UGUI 开发使用技巧

技术开放日 广州站

侑虎科技（上海）有限公司

2016.7.10

Overview

-UI开销分析
-界面切换
-背包系统

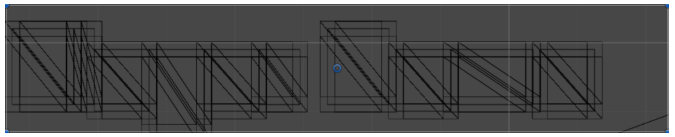
开销分析

- [UI元素更新]
- [网格合并]
- 渲染
- [事件检测]

开销分析

- [UI元素更新]
  - Instantiate/GameObejct.Activate
  - RectTransform 的数值修改
  - Image/Text 的属性修改
  - Layout 属性修改

```
[SerializeField] private Color m_Color = Color.white;
public Color color { get { return m_Color; } set { if (SetPropertyUtility.SetColor(ref m_Color, value)) SetVerticesDirty(); } }
```



开销分析

- [UI元素更新]
  - Canvas.SendWillRenderCanvases()
    - (ScrollRect.LateUpdate())

Canvas.SendWillRenderCanvases()	56.1%	0.0%	1	56.4 KB	73.97	0.00
CanvasUpdateRegistry.PerformUpdate()	56.1%	0.8%	1	56.4 KB	73.97	1.10
Graphic.Rebuild()	43.9%	0.1%	414	5.8 KB	57.81	0.21
Text.UpdateGeometry()	20.5%	0.0%	84	80 B	27.09	0.10
Graphic.UpdateGeometry()	12.6%	0.7%	123	120 B	16.65	1.03
Graphic.UpdateMaterial()	10.5%	0.5%	207	5.6 KB	13.84	0.71
LayoutRebuilder.UnityEngine.UI.ICanvasElement.Rebuild()	9.1%	0.0%	6	50.2 KB	12.02	0.00
LayoutRebuilder.PerformLayoutCalculation()	5.0%	0.1%	4	47.8 KB	6.63	0.23
LayoutRebuilder.PerformLayoutControl()	4.0%	0.1%	4	2.3 KB	5.38	0.24

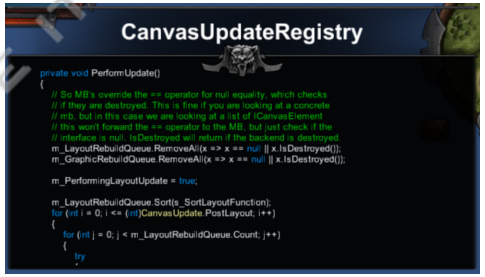
Text.UpdateGeometry()	20.5%	0.0%	84	80 B	27.09	0.10
Graphic.UpdateGeometry()	20.5%	0.6%	84	80 B	26.99	0.85
Outline.ModifyVertices()	13.7%	0.6%	81	0 B	18.07	0.80
Text.OnFillVBO()	5.2%	2.8%	84	0 B	6.91	3.80
Graphic.get_canvasRenderer()	0.2%	0.0%	84	80 B	0.34	0.11
CanvasRenderer.SetVertices()	0.1%	0.0%	84	0 B	0.21	0.03

Graphic.UpdateGeometry()	12.6%	0.7%	123	120 B	16.65	1.03
Image.OnFillVBO()	10.9%	0.8%	123	0 B	14.36	1.11
Image.GenerateSimpleSprite()	5.5%	0.7%	86	0 B	7.32	0.98
Image.GenerateSlicedSprite()	4.4%	1.5%	37	0 B	5.90	2.07
Image.get_overrideSprite()	0.0%	0.0%	123	0 B	0.02	0.02
Image.get_type()	0.0%	0.0%	123	0 B	0.00	0.00
Object.op_Equality()	0.0%	0.0%	123	0 B	0.00	0.00
Component.GetComponents()	0.2%	0.0%	123	0 B	0.26	0.00
ComponentListPool.Release()	0.1%	0.0%	123	0 B	0.26	0.00

开销分析

- [UI元素更新]
  - Canvas.SendWillRenderCanvases()
    - CPU
    - 堆内存（主要在实例化是出现）
      - UpdateGeometry->OnFillVBO

```
var vbo = s_VboPool.Get();

if (rectTransform != null && rectTransform.rect.width >= 0 && rectTransform.rect.height >= 0)
    OnFillVBO(vbo);
```

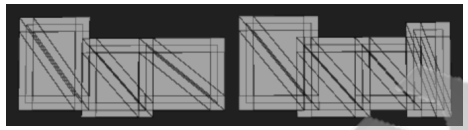



```
private void PerformUpdate()
{
    // So MB's override the == operator for null equality, which checks
    // if they are destroyed. This is fine if you are looking at a concrete
    // mb, but in this case we are looking at a list of ICanvasElement
    // this won't forward the == operator to the MB, but just check if the
    // interface is null. IsDestroyed will return if the backend is destroyed.
    m_LayoutRebuildQueue.RemoveAll(x => x == null || x.IsDestroyed());
    m_GraphicRebuildQueue.RemoveAll(x => x == null || x.IsDestroyed());

    m_PerformingLayoutUpdate = true;

    m_LayoutRebuildQueue.Sort(s_SortLayoutFunction);
    for (int i = 0; i <= (int)CanvasUpdate.PostLayout; i++)
    {
        for (int j = 0; j < m_LayoutRebuildQueue.Count; j++)
        {
            try
            {
                if (ObjectValidForUpdate(instance.m_LayoutRebuildQueue[j]))
                    instance.m_LayoutRebuildQueue[j].Rebuild((CanvasUpdate)i);
            }
            catch (Exception e)
            {
                Debug.LogException(e, instance.m_LayoutRebuildQueue[j].transform);
            }
        }
    }
    instance.m_LayoutRebuildQueue.Clear();
    m_PerformingLayoutUpdate = false;

    m_PerformingGraphicUpdate = true;
    for (var i = (int)CanvasUpdate.PreRender; i < (int)CanvasUpdate.MaxUpdateValue; i++)
    {
        for (var k = 0; k < instance.m_GraphicRebuildQueue.Count; k++)
        {
            try
            {
                var element = instance.m_GraphicRebuildQueue[k];
                if (ObjectValidForUpdate(element))
                    element.Rebuild((CanvasUpdate)i);
            }
            catch (Exception e)
            {
                Debug.LogException(e, instance.m_GraphicRebuildQueue[k].transform);
            }
        }
    }
    instance.m_GraphicRebuildQueue.Clear();
    m_PerformingGraphicUpdate = false;
}
```

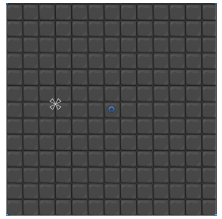
ScrollRect.LateUpdate()	73.8%	0.9%	1	1.6 MB	52.44	0.66	N/A	7.2%	0.0%	1	210.0 KB	5.14	0.00
ScrollRect.EnsureLayoutHasRebuilt()	72.5%	0.0%	1	1.6 MB	51.57	0.00	N/A	0.3%	0.0%	1	0.8 MB	0.28	0.00
Canvas.ForceUpdateCanvases()	72.5%	0.0%	1	1.6 MB	51.56	0.00	N/A	0.2%	0.2%	1	52.5 KB	0.17	0.15
Canvas.SendWillRenderCanvases()	72.5%	0.0%	1	1.6 MB	51.56	0.00	N/A	0.2%	0.0%	1	420.0 KB	0.16	0.00
CanvasUpdateRegistry.PerformUpdate()	72.5%	0.1%	1	1.6 MB	51.56	0.13	N/A	0.0%	0.0%	1	105.0 KB	0.03	0.00
Graphic.Rebuild()	69.1%	0.0%	12	1.6 MB	49.09	0.00	N/A	0.0%	0.0%	1	0 B	0.01	0.00
Text.UpdateGeometry()	61.6%	0.0%	2	1.6 MB	43.80	0.00	N/A	0.0%	0.0%	1	0 B	0.01	0.00
Graphic.UpdateGeometry()	61.6%	0.0%	2	1.6 MB	43.79	0.02	N/A	0.0%	0.0%	1	0 B	0.01	0.00
Text.OnFillVBO()	60.7%	46.7%	2	1.6 MB	43.14	33.22	N/A	0.0%	0.0%	1	0 B	0.01	0.01
List`1.Add()	9.0%	0.6%	12396	1.6 MB	6.43	0.42	N/A	0.0%	0.0%	1	0 B	0.01	0.00
TextGenerator.Populate()	2.8%	0.0%	2	0 B	1.99	0.00	N/A	0.0%	0.0%	1	0 B	0.01	0.00
List`1.get_Item()	1.0%	0.3%	12396	0 B	0.74	0.27	N/A	0.0%	0.0%	1	0 B	0.01	0.00

