

**SunGuide<sup>SM</sup>:**

# **Video Wall Interface Control Document**

**SunGuide-VW-ICD-6.2**



Prepared for:

Florida Department of Transportation  
Traffic Engineering and Operations  
Office 605 Suwannee Street, M.S. 90  
Tallahassee, Florida 32399-0450  
(850) 410-5600

March 16, 2016

<b>Document Control Panel</b>			
File Name:	SunGuide-VW-ICD-6.2.doc		
File Location:	SunGuide CM Repository		
CDRL:	2-7.1		
	Name	Initial	Date
Created By:	Lynne Randolph, SwRI	LAR	11/11/04
Reviewed By:	Steve Dellenback, SwRI	SWD	11/12/04
	Steve Novosad, SwRI	SEN	11/12/04
	Adam Hoffman, SwRI	AGH	3/16/16
Modified By:	Meredith Moczygemba, SwRI	MRM	11/17/05
Completed By:			

## Table of Contents

<b>1.</b>	<b>Scope .....</b>	<b>1</b>
1.1	<b>Document Identification .....</b>	<b>1</b>
1.2	<b>Project Overview.....</b>	<b>1</b>
1.3	<b>Related Documents .....</b>	<b>1</b>
1.4	<b>Contacts .....</b>	<b>2</b>
<b>2.</b>	<b>Data .....</b>	<b>3</b>
2.1	<b>Schema.....</b>	<b>3</b>
2.1.1	Subsystem communication .....	3
2.1.2	Driver communication .....	4
2.2	<b>Examples.....</b>	<b>4</b>
2.3	<b>Subsystem Schemas.....</b>	<b>6</b>
2.4	<b>Driver Schemas.....</b>	<b>9</b>
<b>3.</b>	<b>Notes .....</b>	<b>11</b>
	<b>Attachment 1 .....</b>	<b>12</b>
A.1	activeDesktop.xsd .....	1
A.2	videoTour.xsd .....	2
A.3	wall.xsd .....	3
A.4	wallGeometryUpdateMsg.xsd .....	7
A.5	activateLayoutReq.xsd.....	8
A.6	addTourReq.xsd .....	9
A.7	connectionReq.xsd .....	10
A.8	deactivateLayoutReq.xsd .....	11
A.9	deleteTourReq.xsd .....	12
A.10	modifyTourReq.xsd .....	13
A.11	refreshGeometryReq.xsd .....	14
A.12	retrieveDataReq.xsd.....	15
A.13	startVideoTourReq.xsd .....	17
A.14	stopVideoTourReq.xsd .....	18
A.15	subscribeReq.xsd .....	19
A.16	activateLayoutResp.xsd .....	21
A.17	addTourResp.xsd .....	22
A.18	connectionResp.xsd .....	23
A.19	deactivateLayoutResp.xsd .....	24
A.20	deleteTourResp.xsd.....	25
A.21	modifyTourResp.xsd.....	26
A.22	refreshGeometryResp.xsd.....	27
A.23	retrieveDataResp.xsd .....	29
A.24	startVideoTourResp.xsd.....	33
A.25	stopVideoTourResp.xsd.....	34
A.26	subscribeResp.xsd.....	35



## **List of Figures**

Figure 1.1 - High-Level Architectural Concept.....	1
Figure 2 Sample Transaction .....	5

## **List of Acronyms**

ATMS	Advanced Traffic Management System
DOT	Department of Transportation
FDOT	Florida Department of Transportation
IM	Incident Management
ITS	Intelligent Transportation Systems
ITN	Invitation to Negotiate
SwRI	Southwest Research Institute
TMC	Traffic Management Center
VS	Video Switching
VW	Video Wall
XML	Extensible Markup Language

## **REVISION HISTORY**

<b>Revision</b>	<b>Date</b>	<b>Changes</b>
1.0.0	November 11, 2004	Initial Release
1.0.2	November 17, 2005	Updated for Release 2 software
6.2	March 16, 2016	Updated for Release 6.2

# 1. Scope

## 1.1 Document Identification

This Interface Control Document (ICD) describes the interface between individual SunGuide clients and the Video Wall (VW) subsystem and between the VW subsystem and the associated drivers. The general base architecture of the XML communications including connection information, byte order and base transaction classes is delineated in the general ICD. This ICD defines Extensible Markup Language (XML) schemas upon which XML requests shall be based in communicating amongst the various processes. Refer to the SunGuide-General-ICD-6.2 document for details regarding data transfer.

