

Longwatch HMI Integration Guide

Version4.0

V4.0

Feb 15, 2008

Copyright 2007 Longwatch Inc. ALL RIGHTS RESERVED

Table of Contents

1. Introduction.....	4
2. System Overview	4
2.1. Video Control Center	5
2.1.1. Events.....	5
2.1.1.1. Event Types and Categories.....	5
2.1.1.2. Storage	6
2.1.2. Event Video Clips.....	6
2.1.3. Real-time Video Stream	7
2.2. HMI / PLC Integration.....	8
2.2.1. Installation and Setup.....	8
2.2.1.1. OPC Server (HMI SCADA Node) Installation.....	8
2.2.2. Longwatch OPC Data Server.....	9
2.2.2.1. Setup	9
2.2.2.2. OPC ItemNames	10
2.2.3. iFIX Integration	15
2.2.3.1. The Database.....	15
2.2.3.2. IFIX OPC Client I/O Driver	15
2.2.3.3. UI Integration.....	16
2.2.3.4. ActiveX Control Integration (Live)	16
2.2.4. WonderWare Integration	17
2.2.4.1. The Database.....	17
2.2.4.2. Wonderware OPCLink Application.....	17
2.2.4.3. OPCLink Topic Definition	17
2.2.4.4. OPC Link Tag Creator	19
2.2.4.5. ActiveX Control Installation.....	21
2.2.4.6. Inserting ActiveX Controls In Wonderware	22
2.2.4.7. Setting Control Properties.....	22
2.2.4.8. Scripting the Longwatch Video Control	23
2.2.4.9. Scripting the Web Browser Control.....	24
2.2.5. Other HMI Integration Packages	24
2.3. UI Server – Client Application Components	25
2.3.1. Event List View	26
2.3.1.1. Sorting and Filtering Events	26
2.3.1.2. Simple View.....	27
2.3.1.3. Event View.....	28
2.3.1.4. URL Parameters.....	29
2.3.2. Live Streaming with VideoControl OCX	30
2.3.2.1. Properties	31

- 2.3.2.2. Methods..... 33
- 2.3.2.3. PTZ Operations..... 35
- 2.3.3. Playback View 36
- 2.3.4. Accessing Event ID via Scripting 36
- 2.3.5. Diagnostics View 38

1. Introduction

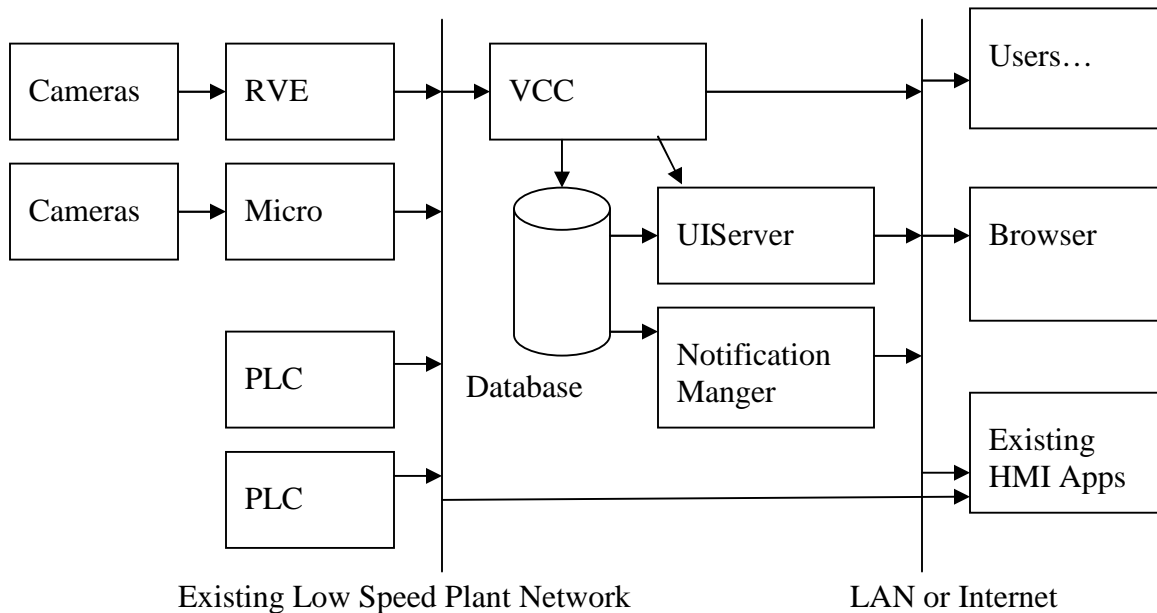
This reference guide will provide a technical description of the web runtime interface for the Longwatch Video System. It assumes the reader is already somewhat familiar with the functionality of the system as a whole and of the RVE (Remote Video Engine), VCC (Video Control Center), UI Server (User Interface Web Server) and Notification Manager (Sends Alarms via Email or Pager) components.

Specifically, it presents a moderately detailed discussion of the exposed runtime web pages that are available from the UI server to be integrated into a browser or HMI application.

This document refers to Remote Video Engine (RVE) as a common term for all three types of video engines, Micro Video Engine (Micro), Remote Video Engine (RVE) and Longwatch Video Engine (LVE). There is no difference in this document between the three types other than the available number of cameras, IO Points, and access point (Doors).

2. System Overview

The system provides video monitoring of remote locations. The system is highly scalable and can support locations with very low bandwidth as well as high bandwidth locations. It can also make use of existing communications infrastructure such as PLC networks. A typical installation is shown below.



The runtime user interface is served by the UI Server to either a browser or an HMI hosting a browser control object.

2.1. Video Control Center

The VCC provides the system with a central repository of unit information. This includes,

- a. Real-time diagnostics information
- b. Security Events detected by the remote (DI or Video Detected)
- c. Maintenance Events detected by the remote
- d. Maintenance Events detected by the VCC (Comm. Error)
- e. Security Event Video Clips
- f. Real time Video Stream

2.1.1. Events

Events from the VCC include:

- Software errors in the VCC system software
- Communications events (communications lost or established to a remote)
- System restart or new configuration loaded

Events from the RVE (remotes) include:

- Software errors in the RVE system software
- System restart (including restart with new configuration)
- Alarms (digital inputs or video sensing)
- Alarm Clips (AVI files capturing an alarm event)

2.1.1.1. Event Types and Categories

All events fall into one of two categories (Maintenance or Security). Security events typically contain one or more event video clips, while maintenance events do not.

Below is a table of event types, category, and typical examples

<u>Event</u>	<u>Source</u>	<u>Event Type</u> (cam=0..11)	<u>Category</u>
DI Hardware Trigger	RVE	DI-(DI# 0..11)	Security

Video Image Trigger	RVE	Video-(cam)	Security
Software Error	VCC/RVE	System	Maintenance
VCC to RVE Communication Lost	VCC	Comm	Maintenance
Restart	VCC/RVE	System	Maintenance
Configuration Download	VCC/RVE	System	Maintenance

2.1.1.2. Storage

All event information including unit, event source, event type, event category, and event timestamp are stored in the event database.

All video files, including VCR archives and event clips, are stored in the following directory.

<install directory>\User Data\CVE_ROOT\ <Unit Name> \Cam(n)

Event video clips files are named starting with an “Exxxxx.avi” where the xxxx is a unique id for this event on this unit.

2.1.2. Event Video Clips

Event video clips can be generated by the Remote unit and sent in near real time to the VCC for review by the operator. The system configuration supports the triggering of these events by a security sensor, direct video analysis, or via remote command. Each RVE can support up to 4 cameras to capture an event video clip per event. The user can specify a BEFORE and AFTER time period that would determine the duration of the event clip.

Events in a remote are immediately sent to the VCC, given a unique Event ID and archived. This process then triggers any configured user notifications. Once the AFTER time has elapsed the video clip associated with that event is archived to the RVE and transmitted up to the VCC. This may take a number of minutes depending on the bandwidth and size of the clip being sent. The VCC will then associate this clip with the previously archived event definition. An event can have up to four clips associated with it (one for each possible camera attached to the RVE). The video clip is stored on the VCC's hard drive for later access.

2.1.3. Real-time Video Stream

The VCC provides an on demand real time video stream to any camera in the network. This stream can be displayed using the Longwatch VideoControl OCX. The VCC will access the single stream from the RVE and distribute this stream to any clients.

The OCX has two live feed modes (Guard Tour and Live).

In Guard Tour mode, the ocx will get the latest cached image of the camera from the VCC. This cached image is minimally updated based on the “slowscan” rate set for the RVE. The “slowscan” setting is found under the Advanced Tab in the VCC Administrator UI for the each RVE.

