

# **TANDBERG MXP and the Virtual Monitor API**

---

TANDBERG

D13796 Rev. 02

## Table of contents

<b>1</b>	<b>DOCUMENT REVISION HISTORY .....</b>	<b>3</b>
<b>2</b>	<b>OVERVIEW.....</b>	<b>3</b>
<b>3</b>	<b>THE ARCHITECTURE .....</b>	<b>3</b>
3.1	LOCAL INPUTS .....	3
3.2	LOCAL OUTPUTS .....	4
3.3	PICTURES.....	4
3.4	VIRTUAL MONITORS.....	5
3.5	BINDING IT TOGETHER .....	5
<b>4</b>	<b>THE API.....</b>	<b>6</b>
4.1	LOCAL INPUTS .....	6
4.2	PICTURES AND VIRTUAL MONITORS.....	6
4.3	PICTURE PROGRAMS .....	6
4.4	LOCATION OF WINDOWS AND CALLS WITHIN A PICTURE .....	7
4.5	LOCAL OUTPUTS .....	7
<b>5</b>	<b>ADDITIONAL INFORMATION .....</b>	<b>7</b>
5.1	PIP API .....	8
5.2	VIRTUAL MONITOR STATUS .....	8
<b>6</b>	<b>EXAMPLES .....</b>	<b>8</b>
6.1	SHOW SELFVIEW ON BOTH MONITORS .....	8
6.2	SHOW REMOTE SITE ON MAIN MONITOR AND SELFVIEW ON SECOND .....	9
6.3	SHOW ONE REMOTE SITE ON EACH MONITOR .....	9
6.4	SHOW THREE SITES + SELFVIEW IN SIDE-BY-SIDE ON TWO MONITOR.....	10
6.5	SHOW FOUR SITES IN SIDE-BY-SIDE ON TWO MONITOR .....	11
6.6	SHOW FOUR SITES IN PICTURE-OUTSIDE-PICTURE PLUS PRESENTATION.....	12
6.7	XML EXAMPLE OF A PICTUREPROGRAM .....	13

# 1 Document Revision History

Rev 02 – Minor corrections

Rev 01 – Initial revision – Grasp the concept

## 2 Overview

The TANDBERG MXP architecture offers a new set of local video layout possibilities, called Virtual Monitors, which did not exist in former TANDBERG endpoints.

This new software architecture provides more flexibility for integrators, as it enables them to design local video layouts, by utilizing the new API.

Although the API provides more flexibility, it also requires that integrators maintain tight control over the local video layout, because by changing the Virtual Monitors / Local Output configurations, the default endpoint behavior can no longer be guaranteed.

For users not familiar with the new API, the endpoint may appear error prone and unstable. In addition, the endpoint remote control cannot be used. Users and integrators who become familiar with the API will soon see the power of the added flexibility and usability.

The purpose of this document is a short introduction to the architecture and API. Some examples are also given.

It is highly recommended to read the TANDBERG MXP API User Manual as well.

## 3 The Architecture

The architecture consists of four (4) main concepts.

- 1) Local Inputs
- 2) Local Outputs
- 3) Pictures
- 4) Virtual Monitors

It is important to grasp how these concepts interact, as a good understanding will make the API more logical and useable.

### 3.1 Local Inputs

These are the actual inputs on the endpoint. The inputs and back plane may differ depending on the model used.

- 1- Main cam
- 2- Aux cam
- 3- Doc cam
- 4- VCR
- 5- PC DVI-I in
- 6- VNC\*

\* There is no physical VNC input. However, a VNC server may be configured and VNC is therefore treated as a local input (try “xConfiguration VNC ?”).

## 3.2 Local Outputs

These are the actual outputs on the endpoint. The output and back plain may differ depending on the model used.

- 1- Single (S-Video)
- 2- Dual (S-Video)
- 3- Single (Composite)
- 4- Dual (Composite)
- 5- Single (DVI-I out)
- 6- Dual (DVI-I out)

The architecture abstracts the Local Output into four (4) outputs. This is because the Single S-Video and Composite output are paired up, by hardware, to display the same video source, but on different video formats. The same applies for the Dual S-Video and Composite output.

