

# **MVD Player User's Guide**

Version 1.0

Copyright © 2004 Emmeskay, Inc.  
All rights reserved.

**EMMESKAY, INC.**  
44191 Plymouth Oaks Blvd.  
Suite 300  
Plymouth, Michigan 48170 USA  
[www.emmeskay.com](http://www.emmeskay.com)

Tool Support Line: [mvdsupport@emmeskay.com](mailto:mvdsupport@emmeskay.com)

## Table of Contents

<b>1. Getting Started .....</b>	<b>3</b>
1.1 What is MVD Player? .....	3
1.2 When do you use MVD Player? .....	3
1.2.1 Screenshots.....	4
<b>2 System Requirements.....</b>	<b>5</b>
<b>3 Installation Instructions.....</b>	<b>5</b>
<b>4 Launching the MVD Player .....</b>	<b>5</b>
<b>5 Quick Tutorial .....</b>	<b>6</b>
5.1 Configuration File Description.....	6
<b>6 Tool Feature Description.....</b>	<b>9</b>
6.1 Player Console Buttons.....	9
6.2 Linking to External Tools.....	12
6.3 Adding your own plots.....	14
6.4 Video Jitter .....	16

# 1. Getting Started

## 1.1 What is MVD Player?

Map, Video and Data (MVD) Player is a MATLAB<sup>®</sup> toolbox for simultaneous and synchronized visualization of several test data streams containing GPS location, video and vehicle data.

The GPS location is displayed as a Microsoft MapPoint<sup>®</sup> **M**ap

The camera captured **V**ideo is displayed using Microsoft Windows Media Player<sup>®</sup>

The vehicle **D**ata is plotted using MATLAB<sup>®</sup>

## 1.2 When do you use MVD Player?

Engineers frequently capture data streams from several sources during a vehicle test such as those from vehicle performance sensors (speed, acceleration, yaw rate, etc.), video with a camera and vehicle position data with a GPS receiver. Such a wide variety of data stream sources are used in several vehicle applications:

- In adaptive cruise control development and testing, vehicle motion is monitored by dashboard-mounted cameras, vehicle sensors (speed, acceleration etc.) and a GPS receiver. Scenarios like road curvature negotiation etc. are analyzed by reviewing maps, data plots and video.
- Driver aid systems like parallel-park assist and fully automatic self-park, use miniature video cameras mounted at multiple locations on the vehicle. Video captured from these cameras is used to design such systems.
- Telematic systems like in-dash vehicle navigation systems, primarily use GPS information to locate the position of the vehicle on maps in real-time.

Designing such systems, involves observing and analyzing data plots, trajectory maps and (possibly multiple) video in the time domain. Different tools can be used for this purpose. For example, MATLAB plots of vehicle sensor data can be used along with MapPoint Maps to analyze vehicle performance and GPS data. Frequently, such analyses require information recorded at a particular time instant. Since the tools are not (time) synchronous with each other while displaying the information on the screen, the user typically juggles between the tools, scanning for information. This is inconvenient and time consuming. This is where MVD Player comes into play. It integrates, synchronizes and presents video, map and data in a user-friendly fashion. Video, map and data are synchronized to a common clock run by the MVD Player. It also allows the user to jump to any time instant. Some of the key features of this tool are listed below.

1. Synchronous playback of multiple data sources (video, map and sensor data)
2. Configurable playback (video, map and data display can be enabled/disabled in any combination)
3. Tested and supported video file formats include AVI and MPEG2, although all formats supported by Media Player 6.1 (and above) are feasible
4. Useful playback control features - Play, Pause, Stop, Step forward, Step backward, Fast Rewind, Fast Forward, Slider bar, View/Change current time
5. Control of user created plots
6. Plot windows, display value of vehicle data corresponding to the current time

7. Saving screen positions of all the MVD Player controlled windows
8. API for linking to external tools

Screenshots of the tool in action are shown below.