In Live mode, the ocx will signal the VCC to turn on Live Feed for this camera. In this mode the VCC will update the cache as fast as possible subject to available bandwidth. NOTE: this may effect overall VCC to RVE bandwidth use. Live feed mode is turned off after 30 seconds of no Live feed requests for that camera.

If the image cache is older than the unit’s “stale timeout” value the “data stale” indicator (red border) is displayed in the OCX.

2.2. HMI / PLC Integration

The Longwatch System has been designed to easily integrate with all of the popular HMI packages. HMI integration consists of integrating the monitoring of security events, playing back of event clips, and displaying of live video. This is accomplished by a set of Longwatch Integration Components, which include:

1. Longwatch OPC Server
2. Microsoft Web Browser control accessing dynamic html pages provided by the Longwatch UI Web Server.
3. Live video streaming with the Longwatch VideoControl OCX

The Longwatch OPC server communicates to the VCC to provide realtime alarm status information for each of the Longwatch Remote Video Units for use in the HMI's process database where it can be used for HMI alarm integration and animation.

The Longwatch UI Web Server, which runs on the VCC node, is a stand alone web server dedicated to providing UI access for configuration and runtime access to the event data. HMI systems that support an embedded web browser can access all UI screens through http.

The Longwatch Video Control OCX is a custom video streamer OCX that needs to reside on any client wishing to view a live stream from a remote unit. With proper security enabled by the client browser, the UI server will allow the remote deployment and registration of the OCX when a camera view is requested. The VideoControl OCX can be directly inserted in any ActiveX container and supports properties and methods to control the live streaming video.

2.2.1. Installation and Setup

The Longwatch HMI integration components are automatically installed on every VCC node and can be optionally installed on a client computer with network access to the VCC.

To enable a client node access to the video events and live streams use the Client Installation option. This option will install both the VideoControl OCX as well as the Longwatch OPC server. Use the Custom option to only install one of them.

2.2.1.1. OPC Server (HMI SCADA Node) Installation

To allow HMI SCADA nodes access to Longwatch data the Longwatch Video Control OCX must be installed. The Client Installation will prompt the user for the nodename (or IP address) of the remote VCC node as well as the VCC's OCX Port that the VCC is

listening to. This information is stored in a file called “opc.ini” and accessed by the Longwatch OPC server to connect to the VCC.

Typically,

C:\Program Files\Longwatch\User Data\opc.ini

Note: If the VCC node name changes you will need to update this ini file to reflect the new computer name for the VCC.

2.2.2. Longwatch OPC Data Server

The Longwatch system includes an OPC Data Access 2.05 server that provides access to most of the system alarm and status information.

The OPC Data server can be installed on the same node as the VCC software or can be installed separately from the VCC software and can run on a different machine. In general it is recommended that it be installed and run on the same machine as the HMI software so that DCOM setup issues can be avoided.

Note: the Longwatch OPC server will be launched automatically when accessed by the HMI.

2.2.2.1. Setup

The OPC Data server needs to know the nodename and port number of the VCC. This is contained in an INI file located at:

C:\Program Files\Longwatch\User Data\OPC.INI

This file is typically created by the install program but can be modified if needed using Notepad. It consists of one line of text indicating the nodename and port number to use to access the VCC. The nodename can be ‘localhost’ if the VCC is on the same computer as the OPC server. The port number should be 7521 unless this conflicts with some other application, and as been changed.

Example:

localhost 7521

2.2.2.2. OPC ItemNames

The server item name syntax is:

rvenue.variablename

Example:

Well1.DI0

Where rvenue is the name assigned to an RVE in the VCC and variable name is selected from the list below.

NOTE: the states of the DI fields show 1 if in the ALARM state – they do NOT show the actual raw input state. A normally OPEN DI will show ALARM (1) if the input CLOSED. A normally CLOSED DI will show ALARM (1) if the input is OPEN.

Name	R/W	Notes
DI(n) n=0..11 Ex. "DI2"	R	The current alarm status of each hardware Digital Input bit range (0..11) on the RVE. This value is 1 when the input is ALARMED (regardless of whether ALARMED means OPEN or CLOSE).
LDI(n) n=0..11 Ex. "LDI2"	R	The 'latched' status of each hardware Digital Input range (0..11) on the RVE. This value is set when the corresponding DI is set but is only cleared by writing to 'CLEAR' (see below). This value will not be cleared if the current DI input state is in alarm when the CLEAR command is sent. HMI Hint: Use this field for input status to make sure you catch the event.
NDI(n) n=0..11 Ex. "NDI2"	R	The "new" alarm status of each of the hardware Digital Input range (0..11). A value of 1 indicates an alarm state. This value is cleared by writing to "CLEAR" (see below). HMI Hint: Use this field to indicate new alarm condition (blinking attribute).
DIFS(n) n=0..11 Ex. "DIFS2"	R/W	Force Set (Force to ALARM state) for each DI (range 0..11). If this is set the DI is forced to be set (1). This can be used to force an alarm event from a remote location. HMI Hint: Use this field to simulate an event. Be sure to clear the force (0) before retriggering another event. You will need to wait the duration of the VCC/RVE poll time to ensure the initial force command has been sent out.
DIFC(n)	R/W	Force Clear (Force to NORMAL state) for each DI

n=0..11 Ex. "DIFC2"		(range 0..11). If this is set the DI is forced to be clear (0). This is essentially ' ALARM DISABLE '. Note this forces clear even if Force Set is active. Also, if the alarm condition still exists when this field cleared a new event is generated.
DOPULSE(n) n=0..12 Ex. "DOPULSE2"	W	Any write operation will cause the corresponding Digital output to be pulsed for a 10 second duration.
DOSET(n) n=0..12 Ex. "DOSET2"	W	A value of 1 will force the output activated. A value of 0 will remove the force. That is, if there is an alarm event is currently driving the same output, then value written (1 or 0) is logically OR'ed with this. To clear the driving of an output from an alarm call CLEAR to clear the latch alarm status.
MASTER	R	Current state of Master Arm (this reflects and force values and/or the state of DI15). 1 = ARMED, 0 = DISARMED. Note that a Closed circuit for DI15 is DISARMED (i.e. 0) and open circuit is ARMED (i.e. 1). This differs from the other DI bits. In the DISARMED state, alarming is suppressed (for details see the Longwatch User Guide).
MASTER_FRC	R/W	This 3 state value controls the master arm state. 0 = Force to DISARMED state 1 = Force to ARMED state 2 = Normal (DI15 controls the state – open = armed, closed = disarmed)
		Note the DI15, DIFS15 and DIFC15 values present in V1 and V2 of the system have been replaced by MASTER and MASTER_FRC
ZONEA ZONEB ZONEC	R/W	These fields indicate the armed/disarmed status of the three zones (A,B,C). 0 = Zone is DISARMED 1 = Zone is ARMED (normal) (See Longwatch User Guide for additional details on Zones)
VDI(n) n=0..11	R	The current status of each Video Cortex TM Analyzer output (one per camera) or IP Camera Event trigger or connection status.

Ex. "VDI2"		
LVDI(n) n=0..11 Ex. "LVDI2"	R	The latched status of each Video Event output. Cleared by writing to CLEAR. Upon clearing the latch status gets the current status of the corresponding video input signal. That is, if there is still an alarm condition the latch status will not be cleared.
NVDI(n) n=0..11 Ex. "NVDI2"	R	The new alarm status of each Video Event output. Cleared by writing to CLEAR
VDIFS(n) n=0..11 Ex. "VDIFS2"	R/W	Force Set for each VDI. If this is set the VDI is forced to be set (1).
VDIFC(n) n=0..11 Ex. "VDIFC2"	R/W	Force Clear for each VDI. If this is set the VDI is forced to be clear (0). This is essentially 'alarm disable'. Note this forces clear even if Force Set is active.
FILEPCT	R	If a file upload is in progress (e.g. and alarm clip) this shows the percent complete. Note there is currently no indication as to whether additional file uploads are queued after this one.
FILECNT	R	Number of Files uploaded from the remote to the VCC since the last time VCC Restarted.
FILEPEND	R	Number of files in the remotes queue that still need to be uploaded.
CPUTEMP	R	The internal CPU temperature of the RVE in DEGC. Note that the CPUTEMP, FREEMEM and FREEDISK values are monitored by the RVE Watchdog Timer task. If that task is not running (as will be the case on a laptop demo system for example) these values will all be 0.
FREEMEM	R	The amount of Free RAM in the RVE in Megabytes. This should normally be approx 100. If it drops below 10 the system we restart itself.
FREEDISK	R	The amount of Free Disk Space in the RVE in Megabytes.
CLEAR	R/W	Read value is always 0. Writing ANY VALUE to this variable (including 0) will clear all LDI and LVDI bits.
CLRQUE	R/W	Read value is always 0. Writing ANY VALUE to this variable (including 0) will clear the event and alarm clip

