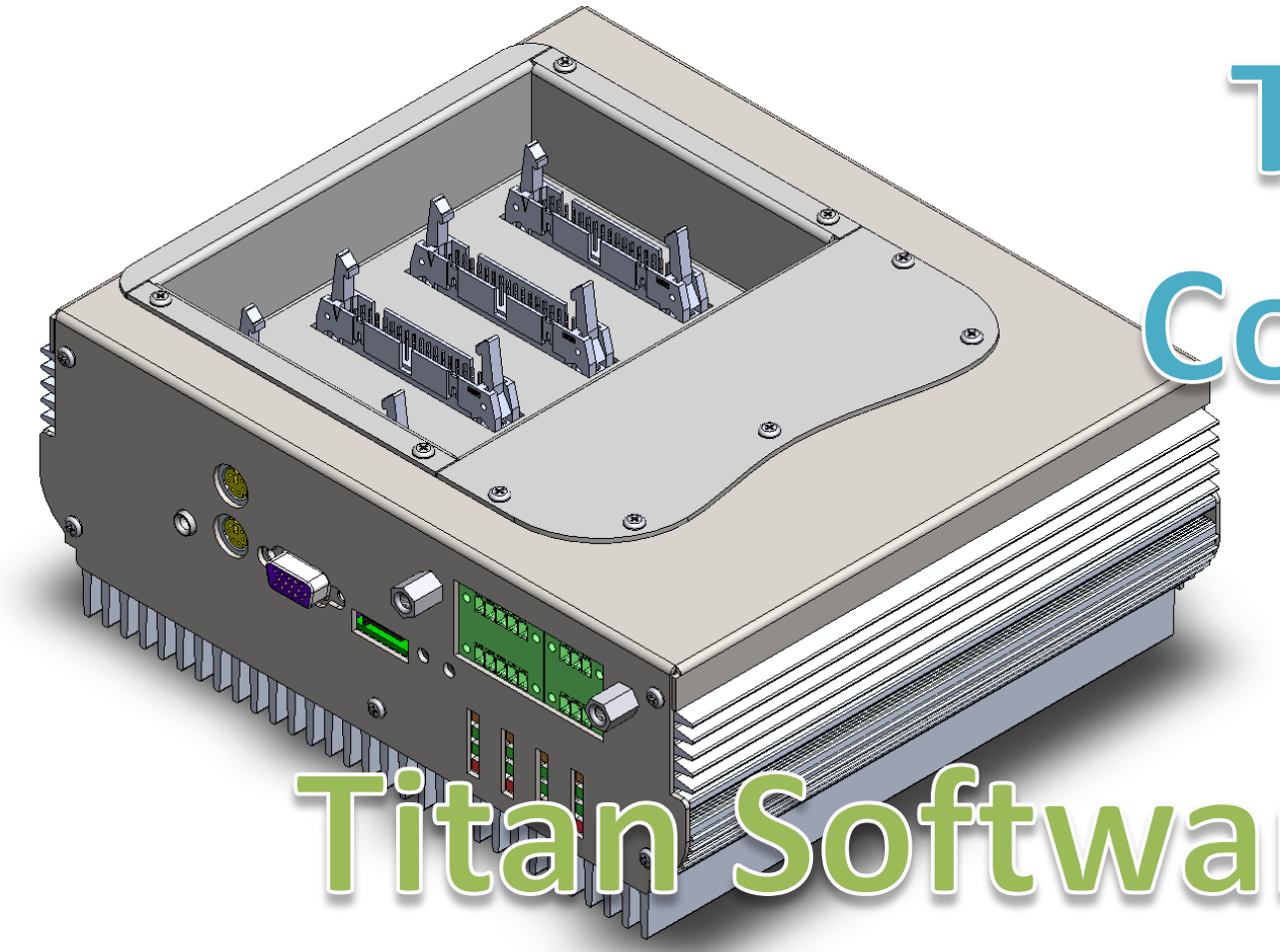
ADAPTIVE

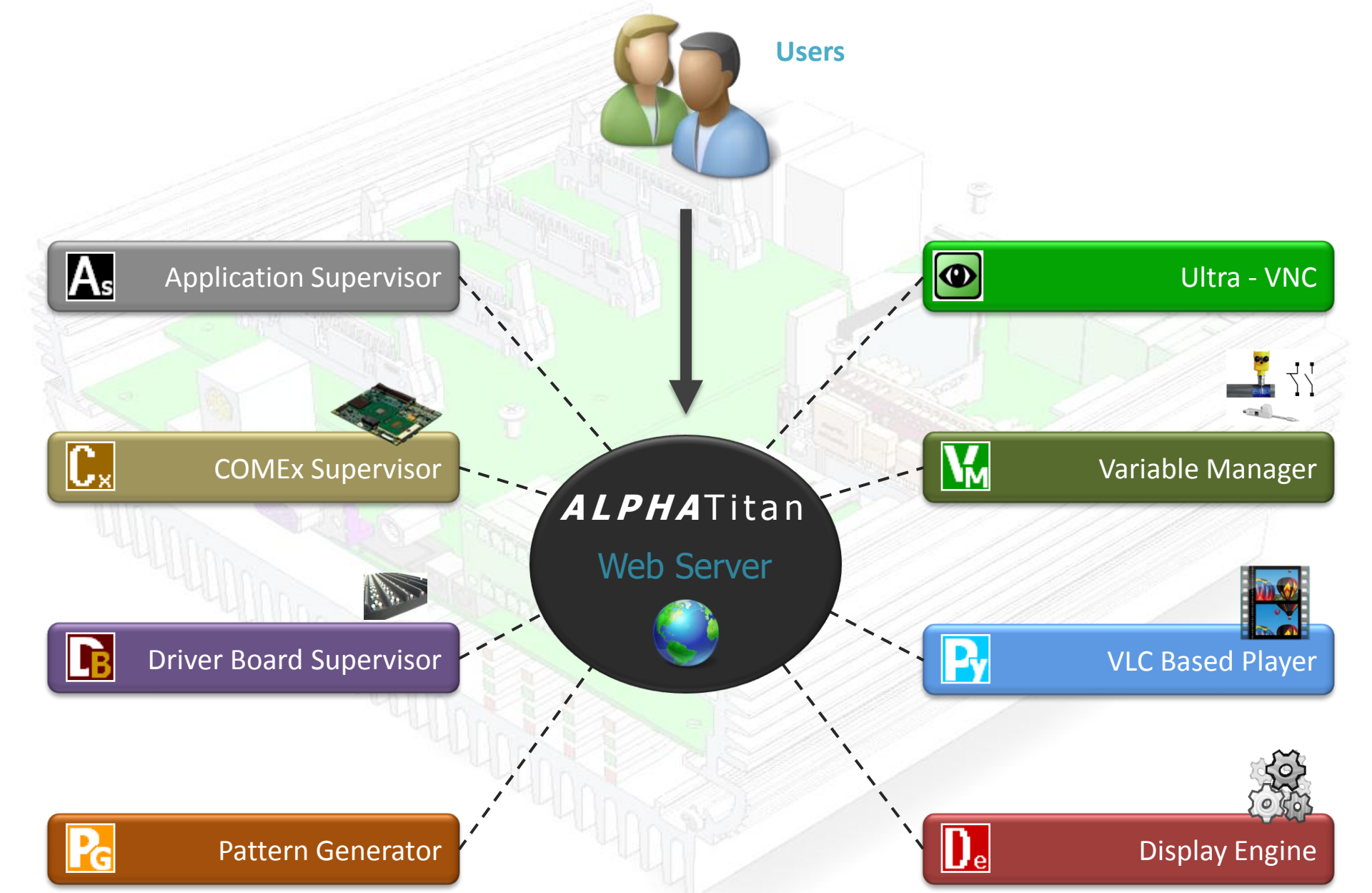
Adaptive Micro Systems Europe

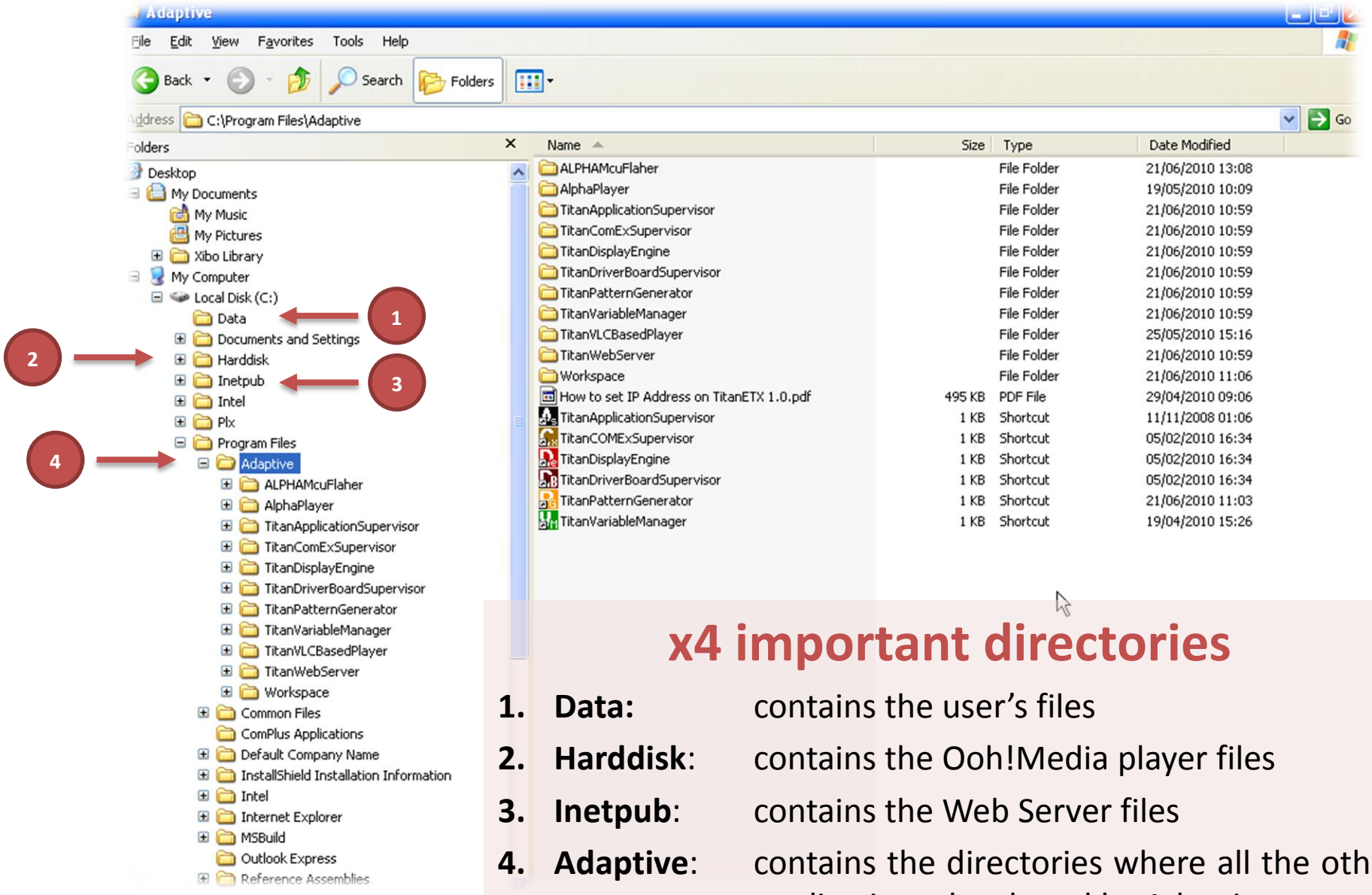


TitanETX Controller

Titan Software Suite

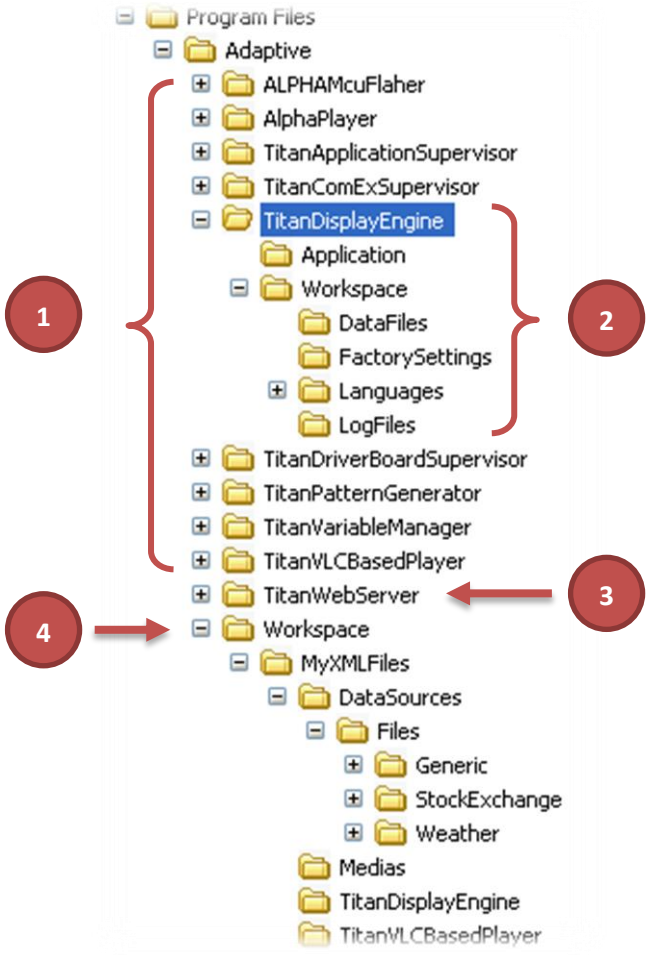
- **Windows XP embedded** operating system 
- Enhanced Write Filter (**EWF**) implemented in order to keep the file system and the OS integrity
- Framework .Net 3.0
- Modular & Opened software architecture
- Inter process communication made via **TCP/IP sockets**
- Possibility to integrate within the existing architecture **customer's applications** for specific requirements & needs
- HMI (multi languages) based on an embedded **Web Server**
- Remote access to the controller through the 3rd party application **Ultra-VNC**
- TitanETX Controller is shipped with a bunch of applications pre-installed on the hard disk


Users
Application Supervisor
Ultra - VNC
COMEx Supervisor
Variable Manager
ALPHATitan
Web Server
Driver Board Supervisor
VLC Based Player
Pattern Generator
Display Engine



x4 important directories

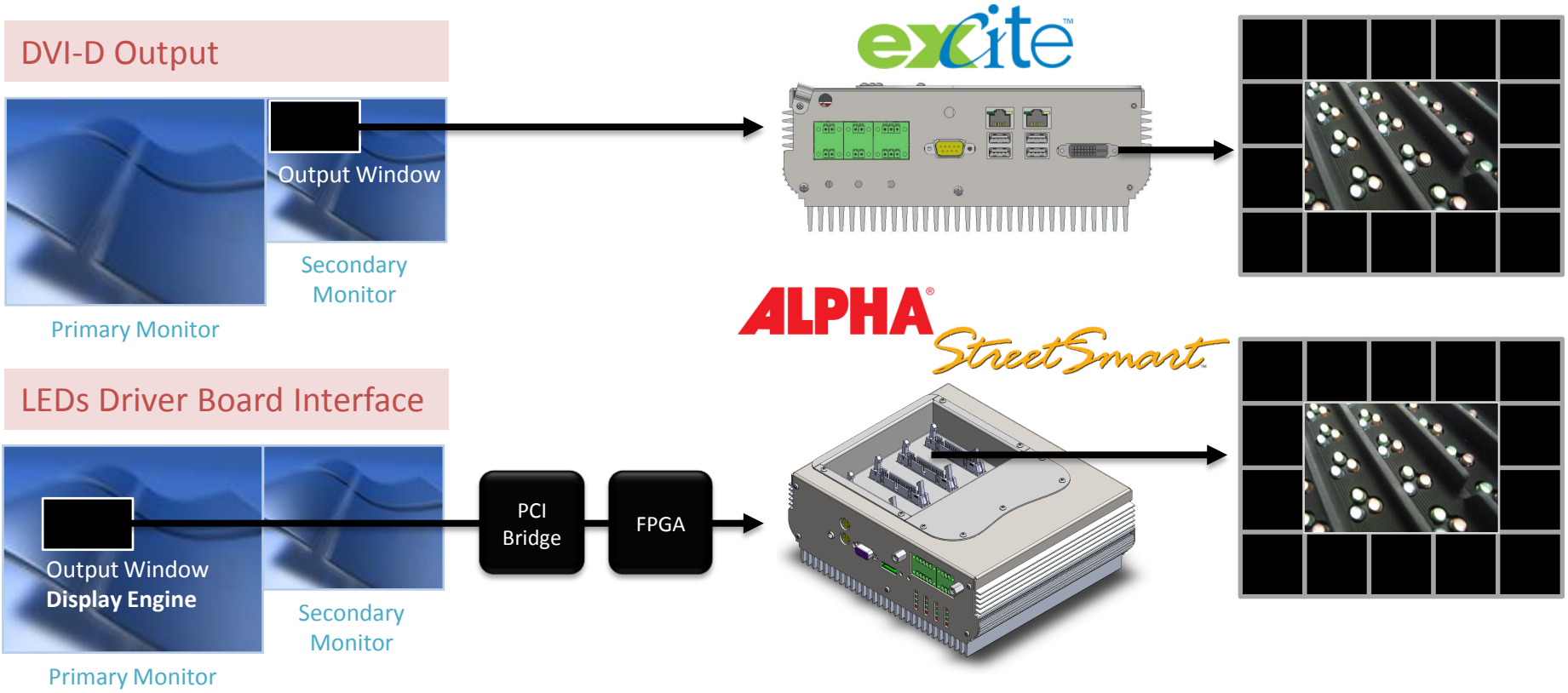
1. **Data:** contains the user's files
2. **Harddisk**: contains the Ooh!Media player files
3. **Inetpub**: contains the Web Server files
4. **Adaptive**: contains the directories where all the other applications developed by Adaptive are installed



1. Directories where the applications developed by Adaptive are installed
2. Each one of these directories has the same file tree architecture
   - Application : contains the .exe, .dll, .ini, ... files
   - Workspace: contains the language, log, ... files
3. Contains the Web Server workspace
4. Workspace shared between all applications

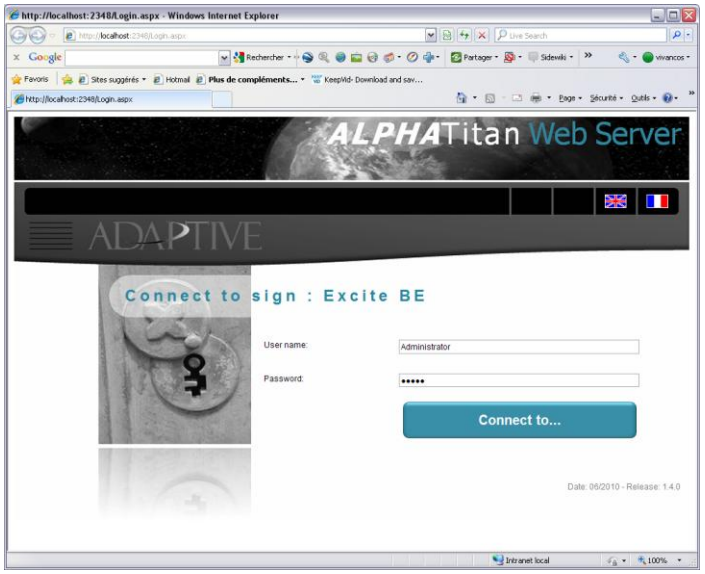
- **The Data and the Adaptive directories are the only ones which are unprotected (Read/Write) by the OS. Those are the only places where the user can copy his own applications**
- **Call the Adaptive's Tech Support to know how to unprotect the rest of the system if needed**

- The **Output Window** contains the data (RGB pixel buffer) sent to either the **DVI-D** output (Excite) or the **LEDs Driver Board Interfaces** (ALPHA & StreetSmart)
- When the **DVI-D output** is used, the Output Window must be always located to the top-left corner of the extended desktop
- When **LEDs Driver Board Interfaces** are used, the Output Window is managed by the **Display Engine** which sends the data to the onboard hardware (PCI Bridge + FPGA)



- The **embedded Web Server** provides the user interface to configure and to control all applications running on the controller

Address : http://192.168.1.150/ALPHATitanWebServer/login.aspx
User name: Administrator
Password : admin

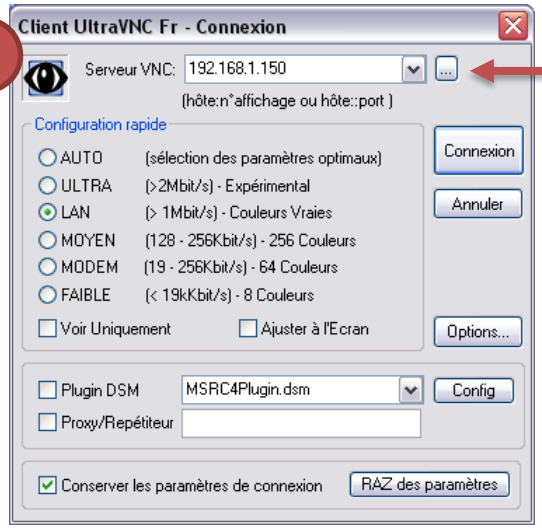


Login page

Home page

Ultra-VNC

- The **3rd party application Ultra-VNC** is used to remote access to the controller in order to perform some specific actions such as the configuration, downloading new files, etc.
- Install the client application on your PC



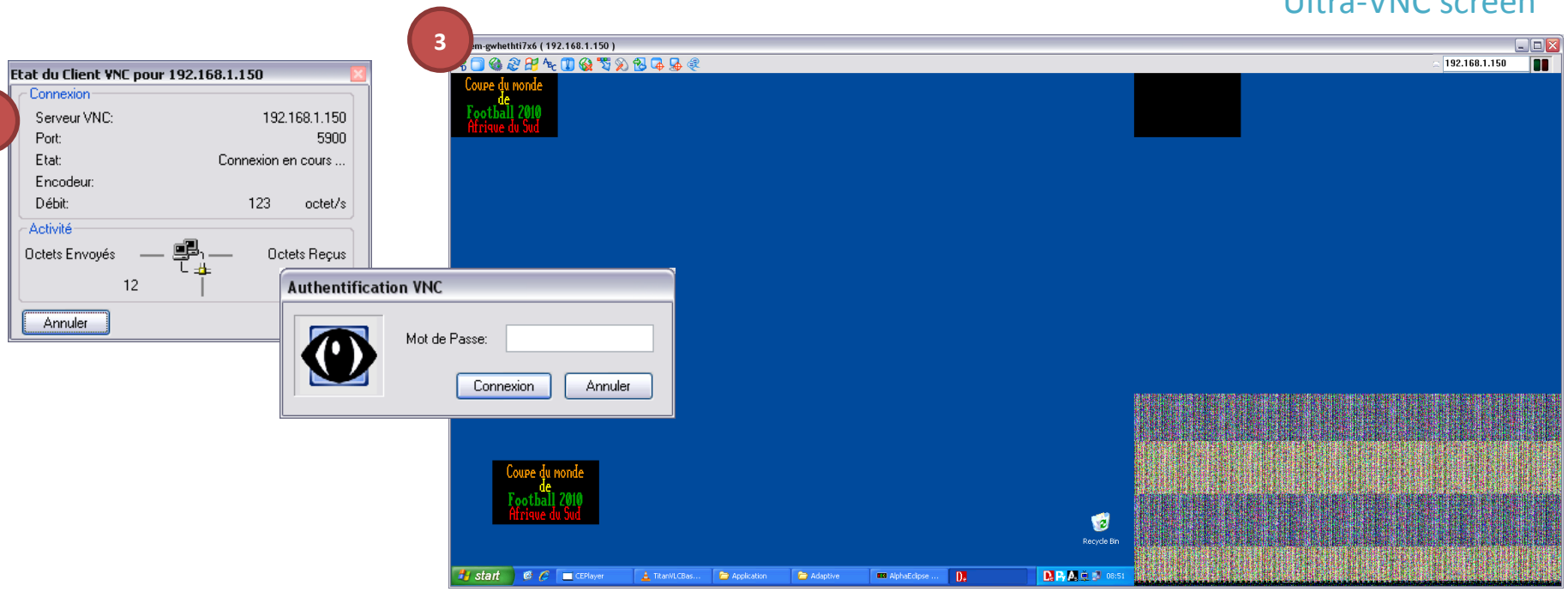
Factory configuration:

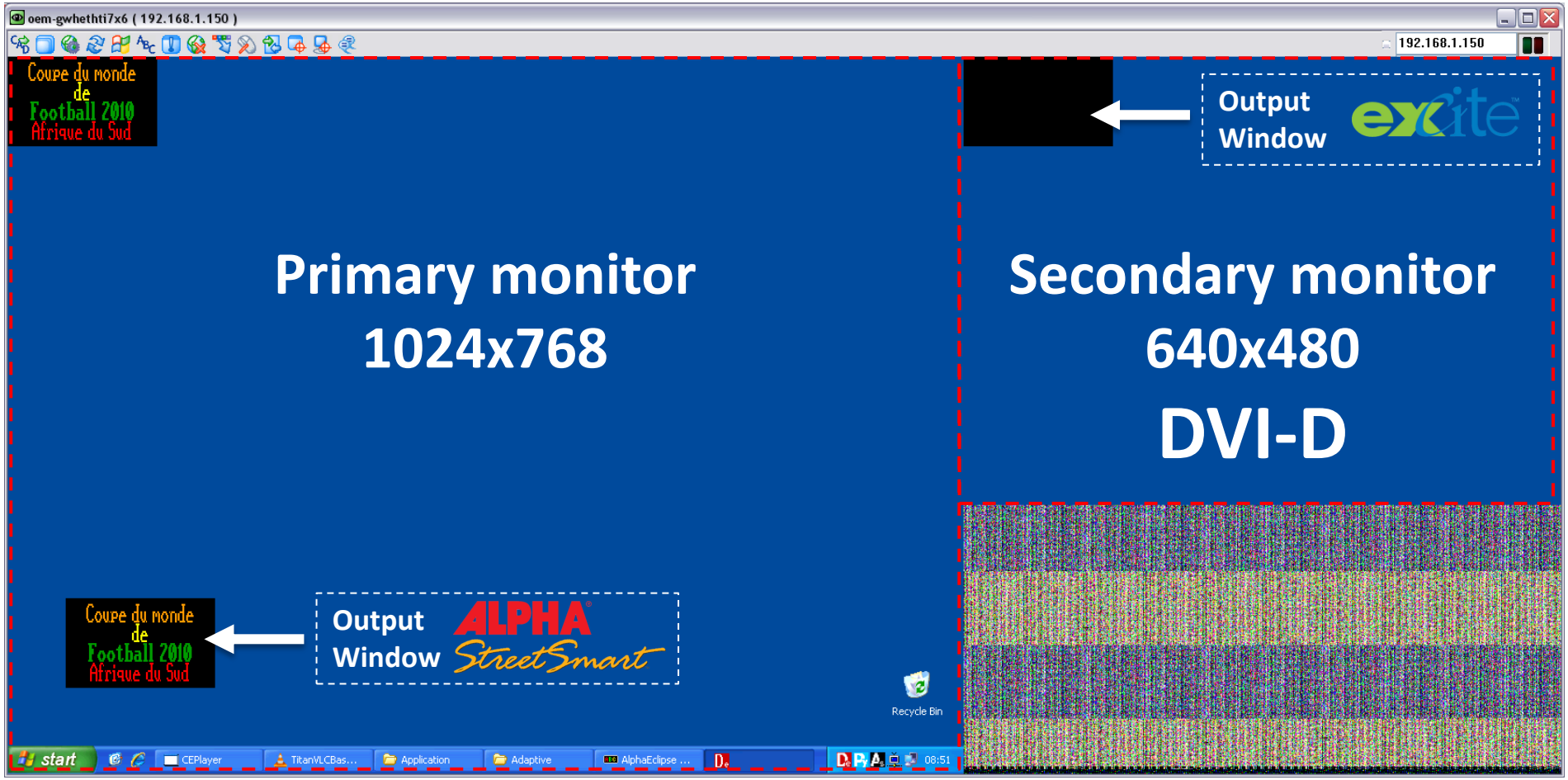
IP address : **192.168.1.150**

Subnet mask: **255.255.255.0**

Password : **admin**

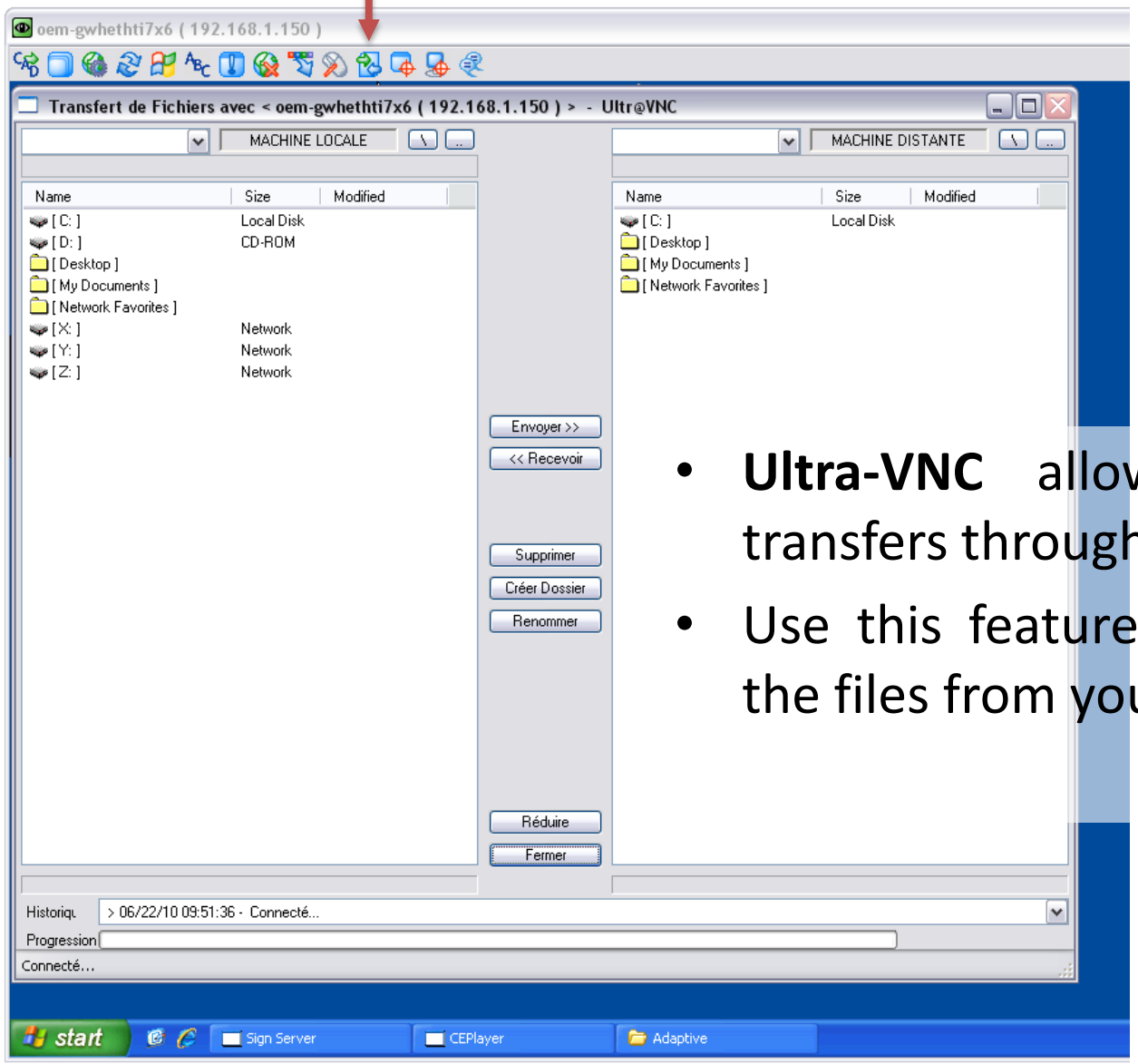
Ultra-VNC screen





- The **Ultra-VNC** window allows to see both the primary and the secondary monitor of the controller
- When the **DVI output** is used (Excite only) the output window must be located on the top-left corner of the secondary monitor
- When **LEDs Driver Board Interfaces** are used (ALPHA & StreeSmart) the output window can be located anywhere on the screens

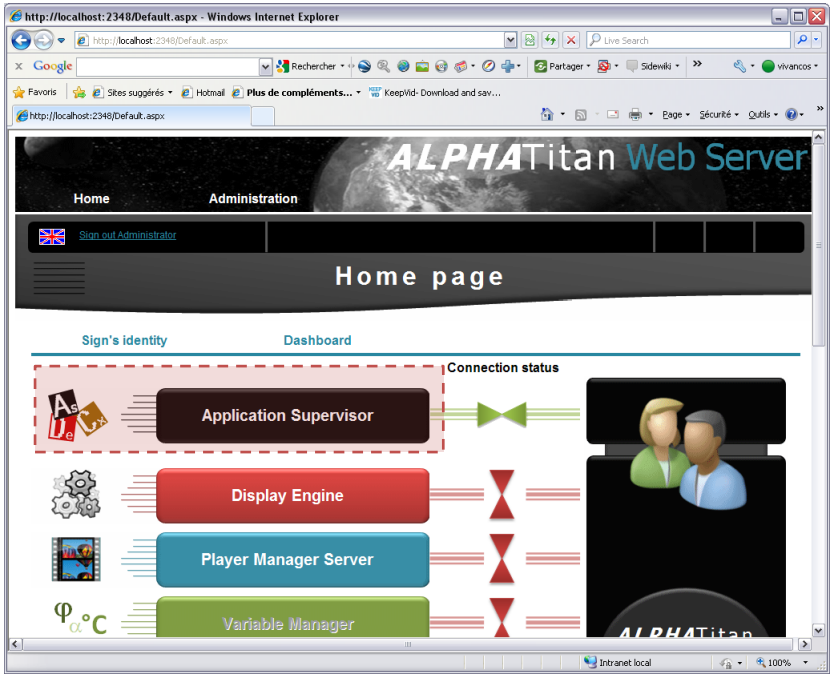
Open the FTP Window



- **Ultra-VNC** allows to perform file transfers through FTP
- Use this feature to download/upload the files from your PC to the controller

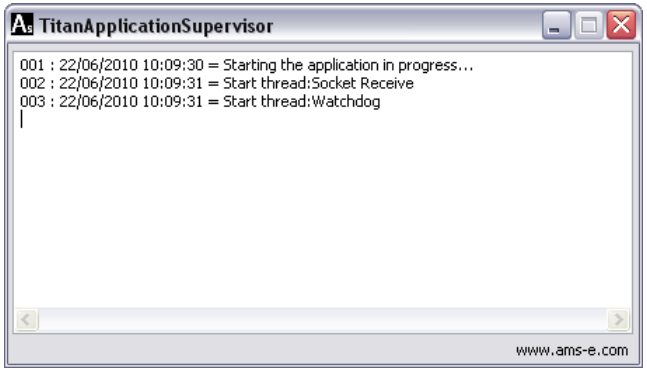
Application Supervisor

- The purpose of the **Application Supervisor** is to manage and to control all the other applications
- It allows:
  - To automatically start a list of applications
  - To position a player to a desired location over the Output Window on the screen
  - To cyclically check if a specific application is still alive (watchdog)
  - To close all the started applications in one click only
- It is configured through the **Web Server**
- Its settings are saved to the file: ApplicationSupervisorINI.xml



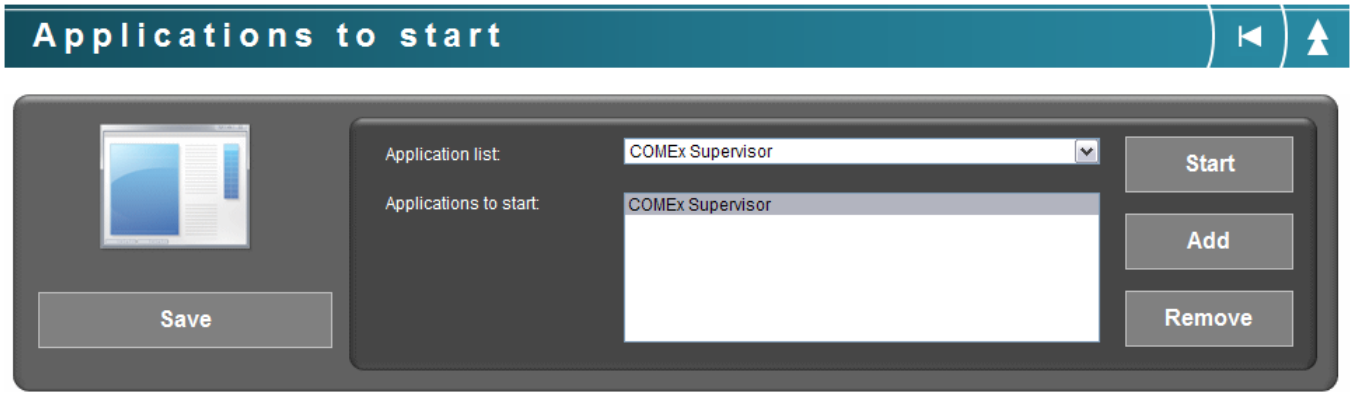
Web Server – Client machine

TCP/IP Commands

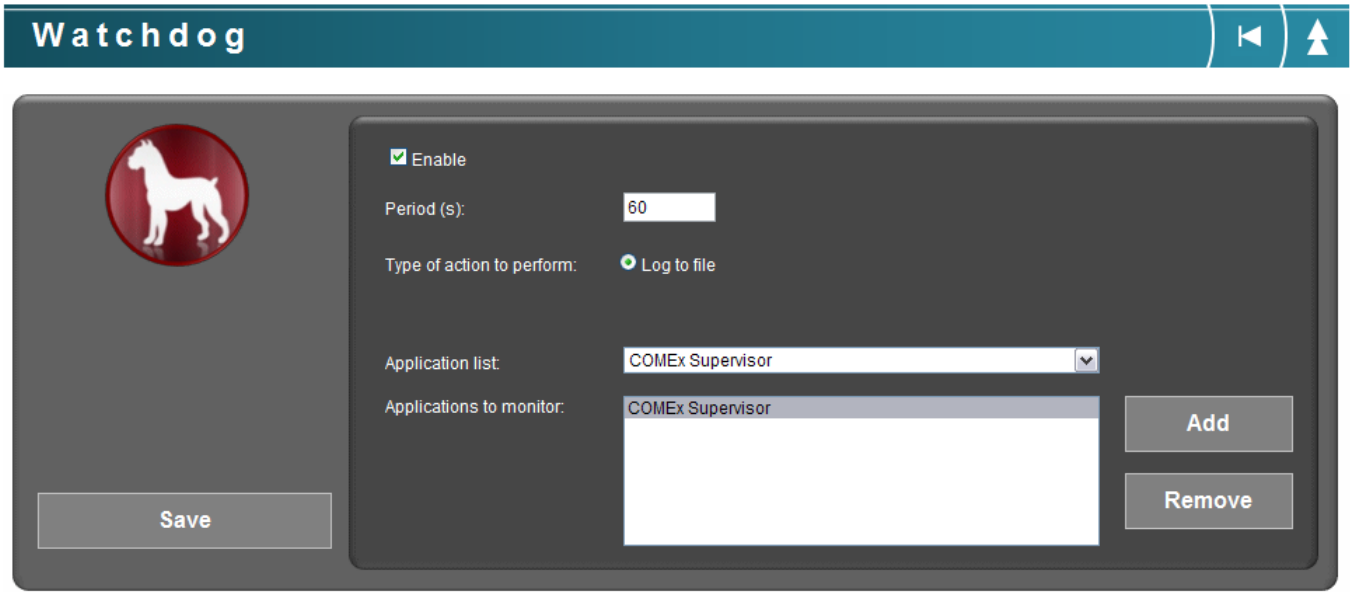


Application Supervisor - Controller

- An application can be manually started from the Web Server
  - Select the application's name from the combo list
  - Click on "Start"
- An application can be automatically started when the **Application Supervisor** is launched
  - Select the application's name from the combo list
  - Add it to the list
  - Save the new configuration



- An application can be cyclically checked to make sure that it is still alive. If not, a warning is traced into the log file ApplicationSupervisor_SW.log
  - Enable the watchdog
  - Set the desired period
  - Select the application's name from the combo list
  - Add it to the list
  - Save the new configuration



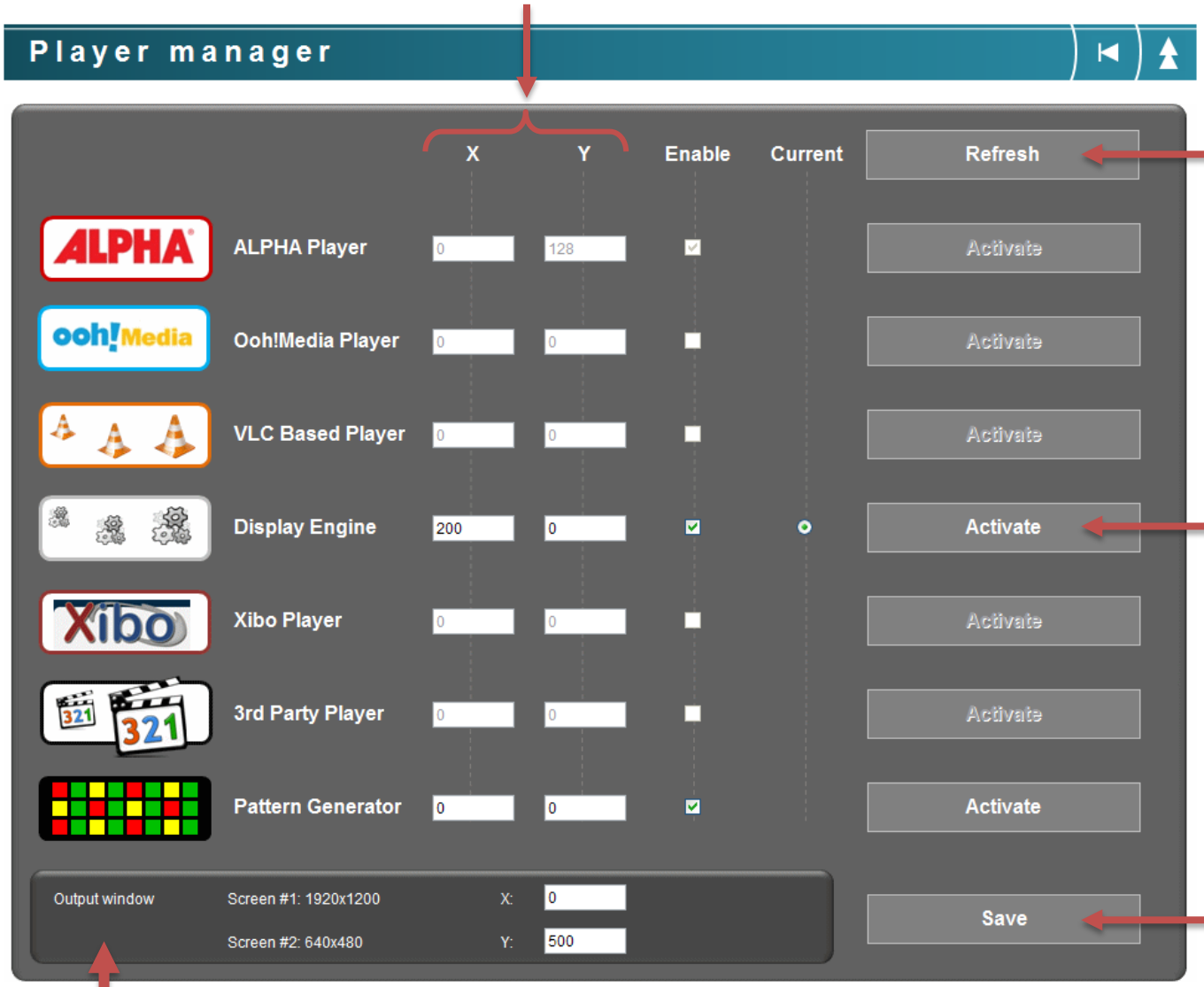

```
2/06/2010 10:48:27 = STS - Start the application
2/06/2010 10:48:27 = STS - CApplicationSupervisorApp::InitInstance-> Init g_oApplicationSupervisorINI called
2/06/2010 10:48:27 = STS - CApplicationSupervisorApp::InitInstance-> Init g_oLanguage called
2/06/2010 10:48:28 = STS - CAMSSocketReceiveProcess::PreRunCommands(...)-> called
2/06/2010 10:48:28 = STS - CAMSSocketReceiveProcess::PreRunCommands(...)-> succeeded
2/06/2010 10:48:33 = STS - Minimize the application
2/06/2010 10:48:38 = STS - CAMSWatchdogProcess::SocketFIFOProcessCommands(...)-> APSP_WATCHDOG_PARAMS
2/06/2010 10:48:38 = STS - CAMSWatchdogProcess::SocketFIFOProcessCommands(...)-> APSP_WATCHDOG_APP_TO_MONITOR_LIST
2/06/2010 10:49:33 = ERR - CAMSWatchdogProcess::ProcessCheckWatchdog(...)-> The following application doesn't answer: COMEx Supervisor
2/06/2010 10:50:38 = ERR - CAMSWatchdogProcess::ProcessCheckWatchdog(...)-> The following application doesn't answer: COMEx Supervisor
```

Log file Warning

- The controller is able to run several Players at the same time on the screens
- Adaptive provides and installs a set of Players:
  - The **ALPHA Player** plays messages (simple texts, variables, graphics & animations) created with ALPHANet Pro
  - The **Ooh!Media Player** plays messages (enhanced texts, graphics & animations) created with Ooh!Media
  - The **VLC Based Player** plays any kind of video medias created from a 3rd party application
  - The **Display Engine** captures multiples regions of the video screen and arranges them of a specific layout
  - A **3rd party Player** such as Window Media Player, Flash Player, VLC, Customer Player, etc.
  - The **Pattern Generator** plays patterns in order to perform diagnostic operations

- The user can quickly select and position his desired player on the **Output Window** (DVI-D vs LEDs Driver Boar interfaces) via the Web Server