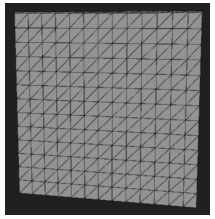
开销分析

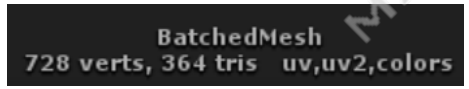
- [UI元素更新]
  - 减少长文本 Text 的变动，慎用 UI/Effect。



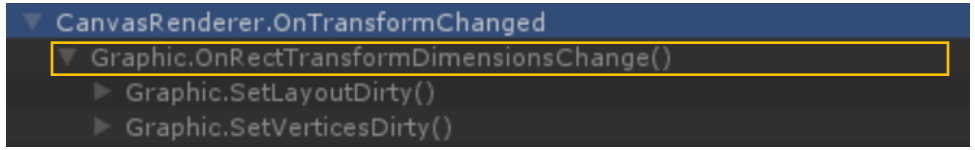
  - 慎用 Tiled 类型的 Image。





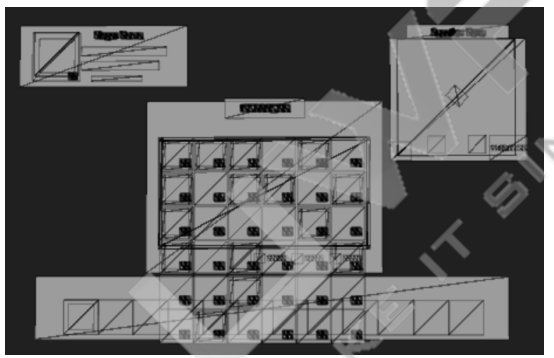


  - 慎用 Pixel Perfect



开销分析

- [网格合并]
  - Canvas 为单位



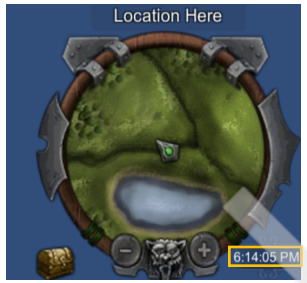
  - Canvas.BuildBatch

Canvas.BuildBatch	21.3%	0.0%	1	0 B	4.22	0.00
Canvas.GenerateBatchedMesh	21.3%	6.6%	1	0 B	4.22	1.32
Canvas.FillBatchMesh	13.6%	1.7%	1	0 B	2.69	0.34
Canvas.FillVerts	10.8%	10.8%	1	0 B	2.14	2.14
Canvas.DoIndicies	0.9%	0.9%	1	0 B	0.19	0.19
Canvas.FillQuads	0.0%	0.0%	1	0 B	0.00	0.00
Canvas.BatchSort	0.9%	0.9%	1	0 B	0.19	0.19
Mesh.AwakeFromLoad	0.0%	0.0%	1	0 B	0.00	0.00

Not Saved (154)	0.7 MB	
Mesh (16)	0.5 MB	
BatchedMesh	489.0 KB	1

开销分析

- [网格合并]
  - 排除不需要的 Canvas.SendWillRenderCanvases()
  - 动静分离，动态元素可按更新频率细分。
    - 坐标、时间
    - HUD 战斗提示
    - 角色头顶信息
    - ……

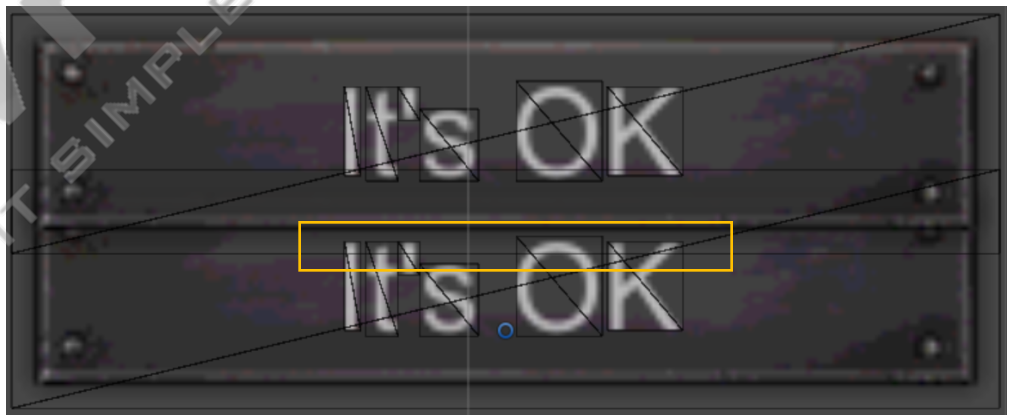


  - 控制“动态 Canvas”中的元素数量（网格大小）。
  - 慎用 CanvasGroup 的 Alpha 渐变。

开销分析

- 渲染
  - 按 Canvas 合并 Mesh（BatchedMesh）

  - 渲染顺序重排



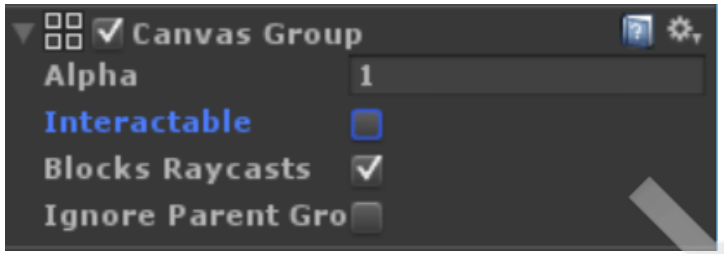
  - UI元素按渲染顺序及Atlas 分成小组（Submesh）

开销分析

- 渲染
  - 合理规划 Atlas
    - 同时同地
    - 适当冗余
  - 尽可能合并静态UI元素
    - 确保静态
  - 去除空 Sprite
    - 改变“增大检测区域”的方式
  - Mask -> RectMask 2D

开销分析

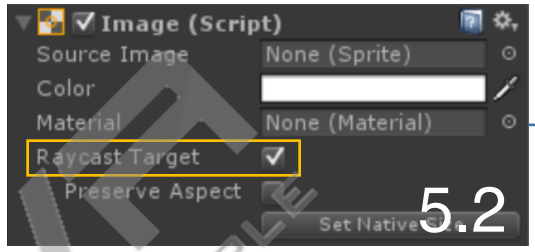
- 事件检测
  - 过滤不需要检测的 Canvas
    - CanvasGroup (Interactable)

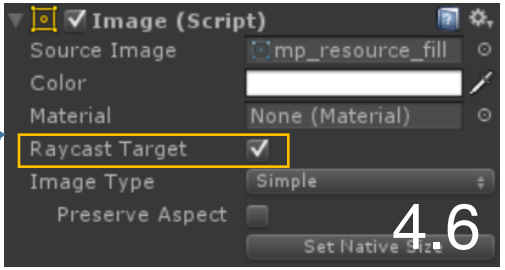


    - GraphicRaycaster (Enable)