		queues in the RVE. This will CANCEL any outstanding messages or clips which have not yet been uploaded and will also abort any upload currently in progress. ** USE THIS WITH CAUTION **
ANYALM	R	Inclusive OR of all DI, VDI, and DR bits
ANYLAT	R	Inclusive OR of all LDI, LVDI, and LDR bits.
		Note: The following variables are obtained from the VCC and are available even if the RVE is offline (status = 1)
STATUS	R	The online/offline status of the remote. 0 indicates ONLINE. 1 indicates OFFLINE (Comm error)
MSGS	R	Number of communications messages so far today.
MSGS24	R	Number of communications messages yesterday.
MSGSR	R	Number of communications messages since reset.
RETRY	R	Number of retries(lost messages) so far today.
RETRY24	R	Number of retries(lost messages) yesterday.
RETRYR	R	Number of retries(lost messages) since reset.
FAILED	R	Number of failed messages so far today
FAILED24	R	Number of failed messages yesterday
FAILEDR	R	Number of failed messages since reset
FRAME	R	Time in seconds required to upload most recent video frame
FRAMEAVG	R	Average time in seconds for last 20 frames
LINK	R	Time in seconds for last individual message to/from RVE.
LINKAVG	R	Average time in seconds for last 20 messages
TRAN	R	Time in seconds for last transaction to/from RVE (similar to LINK time but includes any additional time required to gain access to a shared resource such as a COMM port).
TRANAVG	R	Average time for last 20 transactions
RESET	R/W	Always reads as 0. Write a ANY VALUE to clear the 'resettable' counters (R) listed above.
DROPEN(n) n=1..12 Ex. DROPEN1	R	Door reader open status. Value of 1 indicates door OPEN.

DRCOM(n) n=1..12 Ex.DRCOM1	R	RVE to Door reader communication status. Value of 1 indicates that the communication link is OFFLINE.
DRTAMP(n) n=1..12 Ex.DRTAMP1	R	Door reader onboard tamper status. Value of 1 indicates that the reader has a “tamper” alarm.
DREVENT(n) n=1..12 Ex.DREVENT1	R/W	Writing a 1 to this field will force a onshot manual trigger of a door OPEN event. This field will always read back a value of 0.
DRPULSE(n) n=1..12 Ex.DRPULSE1	R/W	Writing a 1 to this field will set the door unlock relay, wait for the “door delay period”, then relock the door. This field will always read back a value of 0.
DRUNLOCK(n) n=1..12 Ex.DRUNLOCK1	R/W	Writing a 1 to this field will set the door unlock relay. Writing a 0 will lock the relay. Reading this value will indicate the status of the door reader’s unlock relay.
DRNEW(n) n=1..12 Ex.DRNEW1	R	This field indicates the new alarm status of the door reader. A door will generated an alarm on either an OPEN, TAMPER, or BADBADGE event. This field is cleared by a CLEAR command.
DRLAT(n) N=1..12 Ex.DRLAT1	R	This field indicates the latched alarm status of the door reader. A door will generated an alarm on either an OPEN, TAMPER, or BADBADGE event. If the current status is not in alarm, this field can be cleared with a CLEAR command.
PRESET(n) N=0..11 Ex.PRESET2	W	Goto preset for camera number n. The value passed is the preset number sent to the camera.

Notes:

The force buts as well as the master and zone enable bits are persistent in the RVE. They ARE preserved during deploys and restarts including full reboots. (This is a new feature of V3).

There may be additional variables exposed when browsing (such as CBCNT0..CBCNT3 and CFCNT0..CFCNT3). These variables are for internal diagnostic use only.

2.2.3. iFIX Integration

In this section we will outline the iFIX integration to a Longwatch system.

2.2.3.1. The Database

Events can be monitored and alarmed using DI or DA blocks that tie to the Longwatch OPC Server (see Longwatch OPC Server section for details).

Hint: Use the block's alarm extension fields to store the unit name. This can then be used later in the graphics VBA script engine to provide unit context for the alarm in the alarm summary. Using VBA the unit name can be extracted from the currently selected alarm in the alarm summary and used to either filter the event list by unit name, or to bring up a live video view of the unit.

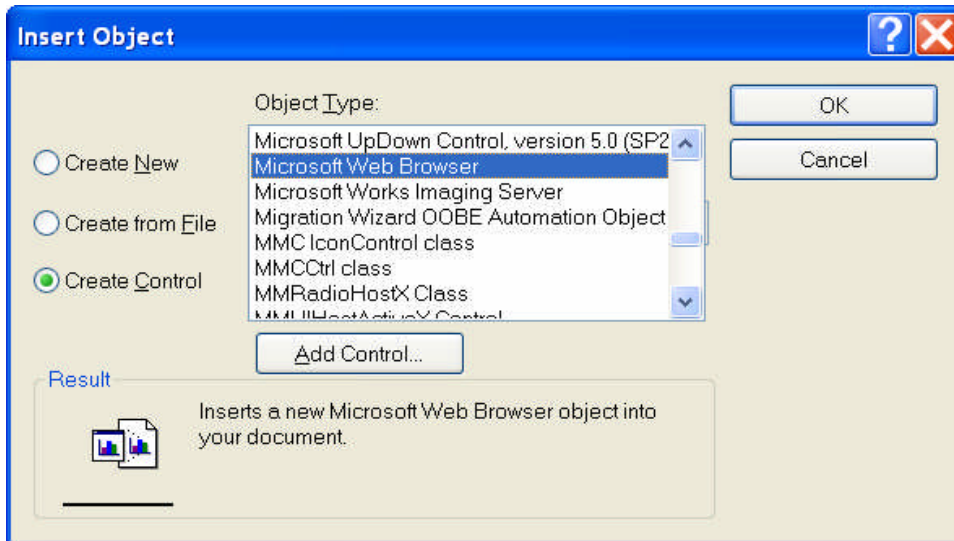
2.2.3.2. iFIX OPC Client I/O Driver

For an iFIX system to access the Longwatch OPC server the iFIX OPC Client IO driver must be installed. Below is some helpful hints to get this running quickly.

- Install it (V7.33 or later)
- Check SCU and add "IOCNTRL.EXE /A" at top of task list if not already there
- Use OPC client power tool to add a single server, group, and item.
- Be sure to ENABLE all servers groups and items (each one has a separate enable checkbox and they are all OFF by default)
- Save the configuration.

2.2.3.3. UI Integration

The Longwatch system exposes snap-in UI components to provide access to video archives, events detected, and live video feed. The UI Server component serves up these UI components as HTML web pages. Any of these UI components can be added to an iFIX screen by using the Microsoft Web Browser ActiveX control.



Once this control is added to an iFIX screen (WebBrowser1) then the following simple script can be added to set its url.

Example VBA script.

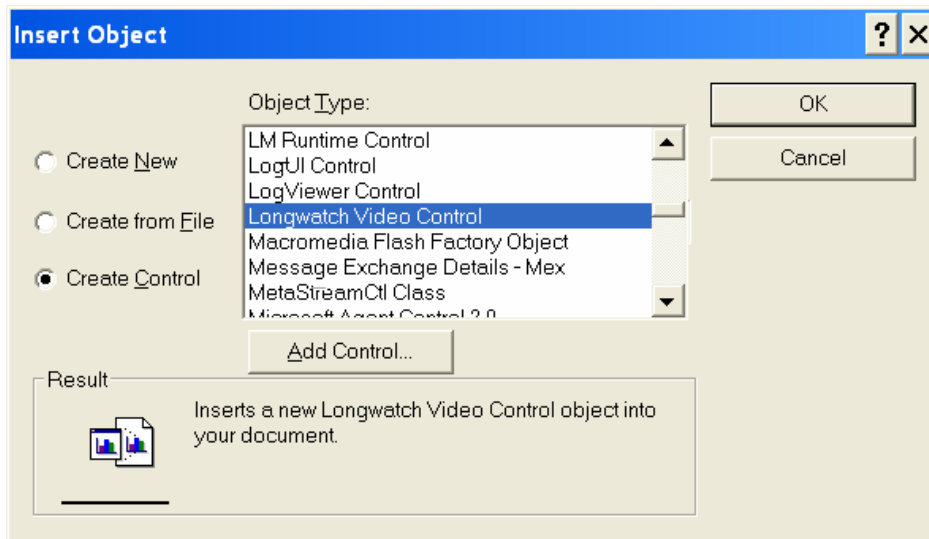
```
WebBrowser1.Navigate2 ("http://localhost:7520/EventView.cgi?unit=test 1")
```

To access the live video stream the VideoControl OCX should be directly embedded into an iFIX screen (see Longwatch Live Video Control)

2.2.3.4. ActiveX Control Integration (Live)

To embed a VideoControl OCX into an iFix screen select OLE Object from the Insert menu item from the iFix application menu. In the Insert Object dialog box select the Longwatch Video Control fro the list of available controls. The Create control radio button should be selected by default. Selecting the OK button will embed a control in the currently active picture.