### 1.2.1 Screenshots

- Player Console

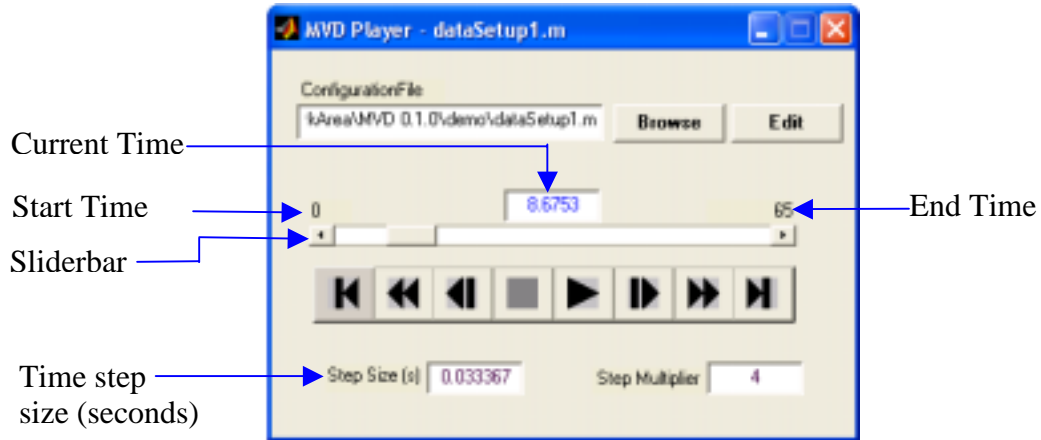


Figure 1: MVD Player Console

- Video, Map and MATLAB plots being played

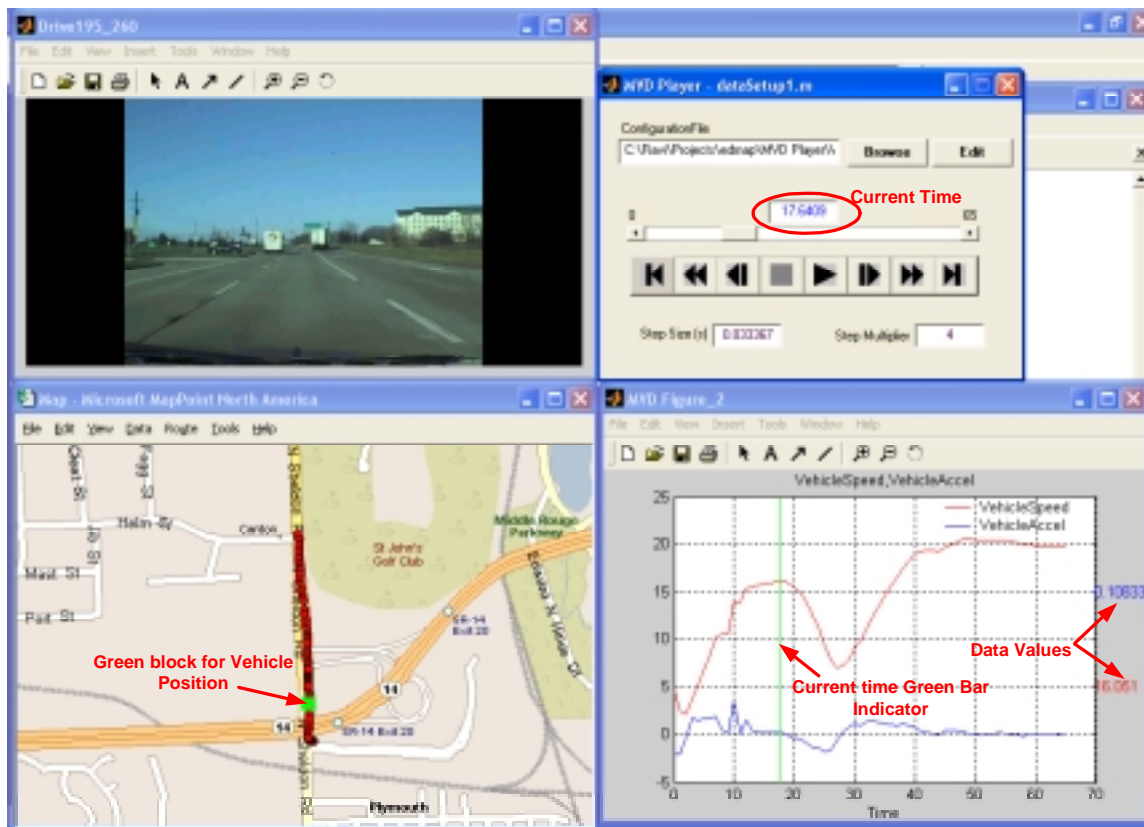


Figure 2: Desktop view of MVD Player in action

The tool also allows the user to view the data to pick the time synchronization instants in different data streams. Except for the video file, the MVD Player assumes that the user has adjusted the initial time (time = 0) in the map data and the vehicle data so that they are already synchronized. For the video file, the user can specify the start time in the configuration file.

