

# Intellex SDK User's Manual Addendum

## Version 2.4

This document discusses the digital signature used to sign specific Intellex API Active X controls and DLLs. It also discusses how to use certain Intellex API interfaces in developing applications that can be used with the Microsoft Internet Explorer browser.

This document discusses Intellex API file redistribution by describing the target system installation and possible system dependencies, including an estimated list of API DLL dependencies. This document also contains a section providing further details regarding the use of specific API functions.

## Browser Safe Controls and Restrictions

This section discusses digital signatures and Microsoft Internet Explorer browser restrictions and usage guidelines for Intellex API objects.

### *Signing for Intellex API Active X Controls and DLLs*

The following DLLs in Intellex API version 2.4, distributed via the World Wide Web, are digitally signed:

```
IntellexAPI.dll  
NtlxPlayerUI.dll  
NtlxVideo.dll
```

This digital signature indicates that these objects are safe to download and install on your system from the web site. The three Active X controls: `NTLXRemoteAx`, `NtlxVideoCtrl`, and `NtlxPlayerUI` are also marked as safe to initialize and safe for scripting so that they can be used in the Internet Explorer browser.

### *INTLXRemoteAx Web Browser Restrictions*

The following restrictions apply when using `INTLXRemoteAx` with a web browser:

1. Video files must have the extension `.TVSV` (Tyco Video Systems Video).
2. AlarmList files must have the extension `.TVSA` (Tyco Video Systems Alarm).
3. You will be **prevented from reading or writing** Video and AlarmList files on the root directory of the drive.
4. Files **will not** be overwritten. You must either specify a new file name each time you save, or you must remove the old file in order to save information to a file with the same name.
5. Actions that require the same resources must run serially. Before you can request an action that uses the same resources as the current action, you must wait until an event signals that the current action is completed. These actions include:
  - `GetVideo`, `GetAlarmVideo`, `GetMediaStreams`, `GetAlarmMediaStreams`, `GetUnitInfo`.
  - `GetAlarmList`
  - `GetStatus`
  - `TimeSync`

For more information on digital signatures and internet security, see the *Microsoft MSDN Library*, or *Microsoft MSDN Online*, at <http://msdn.microsoft.com>.

## **Using the Intellex API with the Internet Explorer Browser**

You can debug your web application with the Internet Explorer browser more effectively by changing the following items in the **Tools/Internet Options/Advanced** dialog:

- Uncheck the Disable Script Debugging box.
- Check the Display a notification about every script error box.

For specifying file names, the `NtlxRemoteAx.htm` sample web page contains edit boxes that are located to the left of a **Browse** button. The button brings up a dialog similar to an **Open/Find** dialog for AlarmList files or Video files. The browser has no way to bring up a **Save** dialog. In those samples where a **Save** is desired, some, or perhaps all, of the file's name and path must be typed by hand into the edit box to upload the file.

**Note:** The Internet Explorer browser does not normally save files, so there is no file "**Save**" (download) style dialog from within the browser. The `NtlxRemoteAx` ActiveX control will save the video and alarm lists to files if the file names follow the restrictions listed above.

## **Using NtlxVideo.dll in a Web Page**

If you are using the `NtlxVideo.dll` object in a web page without bundling it in a CAB file, then the server's copy of this file must be renamed to `NtlxVideo.ocx` in order for this control to be used in the web page.

## **Using the API to Time Sync Multiple Units**

There are two intuitive, but incorrect approaches for using the Intellex API **Time Sync** function to synchronize multiple Intellex units. These approaches and an effective Workaround are discussed in this section.

### **Approach 1**

When using the IntellexAPI to **Time Sync** multiple Intellex units, it is intuitive to follow the approach outlined here:

1. For each unit to be synchronized:  
Connect a session for each unit.
2. For each unit to be synchronized:  
Issue a **TimeSync** request for each unit.
3. For each unit to be synchronized:  
Disconnect from each unit.

This approach is problematic because of a low level object used to manage sockets and various types of messages across those sockets. In early versions of the Intellex API (API version 1.0), the only application that used this approach applied it to only one Intellex unit. The application was not written to handle overlapping requests. This design limitation was discovered through more complex use of the Intellex, and it will be corrected in future Intellex API versions (post-API version 3.0).