X,Y coordinates of the Player when it is not activated



Refresh the list of all Players currently running on the controller

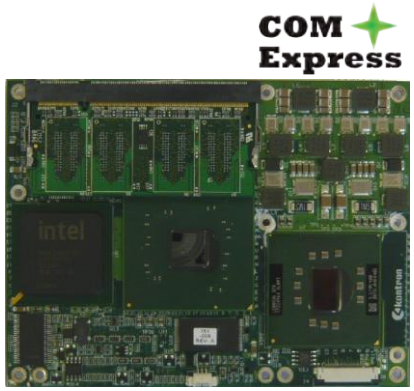
Activate the selected Player

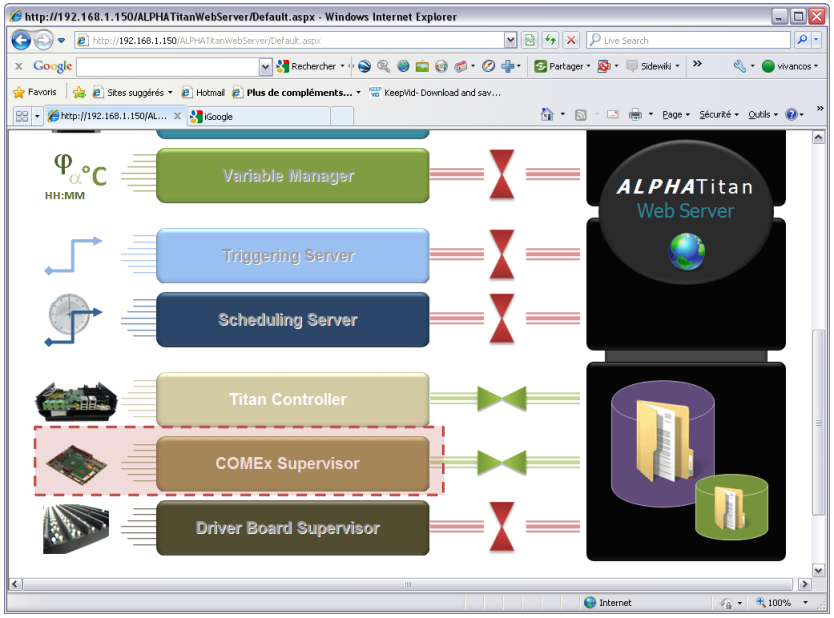
Save the current settings to the file

X,Y coordinates of the **Output Window** on the primary or secondary monitor

COMEx Supervisor

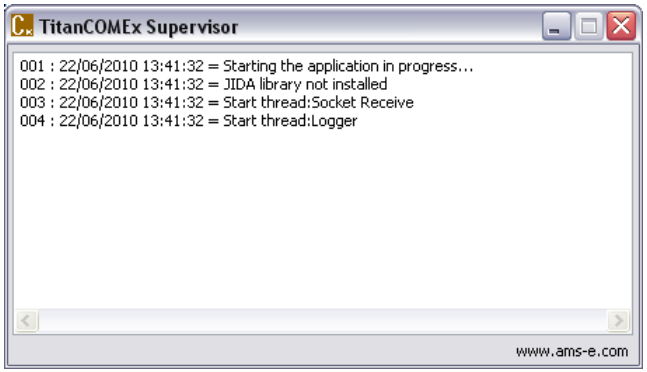
- The purpose of the **COMEx Supervisor** is to monitor the behavior of the COMExpress industrial PC module bought from the selves
- It allows:
  - To get the module information
  - To read the internal temperatures of the processor and the chipset
  - To read the onboard voltage statuses
  - To enable the information to be logged in a file every day
- It is configured through the **Web Server**
- Its settings are saved to the file: COMExSupervisorINI.xml



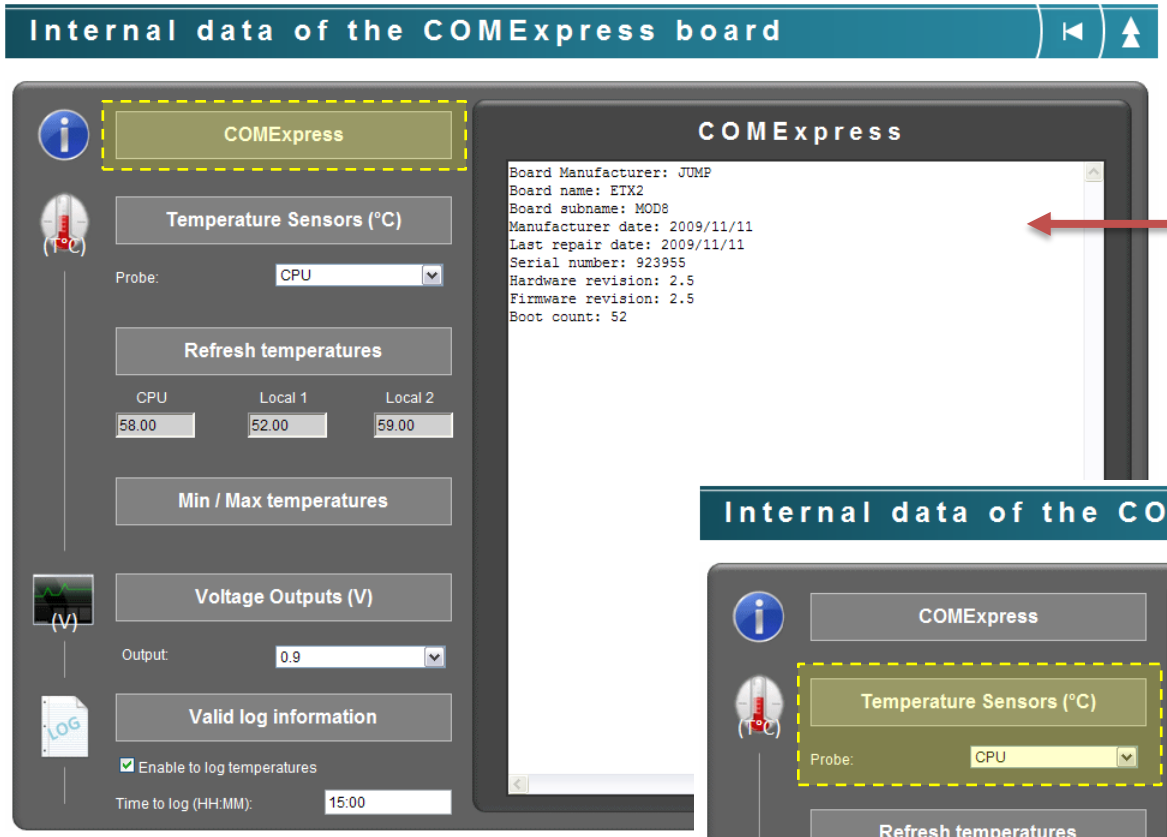


Web Server – Client machine

TCP/IP Commands

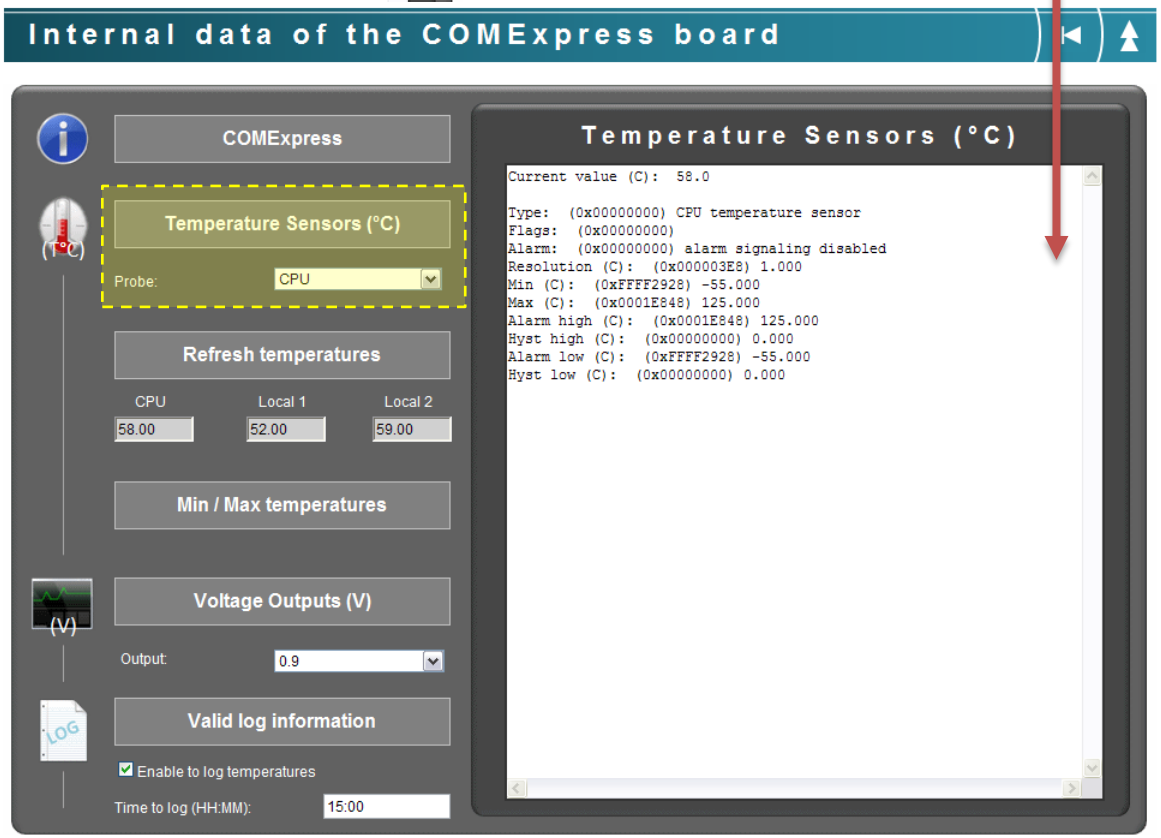


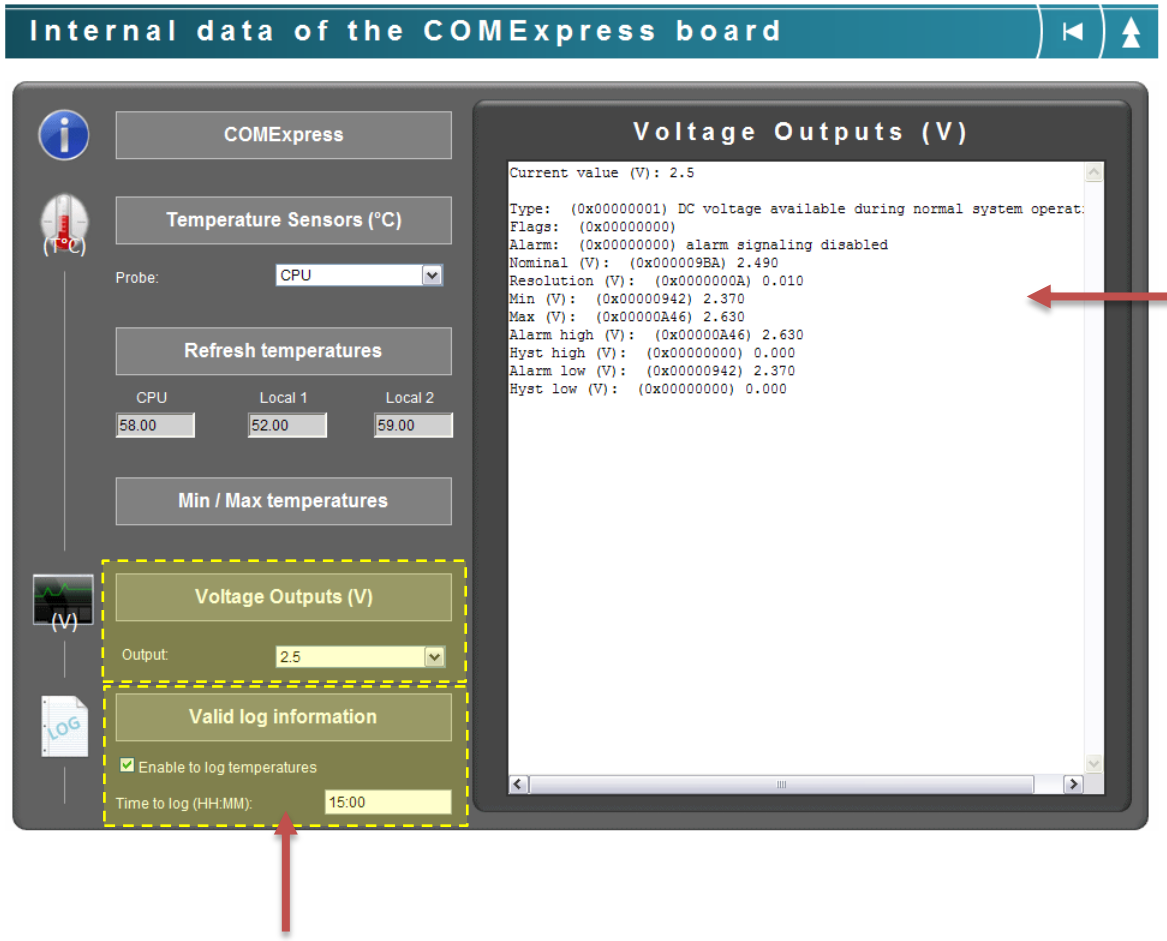
COMEx Supervisor - Controller



COMExpress information

CPU Temperature information



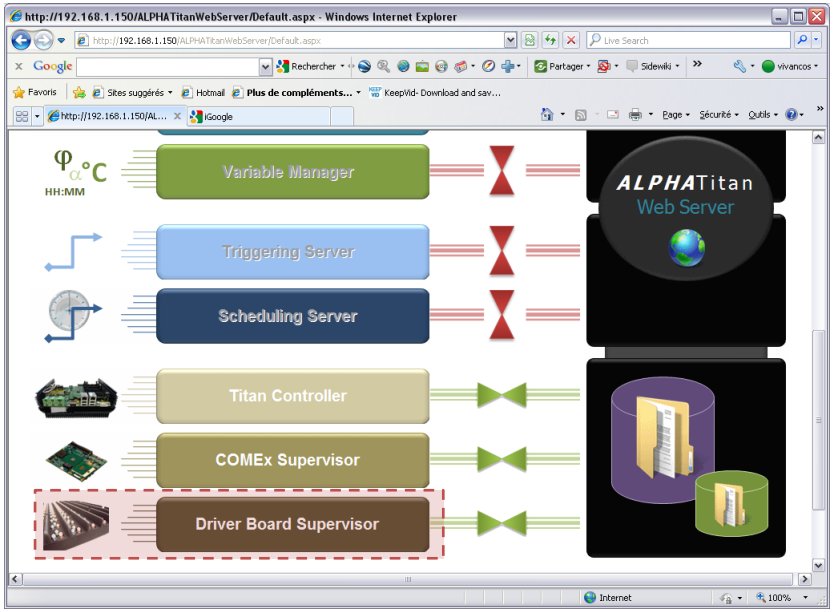


Voltage information

Log information every day at HH:MM

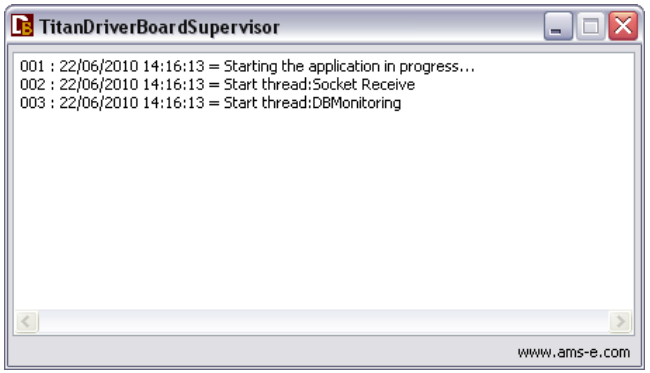
Driver Board Supervisor

- The purpose of the **Driver Board Supervisor** is to monitor the sign (Excite only) and more specifically the driver board operations
- It allows:
  - To setup the sign's luminosity
  - To configure the sleep mode
  - To dialog with both the DVI distributor and the Driver Boards
  - To perform diagnostic operations
- It is configured through the **Web Server**
- Its settings are saved to the file: DBSupervisorINI.xml

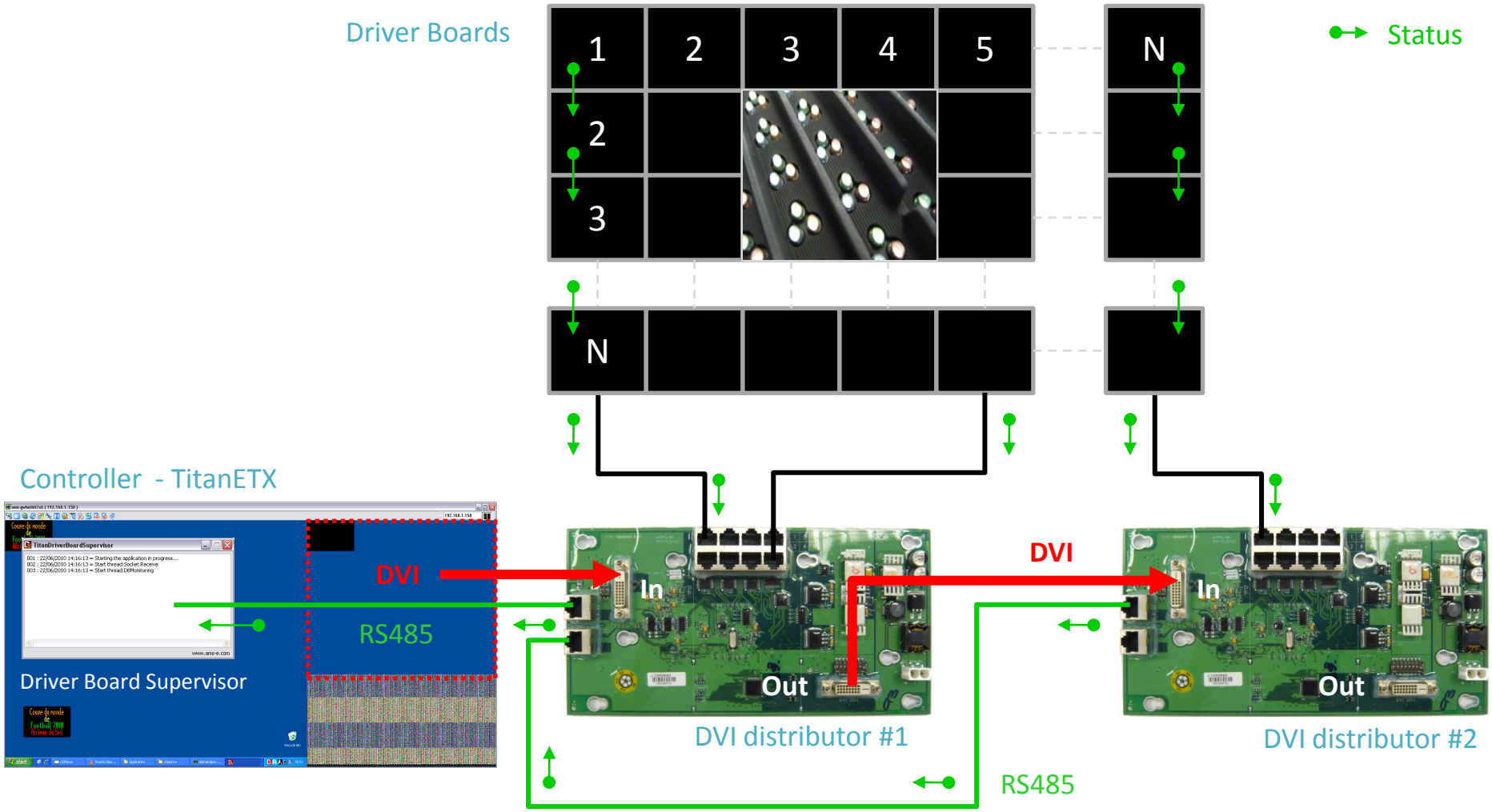


Web Server – Client machine

TCP/IP Commands

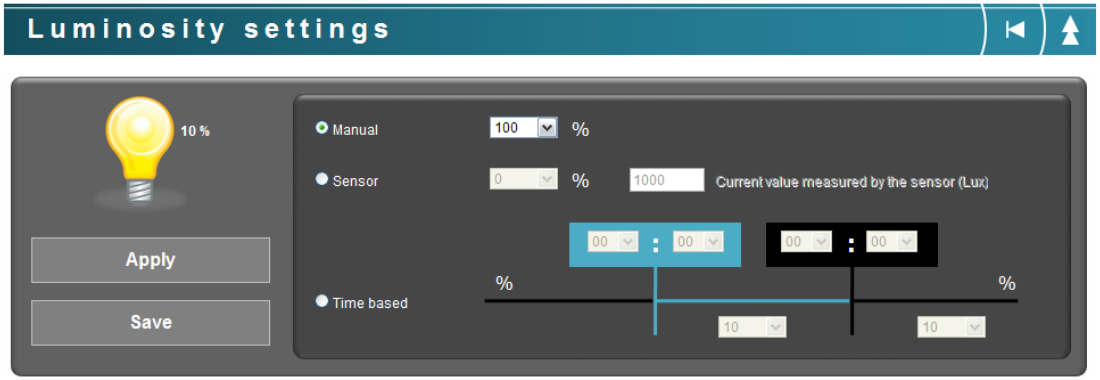


Driver Board Supervisor - Controller

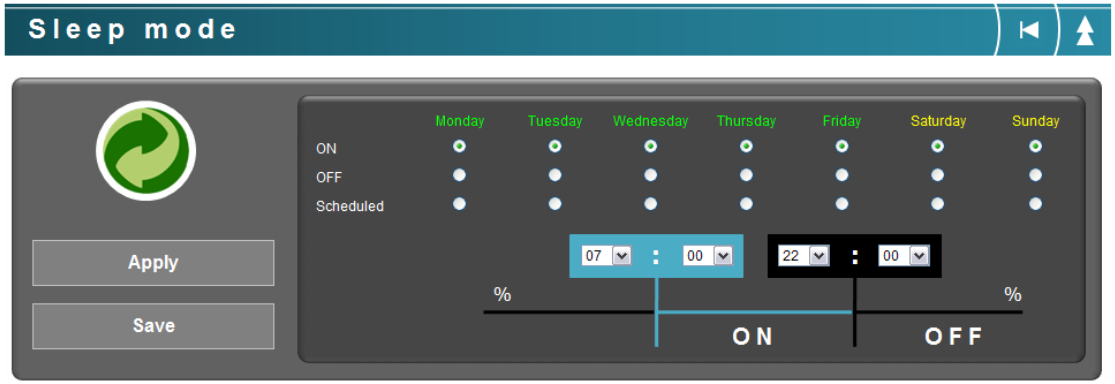


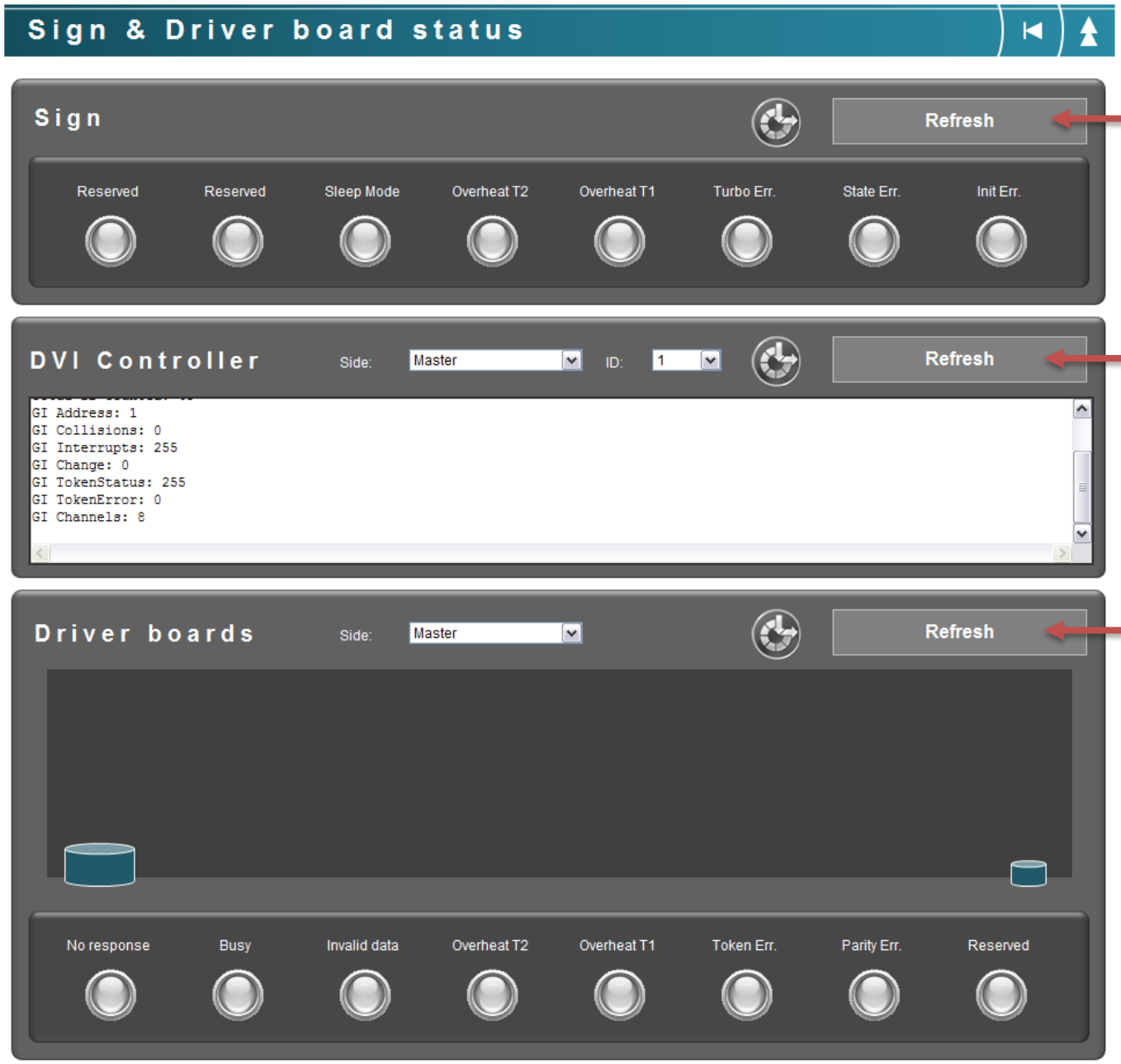
- Each driver board sends its status to the DVI distributor
- Each DVI distributor sends periodically its status to the **Driver Board Supervisor** through the RS485 bus
- The controller's DVI-D output is connected to the 1st DVI distributor

