What is Scratch Card?

This is easy-to-use asset, which allows you to create scratch card objects. So all you need is to add prefab to the scene:

- **ScratchCard.prefab**

choose Main Camera, set Sprite to set your scratch object and Erase(brush) texture.

This works well on Personal and Pro Unity, suitable for all platforms. All takes less then 1 mb on your drive.

Quick Start

As mentioned above, to create scratch object, add prefab «**ScratchCard**» from «**Assets/ScratchCard/Prefabs/ScratchCard.prefab**», you need set Camera, Sprite for scratching, Brush Texture and Scratch Surface will work! That's all!

ScratchCardManager script

ScratchCardManager script creates and configures ScratchCard. **ScratchCardManager** script has parameters:

- **Main Camera** - Main Camera of scene;
- **RenderType** - render type of scratch card: MeshRenderer, SpriteRenderer or CanvasRenderer;
- **ScratchSurfaceSprite** - sprite for scratching;
- **ScratchSurfaceSpriteHasAlpha** - does sprite for scratching has alpha-channel;
- **MaskProgressCutOffValue** - value for comparing sampled RenderTexture alpha values with sampled Source Texture alpha values, using by MaskProgressCutOff shader only with ScratchSurfaceSprite without alpha-channel;
- **EraseTexture** - texture for erasing (erasing brush);
- **EraseTextureScale** - erasing brush scale;
- **Card** - reference to **ScratchCard** script;
- **Progress** - reference to **EraseProgress** script;
- **MeshCard** - reference to GameObject with MeshRenderer;
- **SpriteCard** - reference to GameObject with SpriteRenderer;
- **ImageCard** - reference to GameObject with Image;

After instantiate **ScratchCard.prefab**, you need set **Main Camera**, **CardType**(optional), **ScratchSurfaceSprite** and **EraseTexture**(brush texture)**.**

ScratchCardManager script has methods:

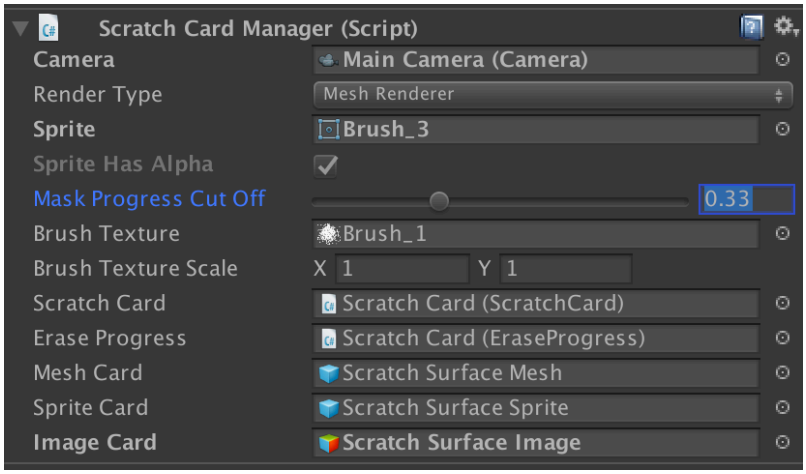
`void Awake()` - setting up the scratch card;

`public void SetEraseTexture(Texture texture)` - setting up the erase texture(brush) for the scratch card;

`public void ResetScratchCard()` - reset scratch card.

Note that if you use Canvas, ScratchCardManager.ImageCard object must be child of Canvas.

If you want to use ScratchCard images with transparent areas, for a more accurate progress result you can setup material for getting better result of progress calculations, using Cut Off field. Cut Off field value is using to compare sampled RenderTexture alpha values with sampled Source Texture alpha values.



ScratchCard script

ScratchCard script creates and configures RenderTexture then draws the quads in RenderTexture. You can use SpriteRenderer, MeshRenderer or Image for scratch effect.

ScratchCard script has such parameters:

- **Main Camera** - Main Camera of scene;
- **Surface** - transform of Surface object, which can contain Render Component;
- **RenderTextureQuality** - quality(size) of RenderTexture texture: High, Medium, Low;
- **Eraser** - Material of Eraser(brush);
- **Progress** - Material of Progress;
- **Scratch Surface** - Material of Scratch Surface.
- **Reset** - Reset Scratch card to original state;
- **RenderTexture** - RenderTexture for scratching;
- **BrushScale** - scale of brush.

ScratchCard script has properties:

`public ScratchMode Mode` - scratch card mode: erase or restore;

`public bool IsScratching` - returns if user scratch surface currently.

ScratchCard script has methods:

`public void FillInstantly()` - fills RenderTexture with white color instantly;

`public void ClearInstantly()` - fills RenderTexture with clear color instantly;

`public void Clear()` - fills scratch card with clear color in the next Update;

`public void Reset()` - re-creates RenderTexture and clears it in the next Update.

EraseProgress script

EraseProgress script creates and configures RenderTexture then calculates the amount of 15x15 pixels alpha-channel from the ScratchCard RenderTexture. Using this script can affect on the performance of device.

EraseProgress script has events:

`public event ProgressHandler OnProgress;` - invoke, when user scratches/restores surface;

`public event ProgressHandler OnCompleted;` - invoke, when user completed scratching/restoring surface.

EraseProgress script has methods:

`public float GetProgress()` - returns scratch erase progress from 0 to 1;

`public void UpdateProgress()` - updates scratch progress;

`public void ResetProgress()` - resets isCompleted flag for for further interaction.

Multiply scratch cards

If you want to use a few scratch card at the same time, you need choose specific Scratch Surface Material for **ScratchCard** script for each scratch card or set it to «None» - in that way, **ScratchCardManager** will create it automatically.

WebGL Build

By some reason, WebGL may ignore .shader and doesn't add files into build. To fix it, add shaders to Graphics Settings or move them to «Assets/Resources/» folder.

Please let me know if you have any questions.
E-mail: unitymedved@gmail.com