructure because the JAVA encapsulated structure class uses the wrong parent class (Structure).

1.5 How to convert between Pointer and JAVA object?

Phenomenon

The user is unclear how to convert a Pointer parameter or variable to a JAVA encapsulated structure; or how to assign a JAVA encapsulated structure object to a Pointer.

Solutions

- Method 1

  Step 1 When converting an object to a Pointer, apply for a memory block of the same size.

  Step 2 Assign a Pointer by starting with Set in the ToolKits.java.

  eg:

```
CFG_SPLIT_INFO pSplits = new CFG_SPLIT_INFO();
pSplits.pSplitChannels = new Memory(pSplitChannels.size());
pSplits.emSplitMode = 1;
pSplits.nReturnChannels = 1;
http://wereview.dahuatech.com/Review/ReviewApproval?ReviewID=90801
pSplits.nMaxChannels = 1;
ToolKits.SetStructDataToPointer(pSplitChannels, pSplits.pSplitChannels, 0);
```

- Method 2

  Step 1 Create a JAVA object when reading the data in the Pointer.

  Step 2 Obtain the object by starting with Get in the ToolKits.java.

  eg:

```
ALARM_VEHICLE_INOUT_INFO stuAlarmInfo = new
ALARM_VEHICLE_INOUT_INFO();
ToolKits.GetPointerDataToStruct(pStuEvent, 0, stuAlarmInfo);
```

2 Dynamic Library Loading

2.1 How to load C language dynamic library?

Phenomenon

Failing to load the dynamic library on Windows platform.

java.lang.UnsatisfiedLinkError: Unable to load library '***': Native library (win32-x86-64/jndadll.dll) not found in resource path

Possible cause:

The dynamic library is not loaded or the loading fails.

Solutions

- Method 1:

Step 1 Rename the folder where the dynamic library is located in a format of "operating system-operating system bit number", such as win32-x86, win32-x86-64, linux-x86, and linux-x86-64.

Step 2 Modify the loading path in NetSDKLib.java to: NetSDKLib NetSDK_INSTANCE = Native.load("dhnetsdk", NetSDKLib.class); NetSDKLib CONFIG_INSTANCE = Native.load("dhconfigsdk", NetSDKLib.class);

The principle of this method is to read the dynamic library from the project, write it to the path corresponding to java.io.tmpdir (usually /tmp under Linux), and load the library therefrom.

You cannot name the folder for the dynamic library the same as others. Instead, name the folder under Windows64 as win64 and that under Linux64 as linux64.

- Method 2: Load the dynamic library using LibraryLoad.java. The principle thereof is the same as that of the method 1.

2.2 How does the Linux operating system load dynamic library?

Phenomenon

When using NetSDK on the Linux platform, the startup project reports an error that the dynamic library cannot be loaded.

Possible cause

Dynamic library loading fails.

Solution

Load the environment variable LD_LIBRARY_PATH in the format of the following script:

Export LD_LIBRARY_PATH=/path/to/so

java –jar myproject.jar

Before starting the project, add the path where the dynamic library is located to the environment variable LD_LIBRARY_PATH.

2.3 Error message “Invalid calling convention 63” reported under Linux

Phenomenon

The complete error report is as follows:

```
java.lang.IllegalArgumentException: Invalid calling convention 63
	at com.sun.jna.Native.createNativeCallback(Native Method)
	at com.sun.jna.CallbackReference.<init>(CallbackReference.java:263)
	at com.sun.jna.CallbackReference.getFunctionPointer(CallbackReference.java:449)
	at com.sun.jna.CallbackReference.getFunctionPointer(CallbackReference.java:426)
	at com.sun.jna.Function.convertArgument(Function.java:558)
	at com.sun.jna.Function.invoke(Function.java:345)
	at com.sun.jna.Library$Handler.invoke(Library.java:265)
	at com.sun.proxy.$Proxy10.CLIENT_Init(Unknown Source)
	at com.netsdk.lib.LoginModule.init(LoginModule.java:34)
	at com.yfqc.facevideo.common.service.CameraInfoService.<init>(CameraInfoService.java:59)
      at com.yfqc.facevideo.common.task.CheckCameraInfoTask.doUpdate(CheckCameraInfoTask.java:46)
	at com.yfqc.facevideo.common.task.CheckCameraInfoTask.run(CheckCameraInfoTask.java:30)
	at it.sauronsoftware.cron4j.RunnableTask.execute(Unknown Source)
	at it.sauronsoftware.cron4j.TaskExecutor$Runner.run(Unknown Source)
at java.lang.Thread.run(Thread.java:745)
```

Possible cause

The Linux platform uses the NetSDKLib.java of Windows version.