- Three possibilities to setup the sign's luminosity
  - Manually from 0% to 100%
  - According to the intensity measured and returned by the sensor (if it is installed)
  - Scheduled



- In order to save energy, the sign can be powered off during a certain period of time in the day
  - **ON:** Always On
  - **OFF:** Always Off
  - **Scheduled:** On/Off based on the time



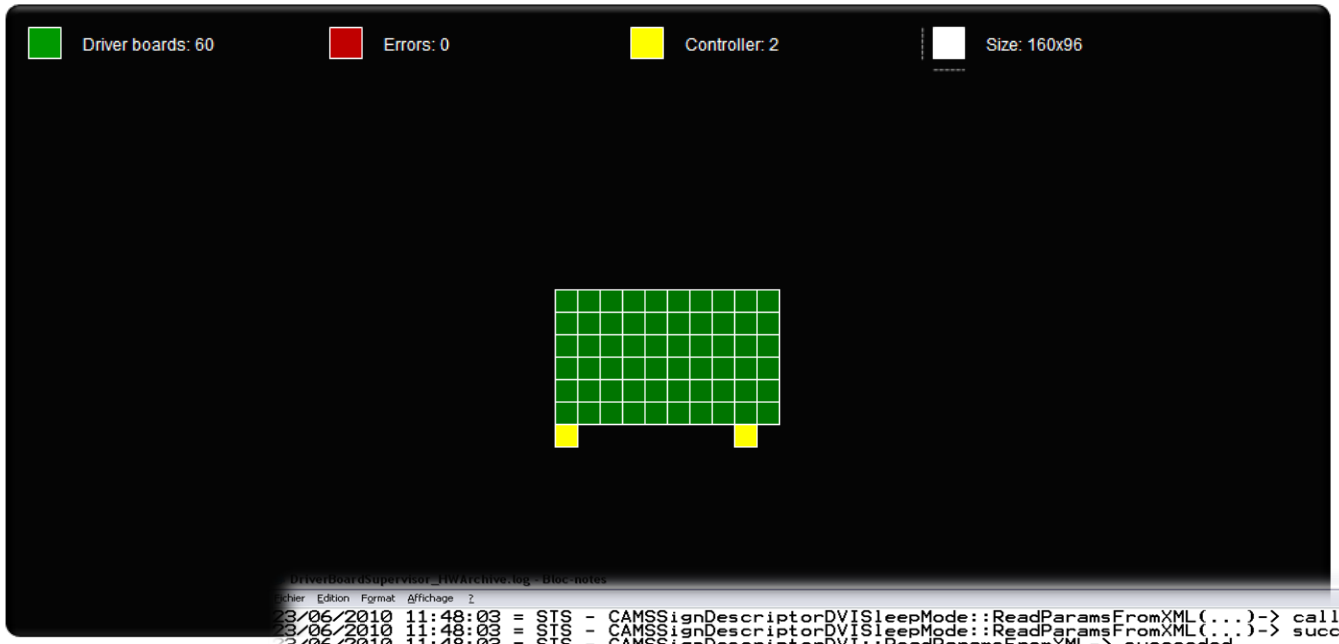


Refresh the main sign's status

Refresh the DVI distributor's status

Refresh the Driver Board's status

- The picture below shows the complete sign's status at a glance
  - The green squares represent each driver board
  - The red squares represent bad driver boards
  - The yellow squares represent the DVI distributors

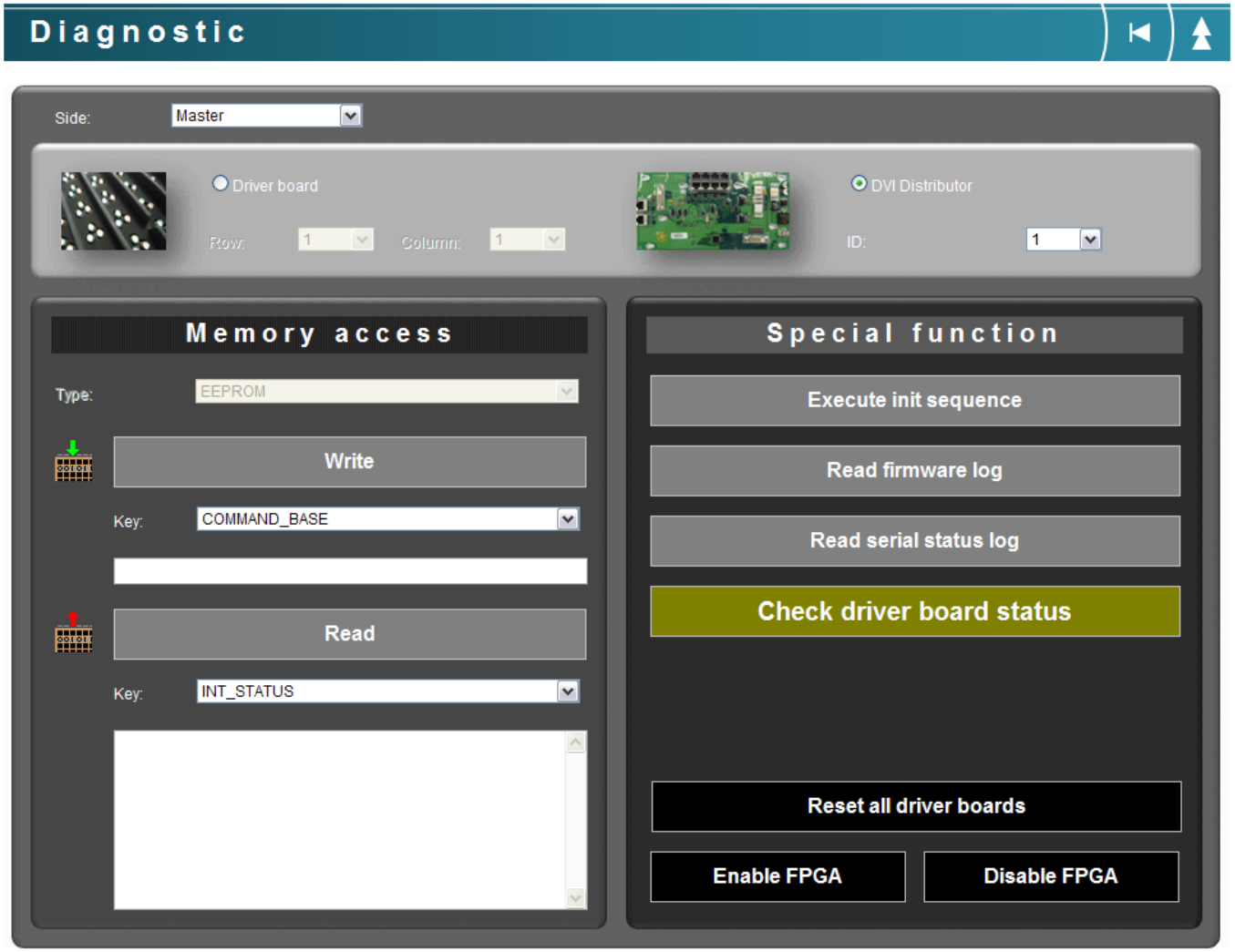



```
Fichier  Edition  Format  Affichage  ?
28/06/2010 11:48:03 = STS - CAMSSignDescriptorDVISleepMode::ReadParamsFromXML(...)-> called
28/06/2010 11:48:03 = STS - CAMSSignDescriptorDVISleepMode::ReadParamsFromXML(...)-> succeeded
3/06/2010 11:48:03 = STS - CAMSSignDescriptorDVI::ReadParamsFromXML-> succeeded
3/06/2010 11:48:03 = STS - CAMSSignDescriptorHWParams::ReadParamsFromXML-> succeeded
3/06/2010 11:48:03 = STS - CAMSSignDescriptor::LoadFromXML(...)-> succeeded
3/06/2010 11:48:03 = STS - CAMSDVIProtocol::Send-> @<000EREG,1,00000117,9C◆
3/06/2010 11:48:05 = STS - CAMSDVIProtocol::Send-> @<000EREG,1,00000117,9C◆
3/06/2010 11:48:05 = STS - CAMSDVIProtocol::Send-> @<000EUV0◆
3/06/2010 11:48:08 = STS - CAMSDVIProtocol::Send-> @<000EUY0◆
3/06/2010 11:48:09 = STS - CAMSDVIProtocol::Send-> @<000EREG,4,00000119,6000A000◆
3/06/2010 11:48:11 = STS - CAMSDVIProtocol::Send-> @<000EREG,4,00000119,6000A000◆
3/06/2010 11:48:12 = STS - CAMSDVIProtocol::Send-> @<010EFv◆
3/06/2010 11:48:14 = STS - CAMSDVIProtocol::Receive-> @0010EEv1199660101GV0908,1133660201A 0906,2611199100020600,1199660601(
3/06/2010 11:48:14 = STS - CAMSDVIProtocol::Send-> @<020EFv◆
3/06/2010 11:48:15 = STS - CAMSDVIProtocol::Receive-> @0020EEv1199660101GV0908,1133660201A 0906,2611199100020600,1199660601(
3/06/2010 11:48:15 = STS - CAMSDVIProtocol::Send-> @<010EF"◆
3/06/2010 11:48:17 = STS - CAMSDVIProtocol::Receive-> @0010E"PACK,00000000,TADR,01,INTS,FF,CHNG,0,TOKN,FF,TERR,00,CHNL,08,N
3/06/2010 11:48:17 = STS - CAMSDVIBoard::LogGeneralInfo-> DVI Board(01): 8 channels detected
3/06/2010 11:48:17 = STS - CAMSDVIBoard::LogGeneralInfo-> DVI Board(01): Channel(1) - 6 driver boards detected
3/06/2010 11:48:17 = STS - CAMSDVIBoard::LogGeneralInfo-> DVI Board(01): Channel(2) - 6 driver boards detected
3/06/2010 11:48:17 = STS - CAMSDVIBoard::LogGeneralInfo-> DVI Board(01): Channel(3) - 6 driver boards detected
3/06/2010 11:48:17 = STS - CAMSDVIBoard::LogGeneralInfo-> DVI Board(01): Channel(4) - 6 driver boards detected
3/06/2010 11:48:17 = STS - CAMSDVIBoard::LogGeneralInfo-> DVI Board(01): Channel(5) - 6 driver boards detected
3/06/2010 11:48:17 = STS - CAMSDVIBoard::LogGeneralInfo-> DVI Board(01): Channel(6) - 6 driver boards detected
3/06/2010 11:48:17 = STS - CAMSDVIBoard::LogGeneralInfo-> DVI Board(01): Channel(7) - 6 driver boards detected
3/06/2010 11:48:17 = STS - CAMSDVIBoard::LogGeneralInfo-> DVI Board(01): Channel(8) - 6 driver boards detected
3/06/2010 11:48:17 = STS - CAMSDVIProtocol::Send-> @<010E*◆
3/06/2010 11:48:17 = STS - CAMSDVIProtocol::Send-> @<010EFW,1,1C,01,01,000F◆
3/06/2010 11:48:18 = ERR - CAMSDVIProtocol::Receive-> No byte received
```

The log file traces all exchanges between the application and the DVI distributor

This information is really helpful to understand issues occurring on the system

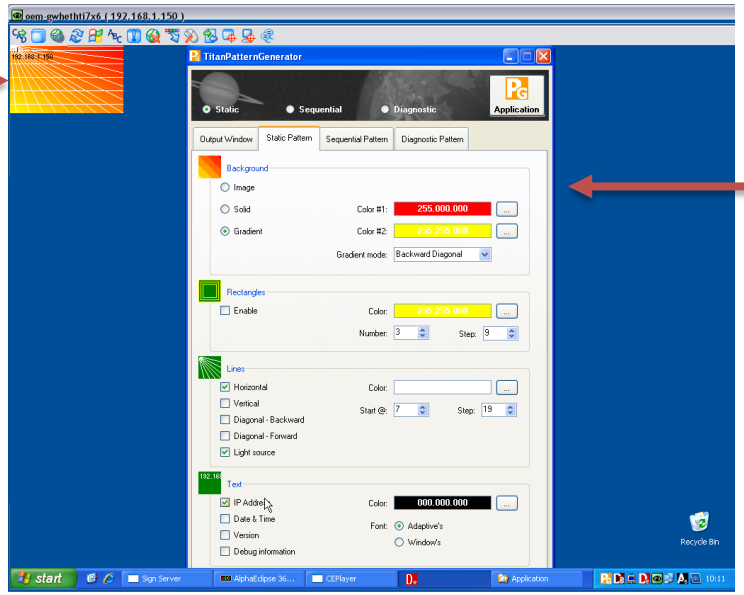
This section is strictly reserved to the Adaptive's Tech Support



Pattern Generator

- The **Pattern Generator** is mainly used to perform diagnostic operations
- The Output Window can be positioned anywhere on both screens by using the Web Server
- It allows:
  - To display different static patterns
  - To run sequential patterns
  - To display patterns specially dedicated to the driver boards
- It has its own user interface which can be opened only when connected on the controller through Ultra-VNC
- Its settings are saved to the file: TitanPatternGeneratorINI.xml

Output Window



Pattern Generator

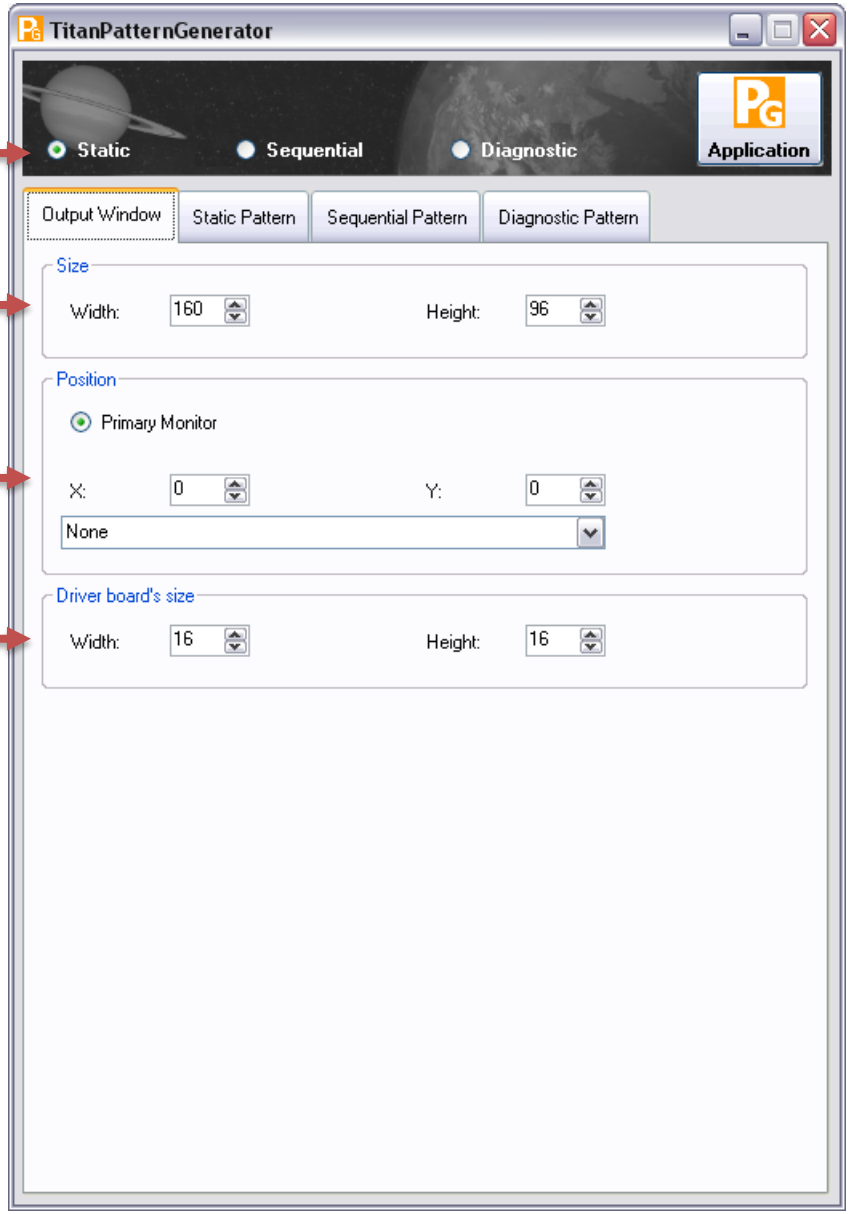
Ultra-VNC

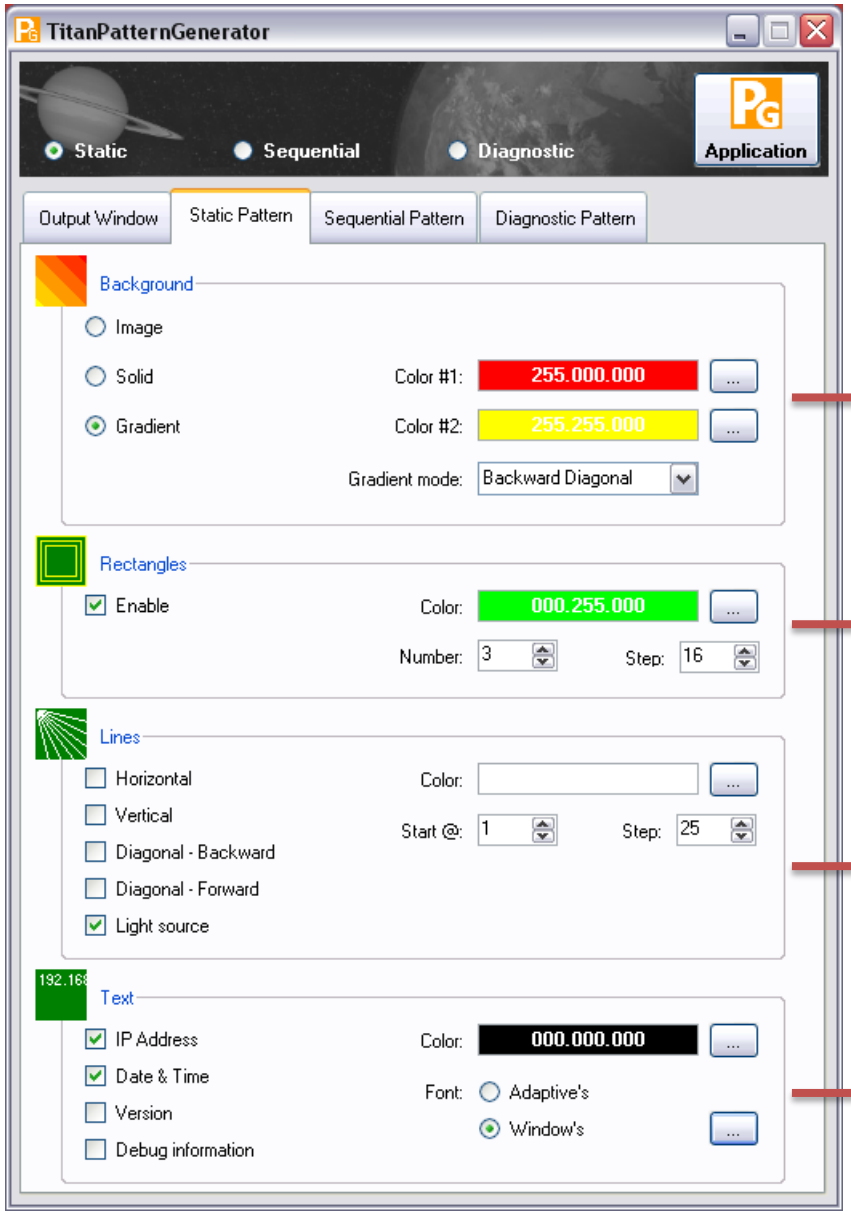
Select the pattern to display on the Output Window

