Inspector Gadgets   Change Log

Inspector Gadgets 5.0

- You can now click on the Position/Rotation/Scale labels and press Ctrl+C/V to copy/paste their values.
- The Transform Inspector now allows you to drag the Position, Rotation, and Scale labels:
    - Dragging the Position moves the object along it's forward vector.
    - Dragging the Rotation rotates the object around the Z axis just like dragging the label of the Z field.
    - Dragging the Scale resizes the object on all axes at once just like in Uniform Scale mode, but it retains the object's aspect ratio instead of setting all axes to the same value.
    - You can hold Ctrl while dragging these labels to snap to the grid.
- Added Scripty Hunter - a system for tracking down and quickly fixing missing script references. It replaces the inspector for missing scripts to get started.
- Added "Find References" functions to the context menu for all Components. There are 3 variants: "In Hierarchy" searches children and parents, "In Scene" searches the whole scene, and "In Assets" searches all assets in the project.
- Added "Watch" function to the context menu for all fields. The function opens an editor window that displays a list of all fields you have used the function on so you can view (and edit) them all at the same time. The Lite version only allows you to watch up to 3 fields at a time.
- Added "Visualise" function to the context menu for float, Vector2, and Vector3 fields. The function opens an editor window that displays a gizmo in the scene to visualise and manipulate the chosen field. Vectors are visualised using a position handle with a line leading back to the origin while floats are visualised using the same wireframe sphere used to display the shape of a SphereCollider.
- [Pro-Only] Added [H], [S], and [A] buttons to object reference fields while they are null to easily find a reference of the correct type in the hierarchy, scene, and project assets respectively.
- Improved Object Reference Context Menus:
    - Renamed Component context menu function "Progressive Search" to "Find Component".
    - Improved the search method to find the component on an object with a name most similar to the property's display name.
    - When multiple objects are selected it will now find the appropriate target object on each selected object individually rather than assigning the same target to all of them.
    - Removed the other "Get Component" functions.
- Added some layout functions to the context menus for RectTransform properties (position and size, but not anchors or pivot).
    - "Set Width = Height" and "Set Height = Width" make the rect square.
    - "Snap to Siblings" has a varient for each direction which moves the rect in that direction so that its leading edge is aligned with the closest edge in any of its siblings (other objects with the same parent).

- [Pro-Only] Added SetHideFlags component which can be added to a scene object to modify its hide flags in the inspector. Adding the component automatically sets the DontSaveInBuild flag.
- [Pro-Only] Renamed [RequireAssignment] to [Required].
- [Pro-Only] Object reference fields will now be drawn in italics when they reference another component on the same GameObject (can be disabled in the Preferences menu).
- [Pro-Only] Added [Inspectable] attribute to draw a field or property that wouldn't normally get drawn in the inspector. Works for static or instance members of any common type (bool, int, float, vectors, Unity Objects, etc).
- [Pro-Only] Italicised the labels for [Inspectable] and [Label] to distinguish them from regular serialized fields.
- [Pro-Only] Added DisplayIndex property to inspectable attributes (Button, Label, Inspectable) to control their positioning amongst the regular serialized fields.
- [Pro-Only] Improved support for [Label] values that take up multiple lines and renamed the Multiline property to LargeMode.
- [Pro-Only] Removed the Log button from [Label] displays. You can now log the value via the context menu.
- [Pro-Only] Added [Scene] attribute for int and string fields to display a dropdown list that allows you to pick a scene from the build settings to assign it's index or name to the field.
- [Pro-Only] Added [ShowPreview] attribute for object reference fields to display a preview of the target object.
- [Pro-Only] Added various validator attributes: [MinValue], [MaxValue], [ClampValue], [HasComponent].
- [Pro-Only] Optimised the gathering of InspectorGadgets.Editor<T>.Target and Targets. They are now instance properties with CurrentTarget and CurrentTargets allowing static access.
- [Pro-Only] The [Toolbar] attribute can now be used on bool fields to specify labels such as yes/no or on/off.
- [Pro-Only] The [Toolbar] attribute now supports tooltips on individual members with the [Tooltip] attribute.
- [Pro-Only] Added [TexDefine(KEYWORD)] which can be used on texture properties in shaders to enable the specified keyword whenever a texture is assigned.
- Added "UI Auto Focus Mode" to the Preferences menu so you can choose if you want the Auto Hide UI system to focus on the selected object or its root RectTransform when you first select a UI object. Focussing on the root also frames it better than the default focus mode to make better use of the available screen space.
- Added preferences field to select which layers will activate the UI Auto Hide feature.
- UI Auto Hide will now also activate when an object with a UIBehaviour (such as an Image or Button component) in its parents or children is selected.
- The Auto Hide UI system will no longer focus the camera on the last stored position when you open a project and is now more reliable in general.
- The "AddComponent"/"Create new Instance" function for Component/ScriptableObject fields will now show a sub menu for all inheriting types so you can make an instance of any child type rather than only the base type.
- Added [R] buttons to the right of all preferences when they aren't at their default value to reset them.
- Standardized all AutoPrefs to have a common set of GUI drawing methods.

