Summary

Hello and welcome to this worked example.

This worked example converts the assets from Synty Studios' Fantasy Heroes pack to use the Textures of their Fantasy Kingdom pack. It also corrects some issues and rescales the character Models.

1. Install the following packages:
PolygonFantasyHeroCharacters
PolygonFantasyKingdom
2. Create a new folder to Hold the Instructions and navigate to it:
Assets/Recipes/PFHC2PFK
3. Create a new Recipe (Recipes/Recipe)
4. Create a new Instruction (Recipes/Instructions/Material Mapping)
5. Create a new Instruction (Recipes/Instructions/Process Prefabs)
6. Create a new Instruction (Recipes/GameObjects/Process Meshes)
7. Create a new Instruction (Recipes/GameObjects/SaveAs)
8. Create a new Instruction (Recipes/Meshes/UV Remap)
9. Create a new Instruction (Recipes/Other/TextureFeatureSet) and rename PolygonFantasyHero_TFS.
10. Create a new Instruction (Recipes/Other/TextureFeatureSet) and rename PolygonFantasyKingdom_TFS.
11. Open the Recipe from step 3 and set the following fields:
    - Title: Fantasy Heroes 2 Fantasy Kingdom
    - Description: Convert the Polygon Fantasy Heroes package to use the Textures from Polygon Fantasy Kingdom
    - Output Path: Assets/PFHC2PFK
    - Instructions: Add instructions from step 4 and 5 in that order
12. Open the Instruction from step 4 and set the following fields:
    - Notes: This will replace the material from PFHC with the one from PFK
    - Source: Assets/PolygonFantasyHeroCharacters/Materials/FantasyHero.mat
    - Destination: Assets/PolygonFantasyKingdom/Materials/PolygonFantasyKingdom_Mat_01_A.mat
13. Open the Instruction from step 5 and set the following fields:
    - Notes: This will load the Prefabs we want to convert
    - Prefabs: Assets/PolygonFantasyHeroCharacters/Prefabs/Characters_Presets/Chr_FantasyHero_Preset_1.prefab
    - Instructions: Add the Instructions from steps 6 and 7 in that order
14. Open the Instruction from step 6 and set the following fields:
    - Notes: These Instructions will modify the Meshes
    - Instructions: Add the Mesh Instruction from step 8
15. Open the Instruction from step 7 and set the following fields:
    - Notes: This causes our loaded prefab to be saved as a new prefab
16. Open the Mesh Instruction from step 8 and set the following fields:

- Notes: This will map UV regions from one texture to another
- Default: Error
- Drop the PolygonFantasyHero_TFS from step 9 onto the Source field.
- Drop the PolygonFantasyKingdom_TFS from step 10 onto the Destination field.

17. Open the Recipe and Build. At this point we should have a Character with incorrect colouring and some SharedMesh is Null errors. We will fix the colouring first then the errors. Take the time to look at the output directory: Assets/PFHC2PFK.
18. Copy the texture Assets/PolygonFantasyHeroCharacters/Textures/Textures/PolygonFantasyHero_Texture _01_A.png and convert to a Sprite Texture.
    - Texture Type: Sprite ("D and UI)
    - Sprite Mode: Multiple
19. Using the Sprite Editor create sprites for the features used.
    - Note: you may have to install the sprite editor from the Package Manager – search for '2D Sprite'.
    - Note: You can use the Texture Inspector (Tools/Recipes/Texture Inspector) to identify the features used by the models.
20. Repeat the process with a copy of the destination Texture. This time consider where you want to map features. Assets/PolygonFantasyKingdom/Textures/PolygonFantasyKingdom_Texture_01_A.png
    - Note: Try to be consistent with naming – the UV Remap Instruction can auto set features with the same name.
21. When Completed open PolygonFantasyHero_TFS and drag the Sprite Texture step 18 to populate the features.
22. Open the Mesh Instruction from step 8 and use the Populate Tool. This will add entries for all Source Features. Any Features that have a matching name in the Destination Feature Set will be pre-filled in. We do however need to make some fixes:
23. Remove :
    - BodyArt_01
    - BodyArt_02
    - BodyArt_03
24. Move to the bottom. The Features are matched in order – we don't want them interfering. These two fixes catch some errant UVs seen as lines across colour blocks in the Texture Inspector.
    - Color_02_Fix
    - Metal_02_Fix
25. Set Mappings. The Mail Feature is larger in the Source, so we have split it into multiple aligned features. The Stubble, Skin and Scar _02 are for body paint – Just mapping to _01 for now. The Fix Features map some errant UVs to better positions.
    - Mail_02 -> Mail_01
    - Mail_03 -> Mail_01
    - Mail_04 -> Mail_01
    - Stubble_02-> Stubble_01
    - Skin_02-> Skin_01
    - Scar_02-> Scar_01
    - Color_02_Fix-> Color_02
    - Metal_02_Fix-> Metal_02

26. The Modular_Characters are missing some Meshes. This is what is causing the Errors on build. The first is on the gameObject Chr_Head_Female_23. Inspection of the models reveals it does not exist – so lets just remove the GameObject. Create a new Instruction (Recipes/GameObjects/Remove Child) and rename to Fix_Chr_Head_Female_23. Open and set the Child field to Modular_Characters/Female_Parts/Female_00_Head/Female_Head_All_Elements/Chr_Head_Female_23. Finally add this to the Instructions List from the Prefab Processor in step 5 and move it to the front – we want to execute this before processing the Meshes.