Output Window's size

Output Window's position

Driver Board's size



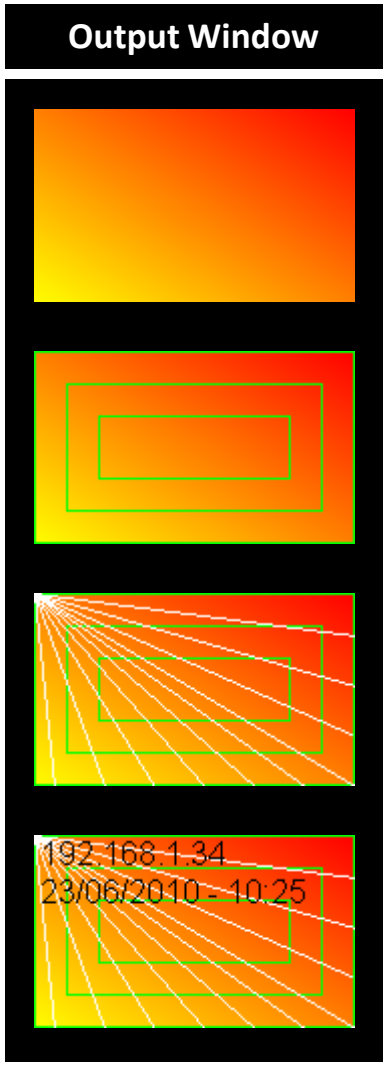


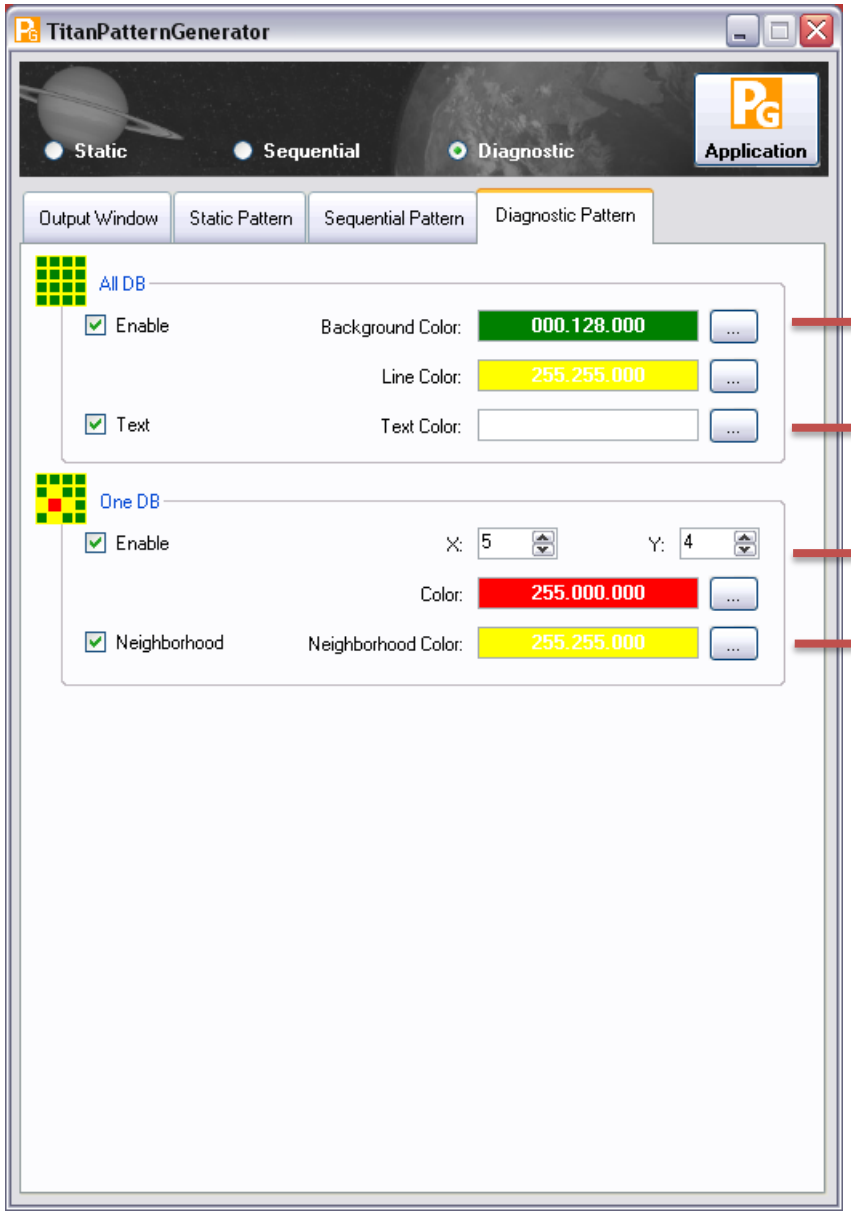
Select a background

Add rectangles

Add light source

Add text



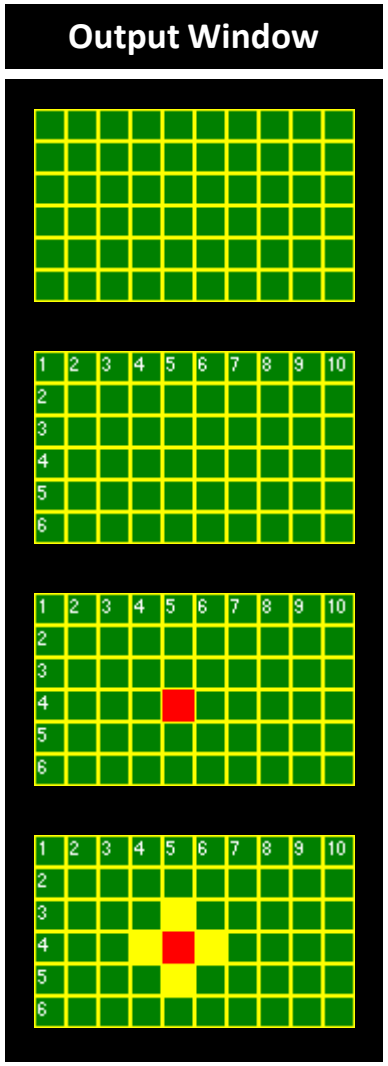


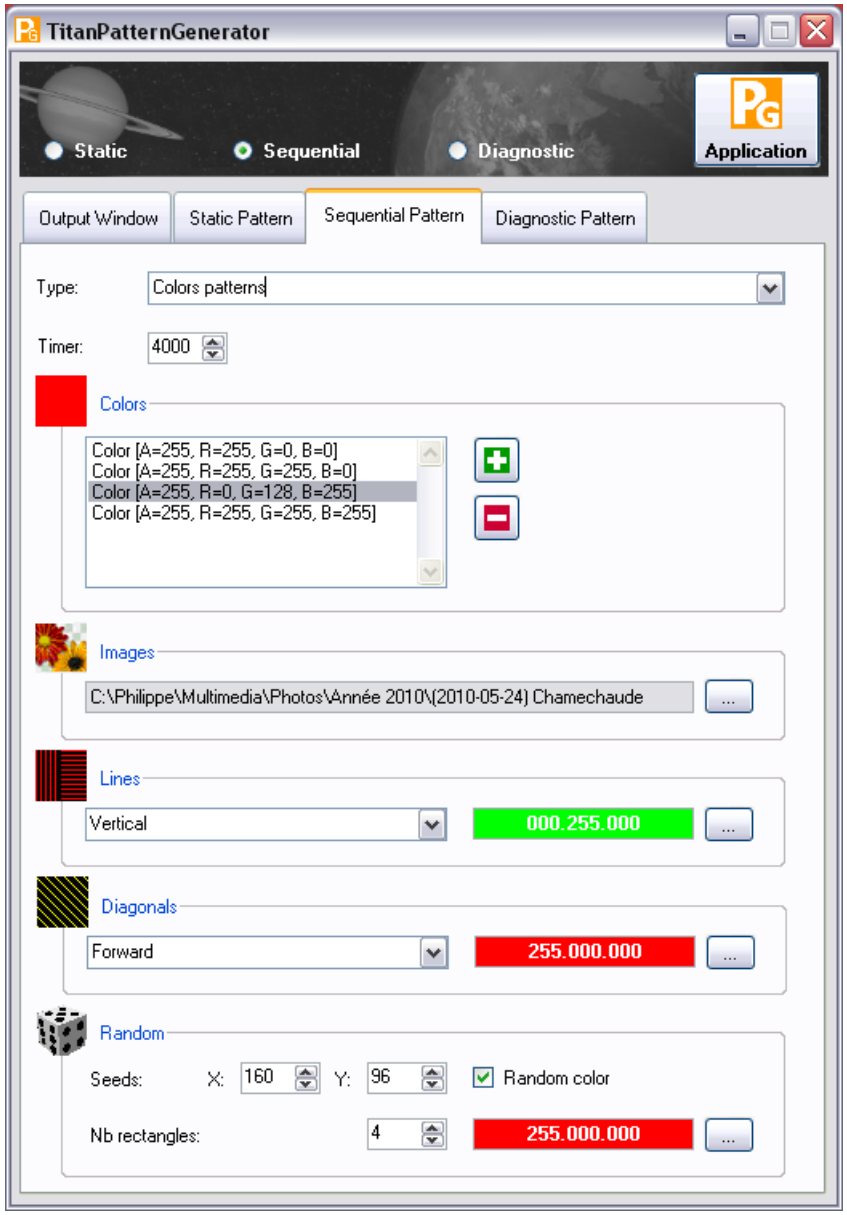
Select a background

Add text on rows & columns

Light a selected DB

Light DB's neighborhoods

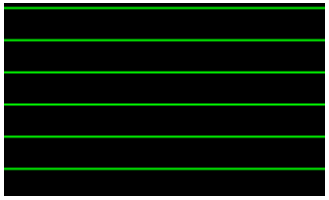




Output Window

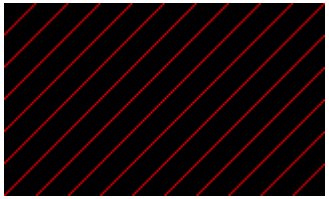
Color sequence

Image sequence



Line sequence

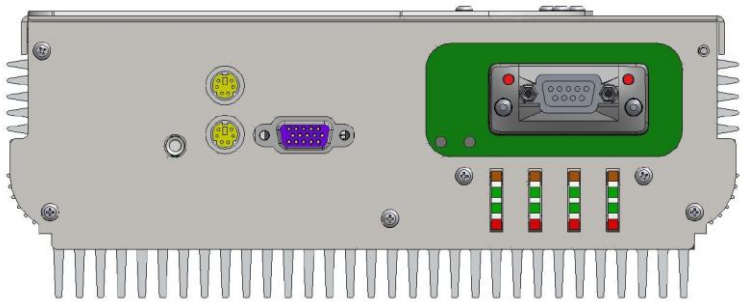
Diagonal sequence

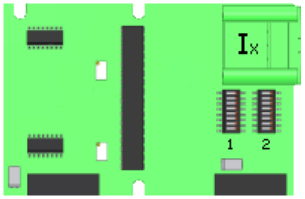


Random sequence

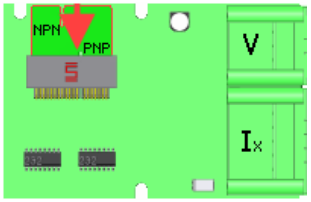
Variable Manager

- The purpose of the **Variable Manager** is to collect data from different hardware sources and to feed the applications which take out a subscription to them
- Data are coming from:
  - The luminosity sensors
  - The temperature sensors
  - The discrete input interface
  - The analog input interface
  - The industrial network interface
- Its settings are saved to the file: TitanVariableManagerINI.xml
- The **Variable Manager** works with a configuration file stored in its own workspace directory

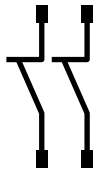


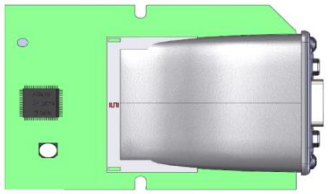


x2 Analog inputs



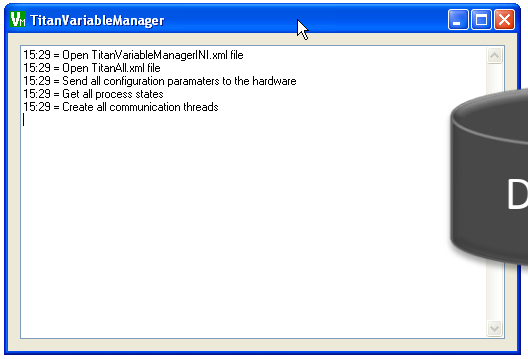
x8 Discrete inputs





x1 Industrial Network

Values



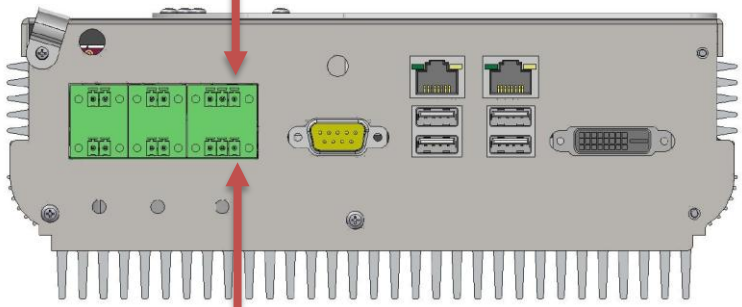
Database

Values

Variable Manager

RS485

x2 Luminosity sensors

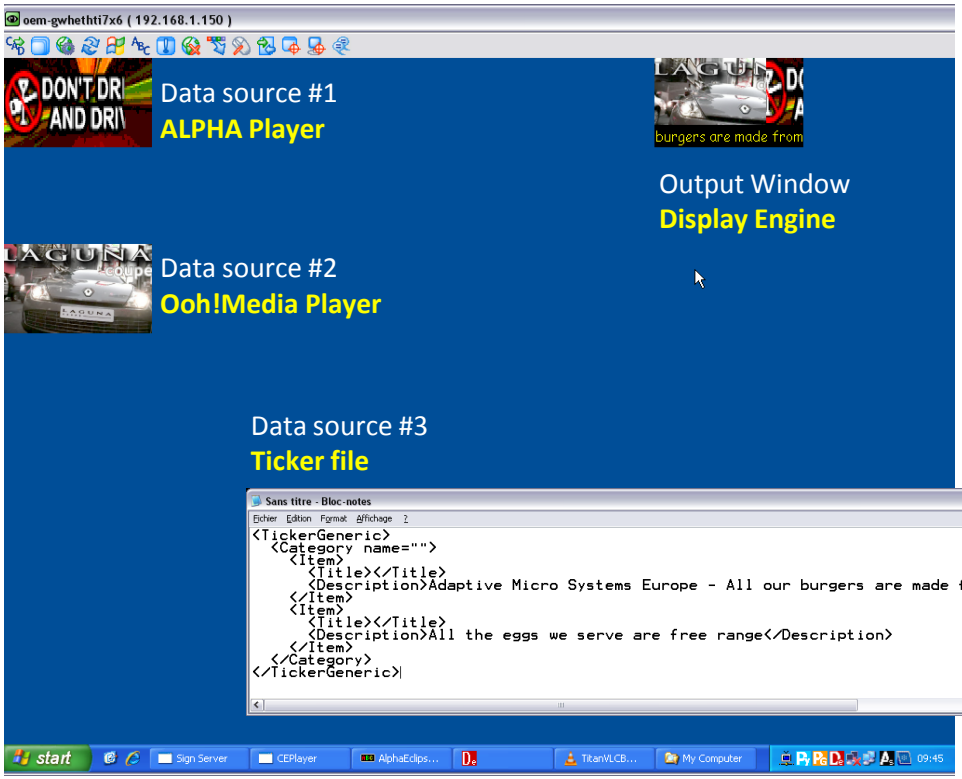


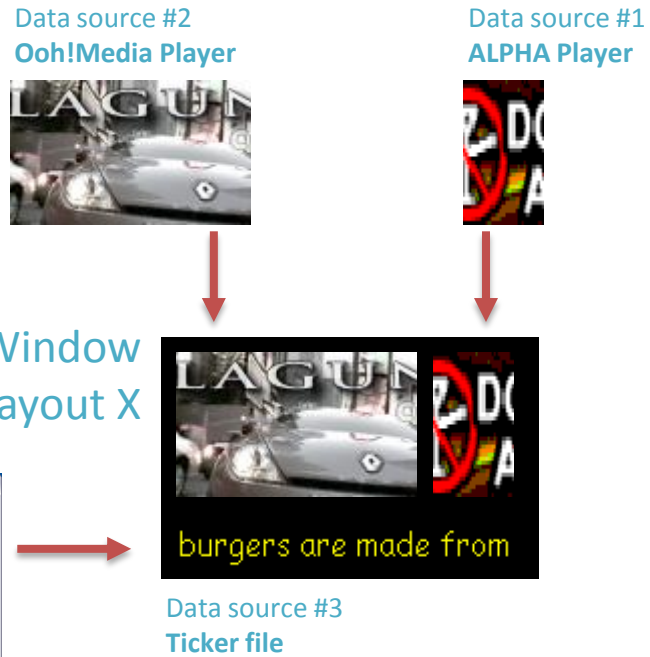
RS485

x2 Temperature sensors

Display Engine

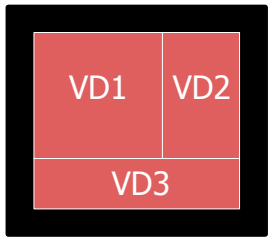
- The **Display Engine** is a powerful application allowing to build and to display images coming from several types of data sources such as **Screen Captures**, **Real Time Variables** and **Tickers**
- Those images are laid out on the Output Window and sent to either the **DVI-D** output (Excite) or the **LEDs Driver Board Interfaces** (ALPHA & StreetSmart)
- The Display Engine is configured through the **Web Server**
- Its settings are saved to the file: DisplayEngineINI.xml





Physical Sign (PS)

- It contains a collection of Virtual Displays, Layouts and Playlists
- It can be divided in xN Virtual Displays ($1 \leq VD \leq 6$)

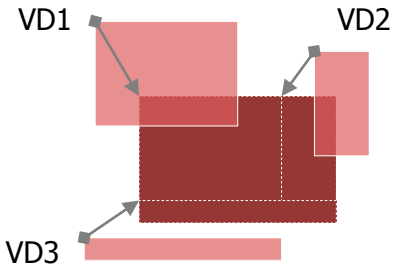


Virtual Display (VD)

- It is a region (WxH) of the physical sign
- Each VD is fully independent from the others
- Each VD has its own operating modes
- x3 types of Virtual Displays : Screen Capture, Ticker, RT Variable

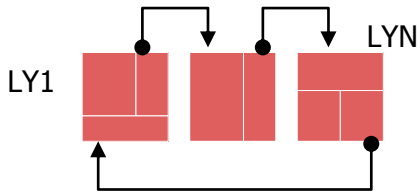
Layout (LY)

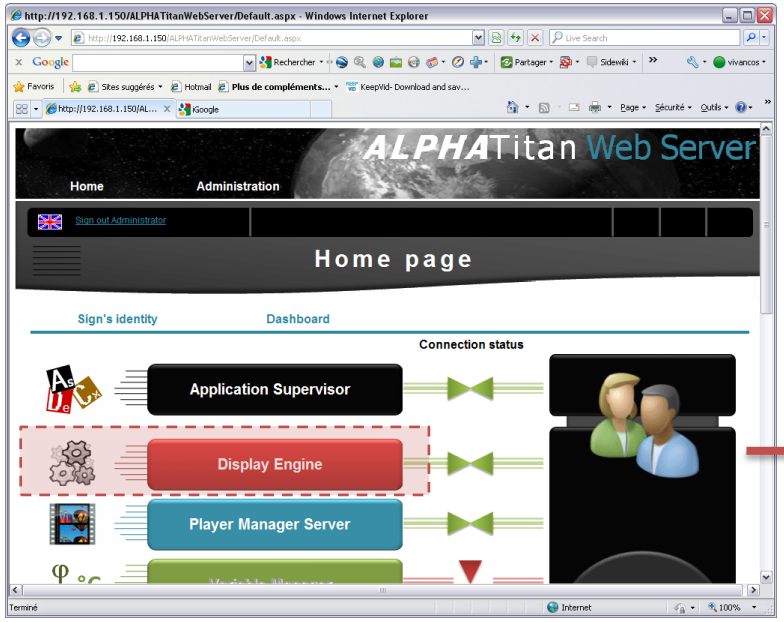
- It defines how several VD are positioned (X, Y) over the PS

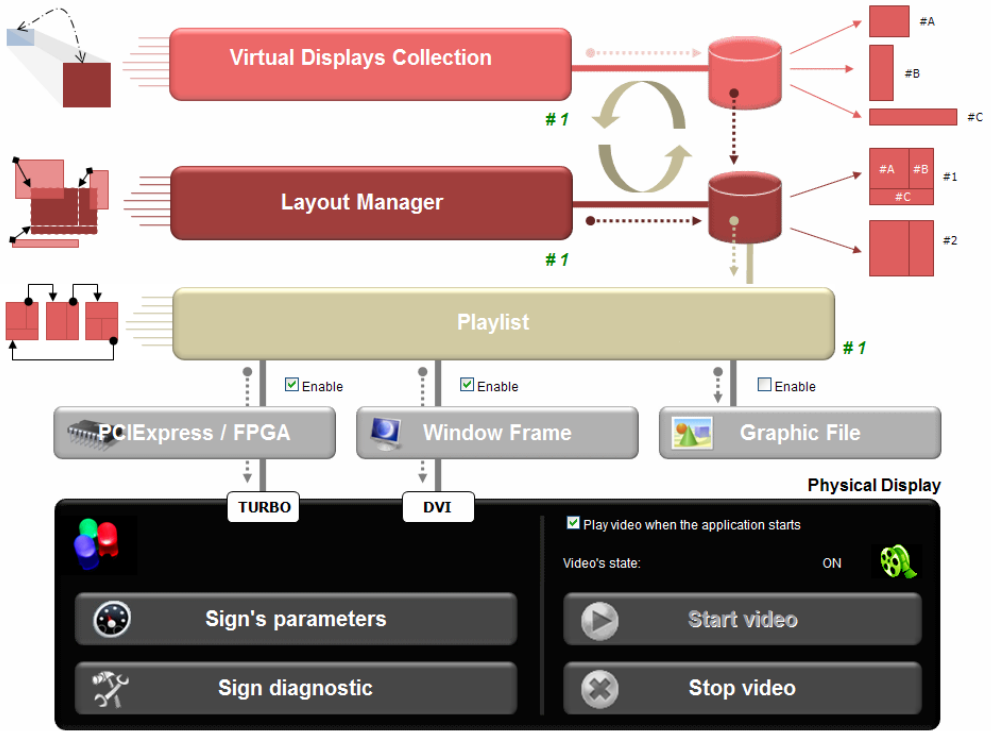


Playlist (PLY)

- It cyclically runs LY one after the other







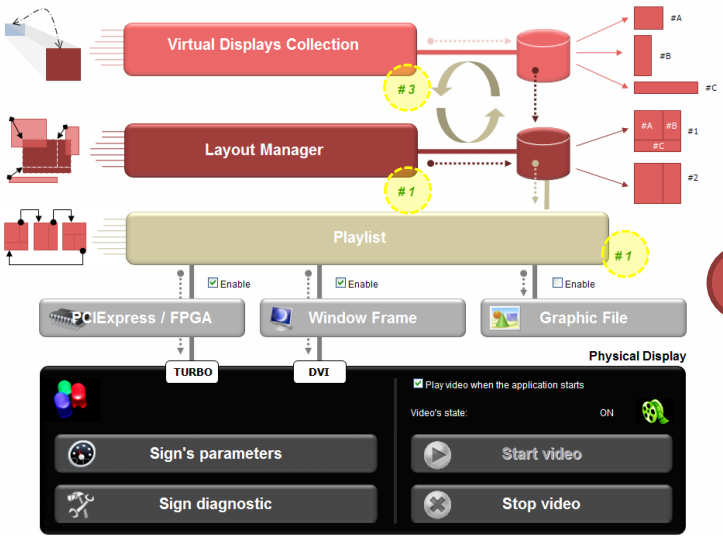
- The Web Server provides the tools to complete a full configuration of the Display Engine
- Going through the configuration process can be complex so it is really important to follow the procedure detailed in the next pages