## 2 System Requirements

- OS - Windows Xp/2000/Me
- MATLAB 6.1 or MATLAB 6.5
- Windows Media Player 6.1 and above
- Microsoft MapPoint2002

## 3 Installation Instructions.

Add the folder “MVDPlayer” and the subfolders (src, lic, sample) to the MATLAB path. The procedure to do this:

1. Open MATLAB and from Main Menu click File -> Set Path ...
2. Click on ‘Add with Subfolders’ button in the set path GUI. This will open another window titled “Browse for folder” containing a directory tree similar to one shown in the windows explored.
3. In the directory tree navigate to the folder where you have installed (copied) the MVD player files. Click on ‘OK’ and this will add the MVD player folder and the subfolders to the MATLAB path, which can be seen now in the set path GUI.
4. On closing the Path Browser window, click on ‘Yes’ to permanently include the new file paths in your MATLAB path (otherwise you have to repeat this procedure at the beginning of every MATLAB session).

**Note:** *You will also need to include your data files (video, mat, csv etc. files) into the MATLAB path to use MVD Player with your data.*

## 4 Launching the MVD Player

- From Command Prompt, enter the following command  
>> mvdplayer

**OR**

- From MATLAB Launch Pad (see below)- Double-click MVD Player under Data Fusion Toolbox. If you do not see Data Fusion Toolbox in the Launch Pad after including the program folders into MATLAB path, right-click on one of the entries in Launch Pad and pick Refresh.

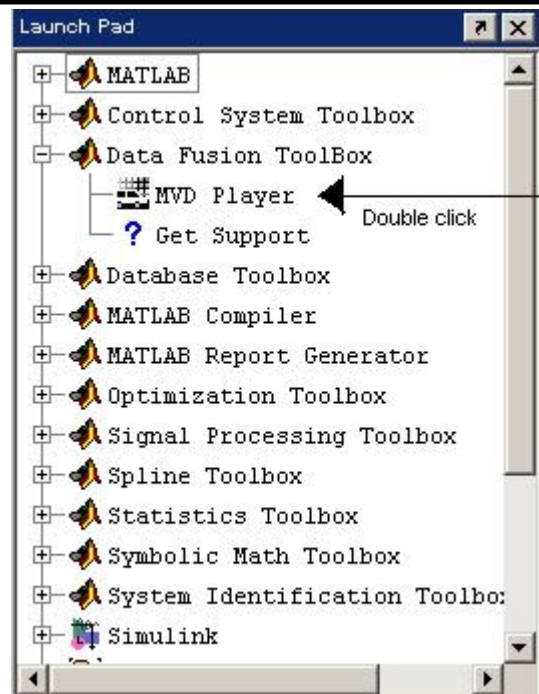


Figure 3: Starting MVD Player from MATLAB's Launchpad

## 5 Quick Tutorial

Tool visualization configuration (names of video files to play, data files for plots, etc.) is obtained from a MVD Player configuration file. This is essentially a MATLAB .m file, with specific line statements, and is described in the next section. It can be used as a template to setup your own configuration file. On loading such a configuration file, MVD Player initializes all the views. The map is initialized with red circles depicting the vehicle trajectory during the test, the video file(s) is loaded (and initialized to the first frame) into Media Player window inside a MATLAB figure and the MATLAB (or CSV) data is plotted in MATLAB figures with initial values (at time=0) displayed on the corresponding figure window.

In the next section, the configuration file is described. In the configuration file, lines in green and beginning with a % indicate MATLAB comments. Lines in blue (without an underline) indicate actual line statements that need to be copied to setup your own configuration file.

### 5.1 Configuration File Description

This section describes the demo configuration file DemoSetup.m in the /demo directory. In this setup function (see below), the contents on the left-hand side of the assignment ("=") statements should not be changed by the user because the tool expects a particular setup data structure (the fields of the variables *out* and *user* in the following m-function). Essentially, the configuration information is partitioned into the following sections. Click on the links for details.

