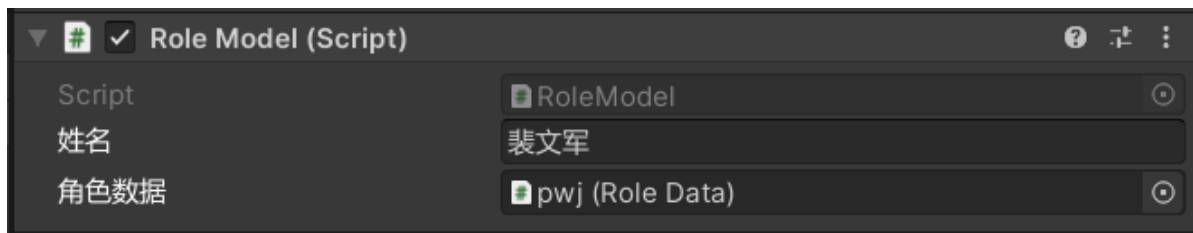
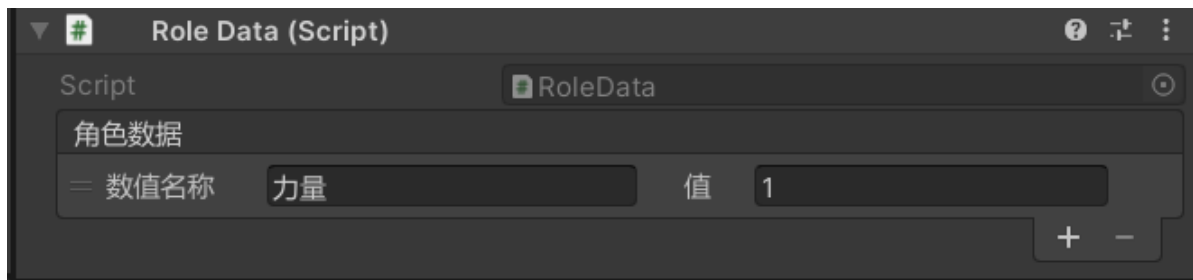


Use of role models

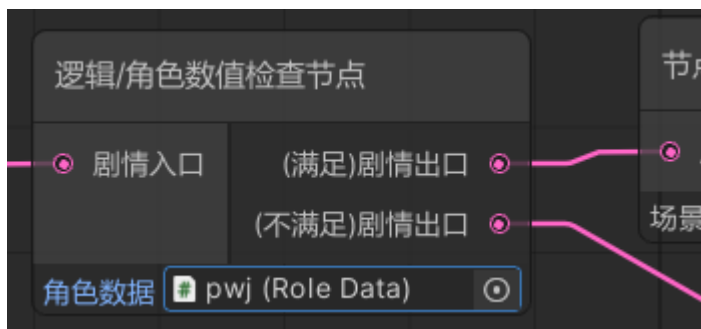


The role model is a class that holds basic information about the role, including an open field name, and a floating point list field, RoleData.

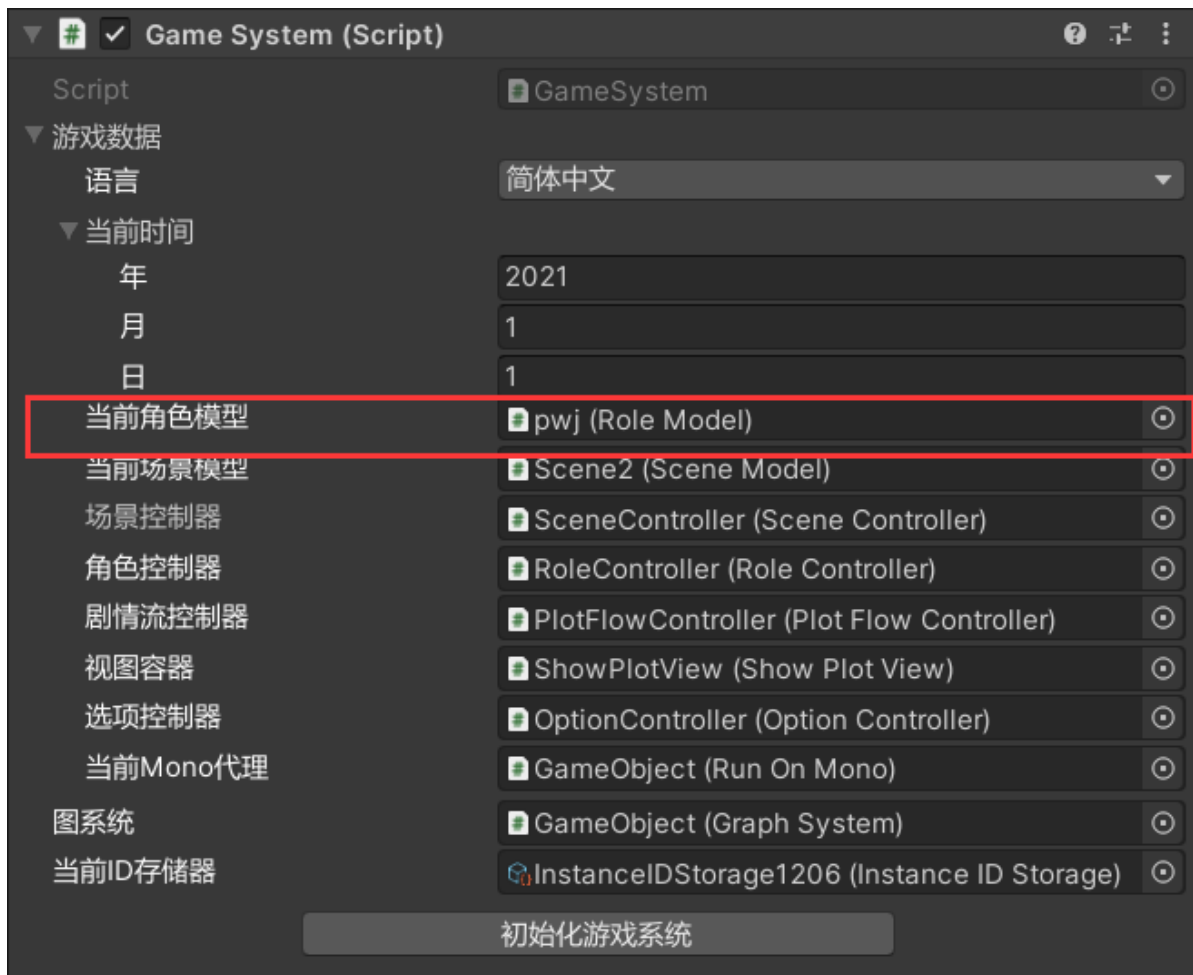


The character value is one of the basic information of the character, you can click the plus sign to add any number of character data, provide various uses in the game, provide various access methods of the underlying API, and can carry out various operations in the story flow.

For example, the story attached to the character data check node to limit the pass, only meet the corresponding character requirements of the value can pass.



The types of plot entry and exit are RoleData, usually the value of the protagonist. Of course, you can also attach another character's character model to replace it with another character's data, dynamically modifying it by attaching the character model to the (GameSystem) GameSystem or script.



Of course from a game development point of view it's usually about attaching the main character's value rather than the other person's. Of course, you can script that to change this value at the right time, even while the plot diagram is executing, because in the underlying code, RoleData is a direct reference to the data in GameSystem

```
GameSystem.Data.CurrentRoleModel.roleData = nodeObj.Execute(GameSystem.Data.CurrentRoleModel.roleData);
```