

Cocell Json Manual

Contents

- Introduction
- Installation
- Save/Load Basics
- Cloud Service
- Cocell Json C# Script
- Examples
- Youtube Video Links

Updated: 7/01/2019

Introduction

Welcome to Cocell Json!

Cocell Json(CJ) is a simple Visual Scripting solution to save and load game data.

Cocell Json C# Script (CJS) is a simple C# Script solution to save and load game data.

CJ is a Custom Unit which can only be used with Bolt 1.4 and .NET4x. Bolt 2 after release.

CJ & CJS creates a Node based Custom Json file when saved, that is editable.

Please Report all bugs and suggestions to cocell@yahoo.com

Installation

Let's get you started. (Bolt 1.4/.NET4x is Required for CJ)

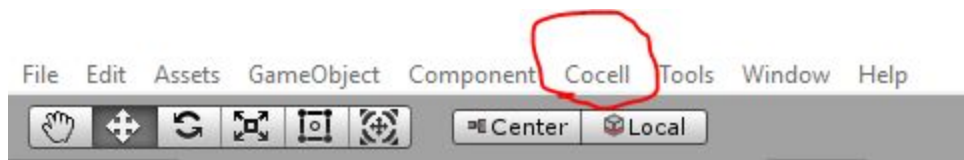
First, download and import Cocells Toolbox Package.

<http://www.cocellproductions.com/Unity/ToolboxDownload/CocellToolbox.zip>

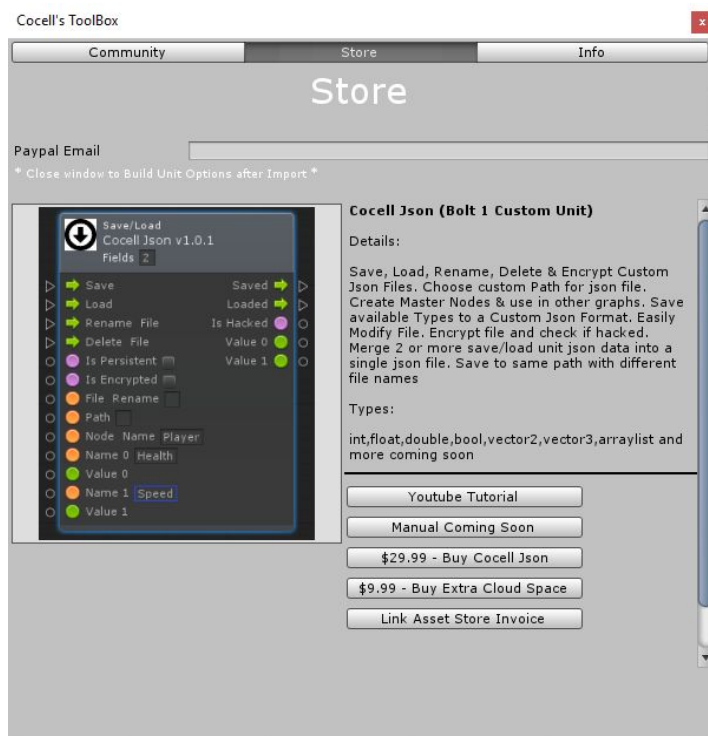
During Free Promotion, Go to link below. Email is Required to register for easy download & updates..<http://cocellproductions.com/Unity/DownloadPage.php>

Then, import it in your project.

After the import completes, open toolbox: Cocell > Cocell Toolbox....



Select Store Tab and then Buy Cocell Json.



After purchase, Enter Paypal/Email and then Import button will appear.



Click Import and files will download and import automatically to project.
Closing window will trigger Build Units Options after Import. You can manually Build Unit Options if needed. **Errors will populate in console which can be cleared. (CJS instructions below)**

You can now add Cocell Json Unit to graph.

Save/Load Basics

Save

Save Fields to Json File or Cloud.