开销分析

- 事件检测
  - 引入 Unity 5.2 的优化方法



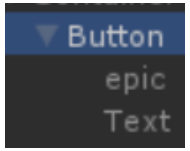


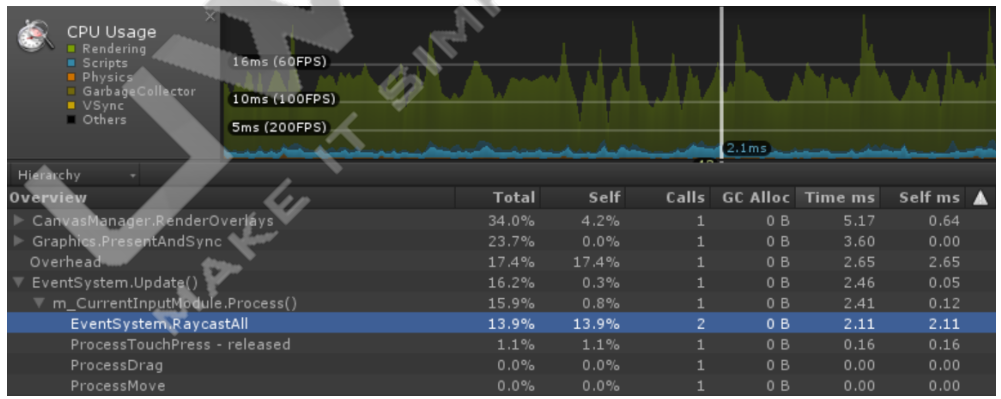

```
[NonSerialized] static readonly List<Graphic> s_SortedGraphics = new List<Graphic>();
private static void Raycast(Canvas canvas, Camera eventCamera, Vector2 pointerPosition, List<Graphic> results)
{
    // Debug.Log("ttt" + pointerPoision + ":::" + camera);
    // Necessary for the event system
    var foundGraphics = GraphicRegistry.GetGraphicsForCanvas(canvas);
    s_SortedGraphics.Clear();
    for (int i = 0; i < foundGraphics.Count; ++i)
    {
        Graphic graphic = foundGraphics[i];

        // -1 means it hasn't been processed by the canvas, which means it isn't actually drawn
        if (graphic.depth == -1 || graphic.raycastTarget == false)
            continue;

        if (!RectTransformUtility.RectangleContainsScreenPoint(graphic.rectTransform, pointerPosition, eventCamera))
            continue;

        if (graphic.Raycast(pointerPosition, eventCamera))
        {
            s_SortedGraphics.Add(graphic);
        }
    }
```

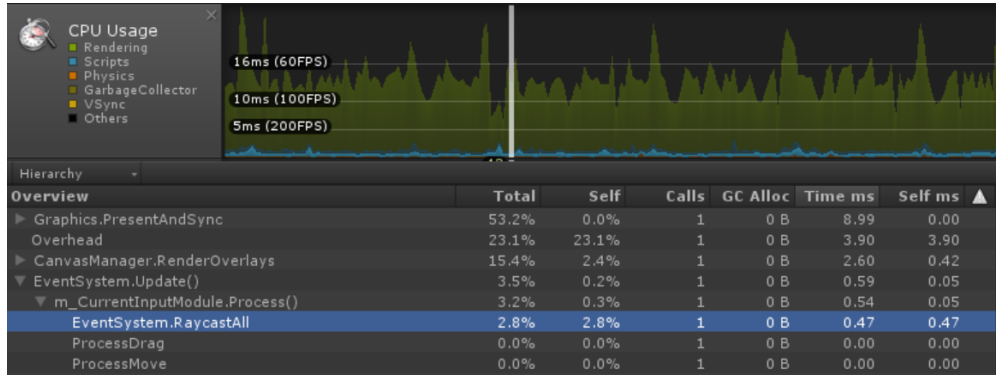
开销分析

- 事件检测
  - 过滤不需要检测的Graphic
  - 一个 Item 只需要一个 Target 进行检测

```
Graphic[] graphics = Resources.FindObjectsOfTypeAll<Graphic>();
foreach (Graphic g in graphics)
{
    if (g.GetComponent<Button>() == null)
    {
        g.raycastTarget = false;
    }
}
```

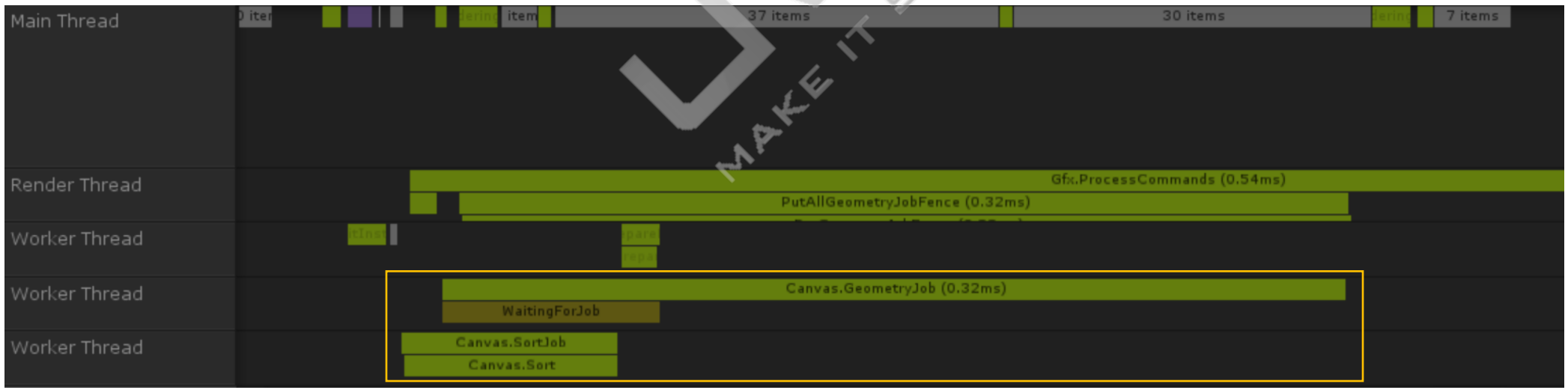






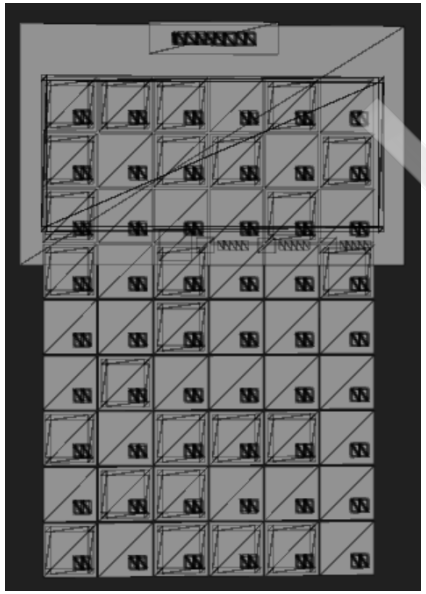
开销分析

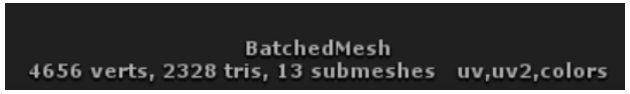
- 多线程 Unity 5.2
  - WaitingForJob
  - PutAllGeometryJob



界面切换

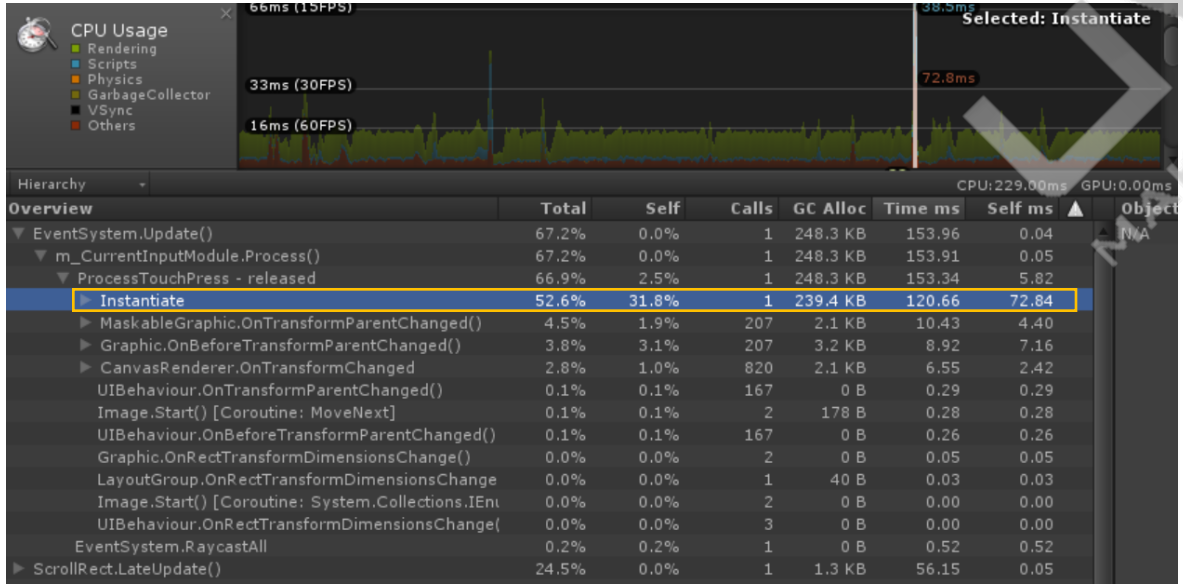
- Instantiate/Destroy
- GameObject.Activate/Deactivate
- CanvasRenderer.OnRectTransformChange
- CullingMask