- You can now add a "private const bool NestedObjectDrawers = false;" field to a class to disable nested object drawers for it in case they don't fit within the layout of your custom editor.
- Moved InspectorGadgetsUtils.GetValue and related functions into SerializedPropertyAccessor and optimised them to be much faster and create much less garbage.
- Fixed a GUI layout error when middle clicking a component to open its script for the first time.
- [Pro-Only] Fixed some layout errors when drawing the [Label] attribute.

Inspector Gadgets 4.4

- [Pro-Only] Added Nested Object Inspector system which draws a foldout for each Object reference field to draw the target object's inspector nested inside the current one. Only applies to fields in custom scripts but not inbuilt components and doesn't support GameObject fields.
- Fixed the Transform position gizmo to properly support vertex snapping.
- Fixed the Transform position gizmo to not move child objects twice as far when both the parent and child are selected.
- Added "Override Transform Gizmos" preference toggle to disable the custom implementation of the scene gizmos that allows features like "Freeze child transforms" and "Draw gizmos for all selected objects" in case there are inconsistencies between the custom and default implementations that cause problems for the user (such as the two bugs this one).
- Added "Remove Component" button to the inspector for missing scripts.
- Added "Open Inspector" function to the context menu for object references so you can easily view the referenced object in a new inspector window without changing your selection.
- Fixed scene gizmos to match Unity's default functionality for static objects.

Inspector Gadgets 4.3

- Added [UniqueCollection] attribute which highlights any fields in a collection that have the same value.
- Added [RequireAssignment] attribute which highlights a field in red when it has its default value so you know it needs to be assigned.
- Added specific context menu functions to RectTransform inspector properties (to replace their generic float/Vector2 menus).
- Transform and RectTransform context menus now display the name of the property as you would access it in code.
- Added Progressive Search context menu item to component fields which looks for a component using GetComponent, and if that fails it uses GetComponentInParent, then GetComponentInChildren, then FindObjectOfType.
- Added separators to the serialized property context menus to improve readability.
- Added "Look At Next Selected Object" function to the Transform Rotation context menu.
- Added Randomize Hue command to Color property context menu.
- Added Multiline property to the [Label] attribute for displaying long values.
- Fixed Missing Script inspector to work like Unity's default.

- Fixed "Find Objects of Type (including Assets)" context menu item for Unity Object arrays.
- Renamed Utils to InspectorGadgetsUtils to avoid name collisions.
- Removed calls to UnityEditor.LocalizationDatabase since 2017.3 made it internal.

Inspector Gadgets 4.2

- Improved integration of the public and private clipboards:
  - Position/rotation/scale each have their own private clipboard so that you can copy a position and then a rotation at the same time without overwriting the previous value.
  - Vector3s also have a public clipboard used by the context menu functions on all Vector3 fields. The public clipboard integrates with the system copy buffer so you can copy values to and from other programs as well.
  - When you copy position/rotation/scale using the [C] button or the context menu function, the value is copied to both the private and public clipboards.
  - You can left/right click the [P] button to paste from the public/private clipboard respectively.
  - The context menu Paste function has also been split into separate functions for public/private.
  - Clipboards now handle multi-object selection with different values better. Previously it copied the value from each object into an array and pasted them to multiple objects by iterating through that array. This led to unpredictable results since the order of selected objects is inconsistent (not based on the order you select them, nor the order they appear in the hierarchy). Now the clipboard holds a single vector with partial values, so if you copy from two objects with the same X and Y but different Z values, the X and Y would be copied and the Z value would be ignored when pasting.
  - Copy and paste for Transform fields now properly respect the Local/World mode and Freeze Child Transforms mode.
- You can now right click the [C] button to log the currently displayed value.
- You can now right click the [S] button to open the Snap Settings window.
- The Transform Inspector will now emphasize fields which aren't at their default value.
- Rect, Bounds, and Color context menu paste functions now parse the system copy buffer to read their string format.
- The Position Snap to Ground context menu function now works with both 2D and 3D physics.
- Improved most tooltips.
- Improved preferences window layout.