1. [DISPLAY SETTINGS](#)
2. [VIDEO SETTINGS](#)
3. [PLOT SETTINGS](#)
4. [MAP SETTINGS](#)

- 
5. [OTHER SETTINGS](#)
  6. [EXTERNAL TOOL SETTINGS](#)

% Leave the two lines below unchanged. These define the outputs and function name to be called  
function [out,user] = dataPlaySetup

user = [];

% \*\*\*\*\* START YOUR MODIFICATIONS BELOW \*\*\*\*\*

% \*\*\*\*\* DISPLAY SETTINGS \*\*\*\*\*

% The flags below control which windows i.e. video, map and plot are displayed

out.Flags.Video = 1; % Set to 0 to prevent displaying video

out.Flags.Plot = 1; % Set to 0 to prevent displaying plots

out.Flags.Map = 1; % Set to 0 to prevent displaying maps

% \*\*\*\*\* VIDEO SETTINGS \*\*\*\*\*

% Video file name(s)

% -----

out.Video.VideoFiles = {'Drive195\_260.mpg'};

% To add more files use a comma. For ex. {'Video1.mpg', 'Video2.mpg'};

% Video file properties

% -----

% Specify the start, end times and position of the video window

% For 1st video

out.Video.VideoFileProps(1).StartTime = 0; % seconds

out.Video.VideoFileProps(1).EndTime = 65; % seconds

out.Video.VideoFileProps(1).Position = [659 45 370 320]; % [x y width height] in pixels

% -----

**% For 2nd video, it could be the following since 30 sec frame corresponds to time = 0 in other data**

**% out.Video.VideoFileProps(2).StartTime = 30;**

**% out.Video.VideoFileProps(2).EndTime = 95;**

**% out.Video.VideoFileProps(2).Position = [10 50 370 320];**

% \*\*\*\*\* PLOT SETTINGS \*\*\*\*\*

% Number of Figures for data plots

% -----

% Specify the number of Figure windows to open. Tip - each Fig. window can hold upto 6 subplots. Each  
% subplot or plot can display about 3 variables with fair granularity

out.Data.NoOfFigures = 2;

% out.Data.PlotFigure(1) indicates 1st Fig. window

% out.Data.PlotFigure(2) indicates 2nd Fig window and so on..

% Settings for 1st Figure window

% -----

% Specify the type - 'SubPlot' or 'SinglePlot'. 'Subplot' puts each variable into a separate subplot on the

% same Fig. window. 'SinglePlot' puts all variables onto a single Fig. window.

```
out.Data.PlotFigure(1).Type = 'SubPlot';
% Specify the position [x y width height] in pixels
out.Data.PlotFigure(1).Position = [-1 38 450 320];

%>>>> Settings for Subplot 1
% Specify the name of the time variable in data
out.Data.PlotFigure(1).Plot(1).TimeVar = 'T';
% File that contains time variable. Should be a .mat or .csv file
out.Data.PlotFigure(1).Plot(1).TimeVarFile = 'DFT_data_195_260.csv';
% Name of 1st variable. 2nd variable would be ...Plot(1).Vars(2).Var = 'varname'
out.Data.PlotFigure(1).Plot(1).Vars(1).Var = 'VehicleSpeed';
% File that contains 1st variable. Should be a .mat or .csv file
out.Data.PlotFigure(1).Plot(1).Vars(1).File = 'DFT_data_195_260.csv';

% >>>> Settings for Subplot 2. Similar to above settings.
out.Data.PlotFigure(1).Plot(2).TimeVar = 'T';
out.Data.PlotFigure(1).Plot(2).TimeVarFile = 'DFT_data_195_260.mat';
out.Data.PlotFigure(1).Plot(2).Vars(1).Var = 'VehicleHeading';
out.Data.PlotFigure(1).Plot(2).Vars(1).File = 'DFT_data_195_260.mat';

% Settings for 2nd Figure window
% -----
% Specify the type - 'SubPlot' or 'SinglePlot'. 'Subplot' puts each variable into a separate subplot on the
% same Fig. window. 'SinglePlot' puts all variables onto a single Fig. window..
out.Data.PlotFigure(2).Type = 'SinglePlot';
% Specify the position [x y width height] in pixels
out.Data.PlotFigure(2).Position = [225 37 450 320];

% >>>> Settings for SinglePlot
% Specify the name of the time variable in data
out.Data.PlotFigure(2).Plot.TimeVar = 'T';
% File that contains time variable. Should be a .mat or .csv file
out.Data.PlotFigure(2).Plot.TimeVarFile = 'DFT_data_195_260.csv';
% Name of 1st variable.
out.Data.PlotFigure(2).Plot.Vars(1).Var = 'VehicleSpeed';
% File that contains 1st variable. Should be a .mat or .csv file
out.Data.PlotFigure(2).Plot.Vars(1).File = 'DFT_data_195_260.csv';
% Name of 2nd variable.
out.Data.PlotFigure(2).Plot.Vars(2).Var = 'VehicleAccel';
% File that contains 2nd variable. Should be a .mat or .csv file
out.Data.PlotFigure(2).Plot.Vars(2).File = 'DFT_data_195_260.csv';

% ***** MAP SETTINGS *****

% Map File Source - 'mat' or 'csv'
% If the data is in csv format, the names of the variables are ignored.
% Instead, it is expected that the csv file will contain time,Lat,Long as
% three comma separated columns with NO header rows
out.Map.MapFileSource = 'mat';
out.Map.Position = [222 4 370 320]; % Map window position [x y width height] in pixels
```

```
% Map Data Variables
out.Map.LatData = 'NavVehPosLat';    %Latitude in deg. Negative for southern hemisphere
out.Map.LongData = 'NavVehPosLong'; %Longitude in deg. Negative behind GMT
out.Map.Time = 'T';                 % Time variable name. Should be in seconds

% Map file name
% If MapFileSource was 'csv', then you would have out.Map.MapDataFile = 'route1.csv';
out.Map.MapDataFile = 'DFT_data_195_260.mat';
out.Map.UpdateTime = 1;            % Map refresh time in seconds

% ***** OTHER SETTINGS *****
% Step Size and Multiplier
out.General.StepSize = 1/29.97;
out.General.StepMultiplier = 4;

% Pause Time scaling for reducing video jitter. More the pause, less the jitter, but slower the video
% Value should be between 0 and 1
out.Video.PauseTime = 0.8;

% ***** EXTERNAL TOOL SETTINGS *****



% Specify the relative path of the .m file, which lists the user applications to be loaded
% Refer to the section on Linking External Tools for details. Uncomment the line below for
% a demo of opening a Simulink model as an external tool

% user.application.path = '../demo/customsetup.m';
```