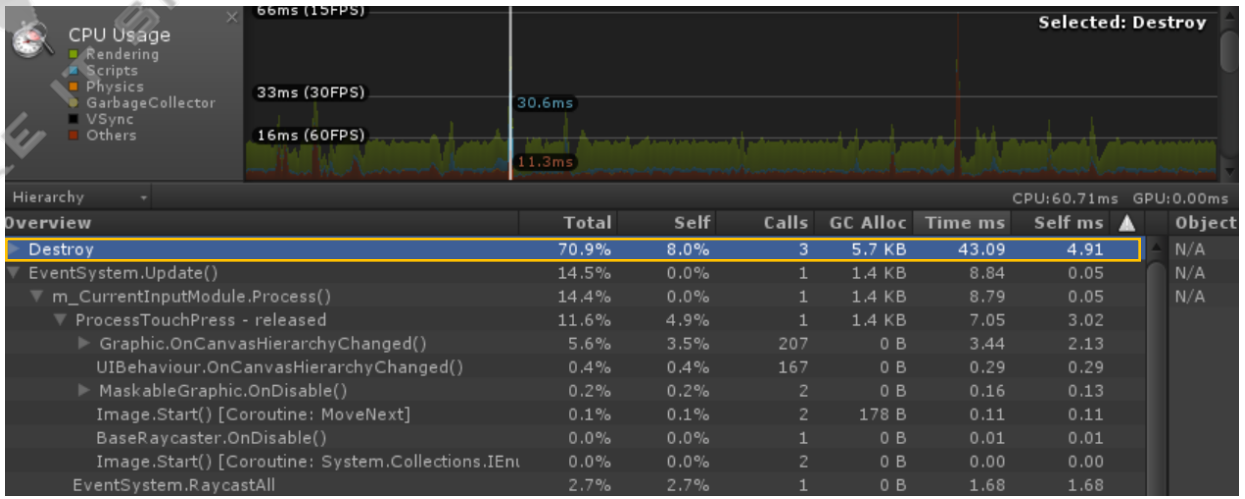




界面切换

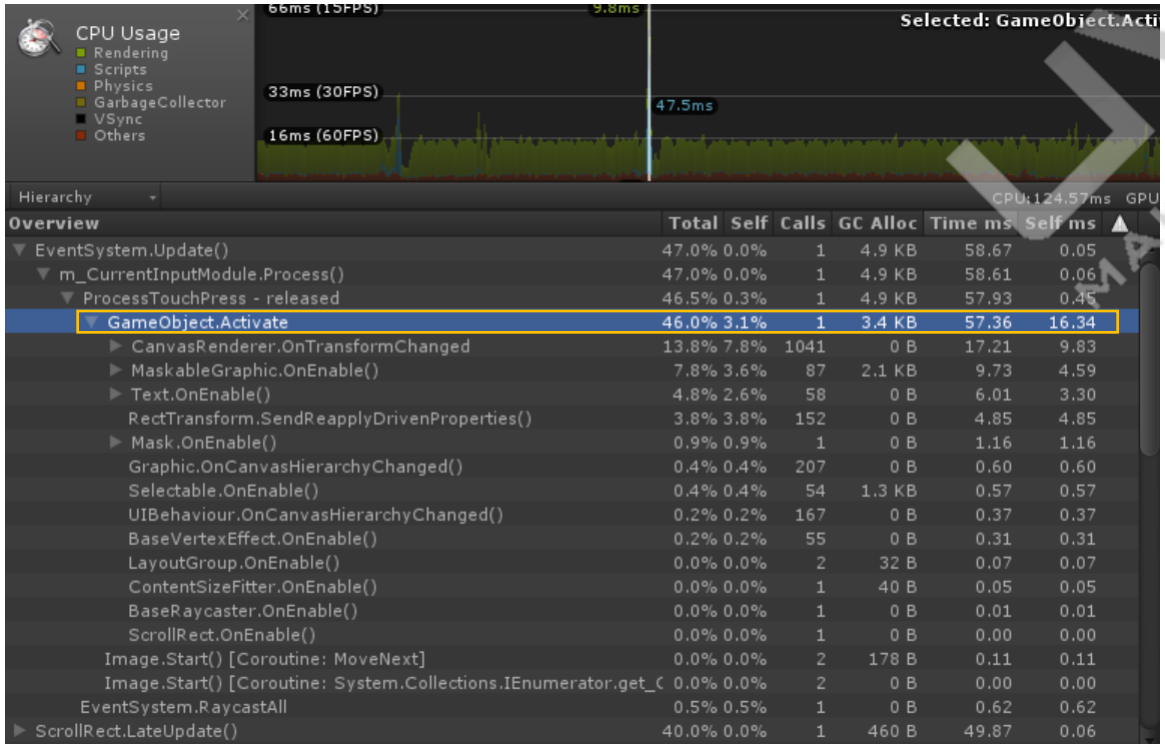
- Instantiate/Destroy
  - 隐藏 60.71ms
  - 显示 229.00ms