Insert Object Dialog Box



To configure the newly embedded Video Control double click on it or right click and select Properties from the Longwatch Video Control Object submenu. In either case the Video Control properties dialog box will open. For more information on setting the Longwatch Video control's properties please refer to section 2.3.2.1.

2.2.4. WonderWare Integration

In this section we will outline an example Wonderware integration to a Longwatch system.

2.2.4.1. The Database

Events can be monitored and alarmed using Discrete or Analog Tagnames that tie to the Longwatch OPC Server by way of the OPCLink application.

2.2.4.2. Wonderware OPCLink Application

Wonderware connects to the Longwatch OPC server by way of an application named OPCLink. To install OPCLink, insert the Wonderware I/O driver disk and select OPCLink version 8.0 from the menu on the left hand side. Optionally you may install the OPCLink Tag Generator, which will assist in the creation of Wonderware Tagnames and DDE Access names.

2.2.4.3. OPCLink Topic Definition

Before accessing data from the Longwatch OPC server you must first define topics in the OPCLink application. First start OPCLink and select the Topic Definition item from the Configure menu. In the Topic Definition dialog press the New button that will open the Topic Definition dialog. The following illustration shows the creation of a DDE Access

name that will communicate with an RVE unit located at a fictitious well house named Well1. For more detailed explanations of these settings please press the Help button and refer to the online help.

Note: You can skip this step if you use the OPCLink Tag Creator

OPCLink Topic Definition Dialog Box

The screenshot shows the 'OPCLink Topic Definition' dialog box with the following settings:

- Topic Name: Well1
- Mode Name: (empty)
- OPC Server Name: Longwatch
- OPC Path: Well1.
- Update Interval: 1000 ms
- Enable access to update interval:
- Poke synchronously:
- Mode After Poke: None
- Transaction Timeout: 10000 ms
- Poke mode: Control mode, Transition mode, Full optimization
- Lifecycle Settings: Lifecycle, Timeout: 0 ms

2.2.4.4. OPC Link Tag Creator

The easiest method of adding Tagnames to a Wonderware application is by way of the OPCLink tag generator.

The Wonderware OPC Tag Creator is a module for Wonderware InTouch WindowMaker. It is capable of connecting to registered local and remote OPC servers and browsing through the exposed address space. It creates InTouch Access Names and tags and configures the Wonderware OPCLink. OPC Tag Creator is intended to be used for InTouch (version 7.1 Patch 06 or later) running on Windows NT 4.0 and Windows 2000.

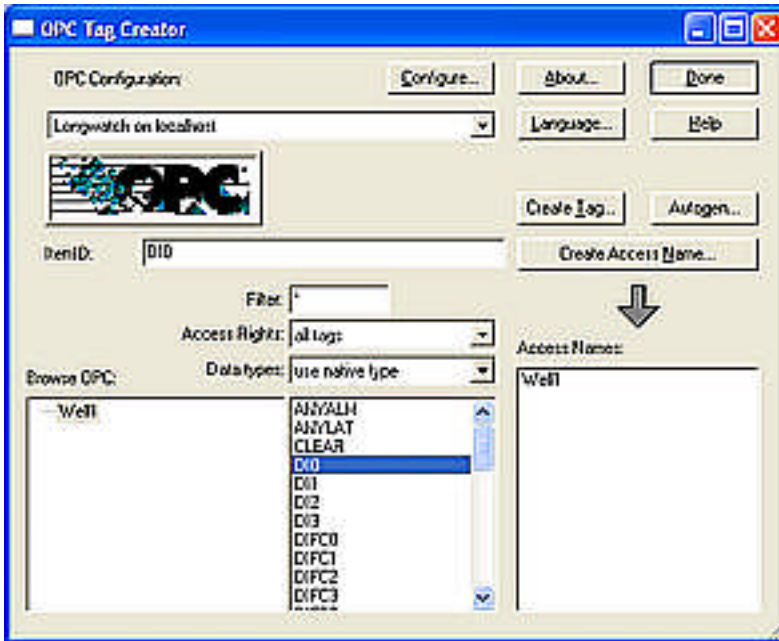
The **OPC Tag Creator** has the following functionality:

- Browsing OPC servers
- Browsing OPC items
- Mapping OPC items to InTouch tags
- Generating tags
- Generating Access Names
- Mapping OPC groups to InTouch Access Names
- Configuring a local and remote Wonderware OPCLink

You can access the Tag Creator from the OPC item of the WindowMaker application explorer.

The following illustration shows the Tag Creator module being used to generate tags for the topic that we created in the previous step.

OPC Tag Creator Module



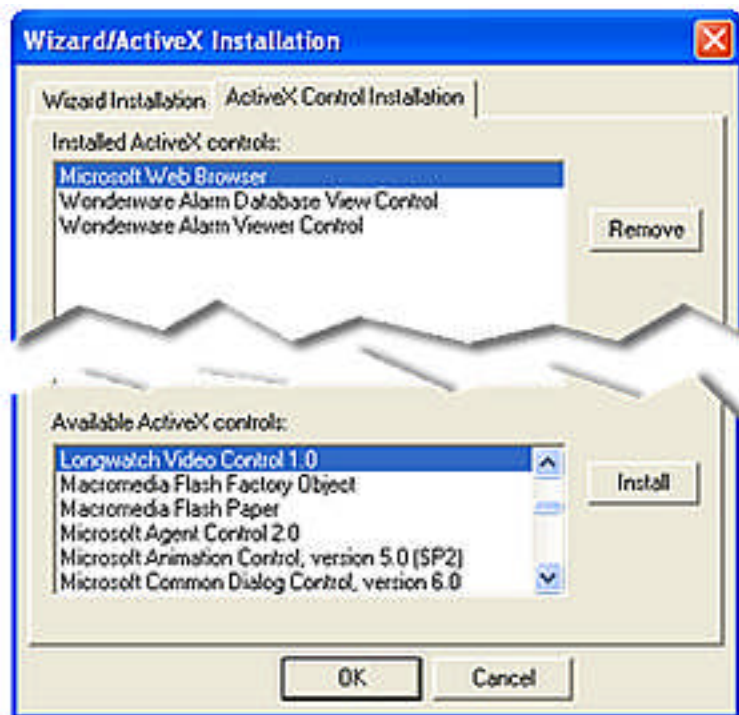
For more detailed instructions on the use on the Tag Creator please refer to the Wonderware online help.

2.2.4.5. ActiveX Control Installation

Before being able to embed an ActiveX in a Wonderware picture it must first be “registered” with WindowMaker. To register a control in WindowMaker double click the Wizard/ActiveX item in the Configure branch of the Application Explorer tree or select the Wizard/ActiveX item from the configure menu.

Once open select the ActiveX Control Installation tab. Select the desired control from the list of available ActiveX controls and press the Install button.

The following illustration shows an example of the Wizard/ActiveX Installation dialog box. As you can see the Microsoft Web Browser is already installed and the Longwatch Video Control is selected in the list of available controls.



2.2.4.6. Inserting ActiveX Controls In Wonderware

Inserting an ActiveX control in a Wonderware picture is done by selecting it from the Wizards toolbar. To insert a control, select the the Wizard button which is represented by an icon resembling a wizard's hat. Then select ActiveX Controls from the list on the left hand side, this will display a list of available controls on the right. Select a control and press the OK button, this will insert the selected control into the currently active picture.

