Auto Values

Introduction

Auto values are values which are internally computed by SceneComposer or Candera or inherited from other objects in certain situations. In these situations, it is useful to show the computed value and an information that the value is computed.

Example: TextNode2D.LineHeight property can have a computed value. Setting it to any other value than 0, will make the value fixed and effectively used by Candera to separate the lines of text. But when setting it to 0, internally, Candera will use instead the line height value (let's say 42) specified by the associated text style. In this case, the value 0 is a trigger for computing a different value internally. And we call the computed 42 value an auto value.

Triggers of Auto-Value Properties

There are several triggers for a property to have an auto value:

1. **Value**. This is the most common one - a certain value of the property makes the property auto (usually the default value is the trigger value – in examples above, for Line Height, 0 is also the default value. In this case, having the default value triggers becoming auto).

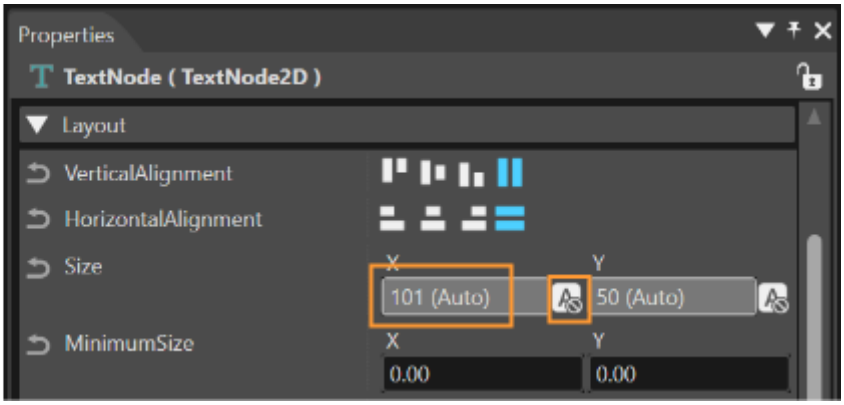
2. **Other configuration** – like properties of parent, parent type, and so on.. (example: when a node is a child of a layout group, the node's position becomes auto because it's computed by the layout group).

3. **Always Auto** – they are values computed by Candera for properties which are not configurable (for example, PreferredSize is a value computed during measuring elements in a Layout process. This is not configurable, but is a value computed based on other values. It may be meaningful for users when understanding of layout is needed).

Functionality of Auto-Values

The text "(Auto)" inside of a read-only properties field indicates that this property has an Auto-Value. The user can interact with auto-values through the switch button which is available in the Properties panel on the right side of the Auto-Value field. In the image below, you can see that the

vertical and horizontal size value is an Auto-Value, each having a corresponding switch button on their right.



As can be seen in the image above, a suffix indicates that the value is "auto". As expected, if the button is used the value will become editable and the user will be allowed to modify it as desired.

Auto-Value Properties

- TextNode2D.LineHeight
  - Type: integer
  - Trigger: having default value (0)
  - Is metainfo property: No
  - Retrieve auto value: retrieve the line height metric from the metrics of the associated TextStyle

- CanvasText.LineHeight
  - Same as TextNode2D.LineHeight

- Node2D.Position
  - Type: Vector2
  - Trigger: parent is layouter
  - Is metainfo property: No
  - Retrieve auto value: query SCHost, requires scene being loaded

- Node2D.Scale
  - Type: Vector2
  - Trigger: parent is layouter and stretch is not None except for when the node has a 9-patch image
  - Is metainfo property: No
  - Retrieve auto value: query SCHost, requires scene being loaded

- CanvasNode.Position
  - Type: Vector3
  - Trigger: parent is layouter. Only X and Y are auto
  - Is metainfo property: No
  - Retrieve auto value: query SCHost, requires scene being loaded

- CanvasNode.Scale
  - Type: Vector2
  - Trigger: parent is layouter and stretch is not None except for when the node has a 9-patch image
  - Is metainfo property: No
  - Retrieve auto value: query SCHost, requires scene being loaded

- Node2D.BoundingRectangle
  - Type: Rectangle
  - Trigger: Width and Height having default value (-1)
  - Is metainfo property: No
  - Retrieve auto value: query SCHost, add method to retrieve the computed bounding box, requires scene being loaded

- Camera2D.Viewport
  - Type: Rectangle
  - Trigger: Width and Height having default value (-1)
  - Is metainfo property: No
  - Retrieve auto value: compute the value based on RenderTarget dimensions and Candera rules

- Camera2D.ScissorRectangle
  - Type: Rectangle (only Width and Height can be auto)
  - Trigger: Width and Height having default value (-1)
  - Is metainfo property: No
  - Retrieve auto value: compute the value based on RenderTarget dimensions and Candera rules

- RenderTarget.Width
  - Type: integer
  - Trigger: having default value (-1)
  - Is metainfo property: Yes
  - Retrieve auto value: compute the value based on display dimensions (-1 means display width)

- RenderTarget.Height
  - Type: integer
  - Trigger: having default value (-1)
  - Is metainfo property: Yes

    - Retrieve auto value: compute the value based on display dimensions (-1 means display height)

- Node.Size (Layout)
    - Type: Vector2
    - Trigger: having default value (-1; -1)
    - Is metainfo property: Yes
    - Retrieve auto value: retrieve the ActualSize value from SCHost, which is computed by the Layouter. SCHost already acquires that value to use it to show the adorner for size (using a layout monitor which is notified about layout processing events and values).

- CanvasNode.Size (Layout)
    - Type: Vector3 (only X and Y can be auto)
    - Trigger: having default values (-1; -1) for X and Y (only X and Y can be auto)
    - Is metainfo property: Yes
    - Retrieve auto value: retrieve the ActualSize value from SCHost, which is computed by the Layouter. SCHost already acquires that value to use it to show the adorner for size (using a layout monitor which is notified about layout processing events and values).

- CanvasText.Size
    - This property is different from the layout size, but has the same name.
    - Type: Vector2
    - Trigger: having default values (-1; -1) for X and Y
    - Is metainfo property: Yes
    - Retrieve auto value: retrieve value computed by the CanvasNode through SCHost.

Improving Other UI Areas

The introduction of auto-value concept has an important consequence: user no longer sees the values if the values are the default ones (like 0 for LineHeight or -1 for Viewport.X). However, there are still cases for which, the default value is shown but auto-value cannot be implemented because doesn’t make sense. For these situations, additional improvements can be made.

- **MaxSize**: This Layout property has a default value of -1 which represents that value is not specified and should be considered infinite. This property can be turned into an optional property, so that by default it shows an unchecked checkbox indicating that MaxSize is not set.
- **MinSize**: Same as for MaxSize, can be turned into optional.

There are also properties which are not configurable in SceneComposer, nor they are visible in the property panel. They could be displayed in the property panel, but stay read-only, only for the purpose of providing information about internals.

- **PreferredSize**: Read-only property, can be added to Layout category to display the corresponding value computed by Candera. Value to be retrieved from SCHost.
- **ArrangeSize**: Read-only property, can be added to Layout category to display the corresponding value computed by Candera. Value to be retrieved from SCHost.
- **ArrangePosition**: Read-only property, can be added to Layout category to display the corresponding value computed by Candera. Value to be retrieved from SCHost.

Revision #7
Created 2023-02-13 13:01:14 UTC by Tsuyoshi.Kato
Updated 2024-05-31 12:00:28 UTC by Elisabeth Zauner