- **Create a new configuration file**
  1. Select the root directory
  2. Click on "New"
  3. Select the new configuration file and rename it
  4. Click on "Update"

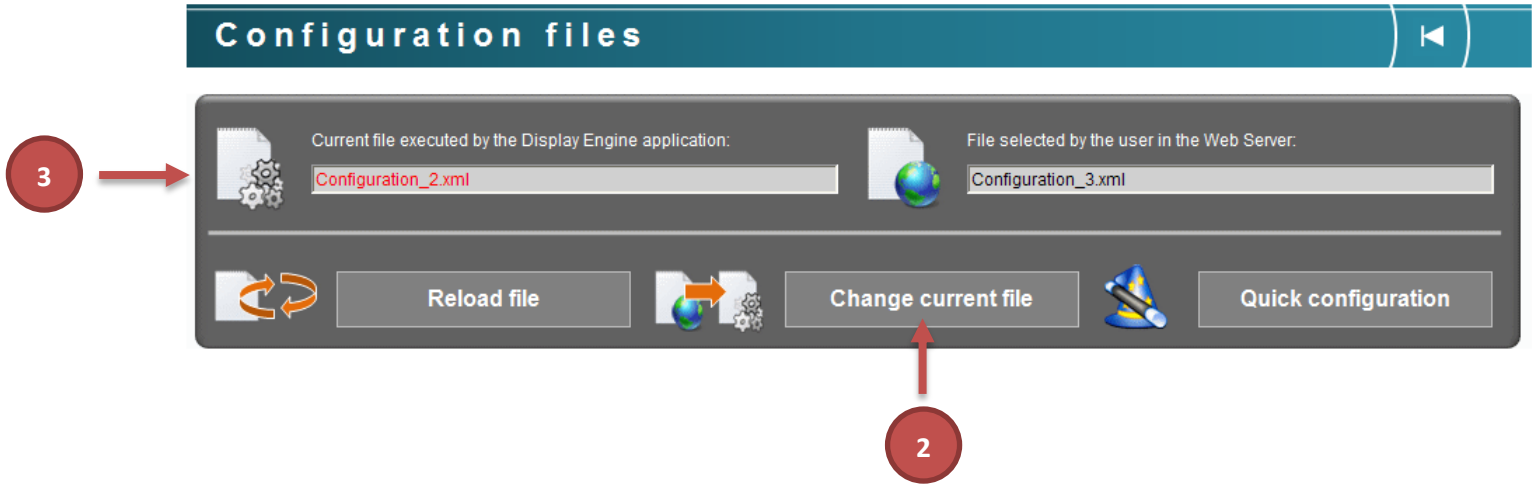
- **Open the wizard configuration tool**
  1. Select the configuration file
  2. Click on "Quick Configuration"
  3. Enter a layout's name
  4. Select a pre-define layout template
  5. Enter the desired X&Y values
  6. Select & Name the virtual displays
  7. Click on "Add"

The Wizard allows to create a full configuration by adding the VD, LY & PLY automatically

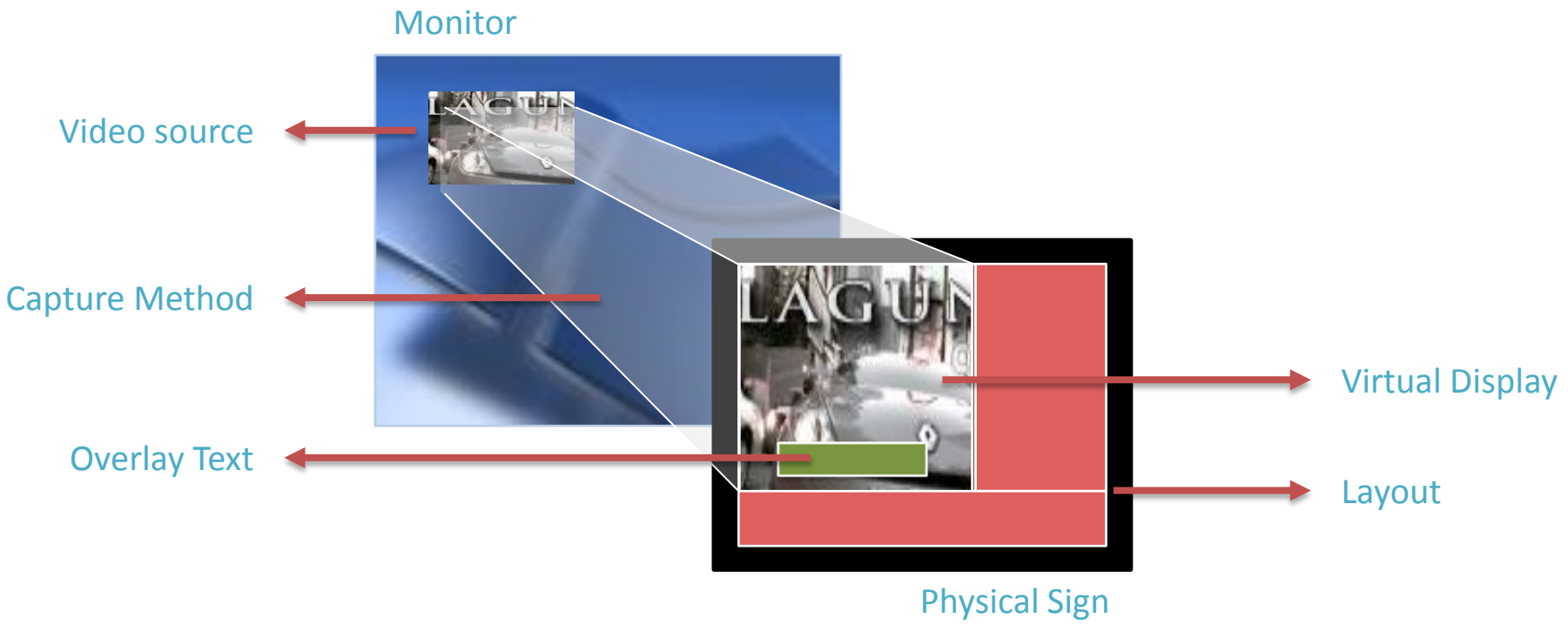
Do not hesitate to use this tool during your first steps with the system



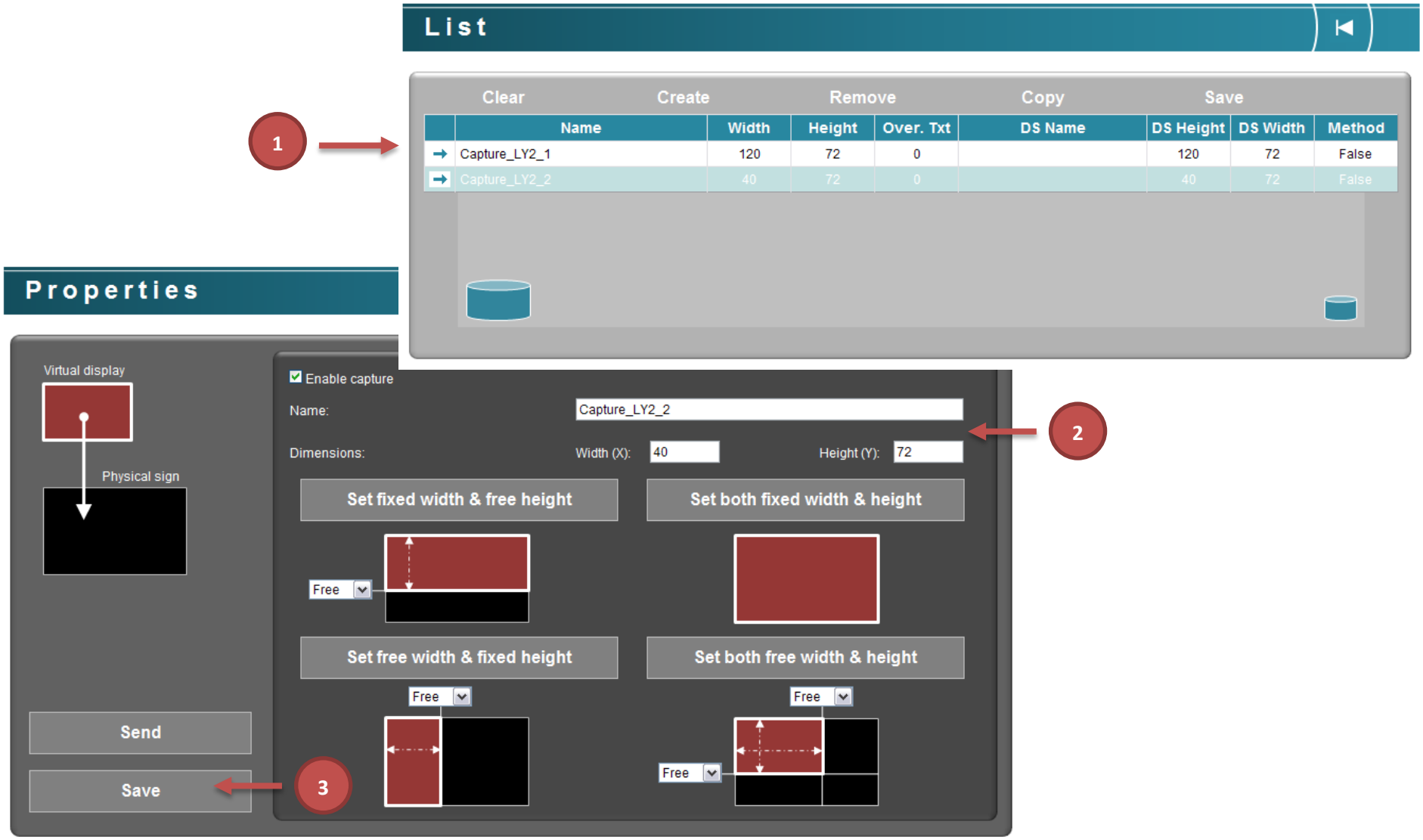
- **Change the current file to the new configuration**
  1. Select the configuration file
  2. Click on "Change current file“
  3. The Display Engine is reloaded with the new file



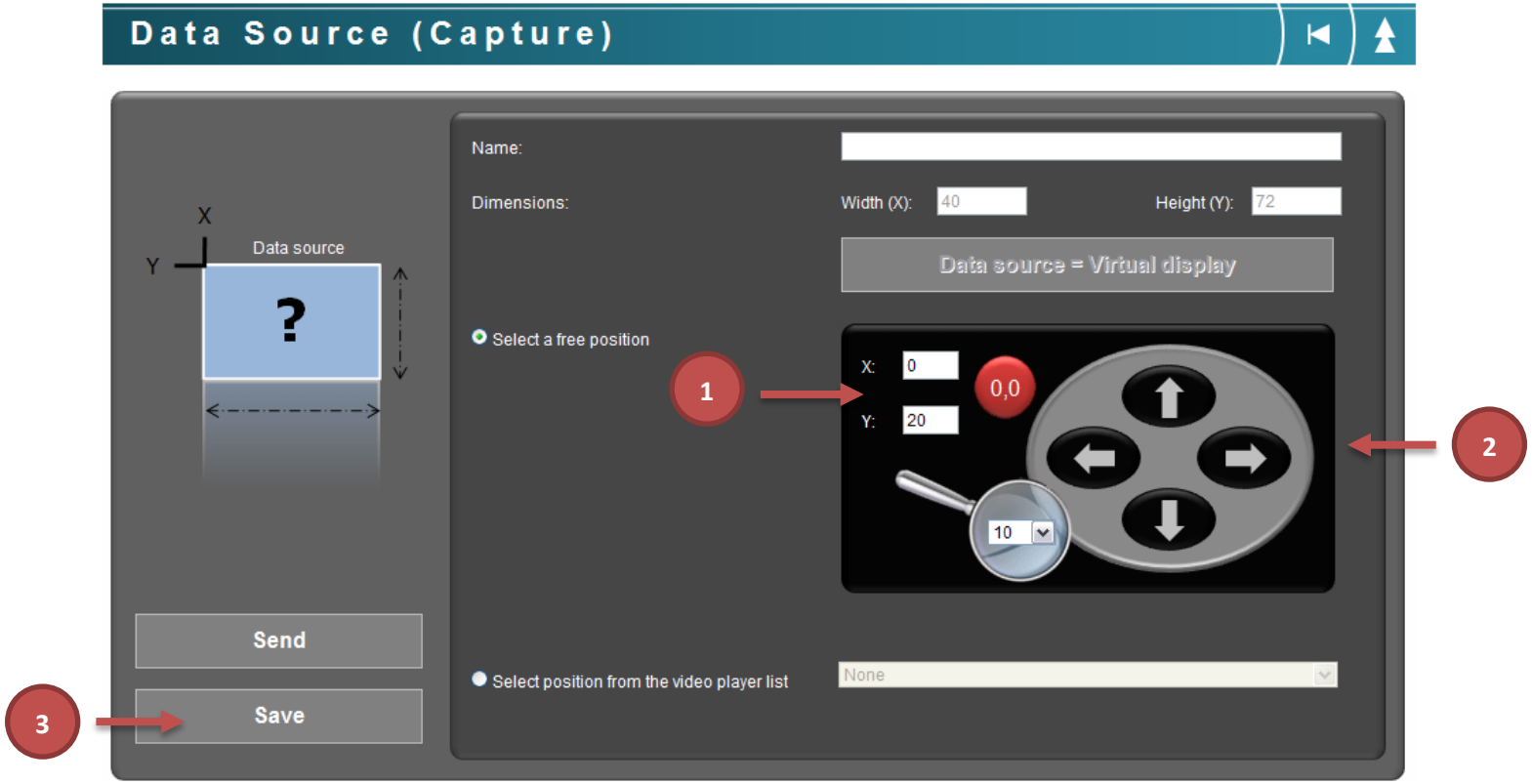
- During the **screen capture** process, the Display Engine goes through the following sequence:
  1. Scans & Captures, using a given method, all video sources located on the screens
  2. Overlays text objects (date, time, T°C, variables, ...) if required
  3. Builds the RGB pixel buffer based on a specific layout
  4. Sends the RGB pixel buffer to the Output Window



- **Screen Capture properties**
  1. Select a VD in the list
  2. If needed, modify the VD's properties

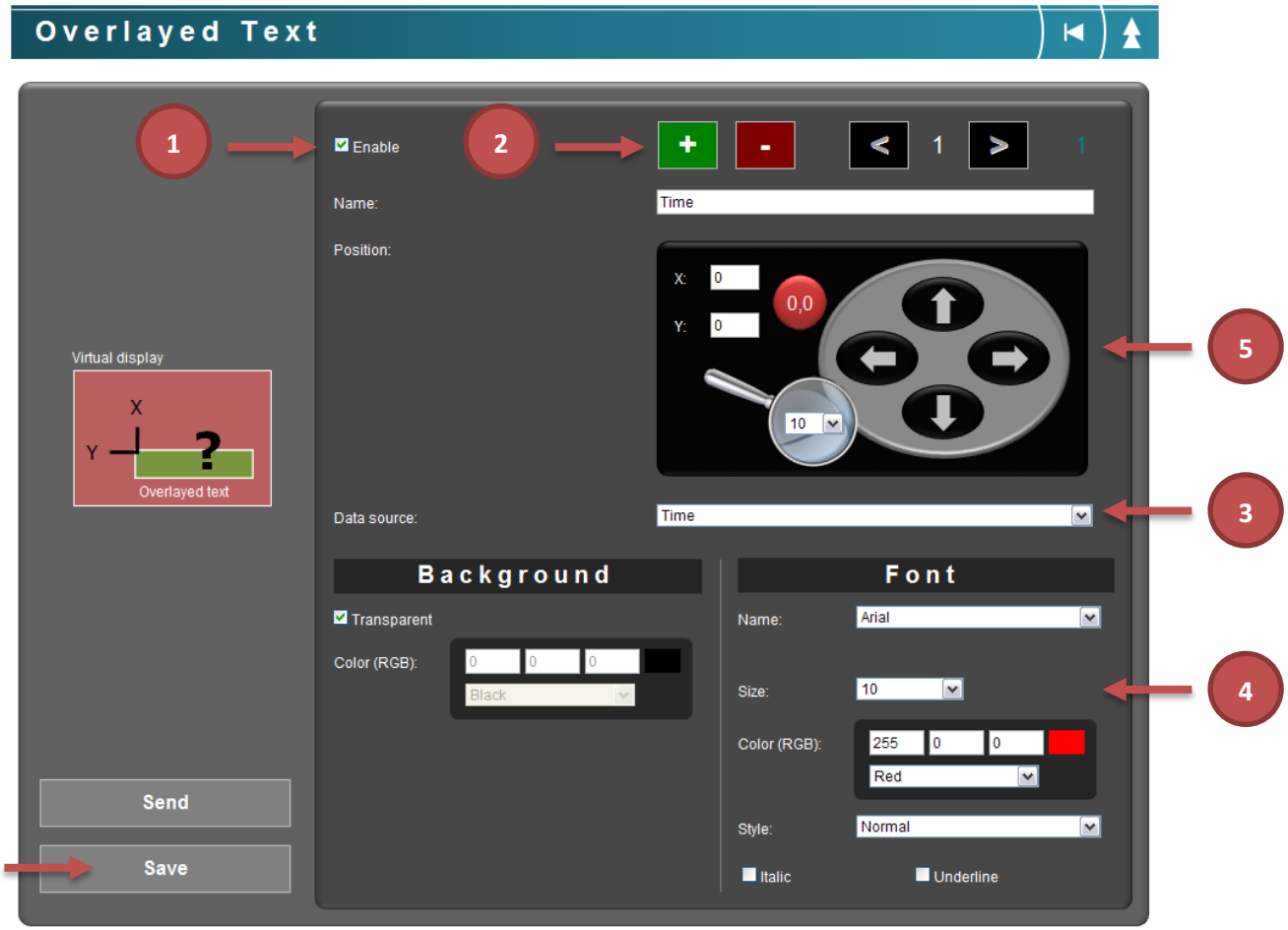


- **Setup the data source's location**
  1. If known, enter the values directly in the edit boxes
  2. If unknown, use the navigation pad to move the screen capture over the data source



We do not recommend to set a capture method because it takes a lot of CPU resources

- **Add overlay texts**
  1. Enable the option
  2. Add an overlay text and enter its name
  3. Select a data feed (Date & Time or Variable)
  4. Set the background and the text properties
  5. Position the overlay text over the virtual display



- A **Ticker** is a long text which rotates from the right to the left in a Virtual Display
- Data are stored in a .XML files located in the shared workspace at ..\MyXMLFiles\DataSources\Files\
- **x3 formats** are available:

Generic format

Generic.xml - Bloc-notes

```
<TickerGeneric>
  <Category name="">
    <Item>
      <Title></Title>
      <Description>Adaptive Micro Systems Europe - All our burgers are made from 100% whole cuts o
    </Item>
    <Item>
      <Title></Title>
      <Description>All the eggs we serve are free range</Description>
    </Item>
  </Category>
</TickerGeneric>
```

Weather format

Meteo.xml - Bloc-notes

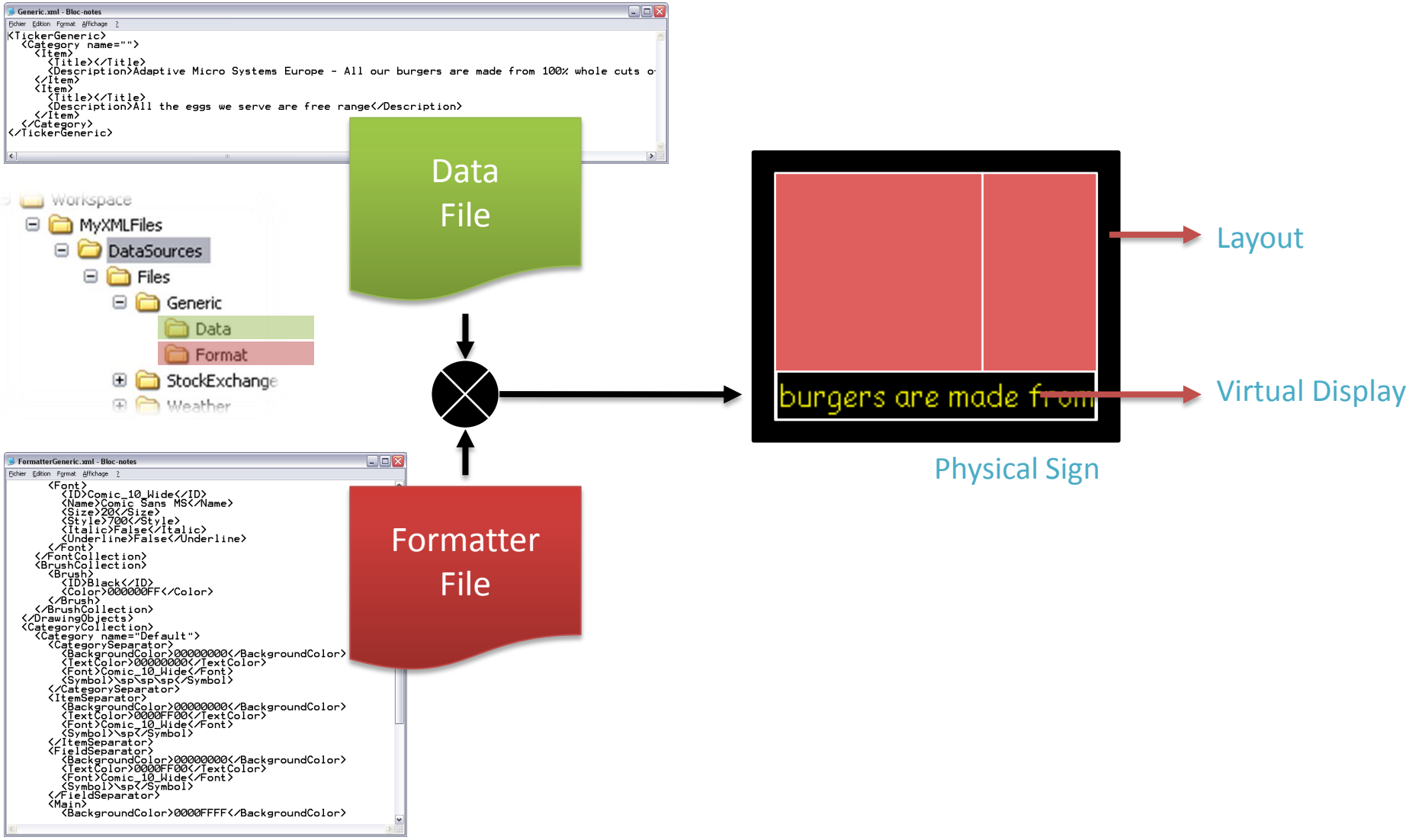
```
<TickerWeather>
  <Category name="Grenoble">
    <Item>
      <Title>Monday</Title>
      <Description>Rain</Description>
      <MaxTemp>Max C = 10</MaxTemp>
      <MinTemp>Min C = 0</MinTemp>
      <WindSpeed>Wind speed= 5 K/H</WindSpeed>
      <WindDirection>Wind direction = South-East</WindDirection>
      <Humidity>Humidity = 95%</Humidity>
    </Item>
    <Item>
      <Title>Tuesday</Title>
      <Description>Sunny</Description>
      <MaxTemp>Max C = 10</MaxTemp>
      <MinTemp>Min C = 0</MinTemp>
      <WindSpeed>Wind speed = 5 K/H</WindSpeed>
      <WindDirection>Wind direction = North</WindDirection>
      <Humidity>Humidity = 95%</Humidity>
    </Item>
  </Category>
</TickerWeather>
```