## 1.2 Project Overview

The Florida Department of Transportation (FDOT) is conducting a program that is developing SunGuide software. The SunGuide software is a set of Intelligent Transportation System (ITS) software that allows the control of roadway devices as well as information exchange across a variety of transportation agencies. The goal of the SunGuide software is to have a common software base that can be deployed throughout the state of Florida. The SunGuide software development effort is based on ITS software available from the state of Texas; significant customization of the software is being performed as well as the development of new software modules. The following figure provides a graphical view of the software to be developed:

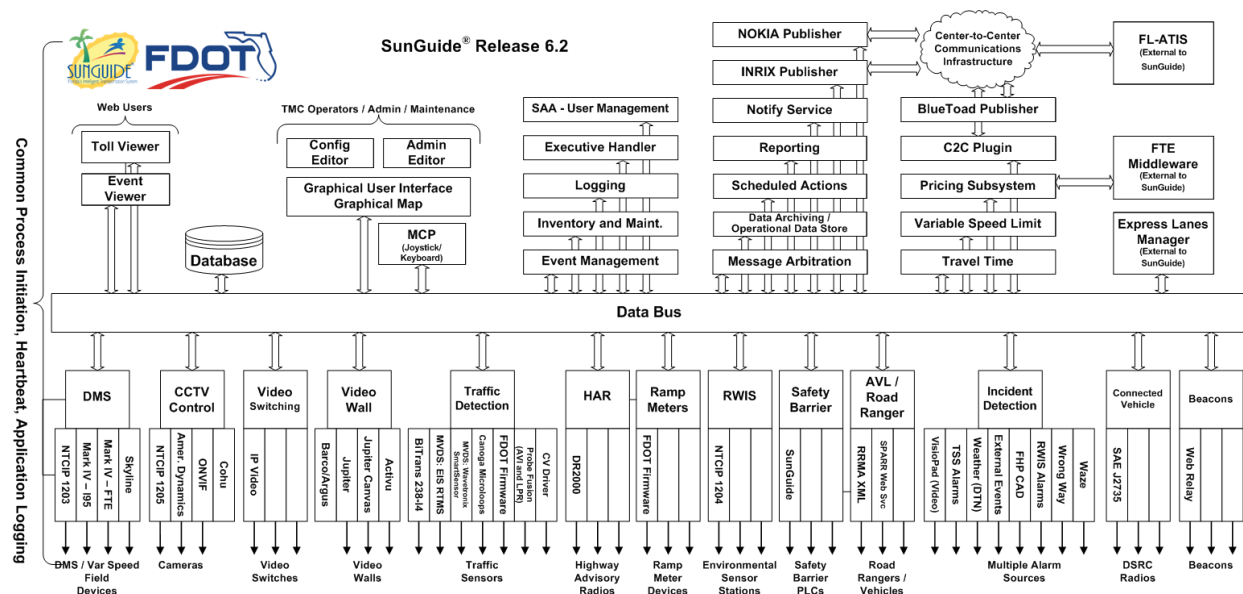


Figure 1.1 - High-Level Architectural Concept

The SunGuide<sup>SM</sup> development effort spans approximately two years. After the development, the software will be deployed to a number of Districts and Expressway Authorities throughout Florida and support activities will be performed.

## 1.3 Related Documents

The following documents were used to develop this document:



- FDOT Scope of Services: *BDQ69, Standard Written Agreement for SunGuide Software Support, Maintenance, and Development, Exhibit A: Scope of Services*. July 1, 2010.
- Notice to Proceed: Letter to Southwest Research Institute® (SwRI®) for BDQ69, July 1, 2010.
- SunGuide Project website: <http://sunguidesoftware.com>.

## **1.4 Contacts**

The following are contact persons for the SunGuide software project:

- Fred Heery, ITS Section, Traffic Engineering and Operations Office Central Office, [fred.heery@dot.state.fl.us](mailto:fred.heery@dot.state.fl.us), 850-410-5606
- Derek Vollmer, ITS Section, Traffic Engineering and Operations Office Central Office, [Derek.Vollmer@dot.state.fl.us](mailto:Derek.Vollmer@dot.state.fl.us), 850-410-5615
- Clay Packard, Atkins Project Manager, [clay.packard@dot.state.fl.us](mailto:clay.packard@dot.state.fl.us), 850-410-5623
- David Chang, Atkins Project Advisor, [david.chang@dot.state.fl.us](mailto:david.chang@dot.state.fl.us), 850-410-5622
- Tucker Brown, SwRI Project Manager, [tbrown@swri.com](mailto:tbrown@swri.com), 210-522-3035
- Roger Strain, SwRI Software Project Manager, [rstrain@swri.org](mailto:rstrain@swri.org), 210-522-6295

## 2. Data

The following sections detail the XML transactions that can be exchanged between client and server applications.

### 2.1 Schema

The schemas for these transactions may be located in the Schemas1.0 directory. The objects directory contains common data schemas that are used by the various request/messages/responses. Schemas are organized in the following tree structure:

- messages
  - removeVideoWallMsg.xsd
  - wallGeometryUpdateMsg.xsd
- objects
  - activeDesktop.xsd
  - dataObjects.xsd
  - wall.xsd
- requests
  - activateLayoutReq.xsd
  - connectionReq.xsd
  - deactivateLayoutReq.xsd
  - refreshGeometryReq.xsd
  - retrieveDataReq.xsd
  - startVideoTourReq.xsd
  - stopVideoTourReq.xsd
  - subscribeReq.xsd
- responses
  - activateLayoutResp.xsd
  - connectionResp.xsd
  - deactivateLayoutResp.xsd
  - refreshGeometryResp.xsd
  - retrieveDataResp.xsd
  - startVideoTourResp.xsd
  - stopVideoTourResp.xsd
  - subscribeResp.xsd

Requests may be sent from a client to a subsystem or from a subsystem to a driver. Responses may be sent from a driver to a subsystem or a subsystem to a client. A message can be sent from any process to another process.