## 6 Tool Feature Description

### 6.1 Player Console Buttons

This section describes the features in the MVD Player Console shown in Figure 1.

- 1. Browse button**   
This button is used to select a setup file (m file).
- 2. Configuration File**  
Should display the m file selected by browse button.
- 3. Edit button**   
This button is used to edit the selected m file.
- 4. Start Time**  
Player start time (cannot be edited). Default value is 0.
- 5. End Time**  
Player end time (cannot be edited). Maximum of the specified video, plot and map time.

## 6. Current Time

Editable box that displays the time where the MVD Player is.

## 7. Slider Bar

Minimum value will be 0. Max value will be the “EndTime” value. Clicking the end arrows will advance MVD Player's time by the specified “StepSize”. Clicking in the empty area between the slider button and the ends, advances MVD Player's time by a value equal to the product of “StepSize” and “StepMultiplier”. While playing, the slider bar's position changes synchronously with the current time.



## 8. Play button

Play button to play the video, to plot the data and to show the location in the map.

The slider bar will be deactivated when the play button is pressed. It would be moving on the screen but the user cannot take control of it.

The data will be played from the Current Time value displayed in the 'Console GUI'

The Current Time will keep advancing forward with the Step Size specified by the user. This value is 'deactivated' in the sense that the user cannot change it manually.

The **green indicator bar** in all the plot windows will start advancing with x-location corresponding to the Current Time.

The data values are displayed in a text box (inside the figure) in each of the subplots corresponding to the Current Time.

The video will advance smoothly. A **green block** will move in the MapPoint window to the Latitude, Longitude position corresponding to the Current Time.

The info in all windows will "freeze" at their individual final times (if data is continued to be played beyond that).

The operation will stop when the Current Time equals the Final Time in the Player Console GUI.



## 9. Stop button

The Current time will reset to Start Time (0).

The Scroll bar will reset to the beginning.

On pressing “Play” after Stop, info in all windows will start from Start Time.



## 10. Pause button

The Current time will pause at the last value in the play mode.

The scroll bar will pause at its current location.

The information on all the windows will pause at the values corresponding to the Current time.

### 11. Fast-forward button



MVD Player plays forward at a faster rate with a time step = Step Size\*Step Multiplier.  
Sliderbar, Current time, StepSize and StepMultiplier are deactivated.