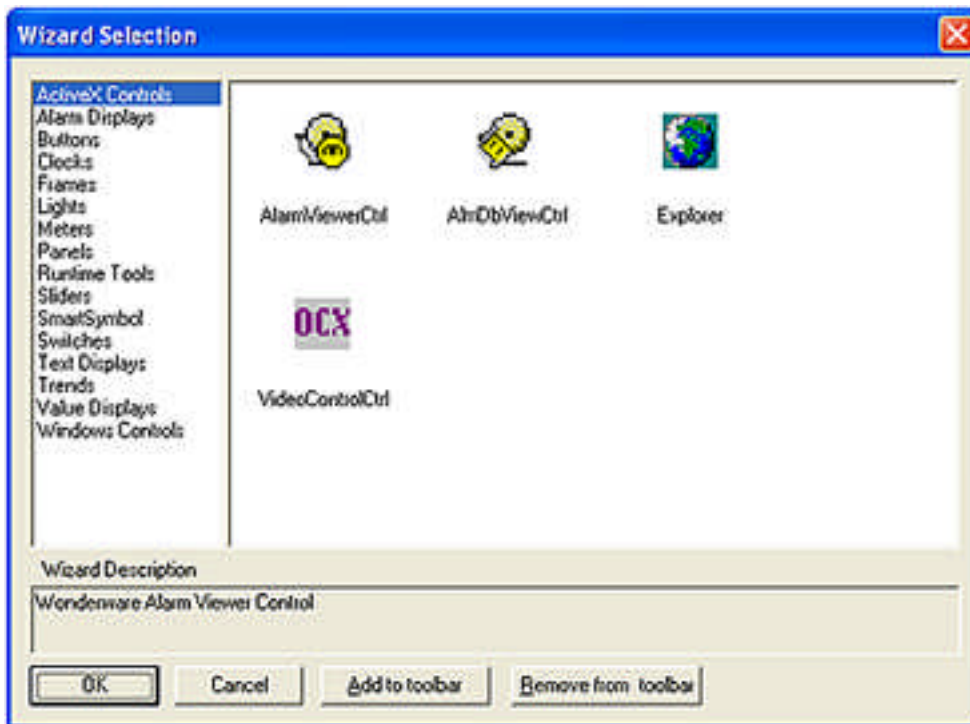
Hint: To add the selected control to the Wizard toolbar press the Add to toolbar button at the bottom of the dialog box. This will make future access to the control(s) easier

The following illustrations show the process of inserting an ActiveXControl:

The Wizard Toolbar



The Wizard Selection dialog box



2.2.4.7. Setting Control Properties

To set control properties double click on the control to open its properties setting dialog box. For information on setting the properties of the Longwatch Video Control please refer to section 3.2.2.1 above

2.2.4.8. Scripting the Longwatch Video Control

Wonderware has its own scripting engine which allows for scripting of the controls either through an event of the control itself or in response to a push button acting upon the control. This concept applies for other ActiveX controls as well.

This example illustrates the script used in the Video Overview picture contained in the Longwatch Wonderware demo.

Example Script:

```
#Video2.StopStreaming();  
#Video2.UnitName = #ThisControl.UnitName; {Only if needing to change unit name}  
#Video2.Cam = #ThisControl.Cam;  
#Video2.StartStreaming();
```

- Place 2 Video Controls on a new picture
 - Name one Video1 and the other Video2
 - Set their properties to different Units and or Cameras
- Double click on Video1 to open it properties dialog
 - Select the Events tab and you will see a list of scriptable events
- To create a *new* script for the Click event
 - Double click in the Script box
 - The script editor will open..
 - Name the script. In this example I named it VideoClick
- Create Script:
 - Select a control name for the script by using the Insert ActiveX Control button. A list of all ActiveX Controls in the application will be displayed
 - Using the example above create the script above.
- Script Notes
 - You cannot have 2 Controls in the application with the same name. Even if they are on different pictures, Wonderware will generate an error.
 - The Keyword #ThisControl refers to the control being clicked on. This makes the script reusable among many controls.
 - Each line in the script ends in a semicolon.
 - The Validate button will tell you if there is an error in the script.

- You can comment your script by placing them between curly brackets
{This is a comment about my script}
- Switch to Runtime mode and click on Video1. You will see Video2 change it's camera and or Unit to match Video1.
- Insert another Longwatch Video Control on the picture
 - Name it Video3 (or anything you like)
 - As in the earlier step double click the control and select the Events tab.
 - Next to the Click event select the ellipses button to the right of the empty cell.
 - Check the VideoClick script we created earlier.
- Switch to Runtime and click on the new Video Control. Video2 will change to match Video3.
- Click alternately between Video1 and Video3 and Video2 will change appropriately.

2.2.4.9. Scripting the Web Browser Control

Scripting this control is similar in the same fashion as scripting the Video Control.

In the following example we will create a picture that opens up the Longwatch Video Events web page in a browser control

- Place a browser control on a new picture. It will be named Explorer1 by default.
- Place a push button on the same picture
 - Double click on the button to open the animation links dialog box
 - Select the Action Touch Pushbutton link.
- Using techniques in the previous example create the following script
 - #Explorer1.Navigate2 ("http://localhost:7520/");
- Switch to Runtime and press the button. The Browser control will navigate to the Video Events page.
- Notes:
 - Placing this script in the picture's OnShow action script would navigate the Browser control to the page without having to press a button.
 - Similar techniques can be used to filter events by modifying the URL of the Nagigate2 method as outlined in previous sections of this document.

2.2.5. Other HMI Integration Packages

Clearly any HMI that supports OPC and ActiveX containment will work well with Longwatch. These include packages such as RSView32, Iconics Genesis32, Citect, etc. Check the Longwatch web site for application notes as they become available.

2.3. UI Server – Client Application Components

The UI Server is a mini HTTP web server that is responsible for providing access to the Longwatch system to client applications (HMI systems or client browsers). The UI server provides access to the following runtime information.

- a. Event List
- b. Video PlayBack UI
- c. Live Video Feed
- d. Unit Diagnostics
- e. Configuration UI

The web pages served by the UI Server can be displayed directly in Internet Explorer (6.0 compatible browsers) or embedded in an HMI via a Web Browser ActiveX control. The UI Server does not require any third party web server (IIS or Apache) but can be used in conjunction with such systems.

Most of the web pages are dynamic and are accessed with the extension (.cgi). This is similar, in concept, to a typical web server CGI page provided by many full web servers. The concept is that html pages are dynamically generated on request (substituting real-time information provided by the DB and VCC). Parameters can be passed to each of these CGI pages to provide application context (ie. UnitName, Cam #, etc)

Below is a quick list of the runtime CGI pages exposed by the UIServer.

<u>CGI Page</u>	<u>Description</u>
SimpleEventView.cgi	Simple table of events. URL parameters provide complete filtering and paging control (no UI controls)
EventView.cgi	Full Event View with UI controls at bottom
Live.cgi	Main Live View page with switching between Unit and Cam live view.
UnitView.cgi	Four camera live view.
CamView.cgi	Single cam live view (use OCX)
Playback.cgi	Video clip playback UI. Displays event video clips associated with a specific event.

DiagView.cgi	Displays a table of diagnostic information on each unit.
--------------	--

2.3.1. Event List View

The Event List view provides a means to,

1. Query the VCC's Event DB for a list of events across the entire system.
2. Playback a clip based on a specific event.
3. Acknowledge the event to halt any outstanding notification scheme in process.

There are two exposed views.

SimpleEventView – this view provides a list of events based on the passed filter parameters. (see URL parameters). The query will return a maximum number of 50 events at one time and then allow paging to the entire list. Sorting of the returned data can be accomplished by clicking on the column headers. A second click will reverse the sorting.

In addition this view provides two clickable regions per event. The first is the ACK field. The ACK field is represented by a grey checkbox. A checked state meaning that the event has been acknowledged. Clicking on the check box will cause an ACK command to be sent to the UI Server. The second clickable area is the Camera column (on the left). If a camera icon is displayed then clicking on this will display the event clip for the event.

EventView – provides a simple view of a list of events plus UI controls to configure all filter parameters, page through the list, as well as acknowledge all events.