Therefore, the architecture offers these local outputs:

- 1- TV 1
- 2- TV 2
- 3- DVI 1
- 4- DVI 2

## 3.3 Pictures

The notion of “Picture” is a logical abstraction introduced in this architecture. A “Picture” may be generic, and hence it may be composed by other “Pictures”. In general terms, a “Picture” is what you would like to display or show.

The architecture provides a set of basic “Pictures”.

- ❑ LocalMain
- ❑ LocalDuo
- ❑ RemoteMain
- ❑ RemoteDuo
- ❑ JPEG
- ❑ TandbergMonitor1
- ❑ TandbergMonitor2
- ❑ PictureProgram1
- ❑ PictureProgram2
- ❑ PictureProgram3
- ❑ PictureProgram4
- ❑ None

**LocalMain** is the local main source. This picture automatically changes when the main source is changed.

**LocalDuo** is the local duo source. This picture automatically changes when the duo source is changed. This source is also accessible when DuoVideo is inactive or when DuoVideo is opened from a remote site.

**RemoteMain** is the remote main source. In a MultiSite call, several RemoteMain sources co-exist. If nothing else is stated, call identification 1 is assumed to be the correct origin of the remote source. See the interface specification on how to get the correct remote source by adding call identification.

**RemoteDuo** is the remote DuoVideo source. At any given time, only one remote DuoVideo source can exist. Remember to use this picture together with the correct call identification.

**JPEG** is the “logo.jpg” or “company logo” file that exists in the endpoint (TANDBERG logo). This file may be personalized and uploaded to the endpoint through its ftp or web interface.

**TandbergMonitor1** is the default monitor behavior of the ‘Single’ output. This makes it possible for a program to display the dynamic behavior of the endpoint (voice switch, MCU layouts and transitions).

**TandbergMonitor2** is the default monitor behavior of the ‘Dual’ output. Please keep in mind that these layouts display TANDBERG logic. TANDBERG does not guarantee that the exact layout in these pictures will be preserved in different software revisions.

**PictureProgram1, 2, 3 and 4** enables customization of the picture layout.

**None** is Black Screen.

For more information about how to use these pictures, please see the API chapter.

### 3.4 Virtual Monitors

The notion of “Virtual Monitor” is also a logical abstraction introduced in this architecture and is the most central concept introduced in the new architecture.

In short, the “Virtual Monitor” is where “Pictures” are displayed. In the video layout architecture we provide four (4), numbered from 1 to 4, virtual monitors that co-exist, and may be used independently and simultaneously.

### 3.5 Binding it together

Both “Pictures” and “Virtual Monitor” are logical software terms. Hence, there have to be some kind of set-up, to be able to see the “Picture” displayed via a “Virtual Monitor”.

The architecture makes it possible for a “Local Output” to address a “Virtual Monitor”, and use the addressed “Virtual Monitor” as the source for its output. Also, a “Virtual Monitor” may be used as the source for multiple “Local Outputs”. This is in fact the case for the default TANDBERG behavior, where TV 1 and DVI 1, and TV 2 and DVI 2 outputs are paired to respectively display the same “Virtual Monitor”.

The architecture also makes it possible for a “Local Input” to be displayed via two specific pictures (LocalMain and LocalDuo); by making sure that the specific “Local Input” is used as the input source for the LocalMain or the LocalDuo “Picture”.

## 4 The API

The API provides a way to work with the new video layout architecture. This chapter will give examples of how to use the API.

### 4.1 Local Inputs

To define the Local Input to the LocalMain picture, use:

```
xConfiguration MainVideoSource: <1..6>
```

To define the Local Input to the LocalDuo picture, use:

```
xConfiguration DuoVideoSource: <1..6>
```

### 4.2 Pictures and Virtual Monitors

Manipulating “Virtual Monitors” is done by a command, not a configuration. Note that when a “Virtual Monitor” is manipulated via the API, this “Virtual Monitor” may only be controlled via the API. Default endpoint behavior will not have any effect on the “Virtual Monitor”. Hence, the API also provides the possibility to reset the “Virtual Monitor” to default TANDBERG behavior.