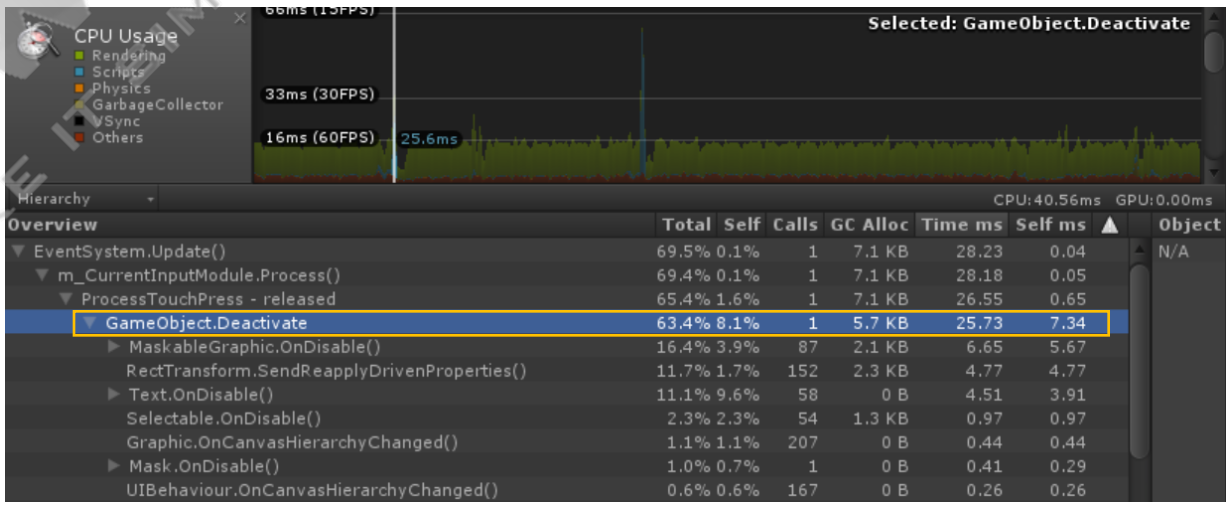




界面切换

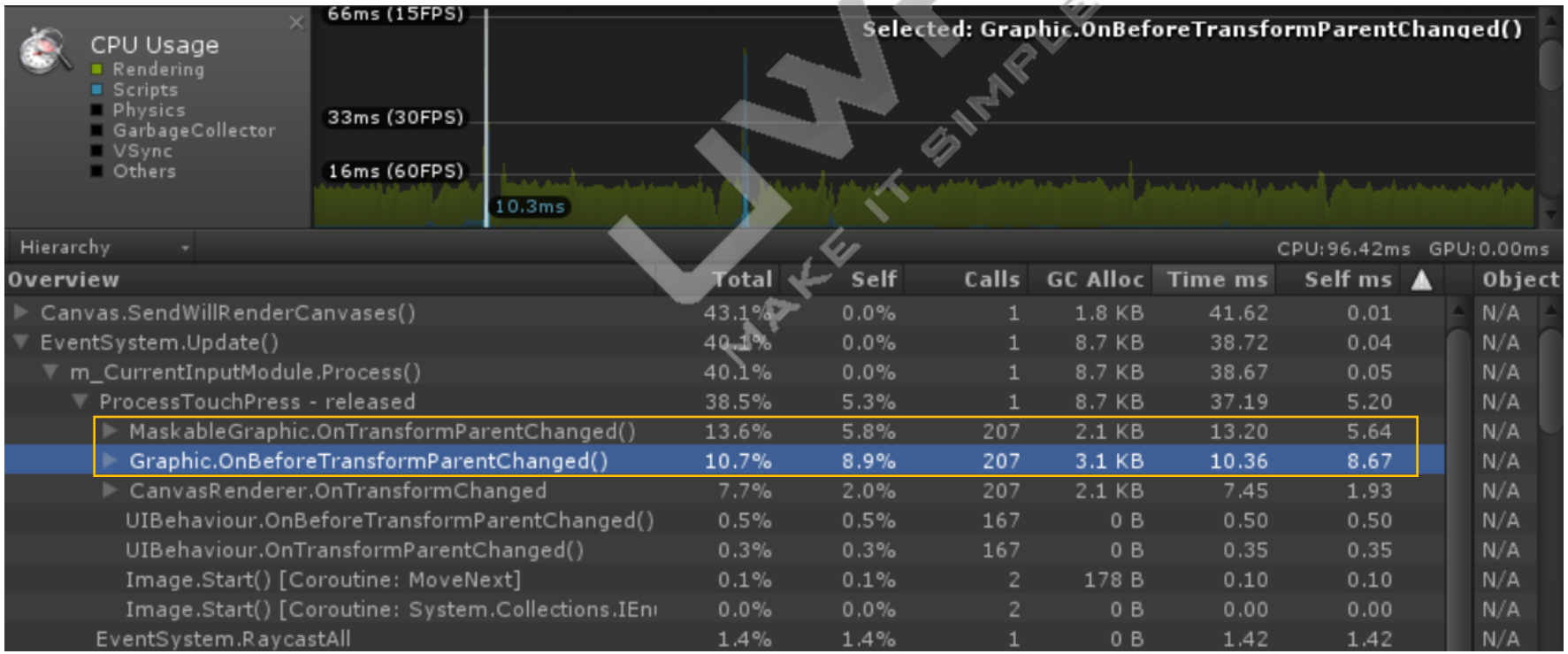
- GameObject.Activate/Deactivate
  - 隐藏 40.56ms
  - 显示 124.57ms