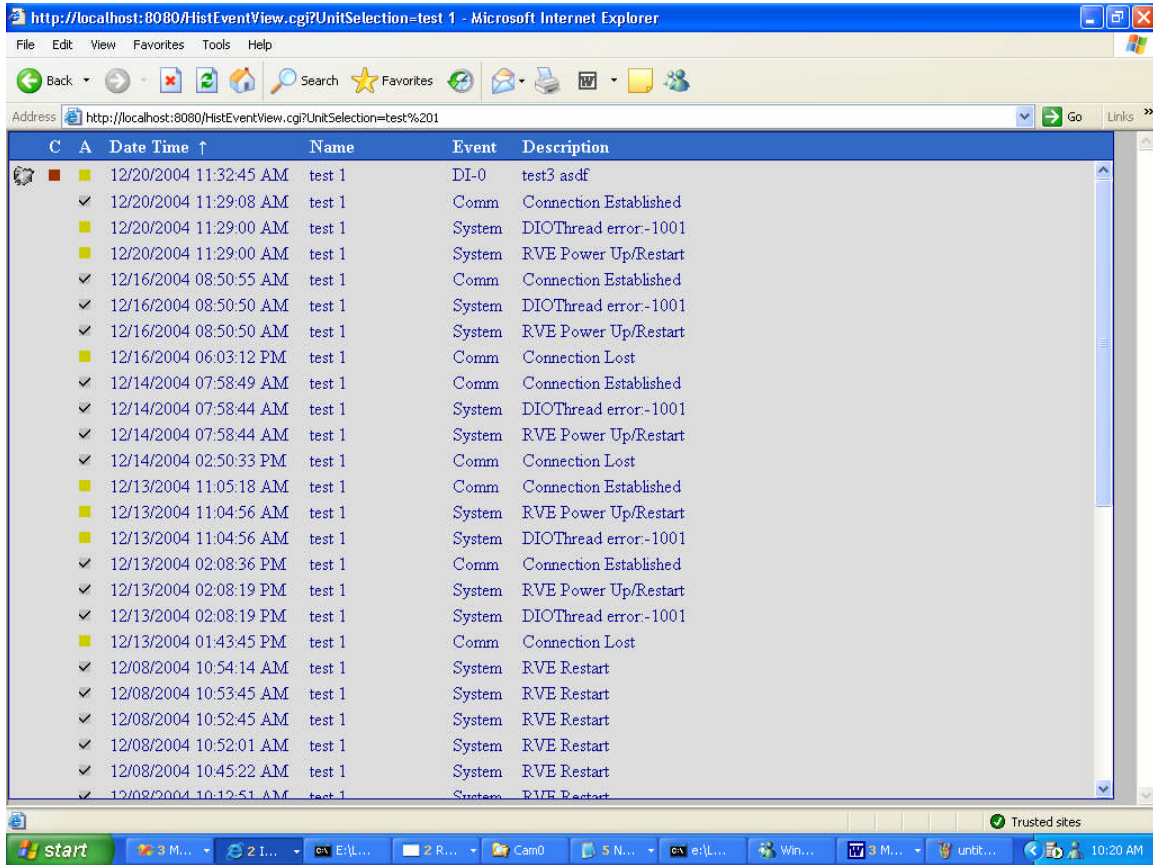
2.3.1.1. Sorting and Filtering Events

All event views have the capability to perform text sorting (ascending or descending) on any text column in the event view. This is accomplished by clicking on the column header. An arrow appears pointing up or down indicating the sorting direction. Sorting on non-text columns is not supported. Sorting will happen in the client and applies only to the items already in the page. The UI server returns a maximum of 50 events per page.

Filtering on specific types of column data is accomplished by either directly specifying the filter list in the filter UI controls of the EventView or by passing url parameters (see below).

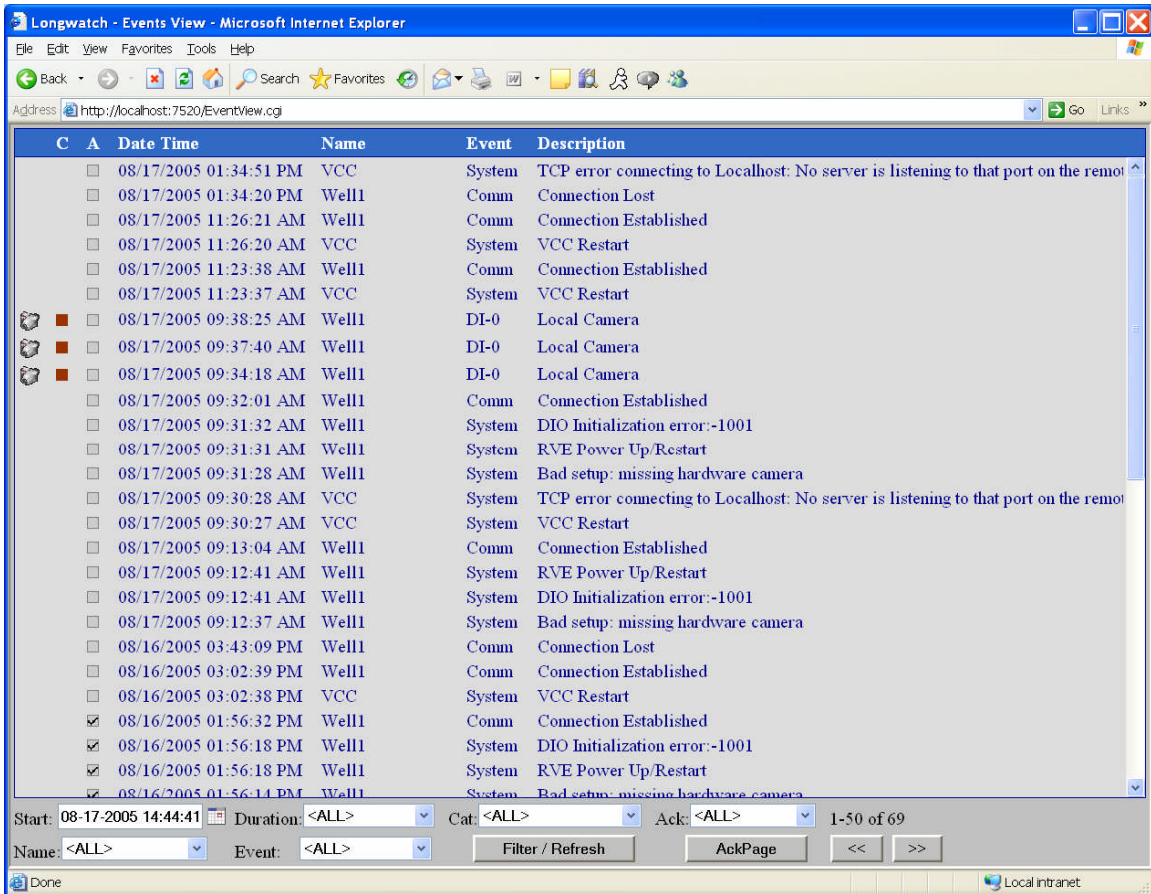
2.3.1.2. Simple View

Below is a Simple View of the event list displayed in Internet Explorer.



2.3.1.3. Event View

Below is an Event View displayed in Internet Explorer. Note the UI controls at the bottom of the display for controlling the filter parameters to the query as well as the page up and page down buttons.



2.3.1.4. URL Parameters

The Event View and Simple Event View provide a rich set of URL parameters that can be used to specify a specific set of filter parameters to the query in the event database.

SimpleEventView - SimpleEventView.cgi
 EventView - EventView.cgi

Below is the list parameters to these url requests.

<u>Parameter</u>	<u>Description</u>	<u>Valid Values</u>	<u>Default Value</u>
startDateText	Starting Date (used in conjunction with Duration)	MM-DD-YYYY hh:mm:ss	Now
duration	Duration before (-) or after (+) the StartDate. If the duration is 0,1,or 2, the Starting Date is ignored.	0 - All 1 - Today 2 - Yesterday 3 - (-) 1 hr 4 - (-) 4 hrs 5 - (-) 24 hrs 6 - (-) week 7 - (-) month 8 - (+) 1 hr 9 - (+) 4 hrs 10 - (+) 24 hrs 11 - (+) week 12 - (+) month >100 – duration value is the amount that exceeds 100. (ex 136 = 136-100 = +36 hour duration value).	0 - All
unit	Name Filter.	<ALL>, VCC's name, or any Unit name.	<ALL>
etype	Event type filter. The filters DI, Video, and VCR will return all Cameras (0..3).	<ALL>, DI, DI-0, DI-1, DI-2, DI-3, Video, Video-0, Video-1, Video-2, Video-3, VCR, VCR-0, VCR-1, VCR-3, VCR-3, Comm, System	<ALL>

cat	Event Category	0 - <ALL> 1 – Maintenance 2 – Alarms Only 3 – Alarm and Event	0 - All
ack	Acknowledge Filter	0 - <ALL> 1 – Ack'd Alarms 2 – UnAck'd Alarms	
pageSize	Number of rows to return	1 – 50	50
startPage	Starting Page	1 to n	1

Below is a few examples of using the above parameters to build a URL string that can query the Simple Event View page. Multiple parameters are separated by a &.