Solution

Use the NetSDKLib.java file on the Linux platform.

2.4 Undefined symbol: SP_LoadLibrary reported when loading SO file under Linux

Phenomenon

When running the Linux program, an error is reported: undefined symbol: SP_LoadLibrary.

Possible cause

The dynamic library lacks dependencies on related functions or macros.

Solution

Step 1 Enter ldd -r *.so on the Linux terminal in the path where the dynamic library is located.
Step 2 View the dependent functions or macros.
Step 3 Replace the original dynamic library with the new one.

3 Other Questions

3.1 Program crash after enabling callback function

Phenomenon

Because JAVA developers rarely contact callback function in NetSD, a program crash occurred when using the function.

Possible cause

The callback was garbage collected and the network was disconnected.

Solution

Definition of the callback:

A callback is a reference to a block of executable code that is passed to other codes through function parameters. This design allows the underlying code to call subroutines defined at a higher layer. See details on NetSDK callback usage.

You can understand the callback function as websocket long connection. For example, when the method in class A calls that in class B, a long connection between servers A and B is connected. Server A monitors the message sent by server B in real time and processes it. That is, the server B triggers the callback function of server A. The callback function is extensively used in Alipay payment. When a bank sends a transfer message, and then another message, Alipay will call the callback function of the bank.

Step 1 Assign the Pointer.

Step 2 Store it temporarily using a custom array.

Step 3 Obtain data from the array and then convert the Pointer data into the structure.

📖

- You cannot call the SDK interface through the callback function. Instead, you should execute another thread.
- You should write the callback function in a singleton or global form because it is a weak reference and features asynchronous. Otherwise, it might be recycled by the JVM.
- The callback is asynchronous and the underlying C++ is basically locked. To prevent deadlock, other NetSDK methods should not be called in the callback.

3.2 Error Code Encapsulation

Phenomenon description

When calling the C layer interface, the returned error code is obscure.

Solution

Step 1 Find the error code encapsulation location com.netsdk.lib.enumeration.ENUMERROR.

Step 2 Print the complete description information corresponding to the error code using getErrorMessage.

eg:

```
System.out.println("query system info failed.error is" +
ENUMERROR.getErrorMessage());
```

3.3 Unmatched library versions of the dynamic library and the encapsulation layer causes garbled characters and crashes

Phenomenon

There occur to incorrect field values, partly garbled characters, completely garbled characters and and crashed JNA layer when parsing data.

Possible cause

The structure encapsulated by JAVA does not match that in the dynamic library because the user uses the dynamic library of another version in the project.

Solution

Encapsulate JAVA classes by libraries and header files (JAVA-encapsulated classes, dll or so should be consistent with the header files).

3.4 Missing dynamic library dependencies in Linux NetSDK Jar package dependencies

Phenomenon

An error occurs to downloading the transcoding video file on the Linux platform and the interface call fails.

Possible cause

Missing transcoding library dependency.

Solution

Add the transcoding library:

Start with the script. See details in NetSDK run.sh file.

📖

The error is more common on Linux platform.

3.5 How to set the encoding format for different operating system platforms?

Phenomenon

The data parsed is garbled.

Possible cause

The encoding format is set incorrectly.

Solution

Step 1 Set the encoding format for Windows platform to GBK.

Step 2 Set the encoding format for Linux platform to UTF-8.

3.6 When getting dwUser data for callback, an error of garbled characters or Invalid Memory Access is reported

Phenomenon

When getting dwUser data for callback, an error of garbled characters or Invalid Memory Access error is reported

Possible cause

dwUser is garbage collected.