The operation will stop when the Current time equals the EndTime.

### 12. Fast-rewind button



MVD Player rewinds with a time step = Step Size\*Step Size Multiplier  
Sliderbar, Current time, Stepsize and StepMultiplier will be deactivated.

The operation will stop when the Current time equals the StartTime.

### 13. Step-forward button



Works only after clicking the Pause or Stop button. Each click of this button advances MVD Player, starting from the current time, by a time step = Step Size\*Step Size Multiplier

Stop button remains deactivated in this mode

The Sliderbar will move in the forward direction. The operation will stop when the Current Time equals the Final Time in the Console GUI.

### 14. Step-backward button



Works only after clicking the Pause or Stop button. Each click of this button **rewinds** MVD Player, starting from the current time, by a time step = Step Size\*Step Size Multiplier

Stop button remains deactivated in this mode

The Sliderbar will move in the backward direction. The operation will stop when the Current Time equals the Start Time in the Console GUI.

### 15. Jump to End button



The Current time is set to the End Time.

The info in all windows jump to their respective EndTimes

The slider bar jumps to the end position

### 16. Jump to Beginning button



The Current time get set to the Start Time.

The info in all windows jump to the StartTime which is 0

The scroll bar jumps to the beginning position

### 17. Step Size and Step Multiplier

---

These edit box values determine the MVD Player's clock time step. The input values, interpreted as seconds, can be a decimal or fraction.

For the **Play** action

$$\text{time step} = \text{StepSize}$$

For the **FastForward**, **FastRewind**, **StepBackward** and **StepForward** actions

$$\text{time step} = \text{StepSize} * \text{StepMultiplier}$$

The default Step Size is 1/29.97, which actually corresponds to one of the supported MPEG video file frame rate of 29.97



## 18. Continuous Play

Context Menu associated with Play button.

If checked, Player will play continuously. If unchecked, Player will stop when the end time is reached. By default, this option checked. It can be checked or unchecked only when the player is in pause or stop state.

## 6.2 Linking to External Tools

MVD Player allows the user to plug-in custom tools. This is useful when the user has proprietary tools whose results need to be synchronously viewed with MVD Player views. A user can link any external tool to MVD Player, by specifying [User Defined Tool Settings](#) in the [Configuration File](#). The figure below shows a schematic perspective of this mechanism and is also explained below. You will also need to refer to the following files in the **demo** directory

1. DemoSetup.m
2. customsetup.m
3. user\_simApp.m

The linking process involves the following 3 steps

1. Edit configuration file (see DemoSetup.m) to point to cutomsetup.m  
Include the line  
`user.application.path = './demo/customsetup.m';`  
in EXTERNAL TOOL SETTINGS. Note that this uses a relative path to specify the file.

