

# ISXEVE - Feature - New #1203

## Scan probe support

2014-10-01 06:16 PM - aj2k88@gmail.com

<b>Status:</b> New	<b>Start date:</b> 2014-10-01
<b>Priority:</b> Normal	<b>Due date:</b>
<b>Assignee:</b>	<b>% Done:</b> 0%
<b>Category:</b>	<b>Estimated time:</b> 0.00 hour
<b>Target version:</b>	
<b>vbulletin_issue_id:</b>	
<b>Description</b>	
Currently, the only method of interacting with anomalies is through scan probes.	
These are used by equipping the launcher, launching them in space, then "triangulating anomaly locations" by successive scans/repositionings.	
- The interface involves moving probes in XYZ space using standard 3d modeling controls, then telling them to warp to that location, selecting a scan range and then hitting "scan", followed by scan results.	
- Most of the complexity involves selecting the various parameters through the graphical interface. The actual commands are more likely to be :WarpTo(X,Y,Z) and Scan(scanRange) for each probe, and "return to bay" for all probes. I'm not sure how the results are represented, though.	

### History

#### #1 - 2023-11-22 07:14 PM - NostraThomas

Here's a video that shows exactly how this process works:

<https://www.youtube.com/watch?v=VRNoyPMBcKU>

Here's an psuedocode example script for logic on how a bot that utilizes this might work:

```
{
  ProbeScannerWindow:LaunchProbes

  ProbeScannerWindow:SetFormation[PinPoint]

  ProbeScannerWindow:Scan

  Wait 10 ;The probe scan takes a few seconds

  var results = ProbeScannerWindow:GetAnomalies

  foreach (var result in results)
  {
    if (result.Coordinates.X > 0) ; Or some other way to detect that the anomaly has been fully scanned down
    scannedResults.Add(result)
  }

  ProbeScannerWindow:RecallProbes
}
```

#### #2 - 2023-11-22 07:18 PM - NostraThomas

Much better logic example from Noobbotter:

1. Get current position of signature
2. Position probes around it using XYZ.
- 2.5. Reduce Probe distance if this is a repeat.

3. Run Scan.
4. Get scan resolution % of signature
5. if == 100, warp to. else goto 1.