**3D Tiles样式和过滤**

本指南介绍如何使用CesiumJS来设置和过滤3DTiles，以高亮显示数据集的重要的要素。

**3D Tiles样式语言**

3D Tiles样式语言允许您定义数据集中的要素应如何显示的规则，例如显示的建筑颜色或建筑类型。这有助于识别城市中的模式或创建信息更丰富的可视化。

样式是用JSON和JavaScript表达式定义的。此外，样式语言提供了一组内置函数来支持常见的数学运算，如min、distance和log。

本指南包括3D Tiles样式语言的这些常用部分的示例：

* 显示和颜色
* 条件测试，包括特征属性测试、空检查和点云中点的分类检查
* 使用定义创建新变量
* 距离函数
* 创建新的颜色定义

**什么是 3D Tiles feature?**

Features可以是诸如单个建筑、门、窗、阀门或点云中的点。

**应用一个基础样式**

有两种常见的方法可以将样式应用于3D tileset：通过show和color视觉特性。一旦确定了样式规则是什么，就可以通过将3D tileset的style成员变量设置为新的Cesium3DTileStyle，将其应用到我们的特性中。

对于下面的代码片段，我们使用OSM建筑3DTiles名字为“Crown Entertainment Complex”的实体涂成红色，其他所有建筑都涂成白色。CesiumJS将有效地将此规则应用于每个feature，替换特性上的任何现有样式。我们选择的特性属性是name，它已经存在于数据集中。

osmBuildingsTileset.style **=** **new** Cesium.Cesium3DTileStyle({

color: {

conditions: [

["${name} === 'Crown Entertainment Complex'", "color('red')"],

["true", "color('white')"],

],

},

});

**显示属性**

除了着色之外，3D Tiles样式语言还允许您根据要素的元数据属性来显示和隐藏要素。

假设我们想显示所有住宅建筑，这些建筑在OSM建筑中被分类为“residential”或“apartments”。

osmBuildingsTileset.style **=** **new** Cesium.Cesium3DTileStyle({

show: "${feature['building']} === 'residential' || ${feature['building']} === 'apartments'",

});

**条件控制颜色**

也可以使用条件来确定要为要素指定的颜色。

例如，您可能需要在墨尔本皇冠娱乐中心周围的区域为游客设计一条巴士路线。在离综合大楼一定距离内有哪些建筑物？

此代码片段根据与皇冠娱乐中心的距离为每个建筑指定颜色：

osmBuildingsTileset.style **=** **new** Cesium.Cesium3DTileStyle({

defines: {

distanceFromComplex:

"distance(vec2(${feature['cesium#longitude']}, ${feature['cesium#latitude']}), vec2(144.96007, -37.82249))",

},

color: {

conditions: [

["${distanceFromComplex} > 0.010", "color('#d65c5c')"],

["${distanceFromComplex} > 0.006", "color('#f58971')"],

["${distanceFromComplex} > 0.002", "color('#f5af71')"],

["${distanceFromComplex} > 0.0001", "color('#f5ec71')"],

["true", "color('#ffffff')"],

],

},

});

* 3D Tiles样式设置条件允许您定义阵列中的颜色等功能。
* 条件表现为if语句。

在这种情况下，如果建筑物距离皇冠娱乐中心大于0.0001度但小于0.002度，则其颜色为橙色（0.0001度=11.1 m）。

* 定义表达式创建新变量。

在这里，distanceFromComplex是一个新的变量，定义为建筑物的经纬度属性与建筑综合体的经纬度属性之间的关系。

* 距离是3DTiles样式语言的内置功能。
* 支持CSS样式颜色定义。例如，十六进制代码color（'#ffffff'）声明了一个新的颜色，与color（'white'）等价。

**点云样式**

点云是遵循上述相同样式规则的点的集合，具有一些附加选项。

**点云样式选项和特性**

除了color和show视觉特性外，点云分片还支持pointSize。默认pointSize为1.0。

在Cesium ion的LAS文件中，点云包含两个每点属性：Classification分类和Intensity.强度。

* Classification分类是指向点的数据类型（如地面、低植被或建筑物）的整数映射。
* Intensity强度是点表面的反射率，是0到255之间的整数。

点云样式也可以应用于其他点特性，例如位置、颜色和法线。例如，您可以使用POSITION\_ABSOLUTE属性，根据点云与中心点的距离来创建点云的热图，类似于我们根据OSM建筑与皇冠娱乐综合体之间的距离对其进行着色（请参见Sandcastle热图）。

**点云样式示例**

假设你想想象一下加拿大蒙特利尔生物圈博物馆周围的环境。如果**Classification**分类属性是植被，则样式规则是将点着色为绿色。

pointCloudTileset.style **=** **new** Cesium.Cesium3DTileStyle({

color: {

conditions: [

["${Classification} === 2", "color('brown')"], *// ground*

["${Classification} === 3", "color('greenyellow')"], *// low vegetation*

["${Classification} === 4", "color('green')"], *// medium vegetation*

["${Classification} === 5", "color('darkgreen')"], *// high vegetation*

["true", "color('white')"]

]

}

});

我们可以直观地看到博物馆周围有多少是植被。

**处理未定义属性**

有时建筑物有缺失的属性。在这种情况下，我们可以添加空检查条件：

Copy to clipboard.Data copied clipboard.

osmBuildingsTileset.style **=** **new** Cesium.Cesium3DTileStyle({

color: {

conditions: [

["${material} === null", "color('white')"],

["${material} === 'glass'", "color('skyblue', 0.5)"],

["${material} === 'concrete'", "color('grey')"],

["${material} === 'brick'", "color('indianred')"],

["true", "color('white')"],

],

},

});

}

在本例中，如果任何建筑不具有material属性，则其颜色为白色。

**使用样式编辑器**

您还可以使用Cesium Stories中的样式编辑器设置3D Tiles的样式。我们上面介绍的相同样式化方法可以在编辑器中应用。

**学习资源**

Sandcastle示例,包括：

* [3D Tiles Point Cloud Styling](https://sandcastle.cesium.com/#c=dVTbjts2EP0VwghgCVBp6i4lziKOu0WMul1j7TQvAgKaorNsaNIgKW3dhf+9pC5OnSB6sDzDmXNm5gw1m20kEwYsuWxqsD+DjZIHCDbUUAUWnFMRgKXkzXHPMPgoWEuVZuYMHuVeGkY0WOM9BFuChWDiiwNYcPo3FrWSYKckIZIzYE3wBzaUgSWTBCvOKKxEixVoGX22RG+BoM9gSTVrjvCvzudNSWcupTCYCaqmAXipBLCPLU1Zl620ZbXNf) — 点云热力图
* [Cesium OSM Buildings Styling](https://sandcastle.cesium.com/index.html#c=rVdtb9pIEP4rK1QpRiJ+JzY0zV2TVG2kNq0CulN19MNiT8yq9i5aryFclf9+s34hdiDlUPsF7PXOPM/MPLMvlkXekhgyQcQ9YVyBpJFiKyDeNZmyFHKSq03KeDLjlkUm1TNJKU8KmgC5FlGRAVdUMcHHZKHUMh9bVsLUopibkcisK8hZkb2fWF58qrRDS0kAK6M5Yln5EiJ2z6LS3pq0oC4LlsYaK6aKkkgUUkG+0Sxvv16Rz0vg1/rDUkhF0) — OSM Buildings样式设置.