Solution

Customize data and transmit them as null.

🕮

When using dwUser, declare it as a global variable. Otherwise, dwUser will be recycled by the jvm.

3.7 What is a handle?

JAVA developers rarely contact or use handles. The handle is a unique integer. It serves as the id to distinguish different objects and different instances of the same type. The program can access partial information about an object through the handle.

You can consider anything in JAVA as an object and the manipulated identifier as actually a "handle" thereto.

3.8 String parameter transmission and extraction causes crash

Phenomenon

Device crashes when sending string data.

Possible causes

Incorrect encoding format.

Solution

- Specify the string encoding format of the Linux platform as UTF-8 when passing and extracting parameters.
- Specify the encoding format of the Windows platform as GBK when passing and extracting parameters.

3.9 How to open the log and specify the path?

Phenomenon

When you analyze SDK problem using log, you are not aware of the existence of a log module.

Solution

You need to enable the SDK log module function in the relevant code module, with the path available to be specified. The relevant sample code is listed as follows:

```
/**
* Open sdk log
  */
public static void enableLog() {
        NetSDKLib.LOG_SET_PRINT_INFO setLog = new
NetSDKLib.LOG_SET_PRINT_INFO();
        File path = new File("sdklog/");
        if (!path.exists()) path.mkdir();

//The log storage address here can be adjusted according to the actual situation
        String logPath = path.getAbsoluteFile().getParent() + "\\sdklog\\" + "sdklog" +
AnalyseTaskUtils.getDate() + ".log";
        setLog.nPrintStrategy = 0;
setLog.bSetFilePath = 1;
        System.arraycopy(logPath.getBytes(), 0, setLog.szLogFilePath, 0,
logPath.getBytes().length);
        System.out.println(logPath);
        setLog.bSetPrintStrategy = 1;
        bLogOpen = netsdk.CLIENT_LogOpen(setLog);
        if (!bLogOpen) System.err.println("Failed to open NetSDK log");
}
```

3.10 When importing a project into idea, the NetSDKLib class reports an error

Phenomenon

After importing the project into idea for the first time, the NetSDKLib class is not detected to be imported thereto, causing the failure to program running.

Solution

Step 1 Open Help->Edit Custom Properties….
Step 2 Copy the code idea.max.intellisense.filesize=4096 and paste it into the file.
Step 3 Close all idea compilers, and reopen them.

3.11 Failing to receive subscription events continuously

Phenomenon description

Subscription events can be received initially, but become invalid after a period of time.