Stock Exchange format

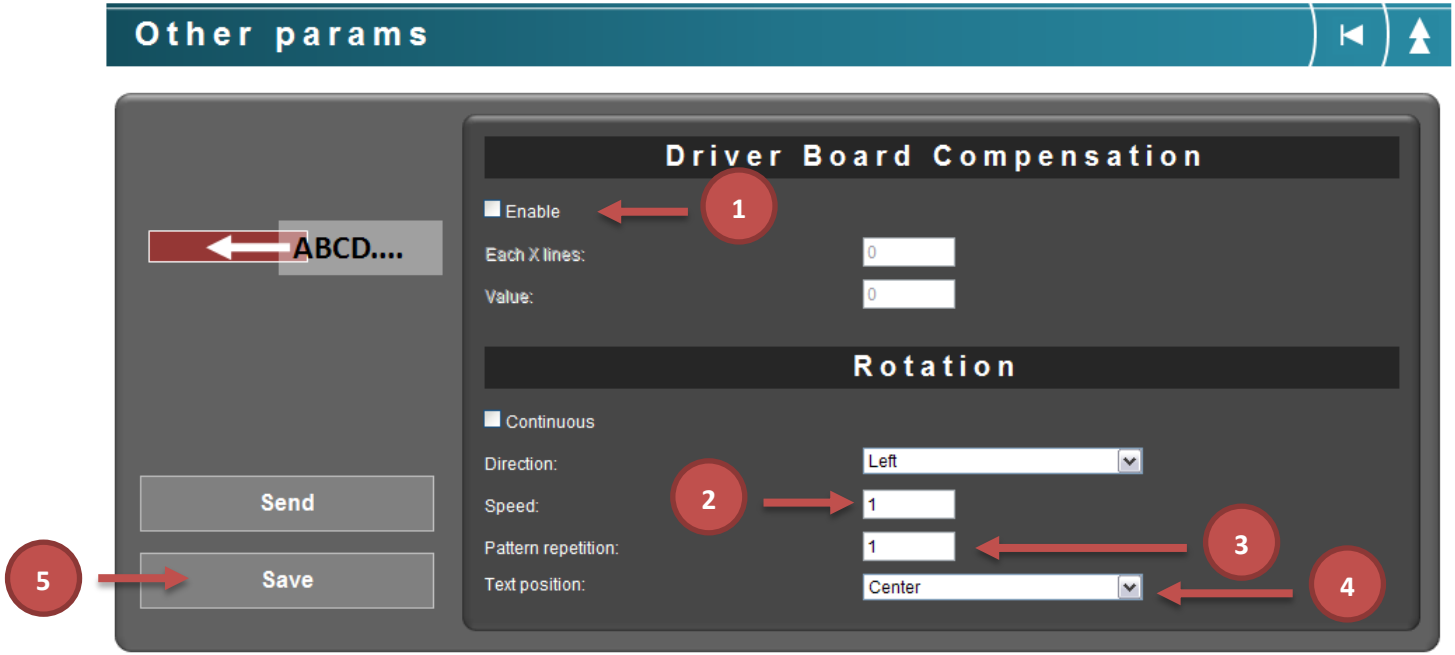
StockExchange.xml - Bloc-notes

```
<TickerStockExchange>
    <Category name=" CAC 40 ">
        <Item>
            <Title></Title>
            <Description></Description>
            <Variance>
              <Sign>-</Sign>
              <Value>0,5</Value>
            </Variance>
        </Item>
    </Category>
    <Category name=" Dow Jones ">
        <Item>
            <Title></Title>
            <Description></Description>
            <Variance>
              <Sign>0</Sign>
              <Value>1</Value>
            </Variance>
        </Item>
    </Category>
</TickerStockExchange>
```

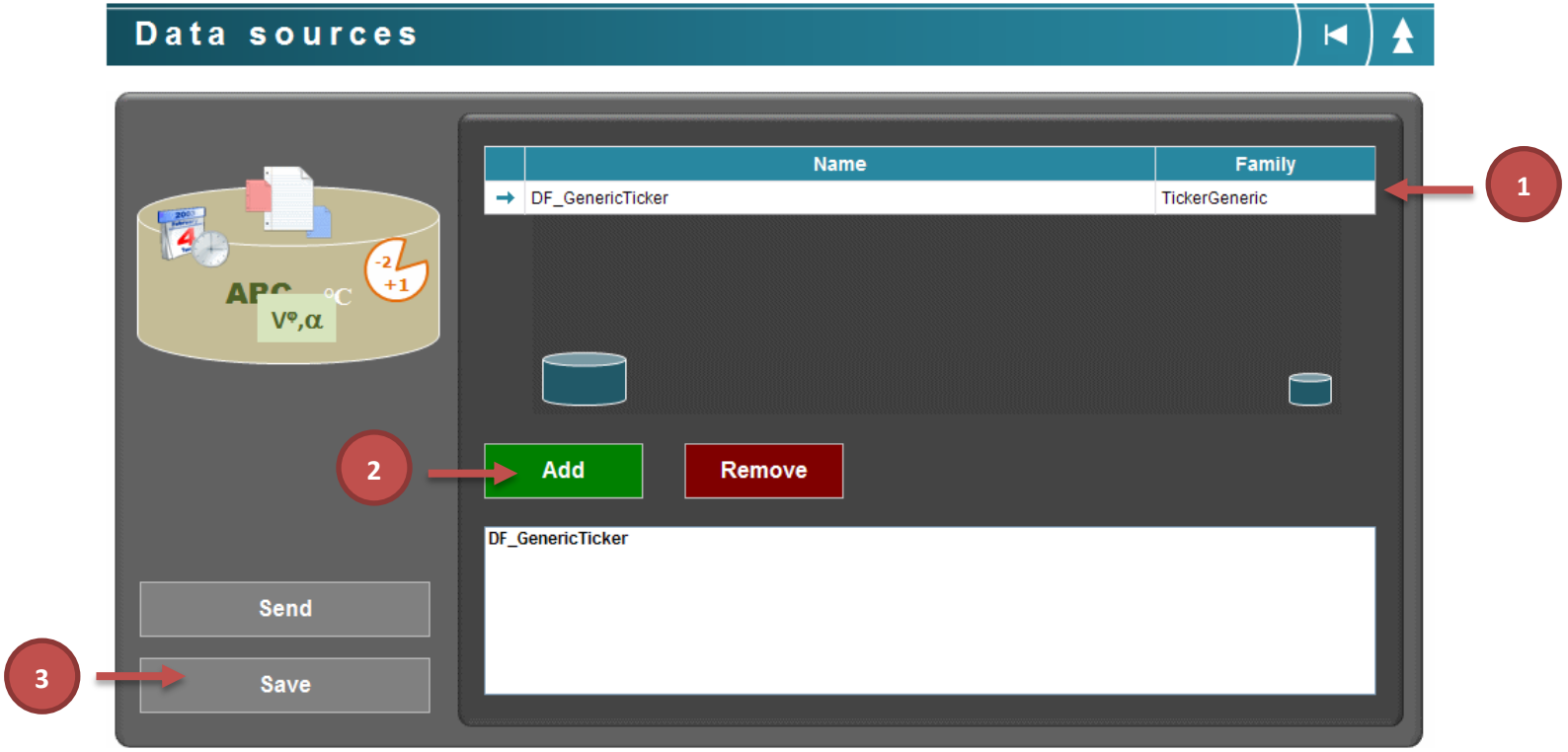
- Each item of the data is displayed according to a certain format defined in a formatter .XML file



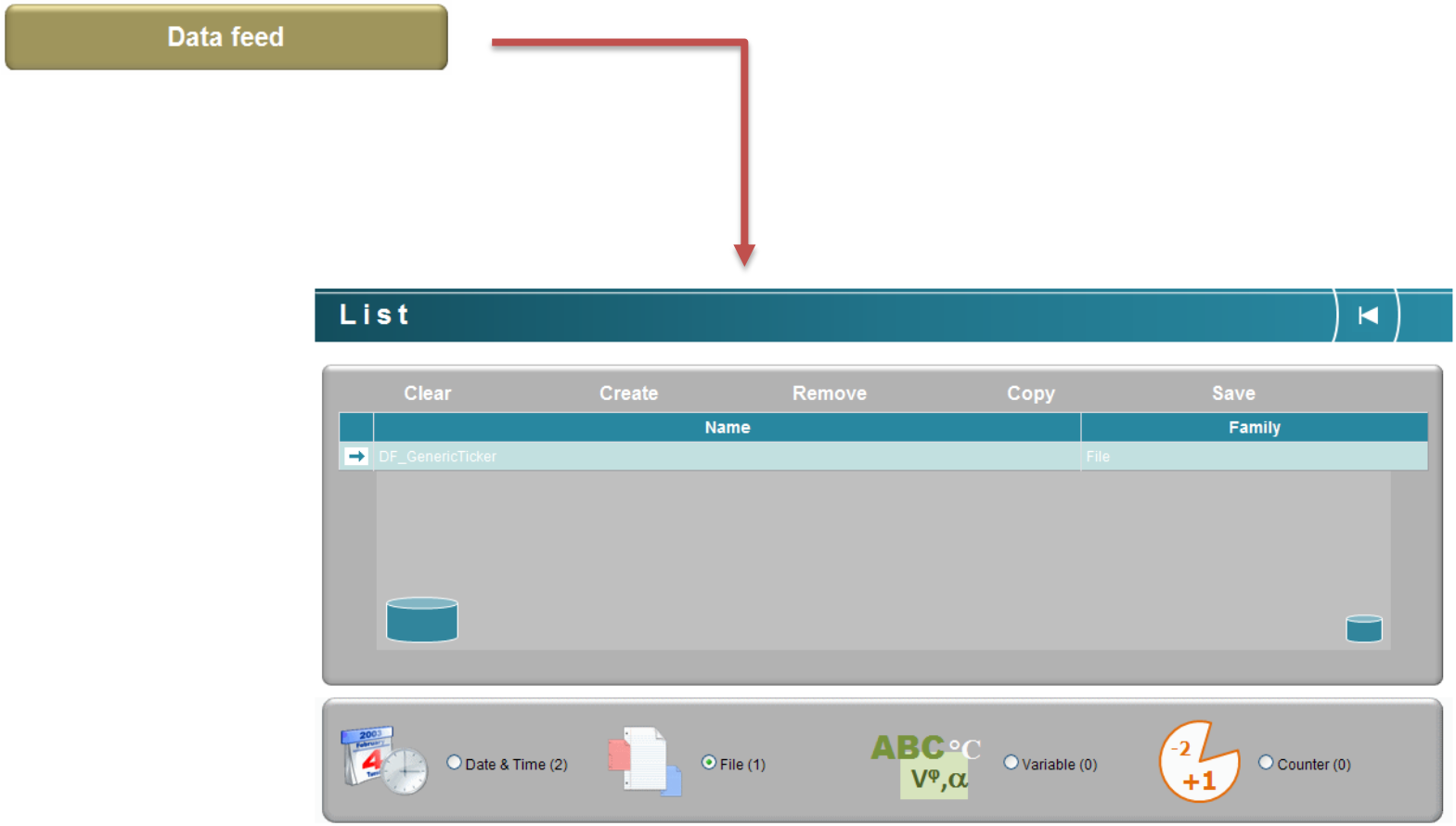
- **Setup the ticker properties**
  1. Enable the Driver Board compensation (ALPHA only) if required
  2. Set the rotation speed (1, 2, 3, ...)
  3. Set the pattern repetition (rotates N times before to empty the RGB pixel buffer and to reload the data .XML file)
  4. Set the text position (Center, Top, Bottom)



- **Assign a data feed to the Ticker**
  1. Select the data feed in the list
  2. Add it to the list

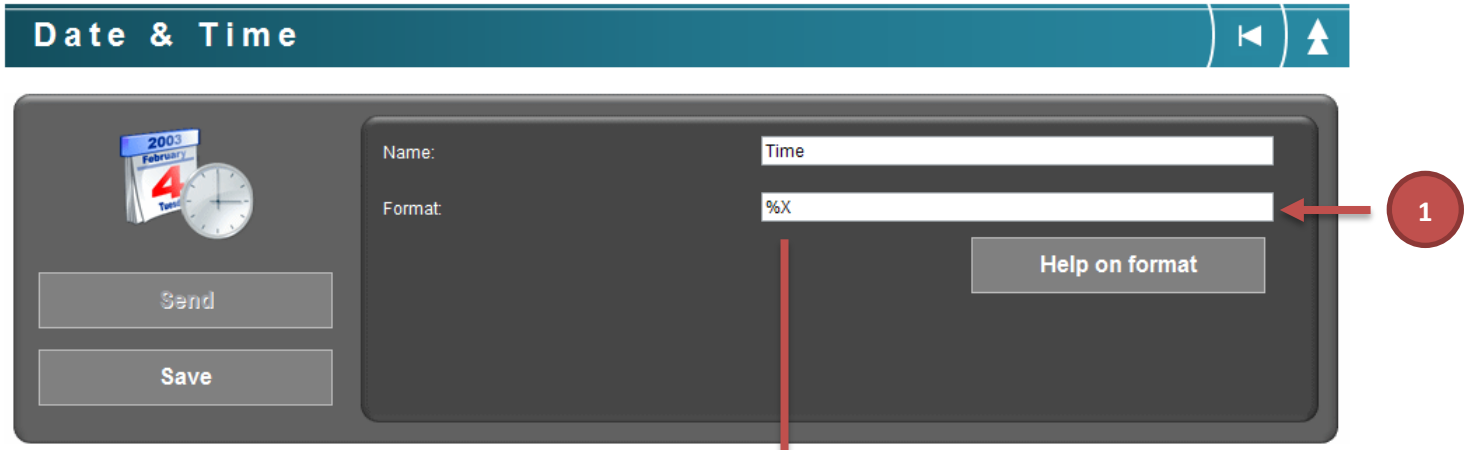


- Data Feed objects manage the data and the way of how they are used in both the **Ticker** and the **Overlay Text**
- **Tickers** use File objects
- **Overlay Text** use Date & Time and Variable objects



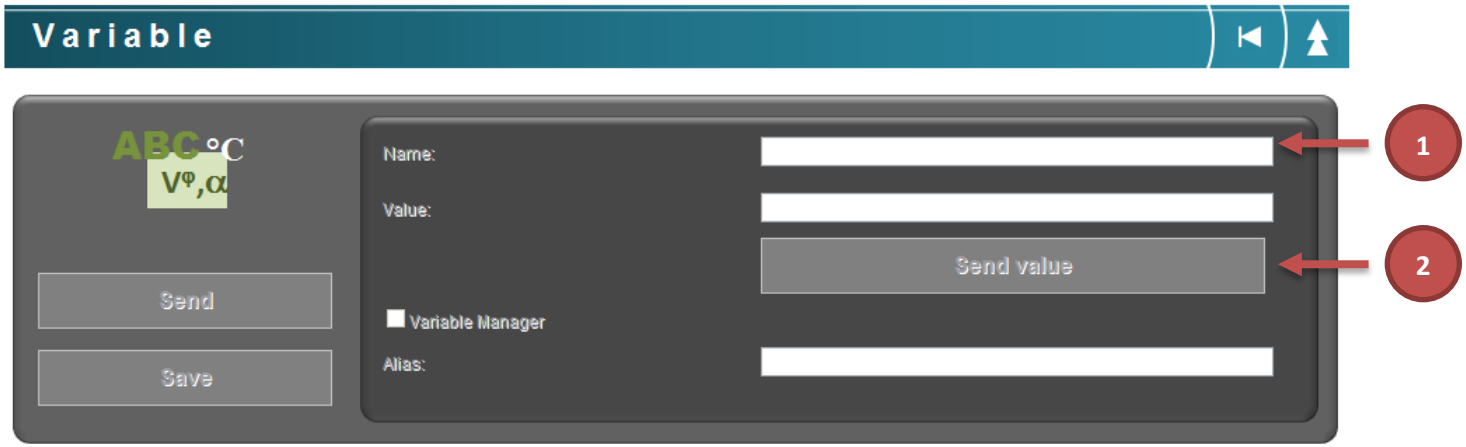
- **Configure a Date & Time data feed**

1. Enter the desired format

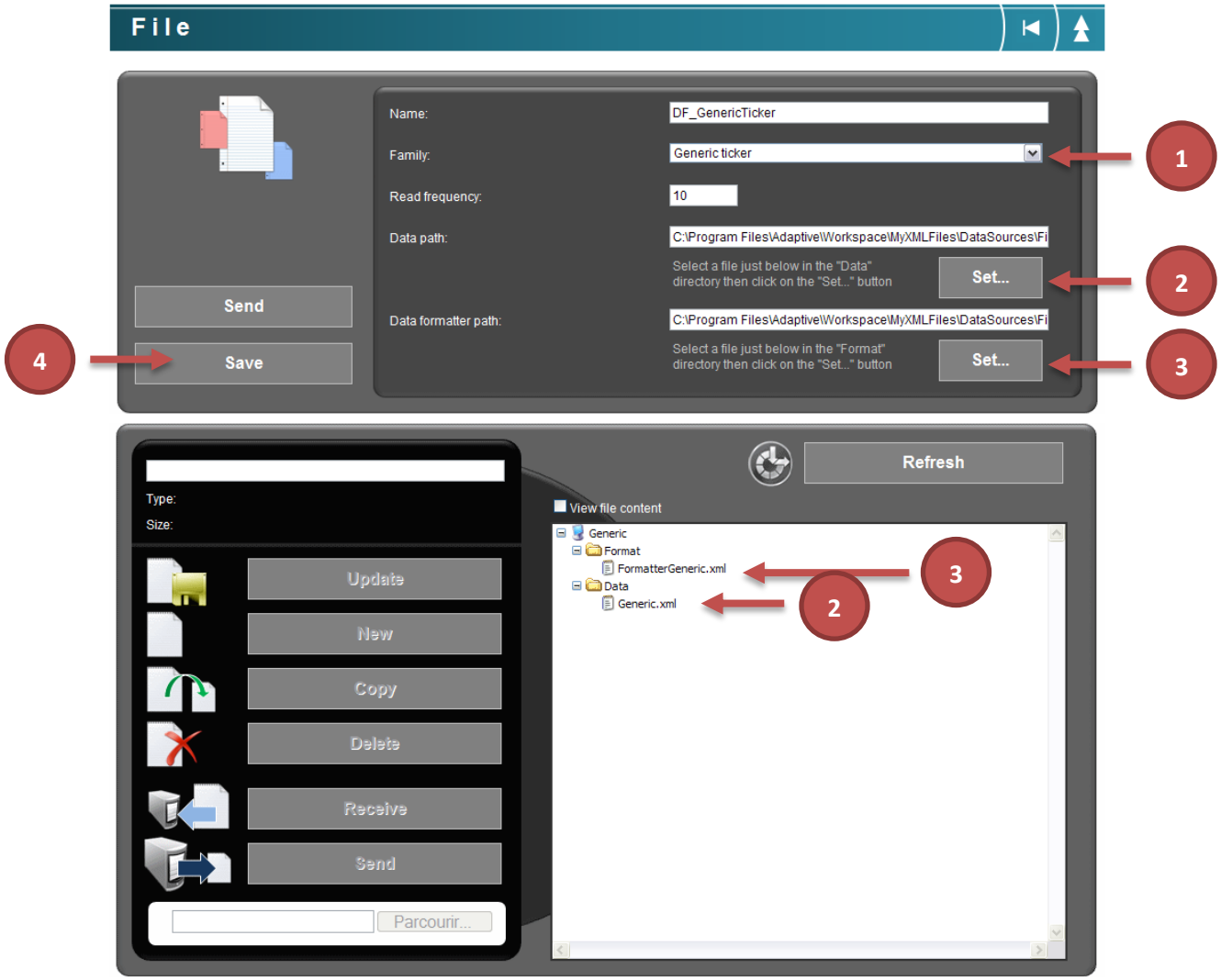


%a :	Abbreviated weekday name
%A :	Full weekday name
%b :	Abbreviated month name
%B :	Full month name
%c :	Date and time representation appropriate for locale
%d :	Day of month as decimal number (01 31)
%H :	Hour in 24-hour format (00 23)
%I :	Hour in 12-hour format (01 12)
%j :	Day of year as decimal number (001 366)
%m :	Month as decimal number (01 12)
%M :	Minute as decimal number (00 59)
%p :	Current locale's A.M./P.M. indicator for 12-hour clock
%S :	Second as decimal number (00 59)
%U :	Week of year as decimal number, with Sunday as first day of week (00 53)
%w :	Weekday as decimal number (0 6; Sunday is 0)
%W :	Week of year as decimal number, with Monday as first day of week (00 53)
%x :	Date representation for current locale
%X :	Time representation for current locale
%y :	Year without century, as decimal number (00 99)
%Y :	Year with century, as decimal number
%#c :	Long date and time representation

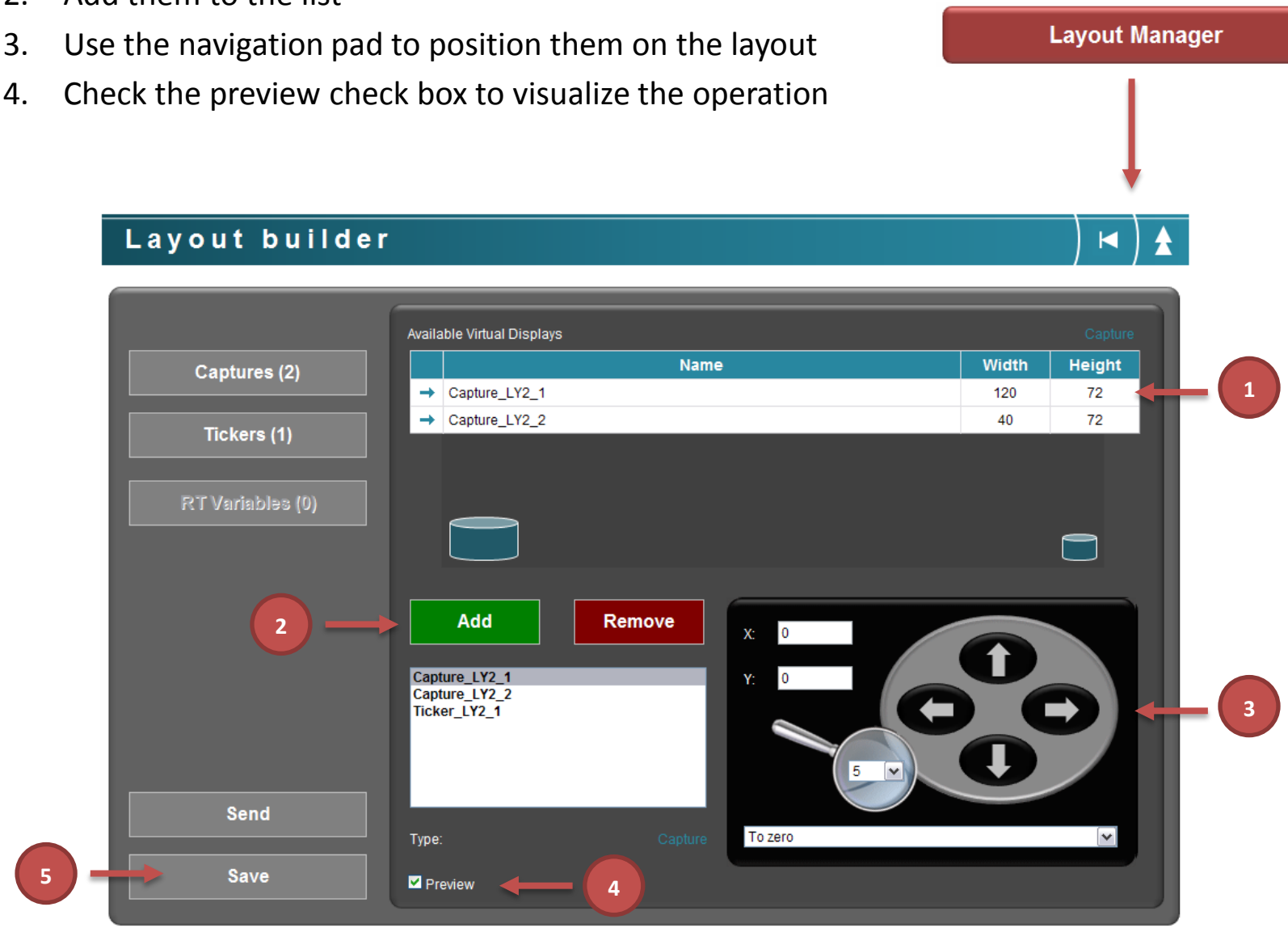
- **Configure a Variable data feed**
  1. Enter the variable's name
  2. Update the its value