## **Approach 2**

Another intuitive approach would be to perform the following steps for each unit to be synchronized:

1. Connect a session for this unit.
2. Issue a **TimeSync** request for this unit.
3. Disconnect from this unit.

However, this approach is also problematic because it does not allow time for the request to get to the Intellex and back again. Depending on what the Intellex unit is doing at the time the **TimeSync** request is made, the unit may reject the request, or it may take a few seconds to stop what it is doing long enough to perform the requested **TimeSync**. The unit will respond with a message that causes an event to be fired if it does the **TimeSync**, or if it totally rejects the request. If the unit proceeds to perform a **TimeSync**, but it is slow about it, there is no event to inform you of the delay: The unit merely sends an event when the process is completed.

## **Workaround (using IntellexAPI 2.4.26):**

To perform a **TimeSync** on multiple Intellex units using Intellex API 2.4.26, follow this sequence of steps:

1. For the current Intellex unit:
  - a. Connect to this Intellex unit
  - b. Record the time the request is being made
  - c. Issue a **TimeSync** request
2. When **OnTimeSyncEvent** is called:
  - a. Record the time the event came in
  - b. Handle the event (error, complete, progress)
  - c. Post a message to self to get the next step in proper context
3. When the posted message comes in
  - a. Disconnect from the current Intellex unit
  - b. Bump the index into the Intellex Unit List
  - c. If there are more units to be synchronized, then repeat from step 1

The time interval between when the request is made and when the event comes in tells you the total turn-around time. You can then estimate the remaining time discrepancy. Generally, the time for the first unit to synchronize is longer because DLLs must be loaded into memory. In a LAN setting, the turn-around time for the second and successive units is less than one second. Typically, the time span is 200 ms. If your units are on a WAN, the time interval will be larger.

If all of your units are on the same network topology (LAN or WAN, same number of hops, etc.), then half the average turn-around time is the probable discrepancy remaining.

Even on a LAN, if the Intellex unit is in a critical process, such as DataBase Conversion, it may take up to 6 seconds for it to respond. That does not count for possible WAN lag-time that may also be added to the total time. The Intellex unit may also respond to that situation by putting off the request. In that case, reschedule time synchronization for later. This would be indicated by an **eTimeSyncReason** of **eTimeSyncError**.

Plan your application's timeouts accordingly.

The sample application **APIMultiGetStatus** employs a similar process for the **GetStatus** command. The sample allows you see what happens when you use the first intuitive approach as opposed to using the event driven Workaround described above.

### ***Workaround (using IntellexAPI 2.4.27):***

The IntellexAPI 2.4.27 version, enhanced for browser safety, contains the **INTLXRemoteAx** interface, which wraps **INTLXRemote** and implements this workaround. Using this interface, the workaround is a little bit simpler and closer to the intuitive approach:

1. For each unit to be synchronized, connect a session for each unit
2. For the current Intellex unit:
  - a. Record the time the request is being made
  - b. Issue a **TimeSync** request for this unit
3. When **OnTimeSyncEvent** is called,
  - a. Record the time the event came in
  - b. Handle the event (error, complete, progress)
  - c. Post message to self to get next step in proper context
4. When the posted message comes in
  - a. Bump the index into the Intellex Unit List
  - b. If there are more units, cause step 1 to happen again
5. For each unit in the list, disconnect from each unit.

In this workaround, all of the session connections are done up front, and the disconnections are all done after the fact. The **NTLXRemoteAx** object manages the internal object cleanup and re-initialization. You should still wait for the event to come in so that the socket doesn't get closed before the Intellex unit receives the message.

## **Intellex API File Redistribution Information**

This section describes the Intellex API installation in terms of its directory structure and DLL file locations and information within that structure. Some of the DLL files are also installed in system directories. An estimated list of the Intellex DLLs with their known dependencies is also provided.

### ***The Intellex API Directory Structure and File Information***

The Intellex API installation program installs files in the following three locations on your system:

```
C:\Program Files\Sensormatic
C:\Windows\System
C:\WinNT\System32
```

## Intellex SDK User's Manual Addendum

C:\Program Files\Sensormatic contains the subdirectories described in the table below. Most, but not all, of the DLLs you may need for distribution are located in these subdirectories.

