Contents

[Developer Quick Start Guide 2](#_Toc352961420)

[Create a New Game from the Template 2](#_Toc352961421)

[Build and Launch the Casino 4/Flash Client (Using the Client Development Harness) 2](#_Toc352961422)

[Build and Launch the Casino 5/HTML5 Client (Using the Client Development Harness) 2](#_Toc352961423)

[How Do I…? 3](#_Toc352961424)

[Get Started Learning the Casino 4 API for building Flash games? 3](#_Toc352961425)

[Get Started Learning about the Casino 5 HTML5 Framework? 3](#_Toc352961426)

[Get Started Learning the Casino 5 API for building HTML5 games? 3](#_Toc352961427)

[Remove the Loading Screen When My Game is Ready? 3](#_Toc352961428)

[Get the Latest Balance From the Server? 3](#_Toc352961429)

[Update the On-Screen Balance? 4](#_Toc352961430)

[Update the On-Screen Wager? 4](#_Toc352961431)

[Send a Request to the Server? 5](#_Toc352961432)

[Get the Betting Limits for My Game? 5](#_Toc352961433)

[Get the Currency Code in Use? 6](#_Toc352961434)

[Get the Language in Use? 6](#_Toc352961435)

[Format a Value for Currency Display? 6](#_Toc352961436)

[Use a Preloaded Asset in My Game? 6](#_Toc352961437)

# Developer Quick Start Guide

## Create a New Game from the Template

Creates a new “empty” game project with partial game-client implementations of both the Casino4/Flash and Casino5/HTML5 game clients. These template clients can be loaded and will run.

1. Open a command prompt and navigate to the GameDevelopmentPlatform/gametemplate directory
2. Run ant –f **gamegenerator**.xml
3. When prompted, enter the Start Game Name (ex. MySlotGame). Alpha-numeric, no spaces.
4. When prompted, enter the Display Game Name (ex. My Slot Game). Spaces allowed.
5. The game is now created at GameDevelopmentPlatform/gametemplate/<Start Game Name>. Move this directory to the preferred location on your system.

## Build and Launch the Casino 4/Flash Client (Using the Client Development Harness)

This process is only applicable for building Flash-based games.

1. Open a command prompt and navigate to the game project’s client/casino4/harness directory.
2. Run ant build.standalone to build the game.
3. Run ant –f webserver.xml start to start the Tomcat webserver.
4. Run ant launch.standalone to launch the game. The game will open in the default browser. A brief loading screen will be displayed followed by a black screen with the Amaya interface bars at the top and bottom. Begin adding code to the game and repeat the build.standalone step to see the changes.
5. Refer to the client/casino4/harness/readme.txt file for full information about the client development harness.

## Build and Launch the Casino 5/HTML5 Client (Using the Client Development Harness)

This process is only applicable for building HTML5-based games.

1. Open a command prompt and navigate to the game project’s client/casino5/harness directory.
2. Run ant build.standalone to build the game.
3. Run ant –f webserver.xml start to start the Tomcat webserver.
4. Run ant launch.standalone to launch the game. The game will open in the default browser. A brief loading screen will be displayed followed by a black screen with the Amaya interface bar at the bottom. Begin adding code to the game and repeat the build.standalone step to see the changes.
5. Refer to the client/casino4/harness/readme.txt file for full information about the client development harness.

# How Do I…?

### Get Started Learning the Casino 4 API for building Flash games?

Unzip the GameDevelopmentPlatform\documentation\client\casino4\libs\api\docs.zip and open index.html. A good place to start is the com.cwh.casino4.api.GameServices class.

Unzip the GameDevelopmentPlatform\documentation\client\casino4\libs\sdk\docs.zip and open index.html. These are additional classes that may be used to make game development easier. A good place to start is the com.cwh.casino4.sdk.singleplayer.game.V2Game class as games typically extend this class.

### Get Started Learning about the Casino 5 HTML5 Framework?

Read the Casino5.docx file in the GameDevelopmentPlatform\documentation\client\casino5 directory.

### Get Started Learning the Casino 5 API for building HTML5 games?

Unzip the GameDevelopmentPlatform\documentation\client\casino5\libs\api\docs.zip and open index.html. A good place to start is the amaya.GameServices class.

Unzip the GameDevelopmentPlatform\documentation\client\casino5\libs\sdk\docs.zip and open index.html. A good place to start is the amaya.sdk.singleplayer.V2Game class. This is a mix-in class used to quickly start a game.

