# Argent BoardRoom Framework

**Confidential Information** 

#### Introduction

Argent BoardRoom is a framework that can load DLLs that implement Argent BoardRoom's interfacing API's. Other than interfacing, the DLL is free to do any logic the programmer wants.

#### Data Flow Between Framework And Registered DLL

- When Argent BoardRoom starts, it reads the DLL inventory from the SQL table BV\_REGDLL.
- For each registered DLL, the framework dynamically loads the DLL, calls API 'GetBVTreeImage' to get the image handle, and then composes the tree structure in the 'Registered DLLs' tab in the left panel.



- Similarly, Argent BoardRoom reads the screen and playlist inventory from SQL tables BV\_SCREEN and BV\_PLAYLIST respectively.
- When a user selects a screen definition, for each frame hosting a Registered DLL, Argent BoardRoom calls API 'InitializeBVDLL', and passing in the definition parameter stored in SQL database to initialize the frame.

#### Data Flow Between Framework And Registered DLL

 When a user does 'Design Currently Selected Frame', Argent BoardRoom calls API 'ShowBVDLLProperties' to bring up a property dialog box etc to define the parameters.

The API is responsible for informing the framework if some properties are changed. This is done by sending the message WM\_USER+100 to framework window. (See sample code)

If the API returns FALSE (0), no change is required and nothing more to do.

If the API returns TRUE (1), some changes exist. It is the DLL that is responsible for applying the changes to the current framework. The framework is only responsible for remembering changes that have been made.

- When the user is leaving the screen definition that has been changed, the framework calls API 'SaveBVDLLProperties' to allow the Registered DLL to dump out the properties to a temporary external file, and then the framework reads information from the file, saves into the database and deletes the file.
- When a user has left a screen definition, the framework calls API 'UninitializeBVDLL' to clean up the internal data structure, and mainly destroy the child windows for each Registered DLL used in the screen.
- When the framework starts to play a screen, it is similar to the design time, except passing in a different Boolean flag. The framework calls API 'InitializeBVDLL' to initialize each frame within the screen.
- When the framework is done with the screen, either because the next screen in a playlist needs to run, or a user stops it, the framework calls API 'UninitializeBVDLL' to clean up.
- If the screen is played as part of a playlist, the framework calls the API 'GetBVDLLState' for each frame to make sure the whole screen is in "ready" mode before moving to the next screen.

#### Appendix A – Argent BoardRoom SQL Tables

#### Registered DLL Table (BV\_REGDLL)

Column	Туре	Size	Description
UUID	varchar	36	Unique identifier and primary key
CREATE_TIME	datetime		Record creation time
MODIFY_TIME	datetime		Last modify time
NAME nvarchar		256	Name
DLL_PATH	nvarchar	512	DLL file path
FOLDER_PATH nvarchar		1024	Folder path within the tree structure
DESCRIPTION ntext			Description
OWNER nvarchar		256	Record owner
CRC_LOW	int		CRC low 16-bit
CRC_HIGH	int		CRC high 16-bit

#### Appendix A – Argent BoardRoom SQL Tables

#### Screen Table (BV\_SCREEN)

Column	Туре	Size	Description
UUID	varchar	36	Unique identifier and primary key
CREATE_TIME	datetime		Record creation time
MODIFY_TIME	datetime		Last modify time
NAME nvarchar		256	Name
PARAM ntext			Parameters
IMAGE_DLL nvarchar		256	DLL name used for the tree item image
FOLDER_PATH nvarchar		1024	Folder path within the tree structure
DESCRIPTION ntext			Description
OWNER nvarchar		256	Record owner
CRC_LOW	int		CRC low 16-bit
CRC_HIGH	int		CRC high 16-bit

#### Appendix A – Argent BoardRoom SQL Tables

#### Playlist Table (BV\_PLAYLIST)

Column	Туре	Size	Description
UUID	varchar	36	Unique identifier and primary key
CREATE_TIME	datetime		Record creation time
MODIFY_TIME	datetime		Last modify time
NAME nvarchar		256	Name
SCREENS	ntext		Screens in the playlist
WAIT_TIME	int		Default wait time between screens
IMAGE_DLL nvarchar		256	DLL name used for the tree item image
FOLDER_PATH nvarchar	-	1024	Folder path within the tree structure
DESCRIPTION ntext			Description
OWNER nvarchar		256	Record owner
CRC_LOW	int		CRC low 16-bit
CRC_HIGH	int		CRC high 16-bit

#### Argent BoardRoom Sample DLL

Argent has a sample DLL that can be implemented as a regular DLL or MFC extension DLL using any modern programming language.

Note: If the Registered DLL is a MFC extension DLL, it should be implemented using VS2005 (SP1) to avoid MFC DLL conflicts between the framework and DLL.