| Subdirectory  | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| COMObjects\   | The DLL files in this directory hold all the Intellex API COM objects and their resources. At least some of these will be required for your distribution list.<br><b>Run:</b> <code>regsvr32 /s %file%</code> on each file.<br><b>Note:</b> When you run <b>regsvr32</b> , there may be some failures; however, the /s flag prevents innocuous failed messages. You may also check to see if the version resource contains the <code>OLESelfRegister</code> flag before running <b>regsvr32</b> . In general, all but the <code>Res*.dll</code> language DLL files will contain the flag. |
| Controls\     | Files in this subdirectory are probably not needed by your redistribution installation. They are used on the server and in Network Client, but not with the API or API samples.                                                                                                                                                                                                                                                                                                                                                                                                           |
| Intellex API\ | Files and subdirectories in this <code>Sensormatic\</code> subdirectory contain sample files and sample source files. These files are <b>not</b> needed in your redistribution installation.                                                                                                                                                                                                                                                                                                                                                                                              |

The Intellex API installation program also installs a set of DLLs in the `\Windows\System` or `\WinNt\System32` directory, depending on the target operating system. The following set of DLLs is installed:

Alarm.dll  
AudioLib.dll  
ResUtility\*.dll  
ToolUtil.dll  
Util.dll  
VideoLib.dll

To check for possible dependencies on other system DLLs, use the Dependency Walker freeware utility (`depends.exe`). For more information on dependency checking, see the URLs listed below for the Dependency Walker utility and the MSDN library.

- Dependency Walker utility: <http://www.dependencywalker.com>
- MSDN information on Microsoft Visual C++ file redistribution:  
<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnvc60/html/redistribvc6.asp>

## Sample Application COM Dependencies

This section contains a series of tables that list the Intellex API sample applications with their dependent Intellex API COM objects. There are tables for each sample application category, including [C++ Samples](#), [Visual Basic Samples](#), and [Web Application Samples](#).

**Note:** An Intellex API sample application may have no COM object dependencies; yet still have some DLL dependencies. Those DLL dependencies are not defined in this list.

### C++ Samples

| Sample Application | COM Dependencies                         |
|--------------------|------------------------------------------|
| APIBasic           | IntellexAPI.dll                          |
| APIFullTest        | IntellexAPI.dll                          |
| APIMultiGetStatus  | IntellexAPI.dll                          |
| APIPlayStoredVideo | IntellexAPI.dll<br>NtlxPlayerUI.dll      |
| IntellexAPITest    | IntellexAPI.dll                          |
| LVBasic            | NtlxVideo.dll                            |
| LVCamCtrl          | NtlxVideo.dll                            |
| LVMulti            | NtlxVideo.dll                            |
| LVPanes            | NtlxVideo.dll                            |
| NtlxAVIExport      | NtlxAVIExportCOM.dll<br>NtlxPlayerUI.dll |
| NtlxPlayerApp      | IntellexAPI.dll<br>NtlxPlayerUI.dll      |
| NtlxVideoOverlay   | NtlxPlayerUI.dll<br>VideoOverlayCOM.dll  |

### Visual Basic Samples

| Sample Application   | COM Dependencies                    |
|----------------------|-------------------------------------|
| GetUnitStatus        | IntellexAPI.dll                     |
| IntellexRemoteDemoVB | IntellexAPI.dll                     |
| LVBASICVB            | NtlxVideo.dll                       |
| VBCameraControl      | NtlxVideo.dll                       |
| LVMultiVB            | NtlxVideo.dll                       |
| LVPanesVB            | NtlxVideo.dll                       |
| vbNtlxPlayer         | IntellexAPI.dll<br>NtlxPlayerUI.dll |

### Web Application Samples

| Sample Application           | COM Dependencies                    |
|------------------------------|-------------------------------------|
| NtlxPlayerUI.htm             | NtlxPlayerUI.dll                    |
| NtlxPlayerUI_PlayBack.htm    | NtlxPlayerUI.dll<br>IntellexAPI.dll |
| NtlxRemoteAx.htm             | IntellexAPI.dll                     |
| NtlxRemoteAx_MultiStatus.htm | IntellexAPI.dll                     |
| NtlxVideoCtrl.htm            | NtlxVideo.dll                       |
| NtlxVideoCtrl_CamCtrl.htm    | NtlxVideo.dll                       |

## COM Dependencies for Intellex API Top Level DLLs

**Note:** This list has not been verified. You may want to test all execution paths of your application when only these DLLs are installed and registered to make sure this set of DLLs is sufficient. By unregistering one DLL at a time, and then re-testing, you can determine which DLLs are not required.