Set “Pictures” on a “Virtual Monitor” with:

```
xCommand VirtualMonitorSet VirtualMonitor(r): <1..4>
    Picture(r): <LocalMain/LocalDuo/Still/RemoteMain/RemoteDuo/JPEG/
                TandbergMonitor1/TandbergMonitor2/PictureProgram1/
                PictureProgram2/PictureProgram3/PictureProgram4/None>
    Call: <1..11>
```

Note that this command has a “Call” parameter, which is used to single out a specific RemoteMain or RemoteDuo “Picture”. The Call number is similar to the call number used in “xStatus Call”. In this manner, it is possible to single out participants in a MultiSite conference on different “Virtual Monitors”.

To reset a “Virtual Monitor”, use:

```
xCommand VirtualMonitorReset VirtualMonitor(r): <1..4>
```

### 4.3 Picture Programs

It is possible to create your own layouts by using pictures in different combinations. Currently, it is possible to create four (4) “Picture Programs”.

The “Picture Programs” must be created in advance and is used when initiating a new layout.

To create a “Picture Program”, use:

```
xConfiguration PictureProgram
```

A “Picture Program” consist of three elements; Layout, Picture and Call.

To select the layout for a “Picture Program”, use:

```
xConfiguration PictureProgram [1..4] Layout:
<Full/2Split//4Split/3+1Split/5+1Split>
```

To specify the Pictures within the selected layout, use:

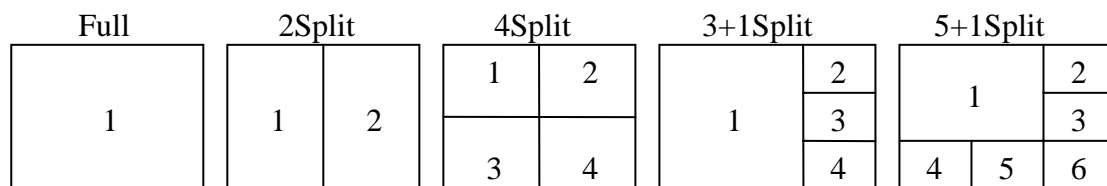
```
xConfiguration PictureProgram [1..4] Window [1..6] Picture:
<LocalMain/LocalDuo/RemoteMain/RemoteDuo/JPEG/TandbergMonitor1/
TandbergMonitor2/None>
```

As an example, if the ‘2Split’ layout is selected, Windows 1 and 2 must be configured with a Picture. In addition, if one or more Windows should show a remote site, which a call identification is needed.

To set the Call identification in accordance with a Window, use:

```
xConfiguration PictureProgram [1..4] Window [1..6] Call: <1..11>
```

## 4.4 Location of Windows and Calls within a Picture



## 4.5 Local Outputs

Now, make sure that the “Virtual Monitor” is displayed on the correct output.

This is controlled by the “xConfiguration Video Outputs TV” and “xConfiguration Video Outputs DVI” configurations.

To set up the TV output, use:

```
xConfiguration Video Outputs TV [1..2] VirtualMonitor: <1..4>
```

To set up the DVI output, use:

```
xConfiguration Video Outputs DVI [1..2] VirtualMonitor: <1..4>
```

**Note!** It does not matter if a “Virtual Monitor” is set before or after the “Local Output” is configured. The result will be the same.

## 5 Additional Information

“Pictures” are displayed on “Virtual Monitors” as full screen pictures. However, “Virtual Monitors” may also have Pictures in PIP’s.

## 5.1 PIP API

PIP, also known as Picture-In-Picture, is set on “Virtual Monitors” in a manner similar to setting “Pictures” on “Virtual Monitors”.

Setting a PIP is done by using the command:

```
xCommand PIPShow VirtualMonitor(r): <1..4>
    Picture(r): <LocalMain/LocalDuo/RemoteMain/RemoteDuo/JPEG/
                TandbergMonitor1/TandbergMonitor2/None>
    Call: <1..11>
    Position: <BottomLeft/BottomRight/TopLeft/TopRight>
```

Removing a PIP is done by the command:

```
xCommand PIPHide VirtualMonitor(r): <1..4>
```

Note! A “Virtual Monitor” currently only supports one PIP. There may however be one PIP on all “Virtual Monitors” at the same time.

