

```

private string strToolName;
private void btnTouch_Click(object sender, EventArgs e)
{
    strToolName = "Touch";
}
private void axMapControl1_OnMouseDown(object sender, IMapControlEvents2_OnMouseDownEvent e)
{
    if (strToolName == "" || e.button != 1 || axMapControl1.LayerCount <= 0) return;
    //获取当前地图并清空
    IActiveView pActiveView = axMapControl1.ActiveView;
    IMap pMap = axMapControl1.Map;
    pMap.ClearSelection();

    //设置目标图层的(这里假设是第一个图层)
    IFeatureLayer pFeatureLayer = (IFeatureLayer)axMapControl1.get_Layer(0);

    //点选
    IPoint pPoint = pActiveView.ScreenDisplay.DisplayTransformation.ToMapPoint(e.x, e.y);
    ISpatialFilter pSpatialFilter = new SpatialFilterClass();
    pSpatialFilter.Geometry = pPoint;
    pSpatialFilter.SpatialRel = esriSpatialRelEnum.esriSpatialRelIntersects;
    IFeatureCursor pFeatureCursor = pFeatureLayer.Search(pSpatialFilter, false);
    IFeature pFeature = pFeatureCursor.NextFeature();

    if (pFeature != null)
    {
        IGeometry pGeometry = pFeature.Shape;
        IRelationalOperator pRelOperator = pGeometry as IRelationalOperator;

        //设置 Touch 要素的颜色
        IFeatureSelection pFeatureSelection = pFeatureLayer as IFeatureSelection;
        pFeatureSelection.SelectionColor = new RgbColorClass() { Red = 255 };

        //返回图层全部要素
        IFeatureCursor compareCursor = pFeatureLayer.Search(null, false);
        IFeature compareFeature = compareCursor.NextFeature();
        while (compareFeature != null)
        {
            IGeometry compareGeometry = compareFeature.Shape;
            switch (strToolName)
            {

```

```
        case "Touch":
        {
            //如果要素与点选要素边界相交,就加入地图选择集
            if (pRelOperator.Touches(compareGeometry))
            {
                pMap.SelectFeature(pFeatureLayer, compareFeature);
            }
            break;
        }
    }
    compareFeature = compareCursor.NextFeature();
}
//刷新地图
pActiveView.Refresh();
}
```