Load

Loads saved Fields from Json File or Cloud.

Rename file (Cloud Disabled)

Renames Json File if using persistent path.



Delete file (Cloud Disabled)

Deletes Json File if using persistent path.

Is Persistent (Cloud Disabled)

PersistentDataPath will point to a public directory on the device.

Is Encrypted (Cloud Disabled)

Protects file from modifications.

Is Cloud

Save Fields to Cloud without any coding or backend work.

Dev ID (Cloud only)

ID that was created and emailed after purchase.

User ID (Cloud only)

ID you create for each player

File Rename (Cloud Disabled)

The File name changing to. Use with Rename File Flow.

Path (Cloud Disabled)

The Path to save/load Cocell Json File.

Node Name (Cloud Disabled)

Splits data into Nodes in File. Ex. <Level1><Data></Level1><Level2><Data></Level2>

Name #

Variable name to save/load

Value #

The Variable Value to save/load

Fields

The amount of Variable Fields Added to Unit

Saved (Output Flow)

Flows out when Save has completed

Loaded (Output Flow)

Flows out when Load has completed

Is Hacked (Cloud Disabled)

When “is Encrypted enabled”, using Loaded Flow, will output True or False.

Use this to check if file was hacked.

Value # (Output)

Loads field Variables from file.

If Node not found, will output false. (For other use)

Cloud Service

The Cloud service gives you a convenient way to save your players' game progression. Your game can retrieve the saved game data to allow returning players to continue a game at their last save point from any device.

The Cloud service makes it possible to synchronize a player's game data across multiple devices. For example, if you have a game that runs on Xbox One, you can use the Cloud service to allow a player to start a game on their Xbox One, and then continue playing on a tablet without losing any of their progress. This service can also be used to ensure that a player's game play continues from where it left off even if their device is lost, destroyed or traded in for a newer model.

The Cloud service Back-End(Server/Database) is configured where you don't need any coding or any configuration. It works right out of the box.

Supported Platforms:

Xbox One

PC

iOS

Android

How to use:


Setting up your project for Unity Services

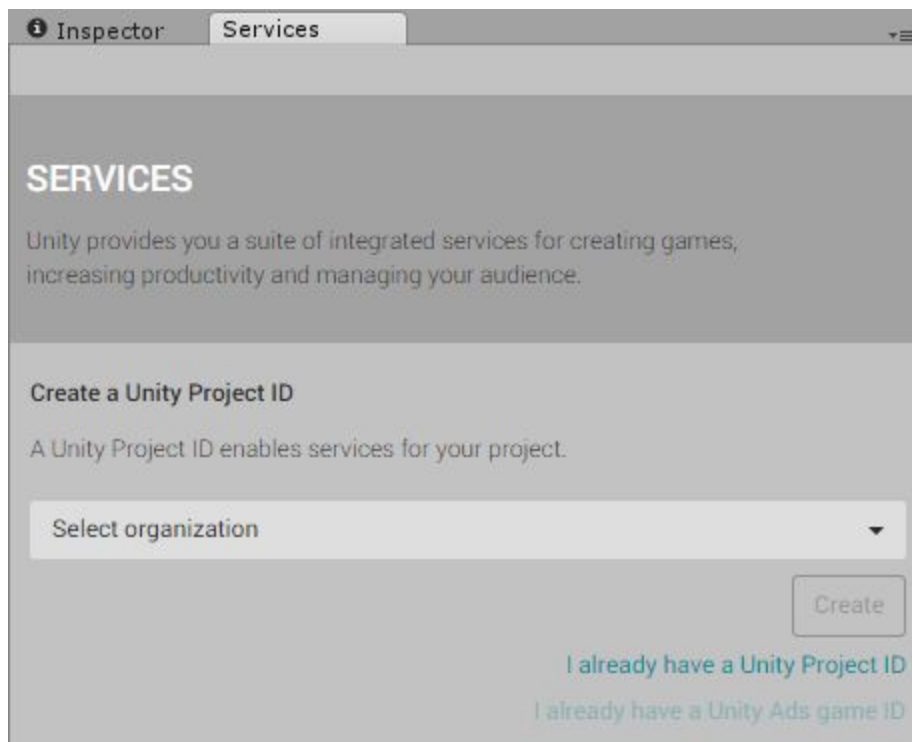
For Unity Full Instructions:

<https://docs.unity3d.com/Manual/SettingUpProjectServices.html>

1. You must first link your project to a **Unity Services Project ID**. A Unity Services Project ID is an online identifier which is used across all Unity Services. These can be created within the Services window itself, or online on the [Unity Services website](#). The simplest way is to use the Services window within Unity, as follows:

To open the Services Window, go to **Window > General > Services**

, or click the cloud button in the toolbar. . If you have not yet linked your project with a Services ID, the following appears:



This allows you to create a new Project ID or select an existing one.

To create a Project ID, you must specify an **Organization** and a **Project Name**.

You are now ready to Use the Cloud service.

2. **Enable "Is Cloud"** in Unit. This will **Disable** Rename File, Delete File, Is Persistent, Is Encrypted, File Rename, Path and Node Name. Creating a Unity Project ID is Required.

Dev ID is a unique ID for developers that is auto-generated when purchase is made. This is sent via Email. You need this ID for every project.

User ID is a unique ID for each player during Log-in/Registration.

Note: Unit will not work without both IDs.

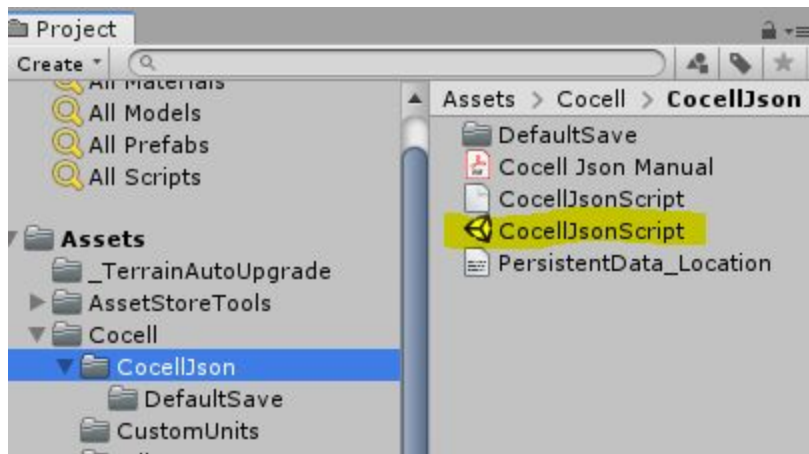


Cocell Json C# Script (Advanced Users)

Cocell Json C# Script gives you the ability to Save/Load game data from C# instead of using Bolt Visual Scripting.

Installation

After following Install instructions above, navigate to and import Assets/Cocell/CocellJson/CocellJsonScript.unitypackage. This will import a .dll file in folder.



How to use?

Create New C# Script, Add Namespace & Member Variables

```
using System.Collections;
using UnityEngine;
using CJ;

public class CocellJsonScriptTest : MonoBehaviour
{
    CocellJsonScript cjs = new CocellJsonScript(); // Create New Object
    private string devid = "XXXXXXXXXXXX"; // (Cloud Only) Devs Unique ID after Purchasing Cloud Space
    public string path; // Path for json file. "blank" Defaults to Assets/Cocell/CocellJson/Defaultsave/CocellJson.json
    public string userid = "12345"; // (Cloud Only) Users id after sign-up
    public object obj = null; // Object Variable that's being saved i.e., string,int,vector3
    public string nodeName = string.Empty; // Master Node for Variables, e.g., (Node){saved variables}/(Node)
    public string varName = string.Empty; // Variable name
    public bool isCloud = false; // If Cloud is enable
}
```