## 5.2 Virtual Monitor Status

Integrators that use the new API should maintain tight control with the configuration of “Virtual Monitors”. Therefore, the API also provides a controlling and monitoring tool.

To receive current status of the “Virtual Monitors”, use

```
xStatus VirtualMonitor <1..4>
```

## 6 Examples

Now that the Virtual Monitor API concept is explained, it is time for some examples.

### 6.1 Show selfview on both monitors

To show the selfview of the local Main Camera on the two monitors connected to DVI 1 and DVI 2 output, the following procedure should be followed:

1. First configure the Main Camera (Local Input 1) to be the Main Video Source.
2. Then Virtual Monitor 1 and 2 must be set up to show the “LocalMain” picture.
3. Finally, the Local Output DVI 1 and DVI 2 must be configured to address Virtual Monitor 1 and 2, respectively.



To proceed, please type the following commands in sequence through the API:

```
xConfiguration MainVideoSource: 1
```

```
xCommand VirtualMonitorSet VirtualMonitor: 1 Picture: LocalMain
```

```
xCommand VirtualMonitorSet VirtualMonitor: 2 Picture: LocalMain
```

```
xConfiguration Video Outputs DVI 1 VirtualMonitor: 1
```



xConfiguration Video Outputs DVI 2 VirtualMonitor: 2

## 6.2 Show remote site on main monitor and selfview on second

To show the remote site on one monitor and the local selfview on the other monitor, the following procedure should be followed:

1. First configure the Main Camera (Local Input 1) to be the Main Video Source.
2. Then the Virtual Monitor 1 must be set up to show the “RemoteMain” picture and Virtual Monitor 2 must be set up to show the “LocalMain” picture.
3. Finally, the Local Output DVI 1 and DVI 2 must be configured to address Virtual Monitor 1 and 2, respectively.



To proceed, please type the following commands in sequence through the API:

```
xConfiguration MainVideoSource: 1
```

```
xCommand VirtualMonitorSet VirtualMonitor: 1 Picture: RemoteMain Call: 1
```

```
xCommand VirtualMonitorSet VirtualMonitor: 2 Picture: LocalMain
```

```
xConfiguration Video Outputs DVI 1 VirtualMonitor: 1
```

```
xConfiguration Video Outputs DVI 2 VirtualMonitor: 2
```

## 6.3 Show one remote site on each monitor

To show one remote site on each monitor, the following procedure should be followed:

1. First configure the Main Camera (Local Input 1) to be the Main Video Source.
2. Then the Virtual Monitor 1 must be set up to show the “RemoteMain” picture from site 1 and Virtual Monitor 2 must be set up to show the “RemoteMain” picture from site 2.
3. Finally, the Local Output DVI 1 and DVI 2 must be configured to address Virtual Monitor 1 and 2, respectively.



To proceed, please type the following commands in sequence through the API:

```
xConfiguration MainVideoSource: 1
```

```
xCommand VirtualMonitorSet VirtualMonitor: 1 Picture: RemoteMain Call: 1
```

```
xCommand VirtualMonitorSet VirtualMonitor: 2 Picture: RemoteMain Call: 2
```

```
xConfiguration Video Outputs DVI 1 VirtualMonitor: 1
```

```
xConfiguration Video Outputs DVI 2 VirtualMonitor: 2
```

## 6.4 Show three sites + selfview in Side-by-Side on two monitor

To show two sites in Side-by-Side on each monitor, the following procedure should be followed:

1. First configure the Main Camera (Local Input 1) to be the Main Video Source.
2. Secondly, configure the "PictureProgram1" to show site 1 and 3 in a Side-by-Side layout and "PictureProgram2" to show site 2 and "LocalMain" picture in a Side-by-Side layout.
3. Then Virtual Monitor 1 must be set up to show "PictureProgram1" and Virtual Monitor 2 must be set up to show "PictureProgram2".
4. Finally, the Local Output DVI 1 and DVI 2 must be configured to address Virtual Monitor 1 and 2, respectively.