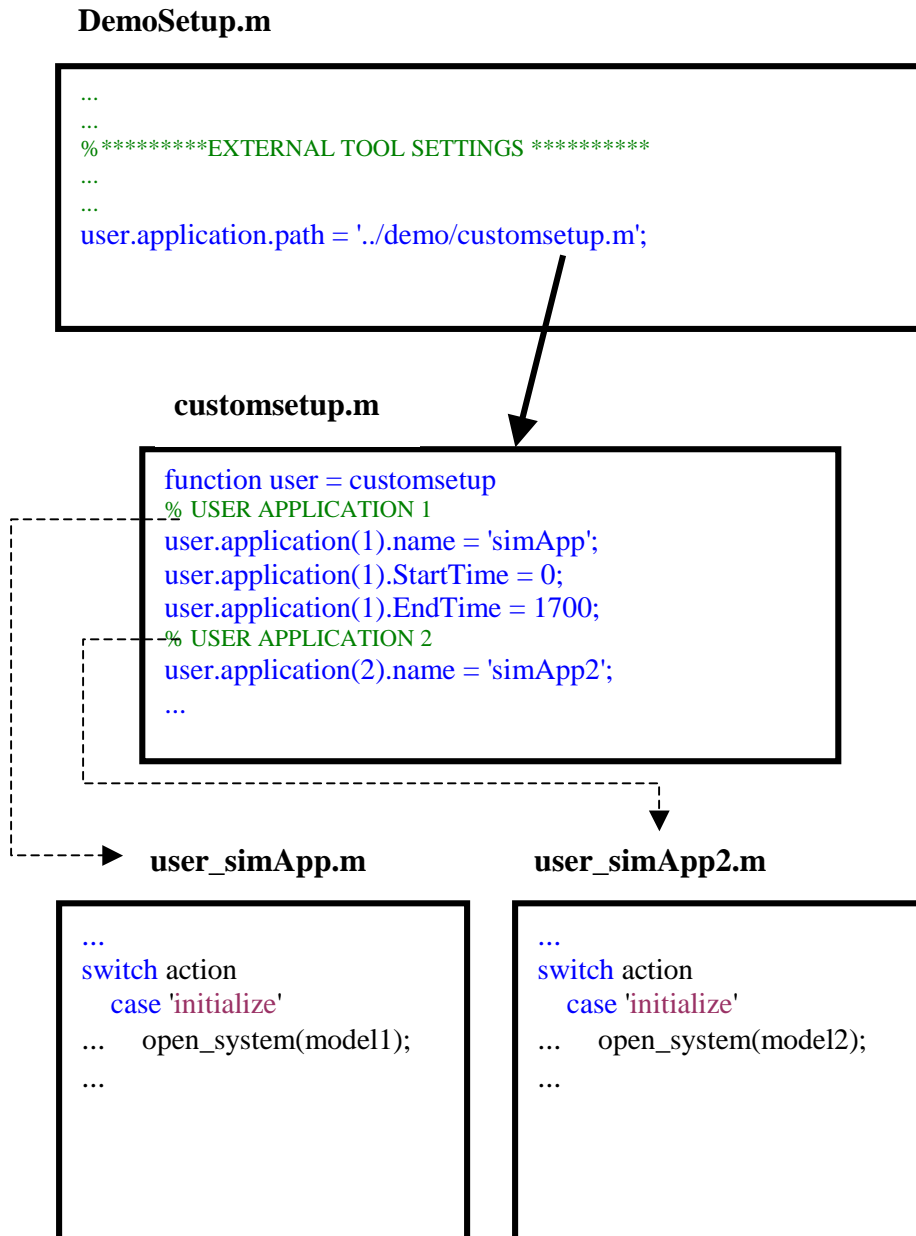


Figure 4: Overview of linking external tools

2. Edit customsetup.m to list user applications  

```
% USER APPLICATION 1  
user.application(1).name = 'simApp';  
user.application(1).StartTime = 0;  
user.application(1).EndTime = 1700;
```
3. Create user\_<AppName>.m files and define the following case statements. See **user\_simApp.m** in the **demo** directory for details.
  - 'open'  
Should be used to open the application. This will be called from MVD Player when the setup file is loaded.

- 'initialize'  
Should be used to initialize the application. This will be called from MVD Player when its time variable hits a value of 0.
- 'getMaxTime'  
Should return the maximum time for which the application will run. This will be called from MVD Player when the setup file gets initialized.
- 'updateTime'  
Should be used to take care of updating the application for given time. The time will be passed from MVD Player.
- 'terminate'  
This will be called from MVD Player when the max time has been reached.
- 'close'  
Should be used to close the application. This will be called from MVD Player when the player console is closed or when the setup file is changed.
- 'refresh'  
Should be used to refresh the application. User should call it from the command line. The call should be like  
`DataFusionTool('user', 'refresh', varargin);`

### 6.3 Controlling User Defined Plots

The command *mvdsbplot* can be used to control any user defined plot from MVD Player. It can be used from both the GUI and the command prompt. This command assumes that the x-axis on the user defined plot is the same time variable that is used by MVD player as the time for synchronizing the map, video and data.

- **From command prompt**

The function call could be any of the following

```
>> mvdsbplot
```

This command adds the last selected Figure window to the list of figures that MVD Player controls. Make sure that you select the correct Figure window before issuing this command.

```
>> mvdsbplot(<figure handle>)
```

Use this command if you know the <figure handle> of the plot, that you want MVD Player to control. The <figure handle> can be obtained by selecting the Figure window and typing `gcf` at the command prompt.

```
>> mvdsbplot('remove')
```

This removes the last selected Figure window from MVD Player's control. Make sure that you select the correct Figure window before issuing this command.

```
>> mvdsbplot(<figure handle>,'remove')
```

Use this command if you know the <figure handle> of the plot that you would like to remove. The <figure handle> can be obtained by selecting the Figure window and typing `gcf` at the command prompt.