The top level Intellex API DLLs are listed in the table below with their dependent DLLs.

| Top Level DLL           | COM Dependencies                                                                                                                                                                        | Notes                                                                                                                                                                                                                                |
|-------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>IntellexAPI.dll</b>  | NCServerCOM.dll<br>NtlxAlarmTransfer.dll<br>NtlxCommon.dll<br>NtlxRemoteUnits.dll<br>NtlxStatusTransferCOM.dll<br>NtlxTimeSync.dll<br>NtlxVideoDatabase.dll<br>NtlxVideoTransferCOM.dll | This DLL is digitally signed and flagged as safe for browser initialization and scripting.                                                                                                                                           |
| <b>NtlxVideo.dll</b>    | -- None --                                                                                                                                                                              | <ul style="list-style-type: none"> <li>• This DLL is designed as a stand-alone DLL.</li> <li>• It requires no other DLLs.</li> <li>• It is digitally signed and flagged as safe for browser initialization and scripting.</li> </ul> |
| <b>NtlxPlayerUI.dll</b> | NtlxCommon.dll<br>NtlxImageTools.dll<br>NtlxPlayer.dll<br>NtlxVideoDatabase.dll                                                                                                         | <ul style="list-style-type: none"> <li>• This DLL is digitally signed and flagged as safe for browser initialization and scripting.</li> <li>• See Note below.</li> </ul>                                                            |

**Note:** Even if you are not downloading a TEXT stream, the \*Text\*.dll files are still required. NtlxPlayerUI knows that it is possible for a media stream to contain video, audio and/or text. It uses the text-related COM objects, even if only to determine whether or not the stream contains text.

The same situation exists with NtlxPlayerUI.dll. NtlxPlayerUI.dll requires NtlxPlayer.dll, which in turn requires TextMgrCOM.dll.

## Supporting DLLs and their COM Dependencies

The following table lists Intellex API supporting DLLs and their COM dependencies.

**Note:** A supporting DLL may have no COM object dependencies; yet still have some DLL dependencies. Those DLL dependencies are not defined in this list.

| Supporting DLL            | COM Dependencies                                                                                                                                                               | Notes                                                                                                                                                                                                                                                                                                                                                   |
|---------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ImageProCom.dll           | -- None --                                                                                                                                                                     |                                                                                                                                                                                                                                                                                                                                                         |
| NCServerCOM.dll           | NtlxSocketLayer.dll                                                                                                                                                            |                                                                                                                                                                                                                                                                                                                                                         |
| NtlxAlarmTransfer.dll     | NtlxCommon.dll<br>NtlxSocketLayer.dll<br>NtlxVideoDatabase.dll<br>NtlxVideoTransferCOM.dll                                                                                     |                                                                                                                                                                                                                                                                                                                                                         |
| NtlxAVIExportCOM.dll      | NtlxCommon.dll<br>NtlxPlayer.dll<br>NtlxVideoDatabase.dll                                                                                                                      |                                                                                                                                                                                                                                                                                                                                                         |
| NtlxCommon.dll            | -- None --                                                                                                                                                                     |                                                                                                                                                                                                                                                                                                                                                         |
| NtlxImageTools.dll        | NtlxImageProCOM.dll<br>NtlxCommon.dll<br>NtlxCommonDlg.dll                                                                                                                     |                                                                                                                                                                                                                                                                                                                                                         |
| NtlxPlayer.dll            | NtlxCommon.dll<br>NtlxV1StreamFilter.dll (1)<br>NtlxV2StreamFilter.dll (2)<br>NtlxV3StreamFilter.dll (3)<br>NtlxVideoDatabase.dll<br>TextMgrCom.dll<br>VideoOverlayCOM.dll (4) | (1) NtlxV1StreamFilter.dll is for older images.<br>(2) NtlxV2StreamFilter.dll is for current images.<br>(3) NtlxV3StreamFilter.dll is for Integra images.<br>(4) The TextRegion object passed into the VideoOverlay object came from NtlxPlayerUI, and in turn from the application. It is NOT AddRef/Release properly by NtlxPlayer. Use with caution. |
| NtlxRemoteUnits.dll       | NtlxCommon.dll<br>NtlxSocketLayer.dll                                                                                                                                          |                                                                                                                                                                                                                                                                                                                                                         |
| NtlxSocketLayer.dll       | NtlxCommon.dll<br>SocketConnect.dll                                                                                                                                            |                                                                                                                                                                                                                                                                                                                                                         |
| NtlxStatusTransferCOM.dll | NtlxCommon.dll<br>NtlxSocketLayer.dll                                                                                                                                          |                                                                                                                                                                                                                                                                                                                                                         |
| NtlxTextStreamCOM.dll     | TextMgrCOM.dll                                                                                                                                                                 |                                                                                                                                                                                                                                                                                                                                                         |
| NtlxTimeSync.dll          | NtlxCommon.dll<br>NtlxRemoteUnits.dll<br>NtlxSocketLayer.dll                                                                                                                   |                                                                                                                                                                                                                                                                                                                                                         |