To proceed, please type the following commands in sequence through the API:

```
xConfiguration MainVideoSource: 1
xConfiguration PictureProgram 1 Layout: 2Split
xConfiguration PictureProgram 1 Window 1 Picture: RemoteMain
xConfiguration PictureProgram 1 Window 1 Call: 1
xConfiguration PictureProgram 1 Window 2 Picture: RemoteMain
xConfiguration PictureProgram 1 Window 2 Call: 3
xConfiguration PictureProgram 1 Window 3 Picture: None
xConfiguration PictureProgram 1 Window 3 Call: 1
xConfiguration PictureProgram 1 Window 4 Picture: None
xConfiguration PictureProgram 1 Window 4 Call: 1
xConfiguration PictureProgram 1 Window 5 Picture: None
xConfiguration PictureProgram 1 Window 5 Call: 1
xConfiguration PictureProgram 1 Window 6 Picture: None
xConfiguration PictureProgram 1 Window 6 Call: 1
xConfiguration PictureProgram 2 Layout: 2Split
xConfiguration PictureProgram 2 Window 1 Picture: RemoteMain
xConfiguration PictureProgram 2 Window 1 Call: 2
xConfiguration PictureProgram 2 Window 2 Picture: LocalMain
xConfiguration PictureProgram 2 Window 2 Call: 1
xConfiguration PictureProgram 2 Window 3 Picture: None
xConfiguration PictureProgram 2 Window 3 Call: 1
xConfiguration PictureProgram 2 Window 4 Picture: None
xConfiguration PictureProgram 2 Window 4 Call: 1
xConfiguration PictureProgram 2 Window 5 Picture: None
```

```

xConfiguration PictureProgram 2 Window 5 Call: 1
xConfiguration PictureProgram 2 Window 6 Picture: None
xConfiguration PictureProgram 2 Window 6 Call: 1
xCommand VirtualMonitorSet VirtualMonitor: 1 Picture: PictureProgram1
xCommand VirtualMonitorSet VirtualMonitor: 2 Picture: PictureProgram2
xConfiguration Video Outputs DVI 1 VirtualMonitor: 1
xConfiguration Video Outputs DVI 2 VirtualMonitor: 2

```

## 6.5 Show four sites in Side-by-Side on two monitor

To show two sites in Side-by-Side on each monitor, the following procedure should be followed:

1. First configure the Main Camera (Local Input 1) to be the Main Video Source.
2. Secondly, configure the “PictureProgram1” to show site 1 and 3 in a Side-by-Side layout and “PictureProgram2” to show site 2 and 4 in a Side-by-Side layout.
3. Then Virtual Monitor 1 must be set up to show “PictureProgram1” and Virtual Monitor 2 must be set up to show “PictureProgram2”.
4. Finally, the Local Output DVI 1 and DVI 2 must be configured to address Virtual Monitor 1 and 2, respectively.



To proceed, please type the following commands in sequence through the API:

```

xConfiguration MainVideoSource: 1
xConfiguration PictureProgram 1 Layout: 2Split
xConfiguration PictureProgram 1 Window 1 Picture: RemoteMain
xConfiguration PictureProgram 1 Window 1 Call: 1
xConfiguration PictureProgram 1 Window 2 Picture: RemoteMain
xConfiguration PictureProgram 1 Window 2 Call: 3
xConfiguration PictureProgram 1 Window 3 Picture: None
xConfiguration PictureProgram 1 Window 3 Call: 1
xConfiguration PictureProgram 1 Window 4 Picture: None
xConfiguration PictureProgram 1 Window 4 Call: 1
xConfiguration PictureProgram 1 Window 5 Picture: None
xConfiguration PictureProgram 1 Window 5 Call: 1
xConfiguration PictureProgram 1 Window 6 Picture: None
xConfiguration PictureProgram 1 Window 6 Call: 1
xConfiguration PictureProgram 2 Layout: 2Split
xConfiguration PictureProgram 2 Window 1 Picture: RemoteMain
xConfiguration PictureProgram 2 Window 1 Call: 2