Inspector Gadgets 4.1

- [Pro-Only] Improved the scene view movement tool:
  - It now shows some extra lines while you are moving an object to indicate where you are moving it from along with a label specifying the distance moved.
  - Labels also show the current position values on each axis.
- Fixed some inconsistencies in the way scene gizmos behave when using Freeze Child Transforms or Draw All Gizmos.

- Added support for the Center pivot mode (enabled via the Pivot/Center toggle in the top left of the Editor) for Freeze Child Transforms and Draw All Gizmos.
- Context menu functions now use the invariant culture for copying float values to the clipboard to avoid problems with cultures that use commas instead of periods.

Inspector Gadgets 4.0

- Middle Click to reset Transform fields is now available in the Lite version and the Reset buttons are now hidden by default.
- Added lots of useful functions to the context menus of properties shown in the inspector.
  - Note that these functions all support multi-object selection, but the Lite version disables them while multiple objects are selected.
  - Copy and Paste most property types using an internal clipboard for each individual type. Most of these types also copy and paste using the system clipboard at the same time so you can copy values to and from other programs.
  - Snap objects to the grid, or raycast down to snap them to the ground.
  - Unity Object acquisition: FindObjectOfType, GetComponent, GetComponentInChildren, GetComponentInParent, AddComponent.
  - Unity Object array acquisition: FindObjectsOfType, GetComponents, etc.
  - Create new instance of any ScriptableObject type.
  - Save any Unity Object as an asset (such as a procedurally generated mesh).
  - Randomize within common ranges: 0-1, 0-100, 0-360 for floats, random Vector2 in a unit circle, random Vector3 on or in a unit sphere, random quaternion, random euler angles for a Vector3.
  - Common vectors: zero, right, up, forward, one.
  - Normalize vector.
  - String to lower or upper case.
  - Log current value.
- Added a system which automatically hides the UI layer in the scene view. When an object on the UI layer is selected in the hierarchy, the layer is automatically shown and the scene camera focuses on that object. Credit to Astral Byte Ltd for the original idea. It will ask if you want to enable this feature the first time you select a UI object with Inspector Gadgets installed.
- The settings window is now accessible as a tab in the Edit/Preferences window.
- Added [OnEditorQuit] attribute which registers a static parameterless method to be called when the Unity Editor is closed.
- Shift + Middle Click in a component inspector will now ping the script asset.
- Added an option in the Settings Window to show the "Script" property at the top of the MonoBehaviour inspector like normal (instead of hiding it to save space).
- Added AutoPrefs.Vector4 and Quaternion, as well as EditorPrefs versions of all of them.
  - Added HasKey method to all AutoPref types.
  - Added implicit cast from string to all AutoPref types.
- Added ConstantlyRepaint property to the [Label] attribute so you can ensure that the displayed value is always up to date rather than relying on the inspector to update naturally.
- Fixed a bug in the Transform Inspector when editing rotations with multiple objects selected.
- Fixed a minor bug that would prevent the world scale warning from showing properly on arbitrarily rotated objects.