27. The rest of the errors need meshes setting. Create a new Instruction (Recipes/GameObjects/Set Mesh) and rename to Fix_Chr_Female_Eyebrow_01. Open and set Target Path to Modular_Characters/Female_Parts/Female_01_Eyebrows/Chr_Female_Eyebrow_01. Add the mesh from Assets/PolygonFantasyHeroCharacters/Models/ModularCharacters.fbx/Chr_Eyebrow_female_01. Finally add this to the Instructions List from the Prefab Processor in step 5 and move it to the front – we want to execute this before processing the Meshes.

28. Repeat step 27 for Eyebrows 02 -07

29. Clean the Scene. Build The recipe. We should now have no errors.

30. Now let us add other prefabs. Rename the Process Prefabs Instruction from Step 5 to Process_Prefabs_Character.

31. Now we will add the other Character presets. It helps if you lock the Inspector for this step. Remove the existing entry from the Prefab list. Now drag-drop the entire contents of Assets/PolygonFantasyHeroCharacters/Prefabs/Characters_Presets onto the [Drop Prefabs to Add] tool at the bottom of the inspector.

32. Now We will add the weapons. Create a new Process Prefabs Instruction (as per step 5) and rename Process_Prefabs_Weapons. Add the Instructions Process_Meshes and Save_Prefab_As from steps 6 and 7. Now drag-drop the entire contents of (Assets/PolygonFantasyHeroCharacters/Prefabs/Weapons) onto the [Drop Prefabs to Add] tool at the bottom of the inspector.

33. The Weapons use a different Material. Create a new Material Mapping Instruction (Recipes/Instructions/Material Mapping). Rename to MaterialMapping_Weapons
    - Source: Assets/PolygonFantasyHeroCharacters/Materials/CustomMaterials/FantasyHero_01.mat
    - Destination: Assets/PolygonFantasyKingdom/Materials/PolygonFantasyKingdom_Mat_01_A.mat

34. Add the 2 new Instructions from steps 32 & 33 to our Recipe. Arrange the two Material Mapping Instructions to the top.

35. Clean the Scene. Build the Recipe. Unfortunately, they are all stacked up on top of each other – not a problem for the prefabs, but we could easily miss problems. Create a new Instruction (Recipes/GameObjects/Position In Line)
    - Padding: 2
    - Dynamically Spacing: false

36. Create a new Instruction (Recipes/GameObjects/Start Position) and rename Start_Position_Weapons
    - Start Position: (0,0,2)

37. Add the Position In Line Instruction from step 35 to the end of the Instruction lists of our two Process Prefabs Instructions – We want these after the Save_Prefab_As Instructions as we want the prefab to be ate the origin.

38. Add the Start_Position_Weapons from step 36 to the Recipe Instructions. Move between the Character and Weapons Process Prefab Instructions.

39. Clean the Scene. Build the Recipe. The Purple Shields are the shields with heraldic motifs. For simplicity we are just going to remove them. The single motif in the destination Texture does not cleanly map so for this example we will just remove them.

40. At this point it is worth dropping an Animation Controller onto an Animator just to check it all works.

41. If we look at the Root GameObject under Chr_FantasyHero_Preset_1, we see it has a scale of (0.01, 0.01, 0.01). To fix this we will scale the Mesh. Start with renaming the Process Meshes Instruction from step 6 to Process_Meshes_Character.

42. Duplicate Process_Meshes_Character and rename to Process_Meshes_Weapons. We only need to fix the Character models. Change Process_Prefabs_Weapons to use Process_Meshes_Character.

43. Create a Mesh Instruction (Recipes/Meshes/Transform) and add it to the beginning of the Process_Meshes_Character Instruction set. Open the new Mesh Instruction and set Scale to (0.01, 0.01, 0.01)

44. Create a new Instruction (Recipes/GameObjects/Set_Scale_To_One) and rename to Reset Scale. Add this new Instruction to the Process_Prefabs_Character Instruction set and reposition to just before the call to Process_Meshes_Character. This will cascade a scale change to (1,1,1) through the children of the GameObject pointed to by target path (the top level as it is blank).

45. We now need to Create a new Avatar as the rescaling will render the original useless. Create a new Instruction (Recipes/GameObjects/Build Avatar). Add this new Instruction to the Process_Prefabs_Character Instruction set and reposition to just before the call to Process_Meshes_Character but after Reset Scale. This will build a new Avatar. It will use the Bone mappings from the original Avatar but only uses the Bones under the root for the skeleton.

46. Clean the Scene and Build The recipe. Again drop an Animation Controller onto an Animator just to check it all works.

47. Looking down the line of characters we see some torsos out of place – specifically the Female Torso 20 on Prefab 59 (and others). To correct this we need to set the Transform for each preset and update the mesh.

48. Create an Instruction (Recipes/GameObject/Transform) and rename to Fix_Chr_Torso_Female_20_ContainerTransform. Add the new instruction to Process_Prefabs_Character's Instruction list before the Set_Scale_To_One instruction (with the other fixes). Open the new Instruction and set
    - Target: Modular_Characters/Female_Parts/Female_03_Torso/Chr_Torso_Female_20

49. Create a new Instruction (Recipes/Mesh/Conditional Set) and rename to Fix_Chr_Torso_Female_20. Add the errant mesh to the Mesh List: Assets/PolygonFantasyHeroCharacters/Models/ModularCharacters.fbx
    Add this Mesh instruction to the beginning of the Process_Meshes_Character Instruction set.

50. Create a new Instruction (Recipes/Mesh/Transform) and rename to Fix_Chr_Torso_Female_20_Transform. Add this new Instruction to the Conditional Set

Instruction set. Open the Mesh Transform Instruction and set the Position to (1099.924, 144.4438,0) – these numbers come from the position of the GameObject in the original prefab.

51. Clean the Scene and Build The recipe. Again drop an Animation Controller onto Chr_FantasyHero_Preset_59's Animator just to check it all works.