```

```

xConfiguration PictureProgram 2 Window 2 Picture: RemoteMain
xConfiguration PictureProgram 2 Window 2 Call: 4
xConfiguration PictureProgram 2 Window 3 Picture: None
xConfiguration PictureProgram 2 Window 3 Call: 1
xConfiguration PictureProgram 2 Window 4 Picture: None
xConfiguration PictureProgram 2 Window 4 Call: 1
xConfiguration PictureProgram 2 Window 5 Picture: None
xConfiguration PictureProgram 2 Window 5 Call: 1
xConfiguration PictureProgram 2 Window 6 Picture: None
xConfiguration PictureProgram 2 Window 6 Call: 1
xCommand VirtualMonitorSet VirtualMonitor: 1 Picture: PictureProgram1
xCommand VirtualMonitorSet VirtualMonitor: 2 Picture: PictureProgram2
xConfiguration Video Outputs DVI 1 VirtualMonitor: 1
xConfiguration Video Outputs DVI 2 VirtualMonitor: 2

```

## 6.6 Show four sites in Picture-Outside-Picture plus Presentation

To show four sites in Picture-Outside-Picture on one monitor and a Presentation on the other monitor, the following procedure should be followed:

1. First configure the Main Camera (Local Input 1) to be the Main Video Source and PC (Local Input 5) to be the Duo Video Source.
2. Secondly, configure the “PictureProgram1” to show a Picture-Outside-Picture layout with four sites in.
3. Then Virtual Monitor 1 must be set up to show “PictureProgram1” and Virtual Monitor 2 must be set up to show “LocalDuo”.
4. Finally, the Local Output DVI 1 and DVI 2 must be configured to address Virtual Monitor 1 and 2, respectively.



To proceed, please type the following commands in sequence through the API:

```

xConfiguration MainVideoSource: 1
xConfiguration DuoVideoSource: 5
xConfiguration PictureProgram 1 Layout: 3+1Split
xConfiguration PictureProgram 1 Window 1 Picture: RemoteMain
xConfiguration PictureProgram 1 Window 1 Call: 1
xConfiguration PictureProgram 1 Window 2 Picture: RemoteMain
xConfiguration PictureProgram 1 Window 2 Call: 2
xConfiguration PictureProgram 1 Window 3 Picture: RemoteMain
xConfiguration PictureProgram 1 Window 3 Call: 3
xConfiguration PictureProgram 1 Window 4 Picture: RemoteMain

```

```
xConfiguration PictureProgram 1 Window 4 Call: 4
xConfiguration PictureProgram 1 Window 5 Picture: None
xConfiguration PictureProgram 1 Window 5 Call: 1
xConfiguration PictureProgram 1 Window 6 Picture: None
xConfiguration PictureProgram 1 Window 6 Call: 1
xCommand DuoVideoStart VideoSource: 5
xCommand VirtualMonitorSet VirtualMonitor: 1 Picture: PictureProgram1
xCommand VirtualMonitorSet VirtualMonitor: 2 Picture: LocalDuo
xConfiguration Video Outputs DVI 1 VirtualMonitor: 1
xConfiguration Video Outputs DVI 2 VirtualMonitor: 2
```

## 6.7 XML example of a PictureProgram

Short example of how to configure PictureProgram1 for a 3+1Split Layout:

```
<Configuration xmlns="http://www.tandberg.no/XML/CUIL/1.0">
  <PictureProgram item="1">
    <Layout item="1">3+1Split</Layout>
    <Window item="1">
      <Picture item="1">RemoteMain</Picture>
      <Call item="1">1</Call>
    </Window>
    <Window item="2">
      <Picture item="1">RemoteMain</Picture>
      <Call item="1">2</Call>
    </Window>
    <Window item="3">
      <Picture item="1">RemoteMain</Picture>
      <Call item="1">3</Call>
    </Window>
    <Window item="4">
      <Picture item="1">RemoteMain</Picture>
      <Call item="1">4</Call>
    </Window>
    <Window item="5">
      <Picture item="1">None</Picture>
      <Call item="1">1</Call>
    </Window>
    <Window item="6">
      <Picture item="1">None</Picture>
      <Call item="1">1</Call>
    </Window>
  </PictureProgram>
</Configuration>
```