Solution

Enable the Singleton Pattern for the callback event. Otherwise, it will trigger the callback failure because the program running is recycled by the system. We take the following smart subscription event as the relevant example code:

```
/** * Subscribe to smart tasks */ public void AttachEventRealLoadPic() { // Unsubscribe
first, the device will not check for duplicate subscriptions, and duplicate events will be
returned after repeated subscriptions this.DetachEventRealLoadPic(); // Picture required
int bNeedPicture = 1; //EVENT_IVS_ALL means subscribing to all smart events
m_attachHandle = netsdk.CLIENT_RealLoadPictureEx(loginHandle, channelId,
EVENT_IVS_ALL, bNeedPicture, AnalyzerDataCB.getInstance(), null, null); /** * //
EVENT_IVS_WORKCLOTHES_DETECT Helmet detection event * //
EVENT_IVS_SMOKING_DETECT Smoking detection event **/ if
(m_attachHandle.longValue() != 0) {           System.out.printf("Chn[%d]
CLIENT_RealLoadPictureEx Success\n", channelId);
     } else {
          System.out.printf("Ch[%d] CLIENT_RealLoadPictureEx Failed!LastError
= %s\n", channelId,
                    ToolKits.getErrorCode());
```

```
    }
}

/**
 * Alarm event (intelligent) callback
 */
private static class AnalyzerDataCB implements NetSDKLib.fAnalyzerDataCallBack {
    private final File picturePath;
    private static AnalyzerDataCB instance;

    private AnalyzerDataCB() {
        picturePath = new File("./AnalyzerPicture/");
        if (!picturePath.exists()) {
            picturePath.mkdirs();
        }
    }

    public static AnalyzerDataCB getInstance() {
        if (instance == null) {
            synchronized (AnalyzerDataCB.class) {
                if (instance == null) {
                    instance = new AnalyzerDataCB();
                }
            }
        }
        return instance;
    }

    @Override
    public int invoke(NetSDKLib.LLong lAnalyzerHandle, int dwAlarmType, Pointer
pAlarmInfo, Pointer pBuffer, int dwBufSize,
                      Pointer dwUser, int nSequence, Pointer reserved) {
        if (lAnalyzerHandle == null || lAnalyzerHandle.longValue() == 0) {
            return -1;
        }

        switch (dwAlarmType) {
            case EVENT_IVS_WORKCLOTHES_DETECT: // Safety helmet detection
event
            {
                NetSDKLib.DEV_EVENT_WORKCLOTHES_DETECT_INFO msg =
new NetSDKLib.DEV_EVENT_WORKCLOTHES_DETECT_INFO();
                ToolKits.GetPointerData(pAlarmInfo, msg);
                if (msg.stuSceneImage != null && msg.stuSceneImage.nLength > 0) {
                    String bigPicture = picturePath + "\\" + System.currentTimeMillis()
+ ".jpg";
                    ToolKits.savePicture(pBuffer, msg.stuSceneImage.nOffSet,
msg.stuSceneImage.nLength, bigPicture);
                              if (msg.stuHumanImage != null &&
msg.stuHumanImage.nLength > 0) {
                                    String smallPicture = picturePath + "\\" +
System.currentTimeMillis() + "small.jpg"; ToolKits.savePicture(pBuffer,
msg.stuHumanImage.nOffSet, msg.stuHumanImage.nLength, smallPicture); } }
System.out.println("Safety helmet detection event (UTC):" + msg.UTC + "Channel
number:" + msg.nChannelID); break; } case
NetSDKLib.EVENT_IVS_SMOKING_DETECT : { // Smoking detection event
(corresponding to DEV_EVENT_SMOKING_DETECT_INFO) System.out.printf("Smoking
detection event"); DEV_EVENT_SMOKING_DETECT_INFO msg = new
DEV_EVENT_SMOKING_DETECT_INFO(); ToolKits.GetPointerData(pAlarmInfo, msg);
String Picture = picturePath + "\\" + "smoking_" + System.currentTimeMillis() + ".jpg"; if
(dwBufSize > 0) { ToolKits.savePicture(pBuffer, msg.stuImageInfo[0].nOffset,
msg.stuImageInfo[0].nLength, Picture); } System.out.println("Smoking detection event
time (UTC):" + msg.UTC + " Start time:" + msg.stuObject.stuStartTime + " End time:" +
msg.stuObject.stuEndTime); break; } case
EVENT_IVS_GRANARY_TRANS_ACTION_DETECTION:{ //Report of grain and surface
movement detection event (corresponding to
NET_DEV_EVENT_GRANARY_TRANS_ACTION_DETECTION_INFO)
System.out.println("Report of grain and surface movement detection event");
NET_DEV_EVENT_GRANARY_TRANS_ACTION_DETECTION_INFO msg = new
NET_DEV_EVENT_GRANARY_TRANS_ACTION_DETECTION_INFO();
ToolKits.GetPointerData(pAlarmInfo,msg); System.out.println("Grain and surface
movement detection event reporting time (stuUTC):" + msg.stuUTC);
System.out.println("Actual number of object list messages (nObjectsCount):" +
msg.nObjectsCount); break; } case
EVENT_IVS_REGION_PROPORTION_DETECTION:{ //Region proportion detection
event (corresponding to
NET_DEV_EVENT_REGION_PROPORTION_DETECTION_INFO)
System.out.println("Region proportion detection event");
NET_DEV_EVENT_REGION_PROPORTION_DETECTION_INFO msg = new
NET_DEV_EVENT_REGION_PROPORTION_DETECTION_INFO();
ToolKits.GetPointerData(pAlarmInfo, msg); System.out.println("Region proportion
detection event reporting Time (stuUTC): " + msg.stuUTC); System.out.println ("The
actual number of object list messages (nObjectsCount): " + msg.nObjectsCount); break; }
default: System.out.println ("Other events--------------------" + dwAlarmType); break; }
return 0; } }
```

Appendix 1 Security Recommendation

1. Account Management

1.1 Use complex passwords

Please refer to the following suggestions to set passwords:

- The length should not be less than 8 characters;
- Include at least two types of characters: upper and lower case letters, numbers and symbols;
- Do not contain the account name or the account name in reverse order;
- Do not use continuous characters, such as 123, abc, etc.;
- Do not use repeating characters, such as 111, aaa, etc.