#### 2.1.1 Subsystem communication

Initial communication to a subsystem is described in the general ICD. For VW, the video tours are retrieved from the database on startup, while the information about the video wall is retrieved from the video wall driver. Once a client has initiated the connection to VW, connection requests may be made, video tours may be started or stopped, and video wall layouts may be activated and

deactivated. If the video wall has been modified outside of the VW subsystem, a client may send a request to refresh the geometry of the wall.

For video tours, the VW subsystem does not update the database for any tours added, modified or deleted. This is a function of the Video Switching (VS) subsystem. Active video tours are not persisted in VW.

The following table shows the various subscriptions a client may request. The last column shows the XML updates that will be received if a client has subscribed to this data.

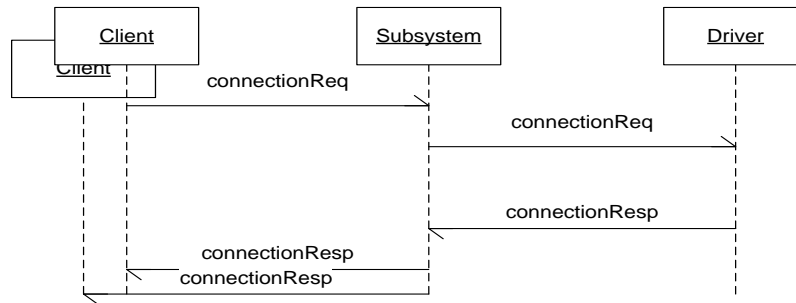
<b>Subscription</b>	<b>Description</b>	<b>Updates Received</b>
wallData	Receive updates to video wall geometry information.	wallGeometryUpdateMsg
connectionData	Receive updates to the sources displayed on the video wall viewers.	connectionResp
tourData	Receive updates to video tours in the system.	addTourResp, modifyTourResp, deleteTourResp, startVideoTourResp, stopVideoTourResp
userData	Receive notification that user permissions have been modified.	updateSystemDataMsg

### *2.1.2 Driver communication*

Initial communication from a subsystem to a driver is described in the general ICD. On the VW driver startup, the driver queries the video wall for the current configuration. Once the driver has current data, the VW subsystem may forward requests from clients to the appropriate driver.

## **2.2 Examples**

For example, if a client wishes to connect a source to a viewer on the video wall, the client sends a connectionReq to the subsystem. Once the subsystem has verified the client has permission to perform this request, the request is then forwarded to the appropriate driver. The driver makes the connection and sends a connectionResp to the subsystem. The subsystem then sends this response back to the appropriate client and to any clients who have subscribed to connection data.



**Figure 2 Sample Transaction**

The tables below show which requests can be sent from client to subsystem and subsystem to driver. The responses sent from driver to subsystem and subsystem to client are also specified. Messages are sent when a response is not required.

### 2.3 Subsystem Schemas

*FC (From client), TC (To client), TD (To driver), FD (From driver)*

Usage Description	Requests	FC	TD	Responses	FD	TC	Messages	TD	FD	TC
Used for a client to activate a layout on a video wall.	activateLayoutReq	X	X	activateLayoutResp	X	X				
Used to add a video wall to the system. (Typically sent when driver connects)							addVideoWallMsg			X
Used for a client to request a source be displayed upon a viewer.	connectionReq	X	X	connectionResp	X	X				
Used for a client to deactivate a layout on a video wall.	deactivateLayoutReq	X	X	deactivateLayoutResp	X	X				
Used to modify an existing video wall.							modifyVideoWallMsg			X

Usage Description	Requests	FC	TD	Responses	FD	TC	Messages	TD	FD	TC
Used to request a refresh of the video wall geometry information. Also used by the VW subsystem when an activate or deactivate layout request is processed.	refreshGeometryReq	X	X	refreshGeometryResp	X	X				
Used to remove a video wall from the system. (Typically sent on driver disconnect)							removeVideoWallMsg			X
Used to retrieve data from VW, data includes wall geometry, connection information, layouts, video tours, sources and users.	retrieveDataReq	X	X	retrieveDataResp	X	X				
Used for a client to request an existing video tour be started on a viewer.	startVideoTourReq	X		startVideoTourResp		X				

Usage Description	Requests	FC	TD	Responses	FD	TC	Messages	TD	FD	TC
Used for a client to request an existing video tour be stopped on a viewer.	stopVideoTourReq	X		stopVideoTourResp		X				
Used to subscribe to updates from the system. Data includes wall geometry, connection information, layouts, video tours, sources and users.	subscribeReq	X		subscribeResp		X				
Sent from the subsystem when the wall geometry has been updated. The driver also generates this message when a refresh geometry message is received.							wallGeometryUpdate Msg		X	X