### Remove the Loading Screen When My Game is Ready?

**Casino 4:**

var gs:GameServices = new GameServices();  
gs.removeLoader();

**Casino 5:**

var gs = new amaya.GameServices();  
gs.removeLoader();

### Get the Latest Balance From the Server?

Set up a listener on an instance of the GameServices object for the GameFinanceUpdateEvent.

**Casino 4:**

var gs:GameServices = new GameServices ();  
gs.addEventListener(GameFinanceUpdateEvent.GAME\_FINANCE\_UPDATE, this.onGameFinanceUpdate);

private function onGameFinanceUpdate ( event:GameFinanceUpdateEvent ) : void {

trace( event.balance );  
}

**Casino 5:**

var gs = new amaya.GameServices();  
gs.addEventListener( amaya.api.events.GameFinanceUpdateEvent.GAME\_FINANCE\_UPDATE, onGameFinanceUpdate );

function onGameFinanceUpdate ( event ) {  
 console.log(event.data.balance);  
}

### Update the On-Screen Balance?

It is the game’s responsibility to set the balance in the bottom bar as only the game is aware when the game is in the proper state to display winnings, bets etc.  
  
The framework will handle formatting the display in the player’s currency.  
  
**Casino 4:**

var gs:GameServices = new GameServices();  
gs.setDisplayBalance( 100 );

**Casino 5:**

var gs = new amaya.GameServices();  
gs.setDisplayBalance( 100 );

### Update the On-Screen Wager?

The framework will handle formatting the display in the player’s currency.

**Casino 4:**

var gs:GameServices = new GameServices();  
gs.setDisplayWager( 100 );

**Casino 5:**

var gs = new amaya.GameServices();  
gs.setDisplayWager( 100 );

### Send a Request to the Server?

**Casino 4:**

var myRequest:V2Request = new V2Request(“MyRequestName”);

myRequest.appendChildToBody(new XML('<freeSpins total="'+ total +'" used="'+ used +'"/>'));

var gs:GameServices = new GameServices();  
gs.sendV2XMLRequest( myRequest.RequestData );

**Casino 5:**

var myRequest = new amaya.sdk.singleplayer.V2Req(“MyRequestName”);

myRequest.append('<freeSpins total="'+ total +'" used="'+ used +'"/>');

var gs = new amaya.GameServices();  
gs.sendV2XMLRequest( myRequest.getRequestData() );

### Get the Betting Limits for My Game?

Betting limits are part of a Game Profile which is retrieved by the framework at startup. The exact content of each game profile is different depending on the requirements of the game but the values are always parsed and provided to you through the GameServices.profile object.

**Casino 4 (Slot Game):**

var gs:GameServices = new GameServices();  
trace (gs.profile.CreditValues); //Prints the array of available values.

**Casino 4 (Non-Slot Game):**

var gs:GameServices = new GameServices();

//Prints the value of “tableMin” profile attribute or throws an exception if value does not exist.  
trace (gs.profile.getParameter(“tableMin”));

**Casino 5 (Slot Game):**

var gs = new amaya.GameServices();  
trace (gs.profile.getCreditValues()); //Prints the array of available values.

**Casino 5 (Non-Slot Game):**

var gs = new amaya.GameServices();

//Prints the value of “tableMin” profile attribute or throws an exception if value does not exist.  
trace (gs.profile.getParameter(“tableMin”));

### Get the Currency Code in Use?

**Casino 4:**

var gs:GameServices = new GameServices();  
trace (gs.profile.CurrencyCode); //Prints the 3-char currency code. Ex: USD.

**Casino 5:**

var gs = new amaya.GameServices();

trace (gs.profile.getCurrencyCode()); //Prints the 3-char currency code. Ex: USD.

### Get the Language in Use?

**Casino 4:**

var gs:GameServices = new GameServices();  
trace (gs.session.Lang); //Prints the 2-char language code. Ex: en.

**Casino 5:**

var gs = new amaya.GameServices();

trace (gs.session.lang()); // Prints the 2-char language code. Ex: en.

### Format a Value for Currency Display?

**Casino 4:**

var gs = new amaya.GameServices();

myText.text = CurrencyFormat.format( 100.50, CurrencyFormatStyle.STANDARD, gs.profile.CurrencyCode );

**Casino 5:**

myText.text = amaya.sdk.currency.Currency.standardFormat(100.50);

### Use a Preloaded Asset in My Game?

**Casino 4:**

var gs = new amaya.GameServices();

myDisplayObjectContainer.addChild( gs.assets.getAsset(“path/to/asset.swf”) );

**Casino 5:**

var gs = new amaya.GameServices();

var myAsset = gs.assets.getAsset(“path/to/asset.png”);

// myAsset now contains a reference to an HTML IMG element.