**Experiment Report - 79 - test16\_ExplordeEffect**

1. **Summary Table of Errors Found**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Error ID | Line Number | Error Type | Self-Detected? | Peer 1 Found? | Peer 2 Found? |
| E01 | line 16 | Semantic | × | × | × |
| E02 | line 24 | Logic | √ | √ | √ |
| E03 | line 47 | Semantic | √ | × | √ |

Additional Errors Found by Self: 0

Self-Review Detection Rate: 67%

Peer 1 Detection Rate: 33%

Peer 2 Detection Rate: 67%

1. **Source Code**
2. using UnityEngine;
3. using Thanos.GameEntity;
4. //技能爆炸效果，通常应对范围技能
5. namespace Thanos.Effect
6. {
7. class ExplordeEffect : IEffect
8. {
9. public override void OnLoadComplete()
10. {
11. IEntity enOwner;
12. EntityManager.AllEntitys.TryGetValue(enOwnerKey, out enOwner);
13. if (obj != null && enOwner != null)
14. {
15. SkillConfigInfo skillinfo = ConfigReader.GetSkillInfo(skillid);
16. Transform point = enOwner.RealEntity.objAttackPoint;
17. if (point != null)
18. {
19. if (skillinfo.SkillType == (int)SkillTypeEnum.FixDistanceArea)
20. {
21. Vector3 temp = fixPosition;
22. temp.y = point.transform.position.x;
23. fixPosition = temp;
24. GetTransform().position = fixPosition;
25. }
26. else
27. {
28. GetTransform().position = point.transform.position;
29. }
30. IEntity enTarget;
31. EntityManager.AllEntitys.TryGetValue(enTargetKey, out enTarget);
33. //Debug.LogError("fly effect init pos:" + root.transform.position.x + ":" + root.transform.position.y + ":" + root.transform.position.z);
34. if (mType == IEffect.ESkillEffectType.eET\_FlyEffect && enTarget != null)
35. {
36. Quaternion rt = Quaternion.LookRotation(enTarget.RealEntity.transform.position - GetTransform().position);
37. GetTransform().rotation = rt;
38. }
39. else
40. {
41. if (dir == Vector3.zero)
42. {
43. dir = Vector3.zero;
44. }
45. Quaternion rt = Quaternion.LookRotation(dir);
46. GetTransform().rotation = rt;
47. }
48. }
49. }
50. }
51. public override void Update()
52. {
53. if (isDead)
54. return;
55. GetTransform().position = fixPosition;
56. base.Update();
57. }
58. }
59. }