Example 1:

<http://localhost:7520/SimpleEventView.cgi?unit=Unit1>

Returns all events from “Unit1”

Example 2:

<http://localhost:7520/SimpleEventView.cgi?startDate=12-22-200410:00:00&duration=4&unit=Unit1&etype=DI>

Returns all hardware generated events (DI) from “Unit1” that have occurred between 12-22-2004 06:00:00 and 12-22-2004 10:00:00.

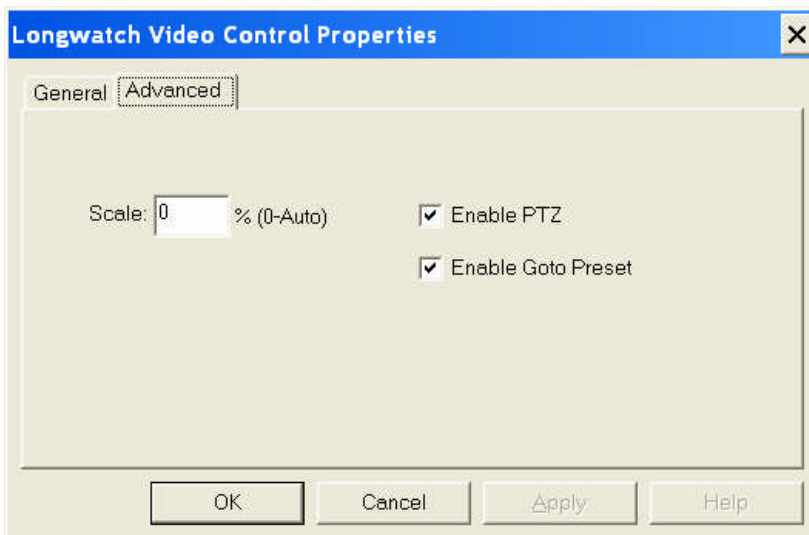
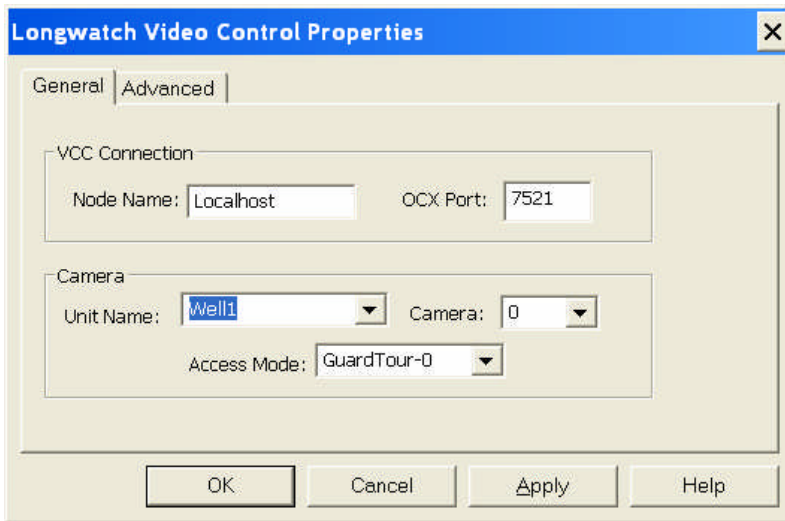
2.3.2. Live Streaming with VideoControl OCX

The Longwatch VideoControl OCX provides a live feed stream from any of the remote units. This control can be directly embedded in any ActiveX container.

The control can be inserted into a container with the OLE Control Insert Dialog box by selecting “**Longwatch Video Control V1.0**”.

2.3.2.1. Properties

The control provides a property page for easy configuration.



The VideoControl OCX has the following properties

<u>Property</u>	<u>Description</u>	<u>Valid Values</u>	<u>Default</u>
UnitName	Unit Name	String	None – must be supplied.
Cam	Camera number	0..3	0

UpdateRate	Hint to VCC on how fast to send image updates to the OCX if available. (milliseconds)		500
Access Mode	Access Mode	0 – Guard Tour mode (from VCC cache). 1 – Live feed mode – turns on RVE live feed streaming. (Caution!)	Guard Tour
DisplayMode	Display Mode	0 – Show image view 1 – Show diagnostics view	Image View
Scale	Image Scale Mode	0 – Scale image to do a best fit within containing window without clipping and without changing aspect ratio. (will leave black borders). >0 – Use this value to calculate a fixed size scale factor from the image's original live feed size (120 by 160). $\text{newW} = \text{orgW} * \text{scale} / 100$ $\text{newH} = \text{orgH} * \text{scale} / 100$ Note: in the event the new values exceed the size of the window, the system will assume a best fit implementation (0 above).	0
NodeName	Computer name or IP address of VCC computer	Something you can ping	Localhost
Port	OCX Port exposed by VCC	Valid TCPIP Port	7521
StreamOnUnitUpdate	Flag indicating if the control should automatically start streaming when the UnitName property is changed.	0 – Wait for StartStream() method or container transitions from Design to Run mode. 1 – Automatically call StartStream() when the UnitName property is Set. Use this if your container does not support a Design/Run mode property and does not support direct method access to the OCX.	0
Stale	Flag indicating if image data is "Stale"	(available only in RUN mode) 0 – not stale 1 – indicates stale	
LastError	Error Code for last error	(available only in RUN mode) 0 – indicates no error non 0 – error code	
Status	Streaming Status	(available only in RUN mode)	

		0 – Active connection to VCC – streaming from VCC. 1 – Idle – no connection or streaming stopped.	
ImageWidth	Size of original image width in pixels	(available only in RUN mode) -1 - unknown	
ImageHeight	Size of original image height in pixels	(available only in RUN mode) -1 - unknown	
FrameUpdateRate	Number of milliseconds between last data update. Diagnostics purposes.	(available only in RUN mode)	
PTZEnable	Controls on screen PTZ actions with the left mouse button	1 – Enable on screen PTZ cursors and mouse interaction enabled. Note: only operational if camera configuration has PTZMode set to FULL. Lastly, this control must be in LIVE access mode for PTZ to be operational. 0 – Disabled for this instance	1
PTZGotoEnable	Controls right mouse Goto PTZ preset operation.	1 – Enables RM Button pop up for Goto Preset. Note: only operational if the camera configuration has PTZMode set to either “Preset” or “Full” AND there are valid preset strings defined. Lastly, this control must be in LIVE access mode for PTZ to be operational. 0 – Disable for this instance	1
IsPTZAvailable	Is PTZ control available (only available at RUNTIME).	1 - PTZ operations can be performed on this camera connection. 0 – PTZ operations are not enabled for this camera connection.	

2.3.2.2. Methods

Method Name	Description	Parameters
StartStreaming()	Apply properties and start streaming video from VCC. Call this function after dynamically changing any of the properties.	None

StopStreaming()	Stop streaming request from VCC. Status goes to IDLE.	None
Freeze()	Freeze the current image from updating.	None
Continue()	Continue applying updates to image.	None
PTZLeft (n)	Pan camera left	n – (Long) – length of movement (see PTZ Driver manual)
PTZRight (n)	Pan camera right	n – (Long) – length of movement (see PTZ Driver manual)
PTZUp(n)	Tilt camera up	n – (Long) – length of movement (see PTZ Driver manual)
PTZDown (n)	Tilt camera down	n – (Long) – length of movement (see PTZ Driver manual)
PTZIn(n)	Zoom camera in	n – (Long) – length of zoom in (see PTZ Driver manual)
PTZOut (n)	Zoom camera out	n – (Long) – length of zoom out (see PTZ Driver manual)
GetPresetName (pset,pName)	Given a preset number return the preset name.	[in] (Integer) pset – Preset number (1..9) [out] (String) pName – returned name of preset, if not defined then null string.
GoToPreset	Send a Goto preset command to the PTZ driver.	[in] (Integer) pset - Preset number (1..9)
GetPresetByName	Get a preset number by the passed in name.	[in] (String) pName – preset name to lookup. [out] (Integer) pset – returned preset number

SetPreset	Set a preset to the current location of the camera.	([in] (Integer) pset – preset number (1..9)
GetDefaultIncrements	Retrieve the increments sent the PTZ driver on each mouse click, SHIFT mouse click, and CNRL mouse click.	[out] (Long) sm – returned small increment value [out] (Long) med– returned medium increment value [out] (Long) big– returned large increment value (Note: see PTZ driver manual for how these are applied)

2.3.2.3. PTZ Operations

The OCX control supports direct PTZ control operations via the mouse, as well as methods and properties to perform full pan, tilt, zoom, Goto Preset operations. The Set Preset operation is only available through a method on the control.