## 2.4 Driver Schemas

*TD (To driver), FD (From driver)*

<b>Usage Description</b>	<b>Requests</b>	<b>TD</b>	<b>Responses</b>	<b>FD</b>	<b>Messages</b>	<b>TD</b>	<b>FD</b>
Used for a client to activate a layout on a video wall.	activateLayoutReq	X	activateLayoutResp	X			
Used for a client to request a source be displayed upon a viewer.	connectionReq	X	connectionResp	X			
Used for a client to deactivate a layout on a video wall.	deactivateLayoutReq	X	deactivateLayoutResp	X			
Used to request a refresh of the video wall geometry information. Also used by the VW subsystem when an activate or deactivate layout request is processed.	refreshGeometryReq	X	refreshGeometryResp	X			
Used to retrieve data from VW, data includes wall geometry, connection information, layouts, video tours, sources.	retrieveDataReq	X	retrieveDataResp	X			



<b>Usage Description</b>	<b>Requests</b>	<b>TD</b>	<b>Responses</b>	<b>FD</b>	<b>Messages</b>	<b>TD</b>	<b>FD</b>
Sent from the subsystem when the wall geometry has been updated. The driver also generates this message when a refresh geometry message is received.					wallGeometryUpdate Msg		X

### **3. Notes**

Information about XML and schemas can be found at the World Wide Web Consortium (W3) website at <http://www.w3.org>.

## **Attachment 1**

### **VW XML Schema**

## A.1 activeDesktop.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="wall.xsd"/>
  <xs:include schemaLocation="../../common/objects/common.xsd"/>
  <xs:include schemaLocation="../../common/objects/id.xsd"/>
  <xs:element name="activeDesktop">
    <xs:annotation>
      <xs:documentation>Elements that are currently active on the
desktop.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="viewer" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

## A.2 videoTour.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../../common/objects/common.xsd"/>
  <xs:element name="videoTour">
    <xs:annotation>
      <xs:documentation>The video tour contains the data needed to define a
tour.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:sequence>
        <xs:element name="tourId" type="identifier">
          <xs:annotation>
            <xs:documentation>The id of the video tour.</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="description" type="xs:string" minOccurs="0">
          <xs:annotation>
            <xs:documentation>The description of the video tour.</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="dwellTime" type="xs:int">
          <xs:annotation>
            <xs:documentation>The dwell time (in seconds) of the video
tour.</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="sourceId" type="identifier" minOccurs="0"
maxOccurs="unbounded">
          <xs:annotation>
            <xs:documentation>The source ids involved in the video
tour.</xs:documentation>
          </xs:annotation>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

### A.3 wall.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../../common/objects/id.xsd"/>
  <xs:element name="desktop">
    <xs:annotation>
      <xs:documentation>Stores a display wall desktop</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="PaneType">
          <xs:sequence>
            <xs:element ref="region" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
  <xs:element name="region">
    <xs:annotation>
      <xs:documentation>Stores a region of a managed desktop</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="PaneType">
          <xs:sequence>
            <xs:element ref="viewer" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
  <xs:element name="viewer">
    <xs:annotation>
      <xs:documentation>Stores a display wall viewer application</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="id"/>
        <xs:element ref="sourceld"/>
        <xs:element ref="window"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="window">
    <xs:annotation>
      <xs:documentation>Stores a window of a managed desktop</xs:documentation>
    </xs:annotation>
  </xs:element>
</xs:schema>
```

```
</xs:annotation>
<xs:complexType>
  <xs:complexContent>
    <xs:extension base="PaneType"/>
  </xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="wall">
  <xs:annotation>
    <xs:documentation>Stores the display wall</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="wallId" type="identifier">
        <xs:annotation>
          <xs:documentation>The name of the wall.</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element ref="desktop" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="layout">
  <xs:annotation>
    <xs:documentation>Stores a layout on a managed desktop</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="shortcut" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="window" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="shortcut">
  <xs:annotation>
    <xs:documentation>Stores a managed desktop shortcut</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:complexType name="PaneType" abstract="true">
  <xs:annotation>
    <xs:documentation>A type used for a pane that has geometry and
position.</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="identifier" type="xs:string">
      <xs:annotation>
        <xs:documentation>The Id name</xs:documentation>
      </xs:annotation>
    </xs:element>
```

```

<xs:element name="height" type="xs:int">
  <xs:annotation>
    <xs:documentation>Height of the pane</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="width" type="xs:int">
  <xs:annotation>
    <xs:documentation>Width of the pane</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="left" type="xs:int">
  <xs:annotation>
    <xs:documentation>Left or x position of window</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="top" type="xs:int">
  <xs:annotation>
    <xs:documentation>Top or y position of window</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="layer" type="xs:int" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Layer or z position of the window, where topmost panes
are designated 0</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="mapping">
  <xs:annotation>
    <xs:documentation>Combination of the following values:
maximized
2: minimized
4: normal
8: sizeable
16: visible
32: topmost
64: restore; (only for setGeometry () calls)</xs:documentation>
  </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:int">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="128"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:element name="layoutId" type="xs:string">
  <xs:annotation>
    <xs:documentation>The name of a layout</xs:documentation>
  </xs:annotation>

```

1:



```
</xs:annotation>
</xs:element>
<xs:element name="sourceld" type="xs:string">
  <xs:annotation>
    <xs:documentation>The source connected to the viewer</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="wallld" type="xs:string">
  <xs:annotation>
    <xs:documentation>The name of a display wall</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="viewerld" type="xs:string">
  <xs:annotation>
    <xs:documentation>The viewer currently open on the desktop</xs:documentation>
  </xs:annotation>
</xs:element>
</xs:schema>
```

#### A.4 wallGeometryUpdateMsg.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../../common/objects/transaction.xsd"/>
  <xs:include schemaLocation="../../objects/activeDesktop.xsd"/>
  <xs:element name="wallGeometryUpdateMsg">
    <xs:annotation>
      <xs:documentation>Message that is sent when the geometry of the wall has been
modified.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="MessageType">
          <xs:sequence>
            <xs:element ref="wallId"/>
            <xs:element ref="activeDesktop"/>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

## A.5 activateLayoutReq.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../../common/objects/transaction.xsd"/>
  <xs:include schemaLocation="../../objects/wall.xsd"/>
  <xs:element name="activateLayoutReq">
    <xs:annotation>
      <xs:documentation>Retrieves information about the available
layouts</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="RequestType">
          <xs:sequence>
            <xs:element ref="wallId"/>
            <xs:element ref="layoutId"/>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

## A.6 addTourReq.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../../common/objects/transaction.xsd"/>
  <xs:include schemaLocation="../../objects/videoTour.xsd"/>
  <xs:element name="addTourReq">
    <xs:annotation>
      <xs:documentation>The request is sent to add a new video tour to the
system.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="RequestType">
          <xs:sequence>
            <xs:element ref="videoTour" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

## A.7 connectionReq.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../objects/wall.xsd"/>
  <xs:include schemaLocation="../../common/objects/transaction.xsd"/>
  <xs:include schemaLocation="../../common/objects/common.xsd"/>
  <xs:element name="connectionReq">
    <xs:annotation>
      <xs:documentation>Connects a source to a viewer or disconnects a source from a
viewer.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="RequestType">
          <xs:sequence>
            <xs:element ref="wallId"/>
            <xs:element name="sourceId" type="identifier" minOccurs="0">
              <xs:annotation>
                <xs:documentation>The source to be connected If no sourceId,
then this should be a disconnect.</xs:documentation>
              </xs:annotation>
            </xs:element>
            <xs:element name="destinationId" type="identifier">
              <xs:annotation>
                <xs:documentation>The viewer to which the connection should be
made.</xs:documentation>
              </xs:annotation>
            </xs:element>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

## A.8 deactivateLayoutReq.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../../common/objects/transaction.xsd"/>
  <xs:include schemaLocation="../../objects/wall.xsd"/>
  <xs:element name="deactivateLayoutReq">
    <xs:annotation>
      <xs:documentation>Deactivates an active layout.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="RequestType">
          <xs:sequence>
            <xs:element ref="wallId"/>
            <xs:element ref="layoutId"/>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

## A.9 deleteTourReq.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../../common/objects/transaction.xsd"/>
  <xs:element name="deleteTourReq">
    <xs:annotation>
      <xs:documentation>The request is sent to delete a video tour from the
system.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="RequestType">
          <xs:sequence>
            <xs:element name="tourId" maxOccurs="unbounded">
              <xs:annotation>
                <xs:documentation>The id of the video tour to delete from the
system.</xs:documentation>
              </xs:annotation>
            </xs:element>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

## A.10 modifyTourReq.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../../common/objects/transaction.xsd"/>
  <xs:include schemaLocation="../../objects/videoTour.xsd"/>
  <xs:element name="modifyTourReq">
    <xs:annotation>
      <xs:documentation>The request is sent to modify a existing tour to the
system.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="RequestType">
          <xs:sequence>
            <xs:element ref="videoTour" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
</xs:schema>
```



## A.11 refreshGeometryReq.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../../common/objects/transaction.xsd"/>
  <xs:include schemaLocation="../../objects/wall.xsd"/>
  <xs:element name="refreshGeometryReq">
    <xs:annotation>
      <xs:documentation>Request the video wall system to retrieve new geometry data
from the wall.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="RequestType">
          <xs:sequence>
            <xs:element ref="wallId"/>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