Intellex SDK User's Manual Addendum

| Supporting DLL                                                             | COM Dependencies                                                                                                           | Notes |
|----------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------|-------|
| NtlxV1StreamFilter.dll<br>NtlxV2StreamFilter.dll<br>NtlxV3StreamFilter.dll | NtlxCommon.dll                                                                                                             |       |
| NtlxVideoDatabase.dll                                                      | NtlxCommon.dll<br>NtlxPlayer.dll<br>NtlxV1StreamFilter.dll<br>NtlxV2StreamFilter.dll<br>NtlxV3StreamFilter.dll<br>VVDB.dll |       |
| NtlxVideoTransferCOM.dll                                                   | NtlxCommon.dll<br>NtlxRemoteUnits.dll<br>NtlxSocketLayer.dll<br>NtlxVideoDatabase.dll                                      |       |
| NtlxWatchDog.dll                                                           | CompressorManager.dll                                                                                                      |       |
| PortMgrCOM.dll                                                             | -- None --                                                                                                                 |       |
| TextDisplayCOM.dll                                                         | NtlxCommon.dll                                                                                                             |       |
| TextHandlerMapping.dll                                                     | -- None --                                                                                                                 |       |
| TextMgrCOM.dll                                                             | NtlxCommon.dll<br>NtlxTextStreamCOM.dll<br>NtlxVideoDatabase.dll<br>TextDisplayCOM.dll                                     |       |
| VideoOverlayCOM.dll                                                        | -- None --                                                                                                                 |       |
| VVDB.dll                                                                   | NtlxCommon.dll<br>NtlxVideoDatabase.dll                                                                                    |       |

## Further Details Regarding API Use

This section provides details that are not found in the current version of the Intellex Software Development Kit User's Manual.

### ***How to Avoid the COM Singleton Problem***

The Intellex API uses COM singletons to manage its system resources. Your MFC application uses COM singletons to manage your system resources. It is the nature of MFC's use of COM singletons that such objects must be shut down in an orderly way to avoid memory leaks and application crashes. Therefore, make sure that your application calls `Release()` on all instances of Intellex API objects before you start the application shutdown. It is especially important to ensure that your application's COM cleanup code is not called from within a class destructor, but in a routine called prior to deleting and destructing the class. Ignoring these cleanup procedures could result in a crash prior to expected shutdown.

### ***Avoiding Dependency Problems***

Be sure to check target machines for screen savers and other applications that may pose difficulty for Intellex API objects and their dependent DLLs.

### ***Pre-Alarm Time***

There are two methods for which pre-alarm time is specified: `INTLXRemote::GetAlarmVideo` and `INTLXRemote2::GetAlarmMediaStreams`. For both of these methods, the `sIncludePreAlarm` parameter is a YES or NO flag, according to the following table:

| Value of <code>sIncludePreAlarm</code> | Description                                                                                                        |
|----------------------------------------|--------------------------------------------------------------------------------------------------------------------|
| 0                                      | Do not retrieve pre-alarm video or pre-alarm media for this camera.                                                |
| Any non-zero value                     | Retrieve pre-alarm video or pre-alarm media for this camera according to the value specified by the Intellex Unit. |

Duration of the pre-alarm time is determined by the alarm configuration on the Intellex unit. If no pre-alarm duration has been configured on the unit, the setting of this parameter is ignored.

### ***Downloading Intellex API DLLs***

Stand-alone Intellex API DLLs can be made web downloadable by changing their file extensions. Remember to rename the `.dll` file extensions to `.ocx` file extensions in order to make them web downloadable. See "Using `NtlxVideo.dll` in a Web Page" on page 2 for an example.

**Note:** At this time, the limitation applies to `NtlxVideo.dll` only, which is a stand-alone ActiveX control DLL. Since the other DLLs have various dependencies, you should build a `.CAB` file for web downloading.

## Notes