1.2 Change passwords periodically

It is recommended to periodically change the device password to reduce the risk of being guessed or cracked.

1.3 Allocate accounts and permissions appropriately

Appropriately add users based on service and management requirements and assign minimum permission sets to users.

1.4 Enable account lockout function

The account lockout function is enabled by default. You are advised to keep it enabled to protect account security. After multiple failed password attempts, the corresponding account and source IP address will be locked.

1.5 Set and update password reset information in a timely manner

Our device supports password reset function. To reduce the risk of this function being used by threat actors, if there is any change in the information, please modify it in time. When setting security questions, it is recommended not to use easily guessed answers.

2. Service Configuration

2.1. Enable HTTPS

It is recommended that you enable HTTPS to access Web services through secure channels.

2.2 Encrypted transmission of audio and video

If your audio and video data contents are very important or sensitive, we recommend you to use encrypted transmission function in order to reduce the risk of your audio and video data being eavesdropped during transmission.

2.3 Turn off non-essential services and use safe mode

If not needed, it is recommended to turn off some services such as SSH, SNMP, SMTP, UPnP, AP hotspot etc., to reduce the attack surfaces.

If necessary, it is highly recommended to choose safe modes, including but not limited to the following services:

- SNMP: Choose SNMP v3, and set up strong encryption and authentication passwords.
- SMTP: Choose TLS to access mailbox server.

- FTP: Choose SFTP, and set up complex passwords.
- AP hotspot: Choose WPA2-PSK encryption mode, and set up complex passwords.

2.4 Change HTTP and other default service ports

It is recommended that you change the default port of HTTP and other services to any port between 1024 and 65535 to reduce the risk of being guessed by threat actors.

3. Network Configuration

3.1 Enable Allow list

It is recommended that you turn on the allow list function, and only allow IP in the allow list to access the device. Therefore, please be sure to add your computer IP address and supporting device IP address to the allow list.

3.2 MAC address binding

It is recommended that you bind the IP address of the gateway to the MAC address on the device to reduce the risk of ARP spoofing.

3.3. Build a secure network environment

In order to better ensure the security of devices and reduce potential cyber risks, the following are recommended:

- Disable the port mapping function of the router to avoid direct access to the intranet devices from external network.
- According to the actual network needs, partition the network: if there is no communication demand between the two subnets, it is recommended to use VLAN, gateway and other methods to partition the network to achieve network isolation.
- stablish 802.1x access authentication system to reduce the risk of illegal terminal access to the private network.

4. Security Auditing

4.1 Check online users

It is recommended to check online users regularly to identify illegal users.

4.2 Check device log

By viewing logs, you can learn about the IP addresses that attempt to log in to the device and key operations of the logged users.

4.3 Configure network log

Due to the limited storage capacity of devices, the stored log is limited. If you need to save the log for a long time, it is recommended to enable the network log function to ensure that the critical logs are synchronized to the network log server for tracing.

5. Software Security

5.1 Update firmware in time

According to the industry standard operating specifications, the firmware of devices needs to be updated to the latest version in time in order to ensure that the device has the latest functions and security. If the device is connected to the public network, it is recommended to enable the online upgrade automatic detection function, so as to obtain the firmware update information released by the manufacturer in a timely manner.

5.2 Update client software in time

It is recommended to download and use the latest client software.

6. Physical Protection

It is recommended that you carry out physical protection for devices (especially storage devices), such as placing the device in a dedicated machine room and cabinet, and having access control and key management in place to prevent unauthorized personnel from damaging hardware and other peripheral equipment (e.g. USB flash disk, serial port).