## A.12 retrieveDataReq.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../../common/objects/transaction.xsd"/>
  <xs:element name="retrieveDataReq">
    <xs:annotation>
      <xs:documentation>A request sent by client to retrieve data from the
system</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="RequestType">
          <xs:sequence>
            <xs:element name="walls" type="xs:boolean" default="false"
minOccurs="0">
              <xs:annotation>
                <xs:documentation>True if the client requests to retrieve the wall
geometry.</xs:documentation>
              </xs:annotation>
            </xs:element>
            <xs:element name="users" type="xs:boolean" default="false"
minOccurs="0">
              <xs:annotation>
                <xs:documentation>True if the client requests to retrieve a list of
vws users.</xs:documentation>
              </xs:annotation>
            </xs:element>
            <xs:element name="statusList" type="xs:boolean" default="false"
minOccurs="0">
              <xs:annotation>
                <xs:documentation>True if the client requests to retrieve the current
viewer information.</xs:documentation>
              </xs:annotation>
            </xs:element>
            <xs:element name="tours" minOccurs="0">
              <xs:annotation>
                <xs:documentation>True if client requests to retrieve the video
tours.</xs:documentation>
              </xs:annotation>
            </xs:element>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
</xs:schema>

```



### A.13 startVideoTourReq.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../../common/objects/transaction.xsd"/>
  <xs:include schemaLocation="../../objects/wall.xsd"/>
  <xs:element name="startVideoTourReq">
    <xs:annotation>
      <xs:documentation>This is a request used to start the video
tour.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="RequestType">
          <xs:sequence>
            <xs:element ref="wallId"/>
            <xs:element name="tourId" type="identifier">
              <xs:annotation>
                <xs:documentation>The id of the video tour to
start.</xs:documentation>
              </xs:annotation>
            </xs:element>
            <xs:element name="destinationId" type="identifier">
              <xs:annotation>
                <xs:documentation>The id of the destination to hold the video tour
on.</xs:documentation>
              </xs:annotation>
            </xs:element>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

#### A.14 stopVideoTourReq.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../../common/objects/transaction.xsd"/>
  <xs:include schemaLocation="../../objects/wall.xsd"/>
  <xs:element name="stopVideoTourReq">
    <xs:annotation>
      <xs:documentation>This is a request used to stop the video
tour.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="RequestType">
          <xs:sequence>
            <xs:element ref="wallId"/>
            <xs:element name="destinationId" type="identifier">
              <xs:annotation>
                <xs:documentation>The id of the destination that video tour will be
stopped on.</xs:documentation>
              </xs:annotation>
            </xs:element>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

### A.15 subscribeReq.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../../common/objects/transaction.xsd"/>
  <xs:element name="subscribeReq">
    <xs:annotation>
      <xs:documentation>A request sent by client to subscribe to data updates from the
system</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="RequestType">
          <xs:all>
            <xs:element name="wallData" type="xs:boolean" default="false"
minOccurs="0">
              <xs:annotation>
                <xs:documentation>Request updates to wall
geometry.</xs:documentation>
              </xs:annotation>
            </xs:element>
            <xs:element name="connectionData" type="xs:boolean" default="false"
minOccurs="0">
              <xs:annotation>
                <xs:documentation>Request updates to connections on the
wall.</xs:documentation>
              </xs:annotation>
            </xs:element>
            <xs:element name="tourData" minOccurs="0">
              <xs:annotation>
                <xs:documentation>Requests updates to video tours and active
tours.</xs:documentation>
              </xs:annotation>
            </xs:element>
            <xs:element name="userData" type="xs:boolean" default="false"
minOccurs="0">
              <xs:annotation>
                <xs:documentation>Requests updates to
users.</xs:documentation>
              </xs:annotation>
            </xs:element>
          </xs:all>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
</xs:schema>

```



## A.16 activateLayoutResp.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../../common/objects/transaction.xsd"/>
  <xs:include schemaLocation="../../objects/wall.xsd"/>
  <xs:element name="activateLayoutResp">
    <xs:annotation>
      <xs:documentation>Response received when a layout is
activated.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="ResponseType"/>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
  <xs:complexType name="layoutData">
    <xs:annotation>
      <xs:documentation>The layout that was activated.</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="responseData">
        <xs:sequence>
          <xs:element ref="wallId"/>
          <xs:element ref="layoutId"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:schema>
```



## A.17 addTourResp.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../../common/objects/transaction.xsd"/>
  <xs:include schemaLocation="../../common/objects/common.xsd"/>
  <xs:include schemaLocation="../../objects/videoTour.xsd"/>
  <xs:element name="addTourResp">
    <xs:annotation>
      <xs:documentation>The response returned when a client adds a video
tour.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="ResponseType"/>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
  <xs:complexType name="tourData">
    <xs:annotation>
      <xs:documentation>The data returned when a tour is added.</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="responseData">
        <xs:sequence>
          <xs:element ref="videoTour" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:schema>
```

## A.18 connectionResp.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../../common/objects/transaction.xsd"/>
  <xs:include schemaLocation="../../objects/wall.xsd"/>
  <xs:element name="connectionResp">
    <xs:annotation>
      <xs:documentation>Response received from the driver with active desktop
data.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="ResponseType"/>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
  <xs:complexType name="connectionData">
    <xs:annotation>
      <xs:documentation>Data for the active desktop</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="responseData">
        <xs:sequence>
          <xs:element ref="wallId"/>
          <xs:element ref="id">
            <xs:annotation>
              <xs:documentation>The viewer to which the connection should be
made.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="sourceId" type="identifier" minOccurs="0">
            <xs:annotation>
              <xs:documentation>The source to be connected If no sourceId, then
this should be a disconnect.</xs:documentation>
            </xs:annotation>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:schema>

```

## A.19 deactivateLayoutResp.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../../common/objects/transaction.xsd"/>
  <xs:include schemaLocation="../../objects/wall.xsd"/>
  <xs:element name="deactivateLayoutResp">
    <xs:annotation>
      <xs:documentation>Response received after a layout is
deactivated.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="ResponseType"/>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
  <xs:complexType name="layoutData">
    <xs:annotation>
      <xs:documentation>The layout that was deactivated.</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="responseData">
        <xs:sequence>
          <xs:element ref="wallId"/>
          <xs:element ref="layoutId"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:schema>
```