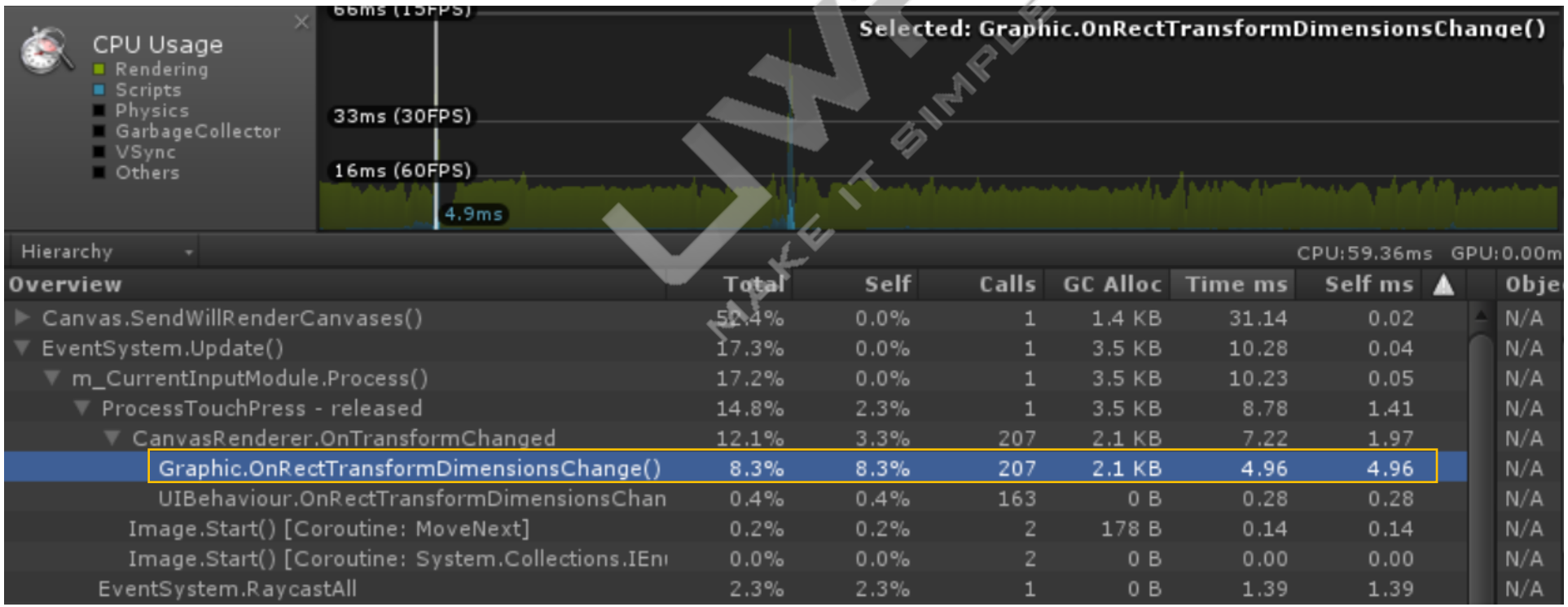
界面切换

- CanvasRenderer.OnRectTransformChange
  - 不修改 parent，避免 XXXParentChanged 回调



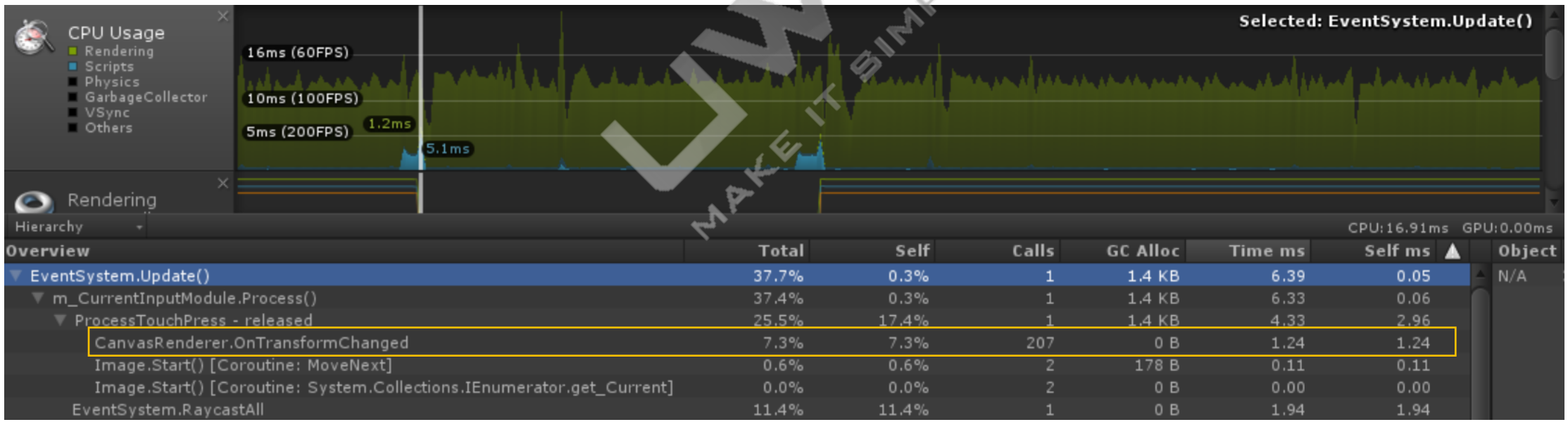
界面切换

- CanvasRenderer.OnRectTransformChange
  - 尽量避免 Pixel Perfect



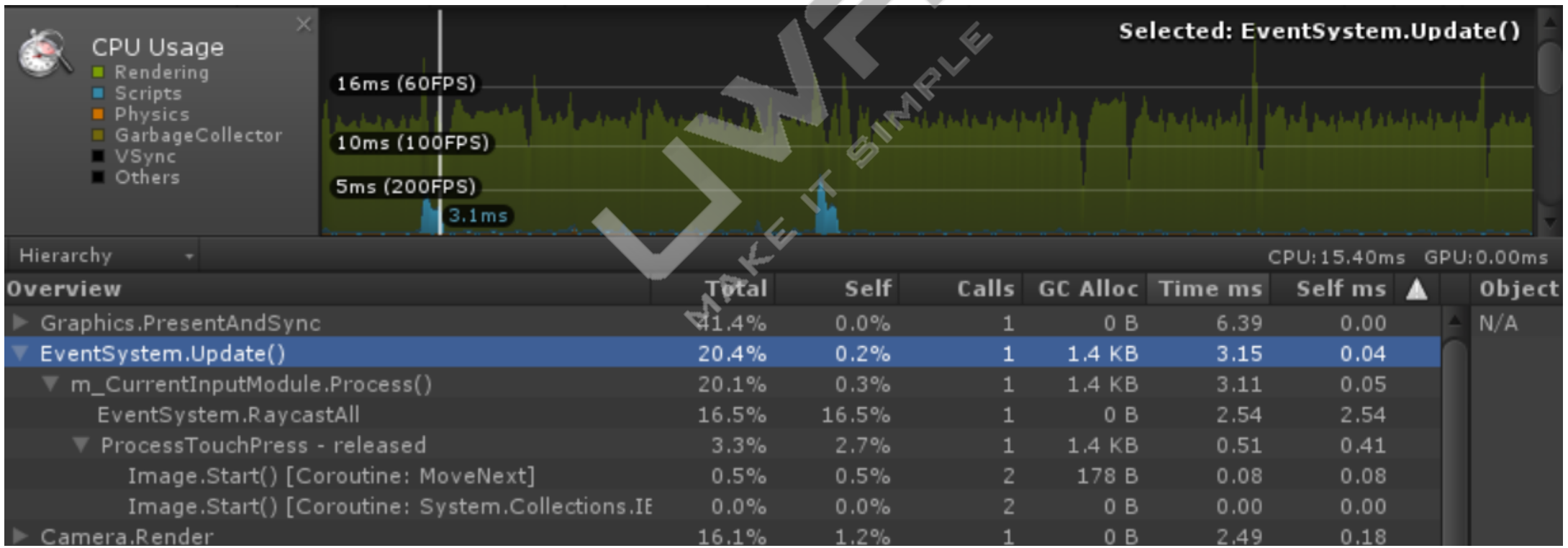
界面切换

- CanvasRenderer.OnRectTransformChange



界面切换

- CullingMask



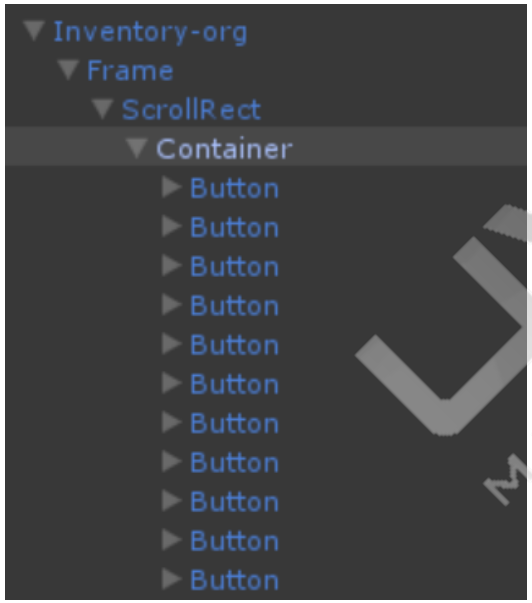
界面切换

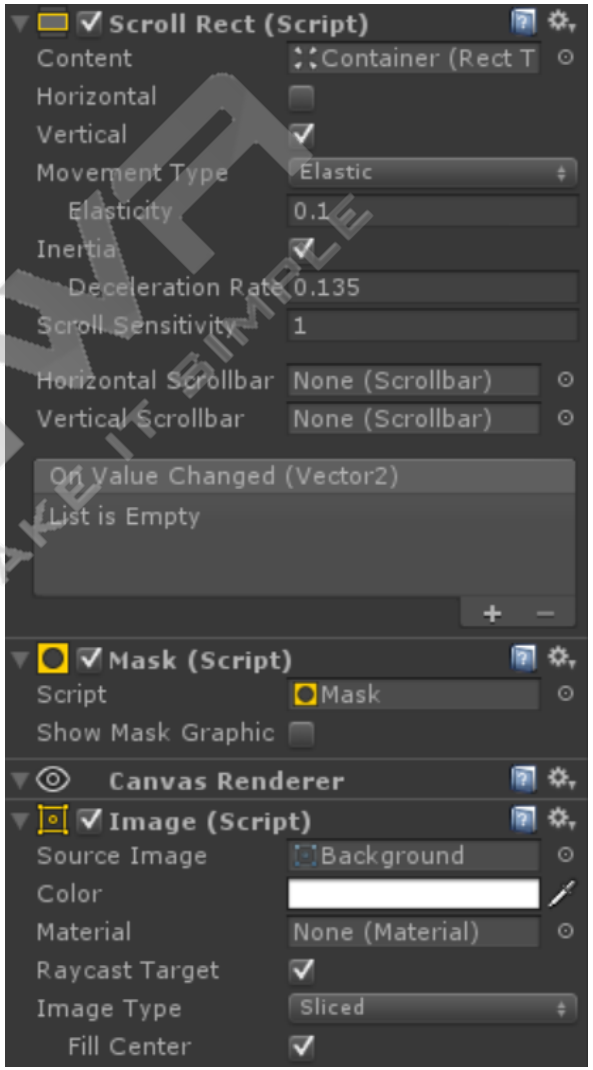
- CanvasRenderer.OnRectTransformChange/CullingMask
  - 移出后，确认 Draw Call 的变化
    - 5.2之后，移出Canvas并不会减少DrawCall
  - 停止更新“隐藏”界面的动态元素
  - 禁用事件点击处理
    - CanvasGroup
    - GraphicsRaycaster

背包界面

- 背包设置

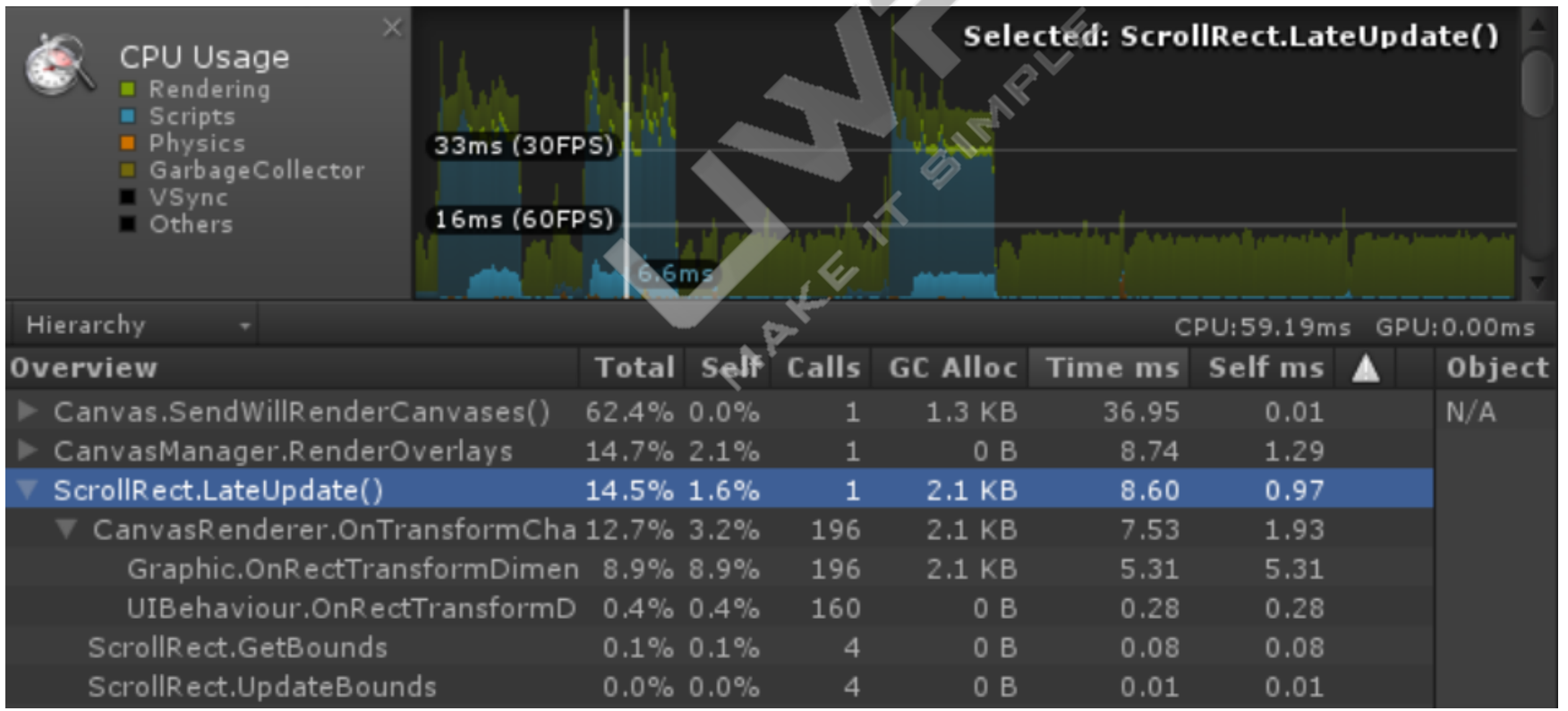

INVENTORY


Inventory-org
Frame
ScrollRect
Container
Button
Button
Button
Button
Button
Button
Button
Button
Button
Button