Add Methods

```
// Update is called once per frame
void Update()
{
    if (Input.GetKeyDown("s"))
    {
        //If cloud is active/inactive
        if (!isCloud)
        {
            //Save to local Drive
            cjs.SaveCocellJson(nodeName, varName, obj, path, this);
        } else {
            //Save to Cloud
            cjs.SaveCocellJsonCloud(varName, obj, devid, userid, this);
        }
    }

    if (Input.GetKeyDown("l"))
    {
        //If cloud is active/inactive
        if (!isCloud)
        {
            //Load from local Drive
            Debug.Log("Load from Local " + cjs.LoadCocellJson(nodeName, varName, path));
        }
        else
        {
            //Load from Cloud
            StartCoroutine(LoadFromCloud());
        }
    }
}
```

Load From Cloud

```
IEnumerator LoadFromCloud()
{
    object valReturn = null;

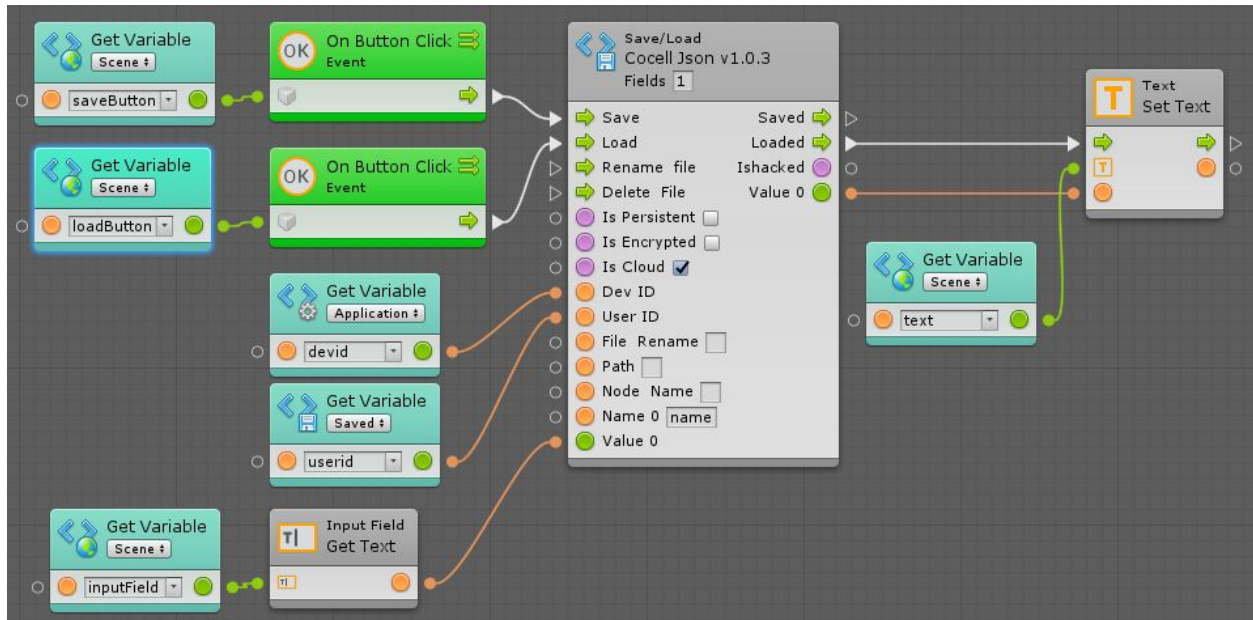
    yield return cjs.LoadCocellJsonCloud(varName, devid, userid, this, result => valReturn = result);

    Debug.Log("Load from Cloud: " + valReturn);
}

}
```

Bolt Examples

Cloud Example:



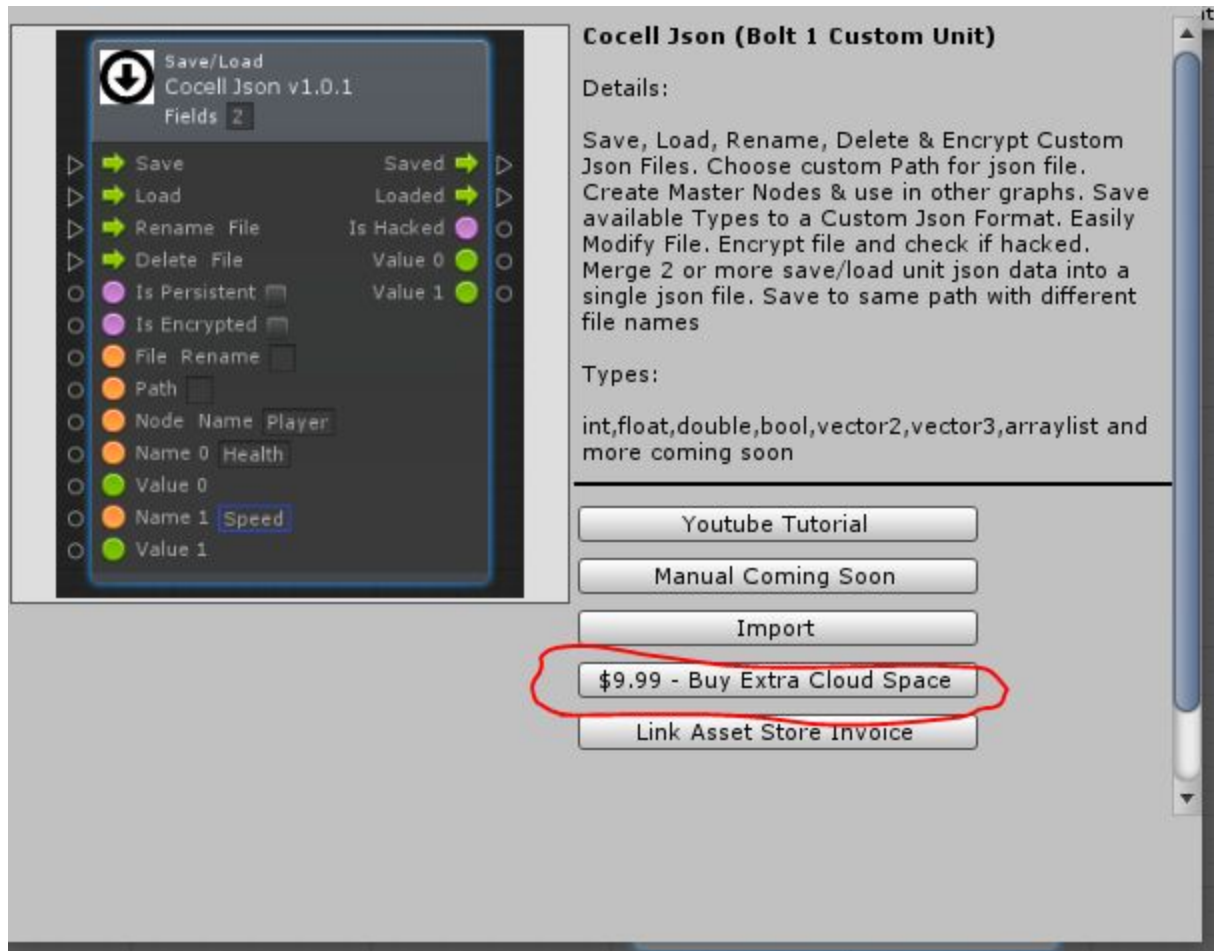
3. Add Extra Cloud Space

You have to purchase Cloud Space for each Project/Game. This will give you space for your specific game data. After purchase, your space is added and available.