- **From GUI**

If the user plotted figure is not named, then MVD Player gives a name like `'non_mvdfigure_1'`;

In the context menu of the configuration file edit box, user can find a menu named 'Control Plots'. The submenu of the 'Control Plots' menu will list the user-plotted figures. See the figure below

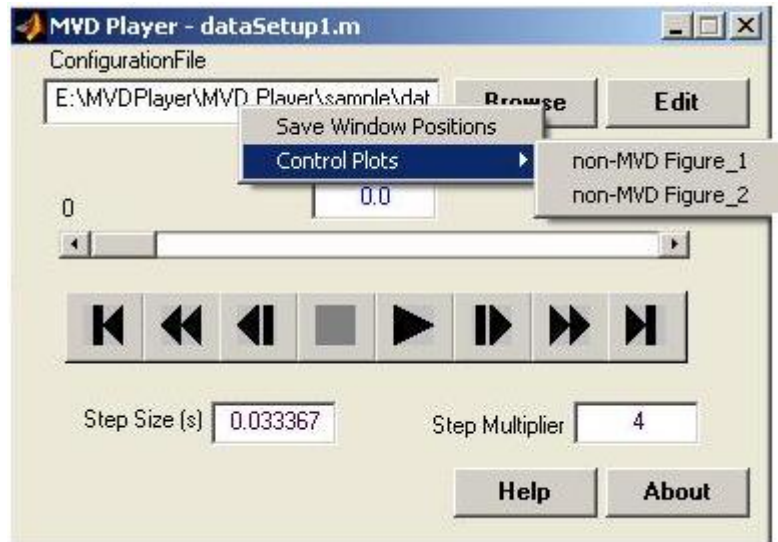


Figure 5: List of user created plots not controlled by MVD Player

When any of the figures is selected, a tick mark appears before it and that figure will be added to the MVD Player's list of figures. See the figure below. Clicking on it again, will remove the Figure from MVD Player's list of figures and also the tick mark.

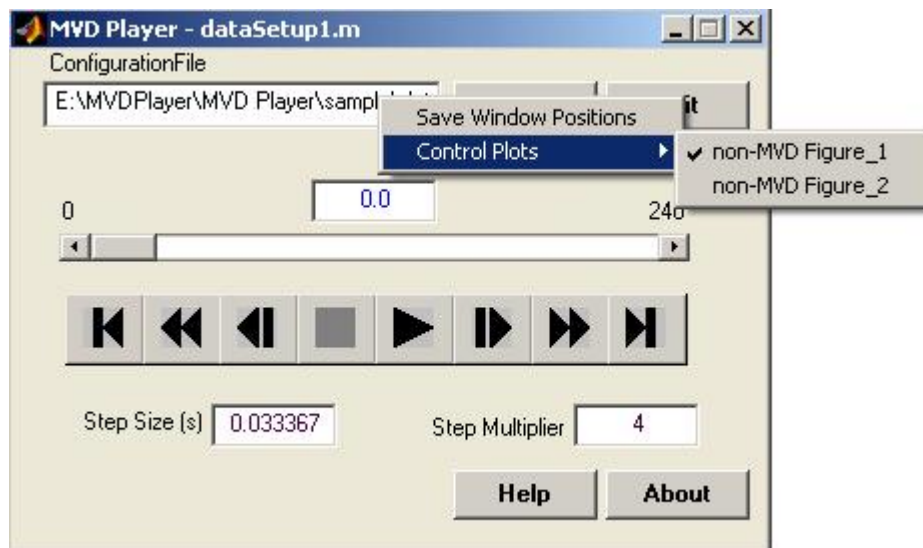


Figure 6: Adding user created plot(s) to MVD Player

## 6.4 Video Jitter

Depending upon the Operating System on the PC and available memory (RAM), there might be some jitters in the Media Player embedded figure window. It is because of the insufficient time to update the window. So, an optional field is included in the setup file.

`out.Video.PauseTime = 0.8;`

It can have a value between 0 and 1 which is used to scale the pause time. The user can adjust this parameter value to suit individual machine performance.

MATLAB<sup>®</sup>, Simulink<sup>®</sup> and Stateflow<sup>®</sup> are Registered Trademarks of The MathWorks, Inc., 24 Prime Park Way, Natick, MA 01760-1500. Visit [www.mathworks.com](http://www.mathworks.com) for details.

MapPoint<sup>®</sup> and Windows Media Player<sup>®</sup> are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.