Scroll Rect (Script)
Content
Container (Rect T
Horizontal
Vertical
Movement Type
Elastic
Elasticity
0.1
Inertia
Deceleration Rate
0.135
Scroll Sensitivity
1
Horizontal Scrollbar
None (Scrollbar)
Vertical Scrollbar
None (Scrollbar)
On Value Changed (Vector2)
List is Empty
Mask (Script)
Script
Mask
Show Mask Graphic
Canvas Renderer
Image (Script)
Source Image
Background
Color
Material
None (Material)
Raycast Target
Image Type
Sliced
Fill Center

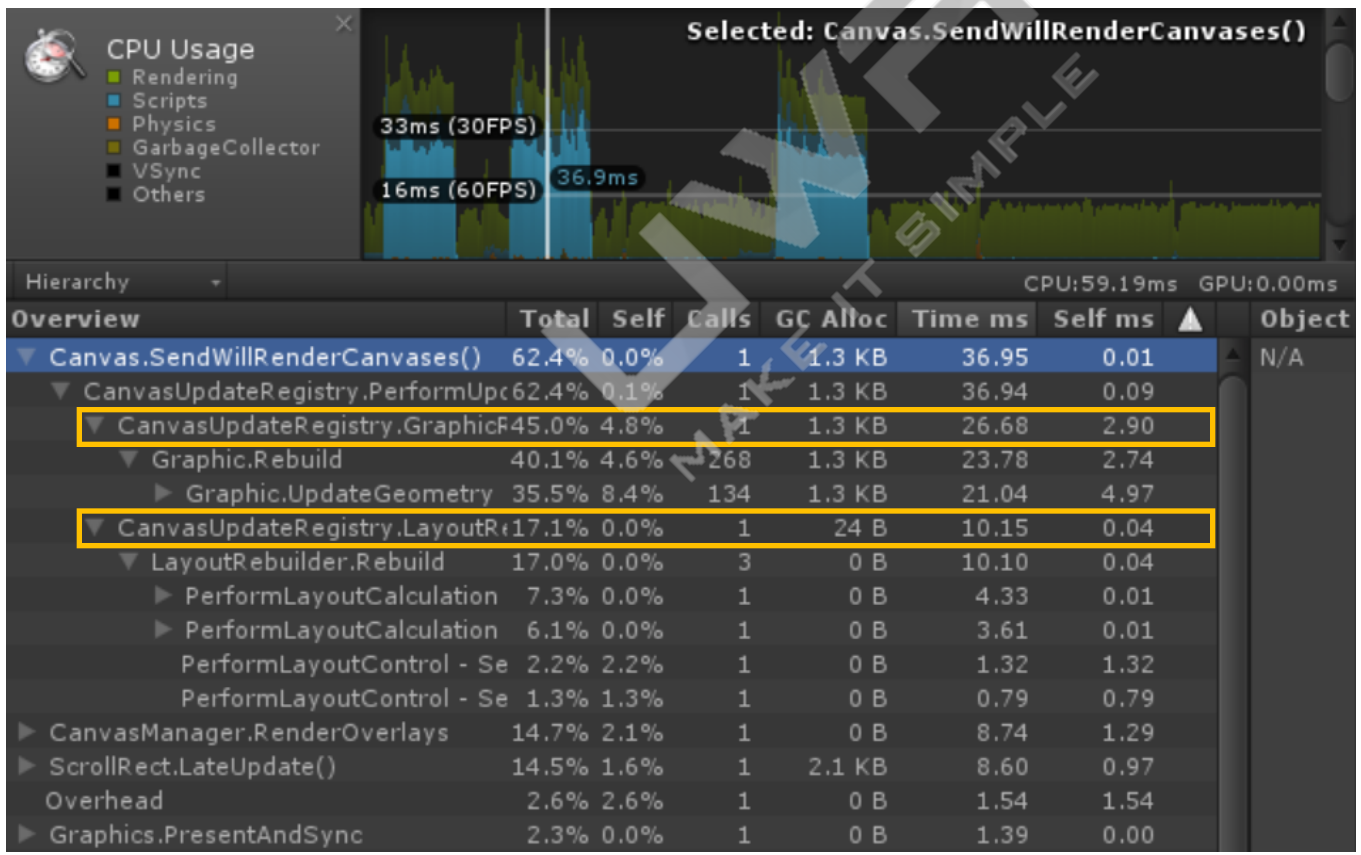
背包界面

- 拖动开销 CPU:59.19ms



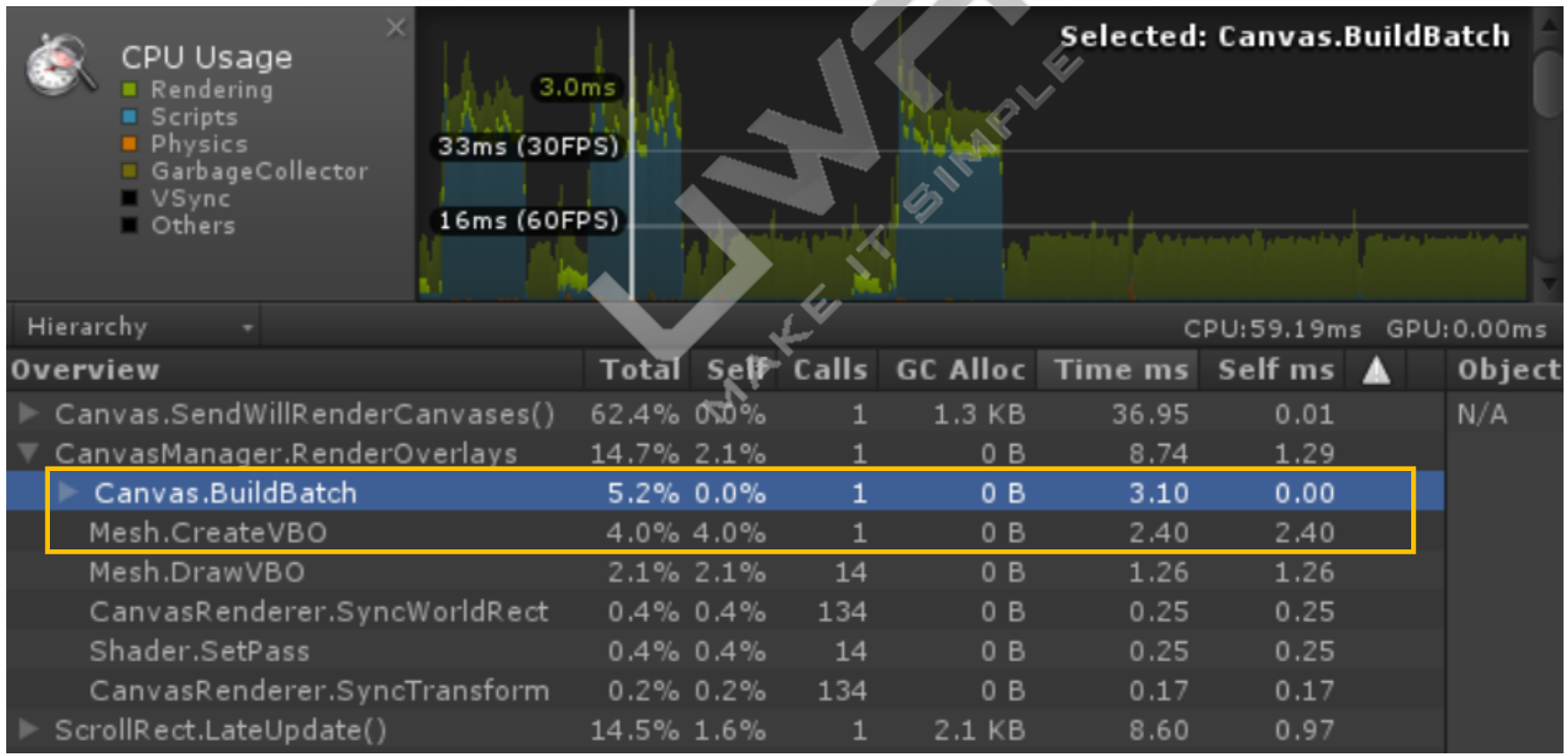
背包界面

- 拖动开销 CPU:59.19ms



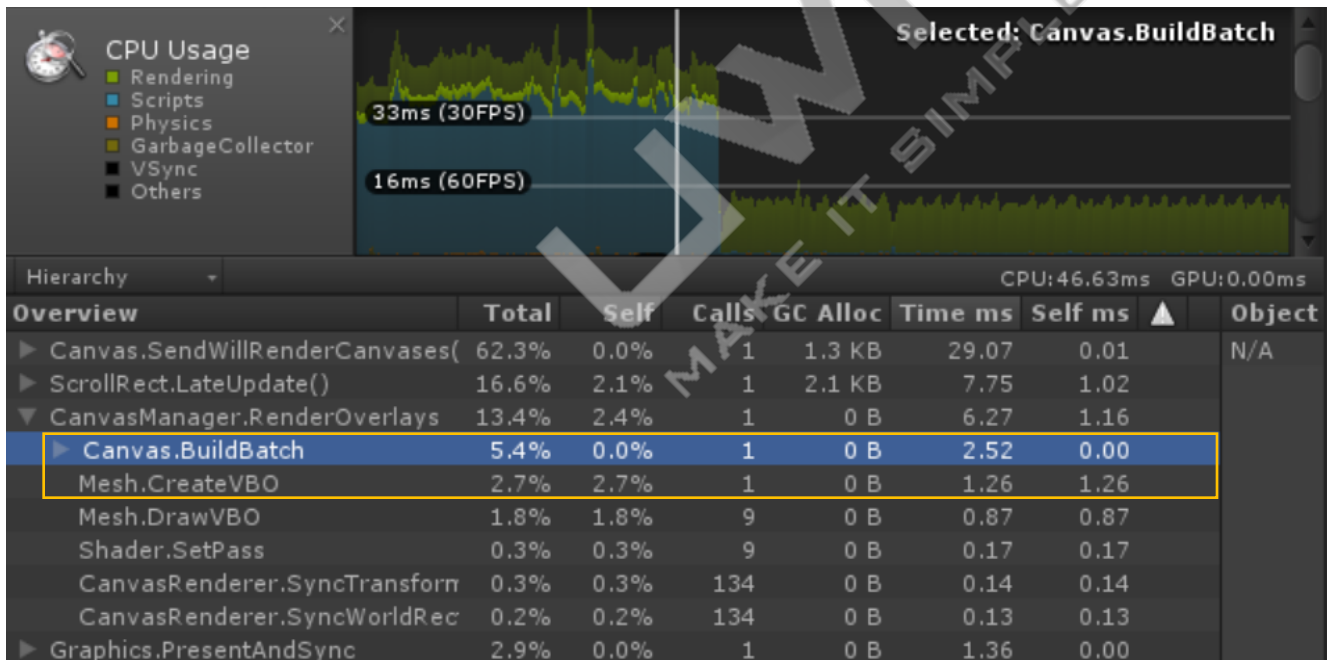
背包界面

- 拖动开销 CPU:59.19ms



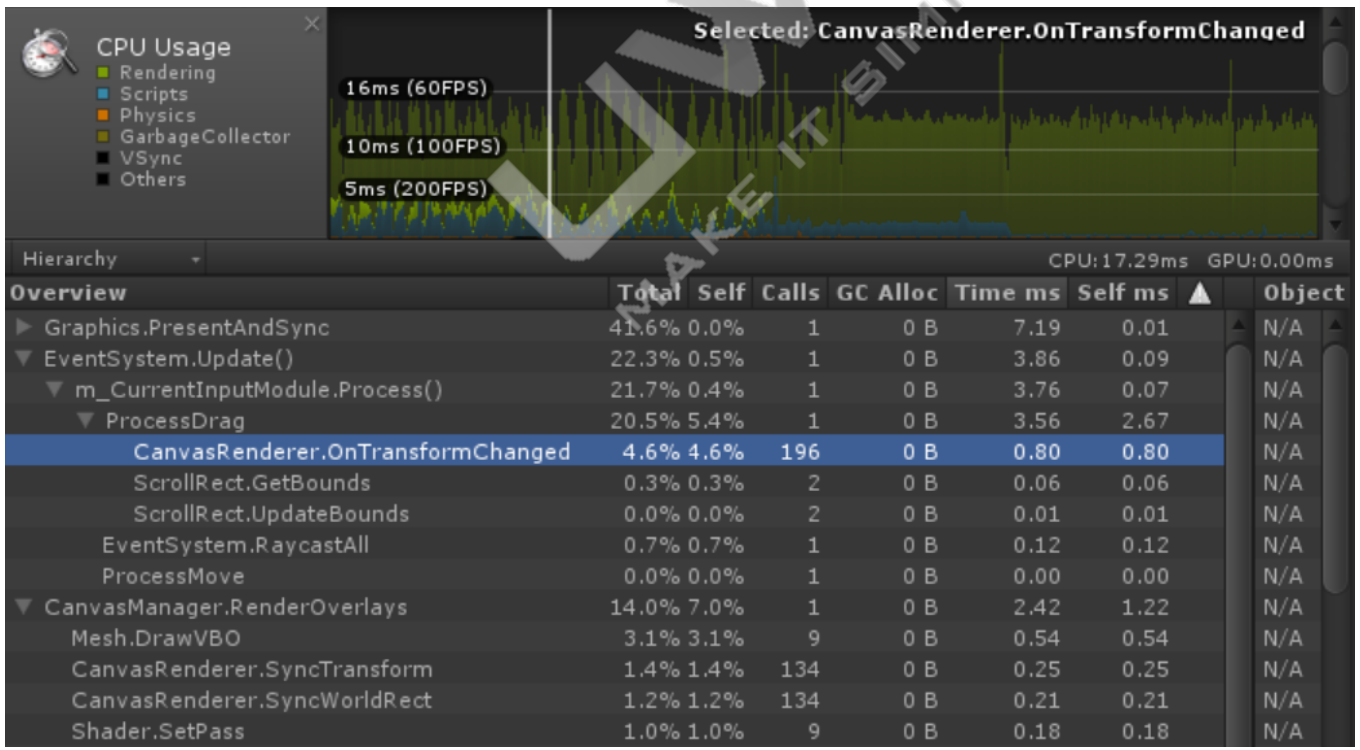
背包界面

- 拖动开销 CPU:46.63ms
  - 将滚动部分独立成 Canvas，以减小 Canvas.BuildBatch 的范围



背包界面

- 拖动开销 CPU:17.29ms
  - 尽可能不用 PixelPerfect，避免 Canvas.SendWillRenderCanvases() 以及 Canvas.BuildBatch



