Geohashes

Geohashes 是一 将 度坐 （ lat/lon ） 成字符串的方式。 做的初衷只是 了 地理位置在 url 上呈 的形式更加友好，但 在 geohashes 已 成一 在数据 中有效索引地理坐 点和地理形状的方式。

Geohashes 把整个世界分 32 个 元的格子 —— 4 行 8 列 —— 一个格子都用一个字母或者数字 。比如 g 个 元覆 了半个格林 ， 的全部和大不列 的大部分。 一个 元 可以 一 被分解成新的 32 个 元， 些 元又可以 被分解成 32 个更小的 元，不断重 下去。 gc 个 元覆 了 和英格 ， gcp 覆 了 敦的大部分和部分南英格 ， gcpuuz94k 是白金 的入口，精 到 5 米。

句 ， geohash 的 度越 ，它的精度就越高。如果 个 geohashes 有一个共同的前 —<code>gcpuuz</code>—就表示他 挨得很近。共同的前 越 ，距 就越近。

也意味着， 个 好相 的位置，可能会有完全不同的 geohash 。比如， 敦 Millenium Dome 的 geohash 是 u10hbp，因 它落在了 u 个 元里，而 挨着它 的最大的 元是 g 。

地理坐 点可以自 索引相 的 geohashes ，更重要的是，他 也可以索引所有的 geohashes 前 。如索引白金 入口位置—— 度 <code>51.501568</code> ， 度 <code>-0.141257</code>—将会索引下面表格中列出的所有 geohashes ，表格中也 出了各个 geohash 元的近似尺寸：

Geohash	Level	Dimensions
g	1	~ 5,004km x 5,004km
gc	2	~ 1,251km x 625km
gcp	3	~ 156km x 156km
gcpu	4	~ 39km x 19.5km
gcpuu	5	~ 4.9km x 4.9km
gcpuuz	6	~ 1.2km x 0.61km
gcpuuz9	7	~ 152.8m x 152.8m
gcpuuz94	8	~ 38.2m x 19.1m
gcpuuz94k	9	~ 4.78m x 4.78m
gcpuuz94kk	10	~ 1.19m x 0.60m
gcpuuz94kkp	11	~ 14.9cm x 14.9cm
gcpuuz94kkp5	12	~ 3.7cm x 1.8cm

{ref}/query-dsl-geohash-cell-query.html[geohash 元 器] 可以使用 些 geohash 前 来 出与指定坐 点（ lat/lon ）相 的位置。

Geohashes 映射

首先，　需要决定使用什　　的精度。　然　也可以使用　12　　的精度来索引所有的地理坐　点，但是　真的需要精　到数厘米　？如果　把精度控制在一个　　一些的　，比如　　1km　　，那　　可以省大量的索引空　：

```
PUT /attractions
{
  "mappings": {
    "restaurant": {
      "properties": {
        "name": {
          "type": "string"
        },
        "location": {
          "type":                "geo_point",
          "geohash_prefix":      true, ①
          "geohash_precision":   "1km" ②
        }
      }
    }
  }
}
```

① 将 geohash_prefix　　true 来告　Elasticsearch 使用指定精度来索引 geohash 的前　。

② 精度可以是一个具体的数字，代表的 geohash 的　度，也可以是一个距　。1km 的精度　　的 geohash 的　度是 7 。

通　如上　置，geohash 前　中 1 到 7 的部分将被索引，所能提供的精度大　在 150 米。

Geohash　元

geohash_cell　　做的事情非常　：　把　度坐　位置根据指定精度　　成一个　geohash　，然后　所有包含　个 geohash 的位置——　是非常高效的　。

```
GET /attractions/restaurant/_search
{
  "query": {
    "constant_score": {
      "filter": {
        "geohash_cell": {
          "location": {
            "lat":  40.718,
            "lon": -73.983
          },
          "precision": "2km" ①
        }
      }
    }
  }
}
```

① precision 字段　置的精度不能高于映射　geohash_precision 字段指定的　。

此　将　<code>lat/lon</code>　坐　点　成　度的　geohash　——　本例中<code>dr5rsk</code>—然后　所有包含　个短　的位置。

然而，如上例中的写法可能不会返回　2km　内所有的餐　。要知道　geohash　上　是个矩形，而指定的点可能位于　个矩形中的任何位置。有可能　个点　好落在了　geohash　元的　附近，但　器会排除那些落在相　元的餐　。

了修　个　，我　可以通　置 neighbors参数　true，　把周　的　元也包含　来：

```
GET /attractions/restaurant/_search
{
  "query": {
    "constant_score": {
      "filter": {
        "geohash_cell": {
          "location": {
            "lat":  40.718,
            "lon": -73.983
          },
          "neighbors": true, ①
          "precision": "2km"
        }
      }
    }
  }
}
```

① 此　将会　的 geohash 和包　它的 geohashes 。

明　的，　2km　精度的　geohash　加上周　的　元，最　致一个　大的搜索区域。此　不是

精度而生，但是它非常有效率，而且可以作 更高精度的地理位置 器的前置 器。

TIP	将 `precision` 参数 置 一个距 可能会有 性。`2km` 的 `precision` 会被 成 度 6 的 geohash 。 上它的尺寸是 1.2km x 0.6km。 可能会 明 的 置 度 `5` 或 `6` 会更容易理解。

此 的 一个 点是，相比 `geo_bounding_box` ，它支持一个字段中有多个坐 位置的情况。 我 在 [optimize-bounding-box] 中 ， 置 `lat_lon` 也是一个很有效的方式，但是它只在 个字段只有 个坐 点的情况下有效。