## A.20 deleteTourResp.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../../common/objects/transaction.xsd"/>
  <xs:include schemaLocation="../../common/objects/common.xsd"/>
  <xs:include schemaLocation="../../common/objects/id.xsd"/>
  <xs:element name="deleteTourResp">
    <xs:annotation>
      <xs:documentation>The response returned when a client requests to delete tour.
</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="ResponseType"/>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
  <xs:complexType name="tourData">
    <xs:annotation>
      <xs:documentation>The data returned when a tour is deleted.</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="responseData">
        <xs:sequence>
          <xs:element name="tourId" minOccurs="0" maxOccurs="unbounded">
            <xs:annotation>
              <xs:documentation>The id of the tour to delete from the
system.</xs:documentation>
            </xs:annotation>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:schema>

```

## A.21 modifyTourResp.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../../common/objects/transaction.xsd"/>
  <xs:include schemaLocation="../../objects/videoTour.xsd"/>
  <xs:include schemaLocation="../../common/objects/common.xsd"/>
  <xs:element name="modifyTourResp">
    <xs:annotation>
      <xs:documentation>The response returned when a client modifies a existing tour
attributes.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="ResponseType"/>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
  <xs:complexType name="tourData">
    <xs:annotation>
      <xs:documentation>The data returned when a tour is
modified.</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="responseData">
        <xs:sequence>
          <xs:element ref="videoTour" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:schema>
```

## A.22 refreshGeometryResp.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../../common/objects/transaction.xsd"/>
  <xs:include schemaLocation="../../objects/activeDesktop.xsd"/>
  <xs:element name="refreshGeometryResp">
    <xs:annotation>
      <xs:documentation>Response received when the geometry has been refreshed
from the wall.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="ResponseType"/>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
  <xs:complexType name="geometryData">
    <xs:annotation>
      <xs:documentation>The updated geometry data that was retrieved from the
wall.</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="responseData">
        <xs:sequence>
          <xs:element ref="wallId"/>
          <xs:element ref="activeDesktop" minOccurs="0"/>
          <xs:element name="layouts" minOccurs="0">
            <xs:annotation>
              <xs:documentation>Layouts available for the wall.</xs:documentation>
            </xs:annotation>
            <xs:complexType>
              <xs:sequence>
                <xs:element ref="layoutId" minOccurs="0"
maxOccurs="unbounded"/>
              </xs:sequence>
            </xs:complexType>
          </xs:element>
          <xs:element name="sources" minOccurs="0">
            <xs:annotation>
              <xs:documentation>Sources available for the
wall.</xs:documentation>
            </xs:annotation>
            <xs:complexType>
              <xs:sequence>
                <xs:element ref="sourceId" minOccurs="0"
maxOccurs="unbounded"/>
              </xs:sequence>
            </xs:complexType>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:schema>
```

### A.23 retrieveDataResp.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../../common/objects/transaction.xsd"/>
  <xs:include schemaLocation="../../objects/activeDesktop.xsd"/>
  <xs:include schemaLocation="../../vs/objects/videoTour.xsd"/>
  <xs:include schemaLocation="../../vs/objects/activeVideoTour.xsd"/>
  <xs:element name="retrieveDataResp">
    <xs:annotation>
      <xs:documentation>Response received for a retrieve data
request</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="ResponseType"/>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
  <xs:complexType name="retrieveData" abstract="0">
    <xs:annotation>
      <xs:documentation>The response data for a retrieve data
request</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="responseData">
        <xs:sequence>
          <xs:element name="walls" minOccurs="0">
            <xs:annotation>
              <xs:documentation>The video walls in the
system.</xs:documentation>
            </xs:annotation>
            <xs:complexType>
              <xs:sequence>
                <xs:element name="wall" minOccurs="0"
maxOccurs="unbounded">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element ref="wallId"/>
                      <xs:element name="wallGeometry" minOccurs="0">
                        <xs:annotation>
                          <xs:documentation>Information about the desktop,
regions and windows of the wall configuration.</xs:documentation>
                        </xs:annotation>
                      </xs:complexType>
                    </xs:sequence>
                    <xs:element ref="desktop" minOccurs="0"
maxOccurs="unbounded"/>