Overview	Total	Self	Calls	GC Alloc	Time ms	Self ms	Object
Graphics.PresentAndSync	41.6%	0.0%	1	0 B	7.19	0.01	N/A
EventSystem.Update()	22.3%	0.5%	1	0 B	3.86	0.09	N/A
m_CurrentInputModule.Process()	21.7%	0.4%	1	0 B	3.76	0.07	N/A
ProcessDrag	20.5%	5.4%	1	0 B	3.56	2.67	N/A
CanvasRenderer.OnTransformChanged	4.6%	4.6%	196	0 B	0.80	0.80	N/A
ScrollRect.GetBounds	0.3%	0.3%	2	0 B	0.06	0.06	N/A
ScrollRect.UpdateBounds	0.0%	0.0%	2	0 B	0.01	0.01	N/A
EventSystem.RaycastAll	0.7%	0.7%	1	0 B	0.12	0.12	N/A
ProcessMove	0.0%	0.0%	1	0 B	0.00	0.00	N/A
CanvasManager.RenderOverlays	14.0%	7.0%	1	0 B	2.42	1.22	N/A
Mesh.DrawVBO	3.1%	3.1%	9	0 B	0.54	0.54	N/A
CanvasRenderer.SyncTransform	1.4%	1.4%	134	0 B	0.25	0.25	N/A
CanvasRenderer.SyncWorldRect	1.2%	1.2%	134	0 B	0.21	0.21	N/A
Shader.SetPass	1.0%	1.0%	9	0 B	0.18	0.18	N/A

www.uwa4d.com