The PTZ on screen operations are enabled if the following conditions are all true.

1. The OCX is in Live access mode.
2. The OCX PTZEnable property is set to 1 (check box enabled)
3. The camera that the OCX is viewing has been defined with FULL PTZ capability. (See VCC administration UI, PTZ tab).

When enabled, the OCX will dynamically change the cursor to an ARROW pointing in the direction of the operation. If the cursor is NOT changing as you move around the image check all three above.

To Pan Left – move cursor to left side of image (cursor changes to left arrow) and click with left mouse. Right, Up, and Down work the same.

The size of the PTZ operation can be changed with keyboard accelerators.

<SHIFT> Left Mouse Click – indicates a BIG movement (note: cursor size change)

<CNRL> Left Mouse Click – indicates a SMALL movement(note: cursor size change)

If you click more than once the system will attempt to accumulate the operation so 2 clicks will move the camera twice the length of one click.

2.3.3. Playback View

The UI server provides access to the playback event clips from the remote unit via a dynamic html page. The playback screen can be accessed by either,

1. Accessing the Event List page and clicking on the camera icon for that event (typical and recommend approach) . The playback UI is displayed inside the same window as the event list.
2. By directly accessing Playback.cgi passing the event id.

URL Parameters

<u>Parameter</u>	<u>Description</u>	<u>Valid Values</u>	<u>Default</u>
id	Event ID	(see note	None – must be supplied.
bcolor	Background Color for html page	Name of color	“silver”
fit	StretchToFit	0 – playback using original size 1 – scale video to autofit in window	1

Example:

<http://localhost:7520/PlayBack.cgi?id=4322445>

2.3.4. Accessing Event ID via Scripting.

There is no direct API that can be used to query the event database to access a list of event ids for playback but you can with a simple trick access the event playback URL of the event the user has selected from the Event List interface. With this information, you can then cancel the request and perform your own redirection to a selected window for playback.

To accomplish this you will need to embed the WebBrowser control in a container that allows you to intercept the browsers "BeforeNavigate2" event. You should set the WebBrowser to initially point to either SimpleEventView.cgi or EventView,.cgi (see above).

The script below will get called when the control switches pages. The code in the event will cancel the request when a "Playback.cgi" is requested.

```
Private Sub WebBrowser1_BeforeNavigate2(ByVal pDisp As Object, URL As Variant, Flags As Variant, TargetFrameName As Variant, postData As Variant, Headers As Variant, Cancel As Boolean)
```

```
Rem This event is called right before going to the passed URL string
```

```
If URL Like "*/PlayBack.cgi?id=*" Then
```

```
    Cancel = True
```

```
    Rem
```

```
    Rem Here you now have the complete URL string including the ID to perform a playback
```

```
    Rem You can save the URL in a global variable and use it in another pop up picture
```

```
    Rem
```

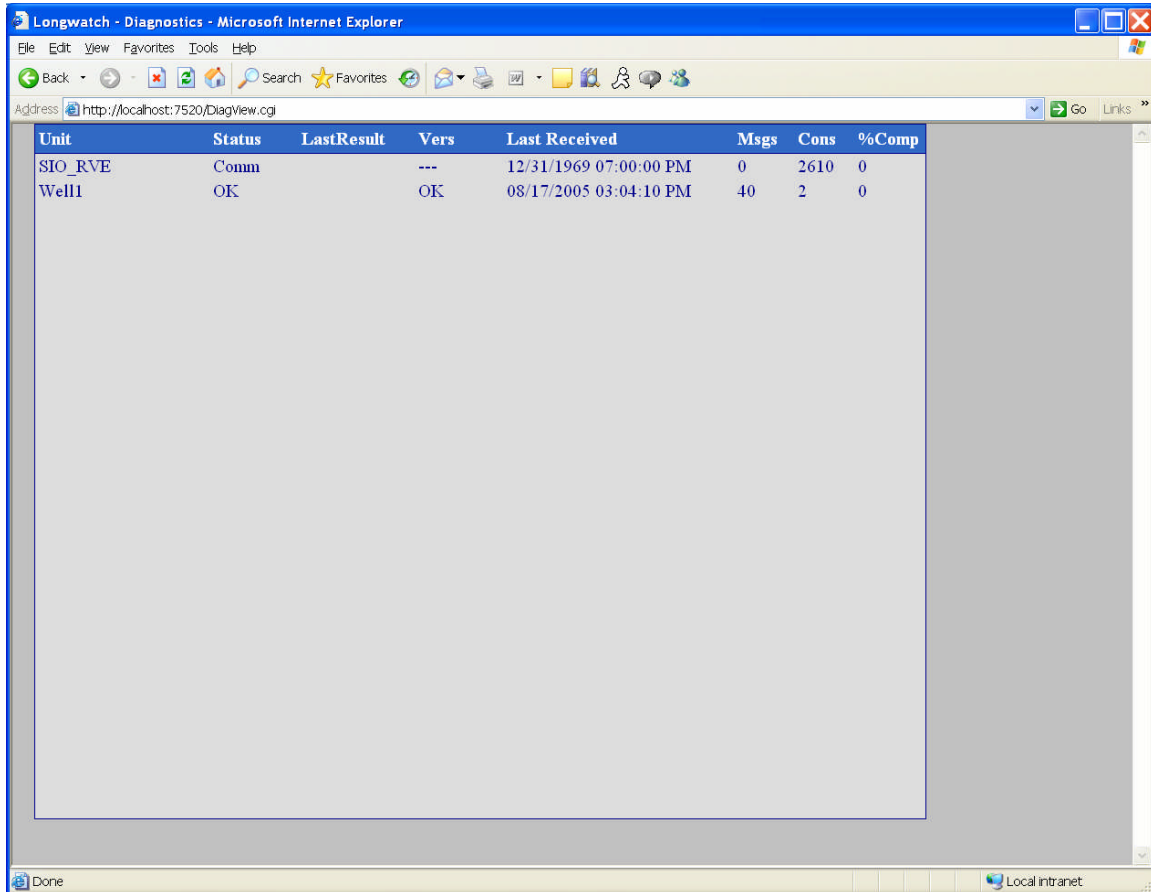
```
    WebBrowser1.GoBack
```

```
End If
```

```
End Sub
```

2.3.5. Diagnostics View

The VCC keeps the current health and communication status of each remote. The UI server accesses the VCC to present this information to the user via the Diagnostics View.



Unit	Status	LastResult	Vers	Last Received	Msgs	Cons	%Comp
SIO_RVE	Comm		---	12/31/1969 07:00:00 PM	0	2610	0
Well1	OK		OK	08/17/2005 03:04:10 PM	40	2	0

The diagnostics view has 8 columns. Each row represents one unit in the configuration.

Column	Description
Unit	Name of remote unit
Status	<p>Current Status of the remote unit</p> <p><u>OK</u> [-<cam#>] - Unit connected and communicating. Optional list of cam numbers indicate which cameras the VCC is currently requesting a live feed video stream.</p> <p><u>Error</u> - Protocol error between UI server and VCC.</p> <p><u>Download</u> - Download from VCC to Remote in process.</p> <p><u>Upload</u> - Upload from Remote to VCC in progress.</p> <p><u>Cntrl</u> - Control request in progress from VCC to Remote</p> <p><u>N/A</u> - Unit configured but not yet deployed to the VCC.</p> <p><u>Comm</u> - VCC to Remote communication error.</p>
Last Result	<p>Displays result of last Upload/Download request. The format of this field is cmd type (Upload, Download, Cntrl) and status. The status will have the format of the following.</p> <p><u>OK</u> – completed successfully</p> <p><u>Busy</u> – operation still in progress</p> <p><u>Read (error)</u> – protocol read file error</p> <p><u>Write (error)</u> – protocol write file error</p> <p><u>Start (error)</u> - protocol restart error</p> <p><u>(error)</u> – remote control command error</p>
Vers	<p>Version status of remote unit's configuration as compared to the VCC central database configuration.</p> <p><u>OK</u> - Remote configuration matches</p> <p><u>Mismatch</u> - May require either a deploy or and upload</p> <p><u>----</u> Unknown. No access to remote</p>
Last Received	Date and time of last message received in local VCC time.
Msgs	Number of messages received from the remote unit since the last reconnect.
Cons	Number of reconnections to remote unit
% Comp	File transfer percentage. Values start at 0 and stop at 100% indicating that the file is uploaded. This field can be used to monitor the progress of an event video clip upload.

Example URL:

<http://localhost:7520/DiagView.cgi>