



```
2 using UnityEngine;
3 using UnityEngine.XR.ARFoundation;
4
5 public class PointCloudScanner : MonoBehaviour
6 {
7     [SerializeField]
8     private ARPointCloudManager pointCloudManager;
9
10    private Dictionary<ulong, Vector3> points = new Dictionary<ulong,
    Vector3>();
11
12    private void Awake()
13    {
14        // はじめはスキャンを停止しておく
15        pointCloudManager.enabled = false;
16    }
17
18    private void OnDestroy()
19    {
20        // GameObjectが破棄されるときにARPointCloudManagerの更新イベントを解除する
21
22        pointCloudManager.trackablesChanged.RemoveListener(OnTrackablesChanged);
23    }
24
25    public void StartScan()
26    {
27        // スキャンを開始する
28        pointCloudManager.enabled = true;
29
30        // ARPointCloudManagerの更新イベントを登録する
31
32        pointCloudManager.trackablesChanged.AddListener(OnTrackablesChanged);
33    }
34
35    public void StopScan()
36    {
37        // ARPointCloudManagerの更新イベントを解除する
38
39        pointCloudManager.trackablesChanged.RemoveListener(OnTrackablesChanged);
40
41        // スキャンを停止する
42        pointCloudManager.enabled = false;
43    }
44
45    private void
    OnTrackablesChanged(ARTrackablesChangedEventArgs<ARPointCloud> args)
46    {
47        // 更新されたポイントクラウドの位置情報を取得する
48        foreach (var pointCloud in args.updated)
49        {
50            if(!pointCloud.positions.HasValue)
51            {
52                continue;
53            }
54
55            if(!pointCloud.identifiers.HasValue)
56            {
57                continue;
58            }
59
60            // ポイントクラウドの位置情報をDictionaryに追加する
61            for (int i = 0; i < pointCloud.positions.Value.Length; i++)
62            {
63                var id = pointCloud.identifiers.Value[i];
64                var position = pointCloud.positions.Value[i];
65
66                if (points.ContainsKey(id))
```

```

65         {
66             points[id] = position;
67         }
68         else
69         {
70             points.Add(id, position);
71         }
72     }
73 }
74 }
75 }
76

```

点群データを公開する

- PointCloudScannerに点の位置情報のListのプロパティを追加
- private：クラス内で使用, public：クラス外に公開

</> C#

```

1  using System.Collections.Generic;
2  using UnityEngine;
3  using UnityEngine.XR.ARFoundation;
4
5  public class PointCloudScanner : MonoBehaviour
6  {
7      [SerializeField]
8      private ARPointCloudManager pointCloudManager;
9
10     private Dictionary<ulong, Vector3> points = new Dictionary<ulong,
Vector3>();
11
12     public Vector3[] Points
13     {
14         get
15         {
16             return new List<Vector3>(points.Values).ToArray();
17         }
18     }
19
20     private void Awake()
21     {
22         // はじめはスキャンを停止しておく
23         pointCloudManager.enabled = false;
24     }
25
26     private void OnDestroy()
27     {
28         // GameObjectが破棄されるときにARPointCloudManagerの更新イベントを解除する
29
30         pointCloudManager.trackablesChanged.RemoveListener(OnTrackablesChanged);
31     }
32
33     public void StartScan()
34     {
35         // スキャンを開始する
36         pointCloudManager.enabled = true;
37
38         // ARPointCloudManagerの更新イベントを登録する
39
40         pointCloudManager.trackablesChanged.AddListener(OnTrackablesChanged);
41     }
42
43     public void StopScan()
44     {
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99

```

```
43         // ARPointCloudManagerの更新イベントを解除する
44
45     pointCloudManager.trackablesChanged.RemoveListener(OnTrackablesChanged);
46
47     // スキャンを停止する
48     pointCloudManager.enabled = false;
49 }
50
51 private void
52 OnTrackablesChanged(ARTrackablesChangedEventArgs<ARPointCloud> args)
53 {
54     // 更新されたポイントクラウドの位置情報を取得する
55     foreach (var pointCloud in args.updated)
56     {
57         if(!pointCloud.positions.HasValue)
58         {
59             continue;
60         }
61
62         if(!pointCloud.identifiers.HasValue)
63         {
64             continue;
65         }
66
67         // ポイントクラウドの位置情報をDictionaryに追加する
68         for (int i = 0; i < pointCloud.positions.Value.Length; i++)
69         {
70             var id = pointCloud.identifiers.Value[i];
71             var position = pointCloud.positions.Value[i];
72
73             if (points.ContainsKey(id))
74             {
75                 points[id] = position;
76             }
77             else
78             {
79                 points.Add(id, position);
80             }
81         }
82     }
83 }
```