```



```

        </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="layouts" minOccurs="0">
        <xs:annotation>
            <xs:documentation>Layouts available for the
wall.</xs:documentation>
        </xs:annotation>
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="layoutId" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="sources" minOccurs="0">
        <xs:annotation>
            <xs:documentation>Sources available for the
wall.</xs:documentation>
        </xs:annotation>
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="sourceId" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="viewers" minOccurs="0">
        <xs:annotation>
            <xs:documentation>Viewers available for the
wall.</xs:documentation>
        </xs:annotation>
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="viewerId" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="activeTours" minOccurs="0">
        <xs:annotation>
            <xs:documentation>The tours which are currently
activated to run on the video wall.</xs:documentation>
        </xs:annotation>
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="activeVideoTour" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>

```

```

        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="users" minOccurs="0">
  <xs:annotation>
    <xs:documentation>A list containing one or more
users.</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="user" minOccurs="0"
maxOccurs="unbounded">
        <xs:annotation>
          <xs:documentation>A user within the cctv
system.</xs:documentation>
        </xs:annotation>
        <xs:complexType>
          <xs:sequence>
            <xs:element name="privileges">
              <xs:annotation>
                <xs:documentation>Contains the privileges for the
user.</xs:documentation>
              </xs:annotation>
              <xs:complexType>
                <xs:attribute name="userData" type="xs:boolean"
use="required"/>
                <xs:attribute name="connectionData"
type="xs:boolean" use="required"/>
                <xs:attribute name="wallGeometryData"
type="xs:boolean" use="required"/>
                <xs:attribute name="createConn" type="xs:boolean"
use="required"/>
                <xs:attribute name="configWall" type="xs:boolean"
use="required"/>
              </xs:complexType>
            </xs:element>
          </xs:sequence>
          <xs:attribute name="name" type="identifier" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="statusList" minOccurs="0">

```

```
<xs:annotation>
  <xs:documentation>List of status data.</xs:documentation>
</xs:annotation>
<xs:complexType>
  <xs:sequence>
    <xs:element name="viewerStatus" minOccurs="0"
maxOccurs="unbounded">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="id"/>
          <xs:element name="status">
            <xs:complexType>
              <xs:sequence>
                <xs:element ref="sourceId"/>
              </xs:sequence>
            </xs:complexType>
          </xs:element>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="tours" minOccurs="0">
  <xs:annotation>
    <xs:documentation>All video tours that have been defined in the
system.</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="videoTour" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:schema>
```

## A.24 startVideoTourResp.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../../common/objects/transaction.xsd"/>
  <xs:include schemaLocation="../../common/objects/common.xsd"/>
  <xs:include schemaLocation="../../objects/wall.xsd"/>
  <xs:element name="startVideoTourResp">
    <xs:annotation>
      <xs:documentation>The response returned when a client requests to start the
video tour.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="ResponseType"/>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
  <xs:complexType name="startVideoTourData">
    <xs:annotation>
      <xs:documentation>The data returned indicating the details of the startVideoTour
request.</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="responseData">
        <xs:sequence>
          <xs:element ref="wallId"/>
          <xs:element name="tourId" type="identifier">
            <xs:annotation>
              <xs:documentation>The id of the video tour
started.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="destinationId" type="identifier">
            <xs:annotation>
              <xs:documentation>The id of the destination that video tour started
on.</xs:documentation>
            </xs:annotation>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:schema>

```

## A.25 stopVideoTourResp.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../../common/objects/transaction.xsd"/>
  <xs:include schemaLocation="../../common/objects/common.xsd"/>
  <xs:include schemaLocation="../../objects/wall.xsd"/>
  <xs:element name="stopVideoTourResp">
    <xs:annotation>
      <xs:documentation>The response returned when a client requests to stop the
video tour.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="ResponseType"/>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
  <xs:complexType name="stopVideoTourData">
    <xs:annotation>
      <xs:documentation>The data returned indicating the details of the stopVideoTour
request.</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="responseData">
        <xs:sequence>
          <xs:element ref="wallId"/>
          <xs:element name="destinationId" type="identifier">
            <xs:annotation>
              <xs:documentation>The id of the destination that video tour was
stopped on.</xs:documentation>
            </xs:annotation>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:schema>

```

## A.26 subscribeResp.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../../common/objects/transaction.xsd"/>
  <xs:include schemaLocation="../../common/objects/common.xsd"/>
  <xs:element name="subscribeResp">
    <xs:annotation>
      <xs:documentation>Response received for a subscribe
request</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="ResponseType"/>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
  <xs:complexType name="subscribeData">
    <xs:annotation>
      <xs:documentation>The response data for a subscribe
request</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="responseData">
        <xs:all>
          <xs:element name="wallData" type="xs:boolean" default="false"
minOccurs="0">
            <xs:annotation>
              <xs:documentation>Request updates to wall
geometry.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="connectionData" type="xs:boolean" default="false"
minOccurs="0">
            <xs:annotation>
              <xs:documentation>Request updates to connections on the
wall.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="tourData" minOccurs="0">
            <xs:annotation>
              <xs:documentation>Requests updates to video tours and active
tours.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="userData" type="xs:boolean" default="false"
minOccurs="0">
            <xs:annotation>

```

```
        <xs:documentation>Requests updates to users.</xs:documentation>
      </xs:annotation>
    </xs:element>
  </xs:all>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:schema>
```