Bridge : Data - Getting Started

QUICKSTART

1. The Bridge : Data expansion allows you to save anything that you can turn into a string to your WordPress database and fetch it back again. (Including binary data converted to base64, in case you were wondering)
2. Returned data can be extracted as:
   int, long, float, Rect, Color, Vector2, Vector3, bool and string
3. Name your variables anything you want when you save it then use the same value when you extract it. Job done!
4. When you save data to your server it is categorized for easier bulk retrieval.
   Name the category anything you want (even string.Empty) then use the same name when you retrieve it. Done!

Example: How to save info relating to your player and fetch it back again

Saving info to the database (many types are converted to string for you so just call "Set"):

```
void SaveMyData()
{
	var Data  = new CMLData();
	Data.Set("position", transform.position);
	Data.Set("rotation", transform.rotation);
	Data.Seti("health", 200);
	Data.Set("mana", "200");
	Data.Set("poisoned", true);
	WUData.UpdateCategory("player", Data);
}
```

Fetching your player's data from the database:

```
void Start() => WUData.FetchCategory("player", ParsePlayerInfo);
void ParsePlayerInfo(CML data)
{
	transform.position = data[1].Vector3("position");
	transform.rotation = data[1].Quaternion("rotation");
	var health = data[1].Int("health");
	var mana = data[1].Int("mana");
	var poisoned = data[1].Bool("poisoned");
}
```

That is all you need in order to get started. Below I explain a lot of stuff in great detail so feel free to read through it in order to get the most out of the kit but, ultimately, it boils down to doing exactly like the demo:

1. Decide which fields you want to save
2. Pass it to a CMLData object using the various Set() functions
3. Call UpdateCategory() and give your data a label to categorize it under
4. When you want it back call FetchCategory() and specify the category label
5. When the server's response is received, extract your variables using the names you gave them

That is what it boils down to. Easy as can be!

Every game is automatically cloud based

Any data you save online is saved to the user's account. When that user fetches his data back again it

doesn't matter what platform or device he is doing so from. This means a player can play on a TV or mobile phone then later continue playing the same game on a desktop or even a web browser at an internet café.

NOTE: The data is not updated in real time across devices so each active game needs to make a call to fetch the current values but the current and correct values will always be available on any platform/device the player is playing on

So, what can you do with this extension?

1. Save a player's data for the current game - High score, progress, etc.
2. Save a player's data to be shared across games - custom avatar settings, login tokens, etc.
3. Save game info shared by all players - Tweak difficulties/prices/etc and have the changes live immediately. No need for players to re-download the binary to receive your updates
4. Save images - Store/retrieve screenshots to a custom folder per user. Images are available for use on your website
5. Update other player's data - In authoritative games, prevent each user from saving their match results themselves