Inspector Gadgets 3.0

- Shrunk the Local/World button into a tiny toggle (L/W) and added two more feature toggles in the free space:
    - [F] = Freeze child transforms (so that moving a parent object won't also move its children).
    - [G] = [Pro-Only] Draw gizmos for all selected objects (instead of only drawing a gizmo for the first object selected).
- Added [LabelledCollection] which provides labels for the elements of a collection field to use instead of just calling them Element X.
- Added AutoPrefs for more easily managing PlayerPrefs and EditorPrefs. Used for various settings.
- Added Comment component for leaving comments attached to scene objects which are automatically removed from builds.
- Removed MonoBehaviour Constructors (ConstructableBehaviour, [RequireInitialisation], etc) as they were not really related to the inspector. Source code available on request.
- Fixed TransformInspector to not cause the inspector window to constantly repaint.
- Fixed GUI layout error in World space mode with arbitrarily rotated objects.
- [Pro-Only] Fixed [Euler] and [Toolbar] to work properly when used on collection fields (such as an array or list of quaternions or enums respectively).

Inspector Gadgets 2.2

- Added experimental ConstructableBehaviour class as another alternative to MonoBehaviour constructors (separate from [RequireInitialisation]).
- The Copy, Paste, Snap, and Reset buttons will now be greyed out when they would do nothing. I.E. Copy and Paste are greyed out if the clipboard contains the same values as the selected object, Snap is disabled if the object is already snapped, and Reset is disabled if the object is already at the default values.
- [Pro-Only] ScriptableObject and StateMachineBehaviour now benefit from the same features as MonoBehaviour.
- [Pro-Only] Added Edit/Selection/New Locked Inspector (Ctrl + Alt + S) which opens an inspector window locked to the current selection so you can easily compare and copy values between different selection sets.
- [Pro-Only] The EditorState for inspectables ([Button] and [Label]) now defaults to Always instead of Playing.
- [Pro-Only] Added OnInspectorGUI event. Declaring a method with that name in a MonoBehaviour script allows Inspector Gadgets to call it instead of drawing the GUI normally. AfterInspectorGUI works similarly, but is called after the regular GUI is drawn.
- [Pro-Only] Added [Euler] attribute for drawing Quaternions as euler angles.
- [Pro-Only] When the [Button] attribute logs the returned value, it now gives it the proper context so that clicking the log message pings the selected object in the hierarchy.
- [Pro-Only] The [Button] attribute will now only call a static method once instead of once per selected object.
- [Pro-Only] Added tooltips to the [Button] attribute.
- [Pro-Only] Fixed [Button] attribute so it doesn't alter the layout of other properties when present (previously it shifted other properties by a few pixels).

- [Pro-Only] The [Label] attribute now includes the member name in the message when you click the Log button.
- Added Utils.FindComponent and TryFindComponent for doing lazy proximity based searches until a component is found. GetComponent -> GetComponentInParent -> GetComponentInChildren -> FindObjectOfType -> AddComponent.
- Fixed the [RequireAssignment] drawer to work properly for non-public fields and types derived from UnityEngine.Object.
- Fixed an issue where the scale GUI would always be enabled, regardless of the overall GUI state.

Inspector Gadgets 2.1

- Added "Inspector Gadgets Manual" command to the Transform context menu.
- [Pro-Only] Fixed [Color] and [Readonly] attributes to allow nested fields to be expanded properly.

Inspector Gadgets 2.0

- Added experimental [RequireInitialisation] attribute as an alternative to MonoBehaviour constructors (which aren't supported).
- [Pro-Only] Added [RequireAssignment] attribute to ensure that a field is assigned a value (as part of RequireInitialisation).
- [Pro-Only] You can now middle click on a transform field to reset it to its default value.
- Transform Inspector now properly shows bold labels when values are modified from a prefab and you can right click to revert them.
- [Pro-Only] Fields are now drawn in italics while their value that isn't a multiple of the snap threshold.
- Changing an object's rotation on one axis no longer allows floating point imprecision in the euler angle calculation to cause changes to the other axes.
- [Pro-Only] Added an Inspector Gadgets Settings window to the Transform context menu to toggle various options: show copy/paste/snap/reset buttons, use uniform scale, use field colors, italicise un-snapped fields, forget world space mode.
  - Disabling "Forget World Space" in settings will allow the inspector to stay in world space mode when you select a different object (otherwise it reverts to local space automatically).
- [Pro-Only] The [Toolbar] attribute can now be used on string fields. You can specify the available options in the constructor and it displays them as a toolbar.
- Fixed Transform Inspector layout in the Avatar Configuration scene.
- Various performance optimisations and major code structure improvements.

Inspector Gadgets 1.0

- Initial release.