How to add Extra Cloud Space

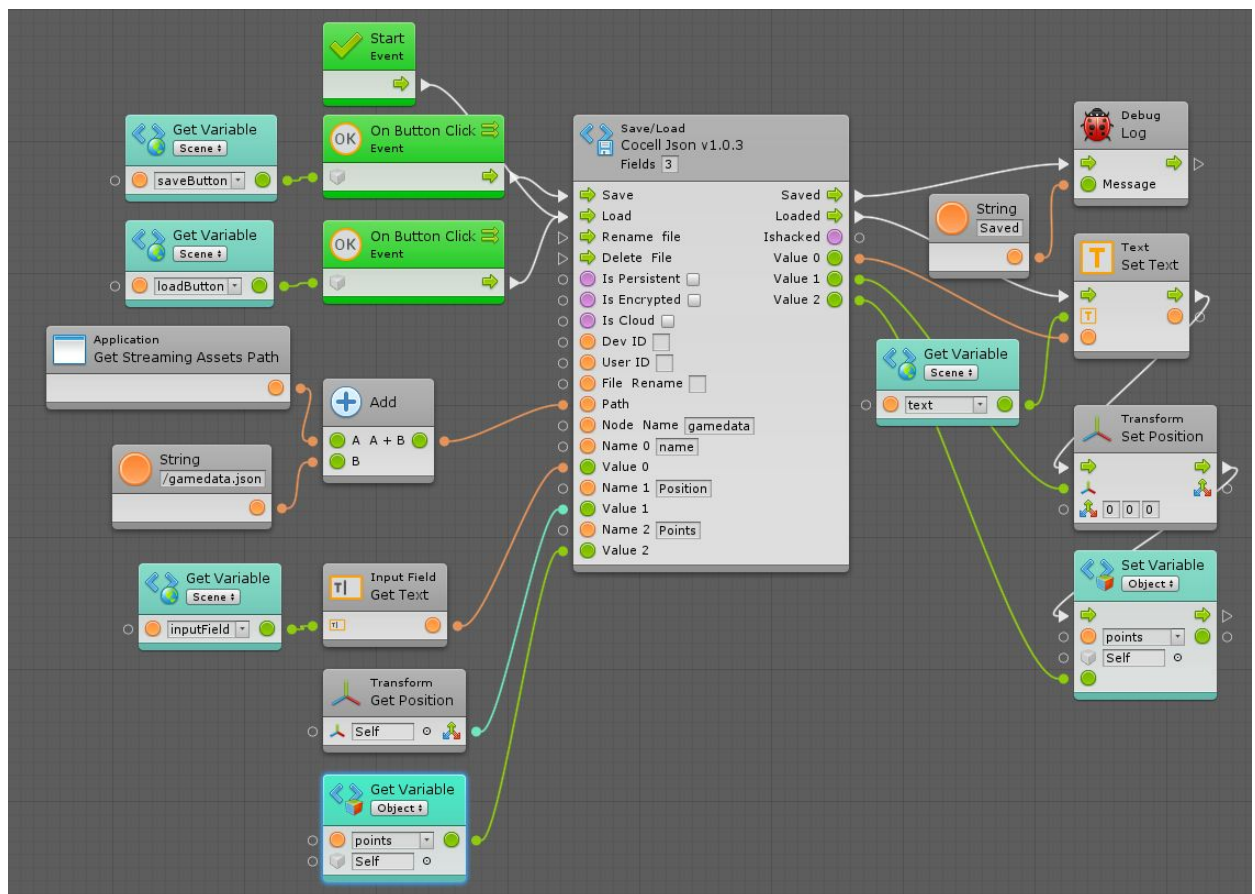
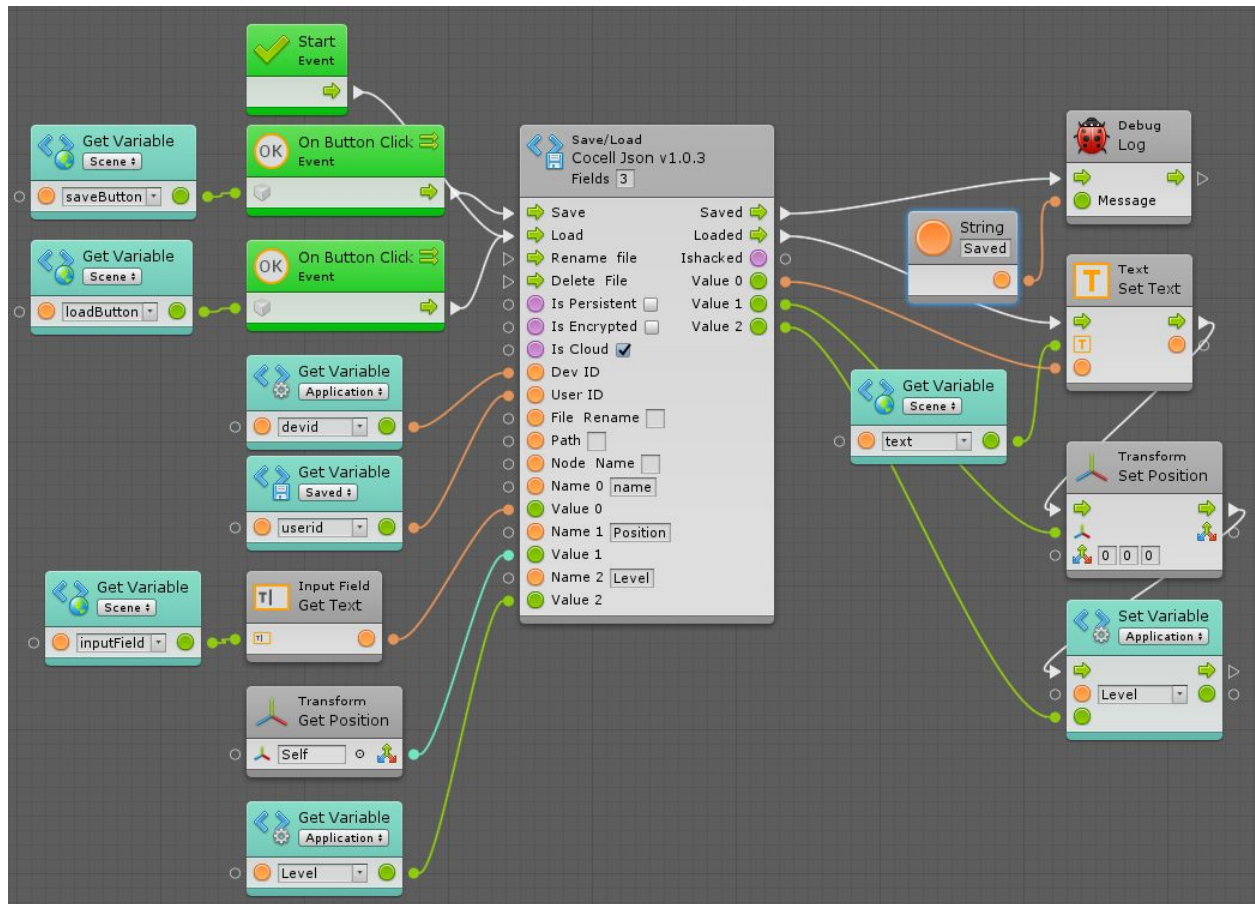
Open Toolbox: Cocell > Cocell Toolbox....

Select Store Tab: Store > Buy Extra Cloud Space Button...



Note: (Coming Soon) Link Asset Store Invoice

Save/Load Examples:



Youtube Video Links

CJ - Save and Load

<https://www.youtube.com/watch?v=T4zCjt3g8r4&t=1s>

CJ - How to Install

<https://www.youtube.com/watch?v=eZpZ0QO53Do>