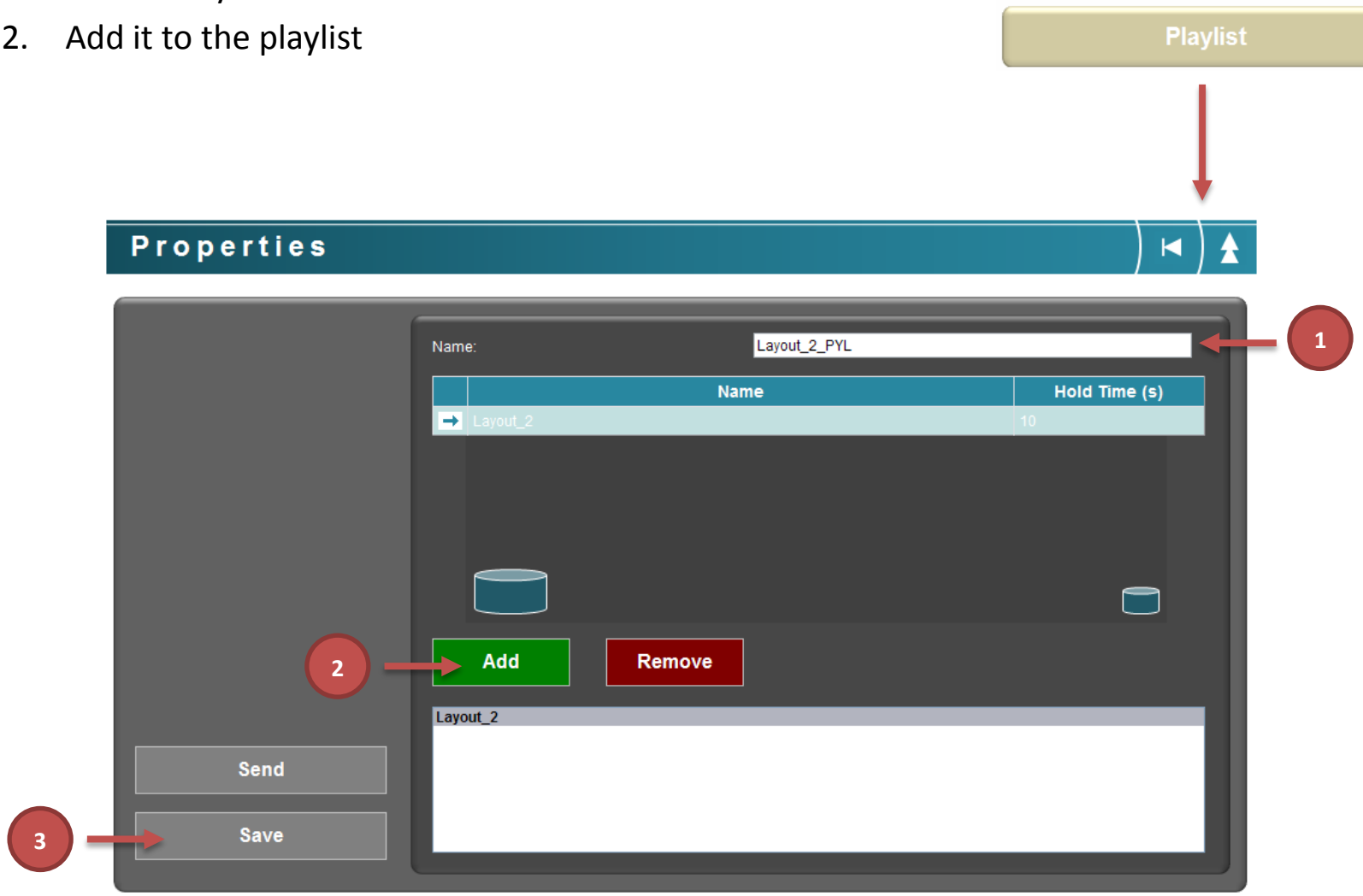
- **Configure a File data feed**
  1. Select the family (Generic, Weather, Stock Exchange)
  2. Select an .XML file in the explorer and assign it to the data path
  3. Select an .XML file in the explorer and assign it to the data formatter path



- **Build a layout**
  1. Select virtual displays among the ones available in the list
  2. Add them to the list
  3. Use the navigation pad to position them on the layout
  4. Check the preview check box to visualize the operation

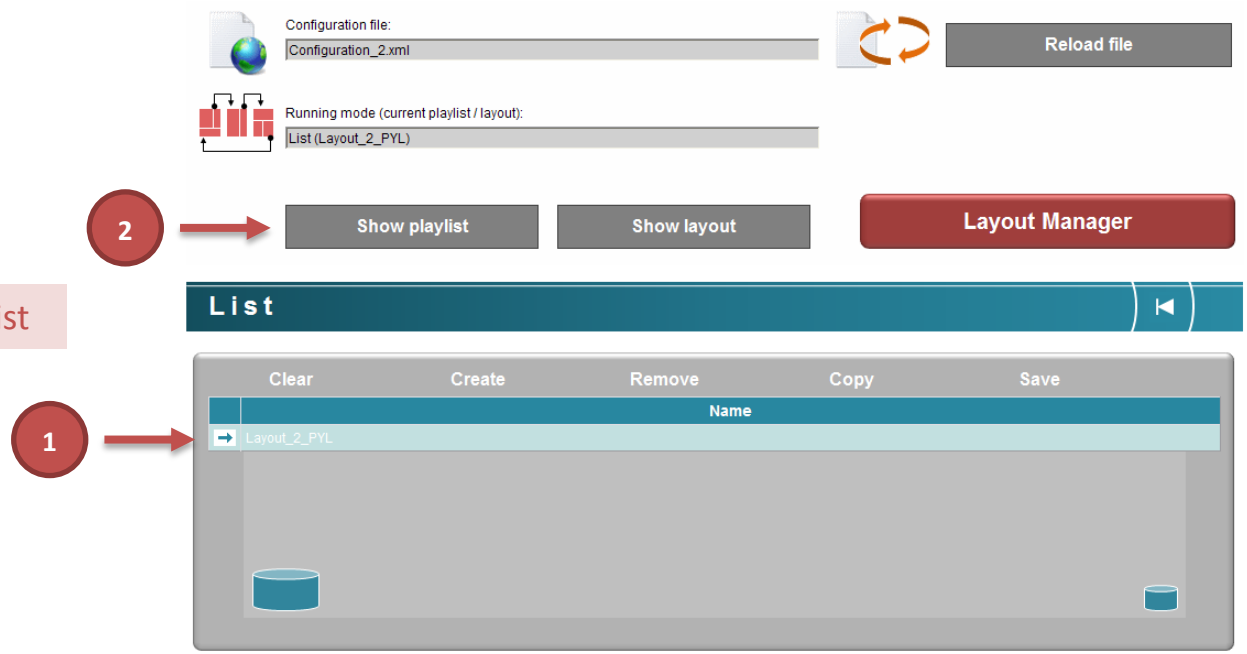


- **Build a playlist**
  1. Select a layout
  2. Add it to the playlist

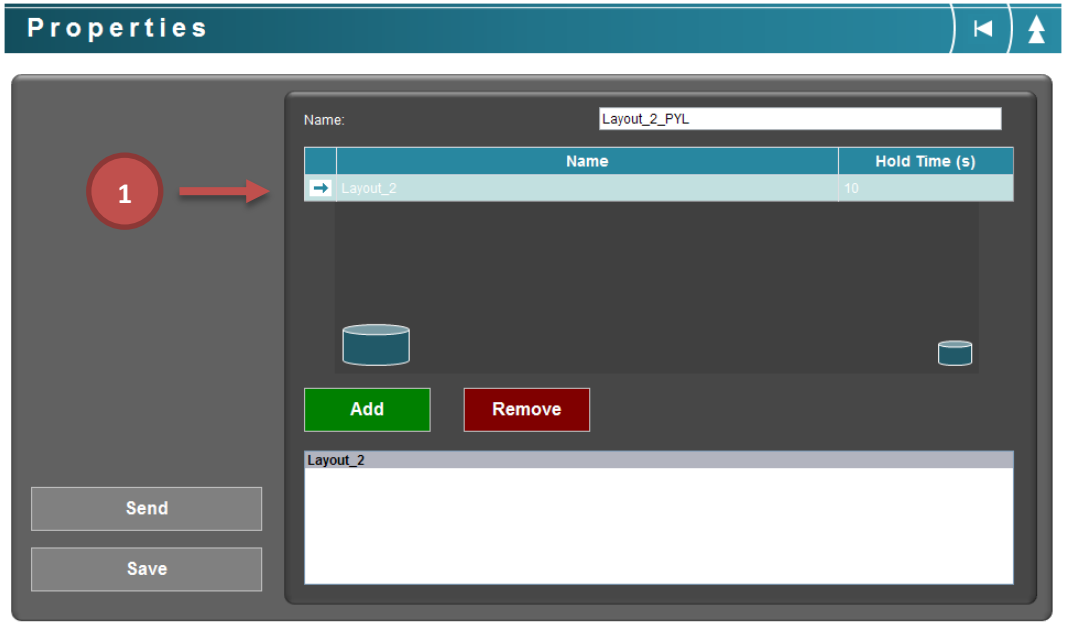


- **Show a playlist**
  1. Select a playlist
  2. Click on “Show playlist“

Show all layouts belonging to the playlist

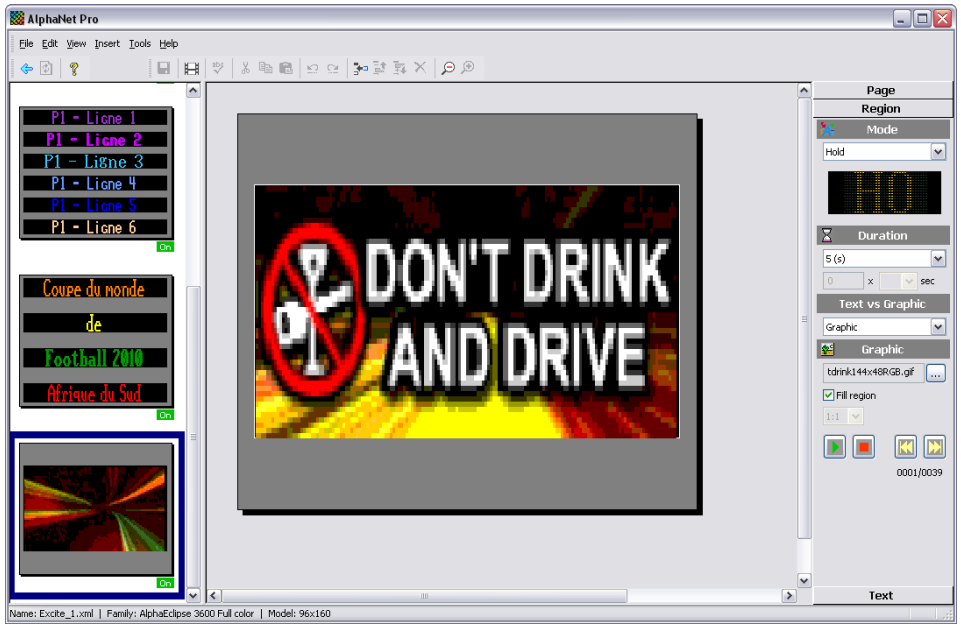


- **Show a layout**
  1. Select a layout
  2. Click on “Show layout“



VLC Based Player

- The user can, according to his immediate needs, select among a set of players provided and installed by Adaptive
  - The **ALPHA Player** plays messages (simple texts, variables, graphics & animations) created with ALPHANet Pro
  - The **Ooh!Media Player** plays messages (enhanced texts, graphics & animations) created with Ooh!Media
  - The **VLC Based Player** plays any kind of video standards created from a 3rd party application
  - The **Display Engine** captures multiples regions of the video screen and arranges them of a specific layout
  - The **Pattern Generator** plays patterns in order to perform some diagnostic operations
- All players can be enabled and activated through the Web Server and placed over the Output Window (ref. to Application Supervisor)
- The user may install and use his own player too

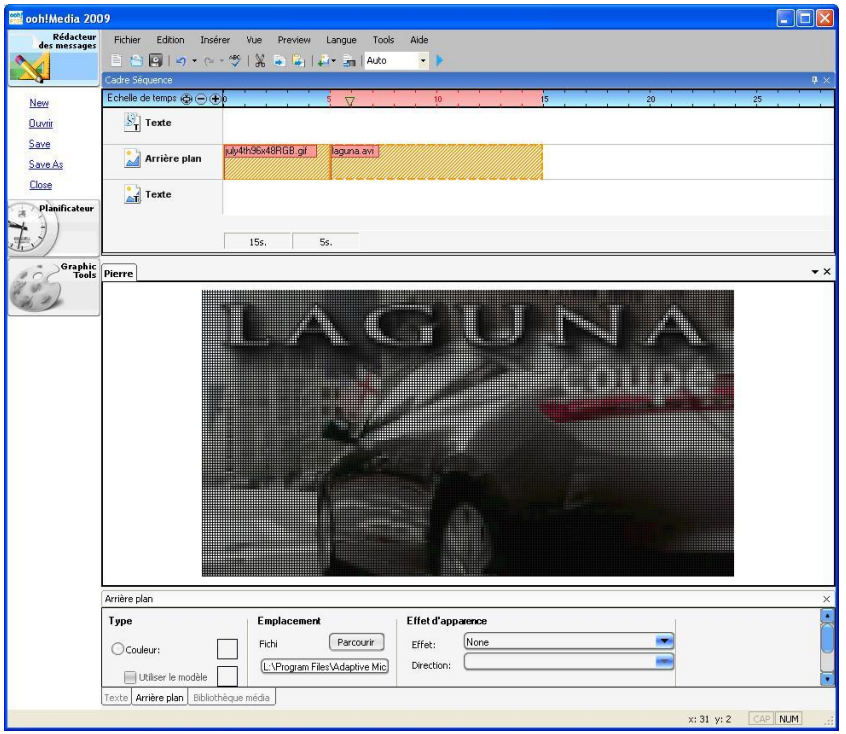


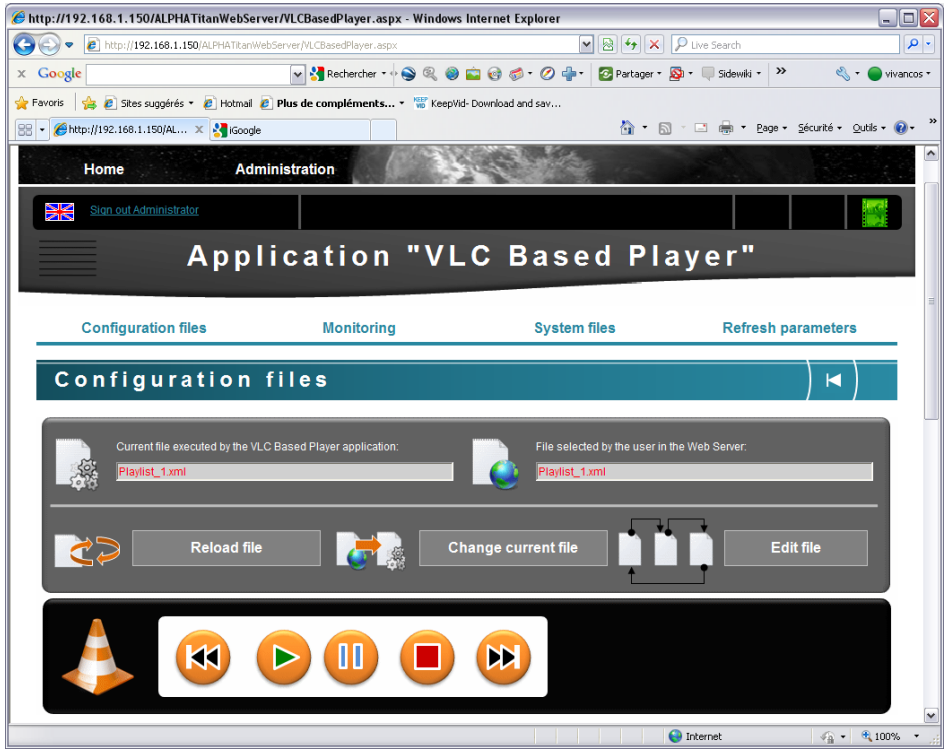
ALPHANet PRO

ALPHA Player

Ooh!Media Player

Ooh!Media

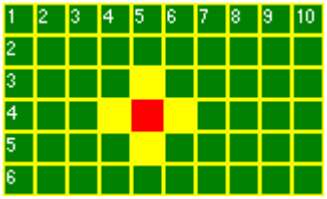


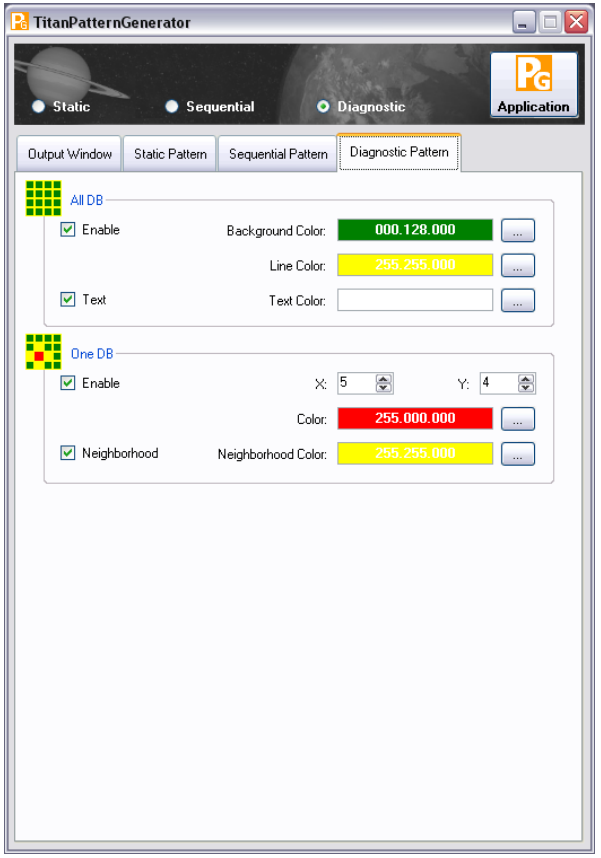


VLC Based Player

VLC Based Player

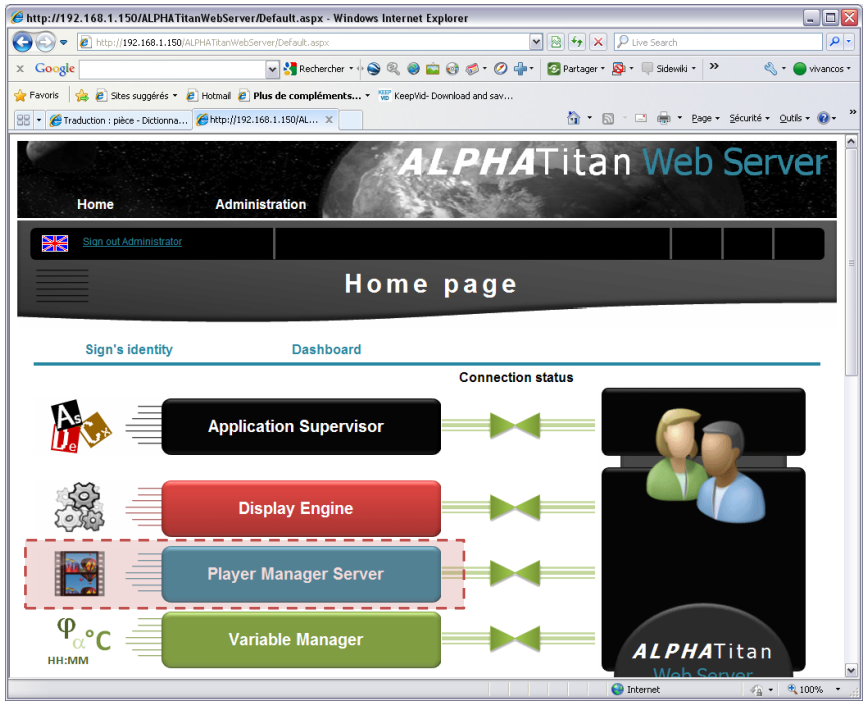
Pattern Generator





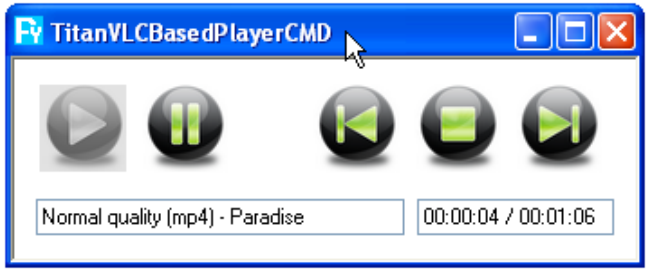
Pattern Generator

- The **VLC Based Player** is the only one which can be fully configured and controlled by the Web Server
- It plays scheduled playlists containing a list of any kind of medias
- Possibility to setup multiple time slots, days of week, start & stop dates per playlist
- Support a wide range of formats for images, movies and streaming



Web Server – Client machine

TCP/IP Commands



VLC Based Player - Controller

Headquarter USA

Milwaukee, WI

USA

Production Plant USA

Jackson, WI

USA

Production Plant Malaisie

Alor Setar, Kendah

MALAYSIA

European Subsidiary Europe

38320 Eybens

FRANCE

Office Ile de France

92300 Levallois-Perret

FRANCE

Office Sud Ouest

33240 Saint Laurent d'Arce

FRANCE