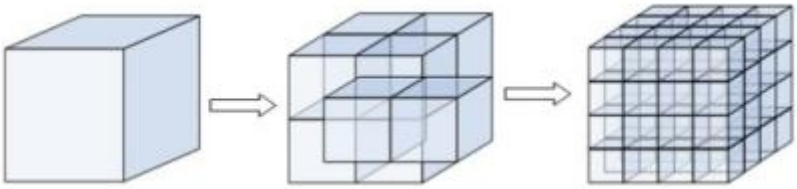
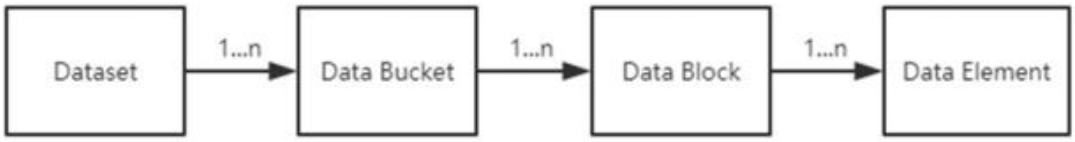
气象网格数据的激增为气象系统带来了宝贵的见解，但也为数据存储和管理系统提出了挑战。因此，基于 HBase 的专用存储系统已经开发出来，以支持分布式方法中的大量数据。现有基于 HBase 优化的气象网格数据存储管理系统，实现了一个分布式气象数据存储系统。其大致流程如下：

1. 构建空间网格并编码：使用气象数据构建空间网格并将其编码，具体包括以下两个步骤
   a) 空间网格经纬度定义：北纬为正，南纬为负，纬度范围为（-90，90）；东经为正，西经为负，经度范围为（-180，180）；
   b) 构建空间网格：使用经纬度将地球划分为东西半球并投影成平面，在投影平面的基础上加上高度，构建空间网格，其中高度为地球半径。即东西半球分别构建一个立方体（空间网格），立方体的底面为半球经纬度的投影面，立方体的高则是从地表往大气方向延伸的长度（地球半径）。构建了初始的两个立方体之后，使用空间八叉树将立方体迭代划分，划分程度自己设定，若划分网格数过多则查询效率变低，反之则每个网格包含的数据多大，超过 Hbase 单个值的存储上限；



   c) 编码空间网格：以前三层为例：
      i. 层级 0：仅有一个初始网格。其编码为"00"。
      ii. 层级 1：将地球划分为东西两个半球，西半球的编码为"10"，东半球的编码为"11"。
      iii. 层级 2：在层级 1 的基础上进行八叉树划分。从层级 2 开始，每个网格的编码基于上一层级并增加三位二进制数，分别表示纬度、经度和高度。其中，纬度、经度和高度较大的部分用 1 表示，较小的部分用 0 表示。

2. 创建两种类型的 Hbase 表，系统模型根据时间序列将数据划分为数据桶，再将数据桶中的数据划分为数据块。因此有两个表，分别为数据桶表和数据块表。



   具体情况如下：
   a) 表的行键（Rowkey）生成
      i. 数据桶表：利用 Hbase 多版本的特性，行键存储数据桶 ID，该 ID 从 0 开始依次递增，时间戳列存储数据桶中数据所属时间戳。查询时不加时间版本，默认查询最新时间戳的数据。
      ii. 数据块表：行键为所属数据桶 id 加数据块编码加数据块级别。
   b) 表的设计：如下图所示，其中数据桶表中的 BlockNum 存储了所划分的数据块数量，BlockCode 存储了数据桶的网格编码，BlockLevel 则存储了数据桶的网格级别；而数据块表中的 BlockCode 存储了该块的编码，BlockLevel 存储

了该块的级别，ElementInfo 存储了该数据块中包含的数据元素（即气象数据）。

Table Name	Rowkey	Timestamp	Column Name	Unit
DataBucket	R1	T1	BlockNum	
			BlockCode	
			BlockLevel	String
DataBlock	R2	T2	BlockCode	
			BlockLevel	String
			ElementInfo	

c) 数据查询：给定一个时间范围 T(t1,t2)和一个位置坐标范围 S(s1,s2)，根据时间范围查询数据桶表，获得一个 BucketList，然后循环 BucketList，根据位置范围和每个 bucket 的 BlockLevel 生成一个 CodeList（位置编码集合）结束循环后，根据 CodeList 查询数据块表，获得结果数据块 BlockList。

要求：

1. 根据上面的描述对 2019 年中国 1km 分辨率逐月平均气温数据集进行空间网格划分和编码，划分程度自行设定，数据集单位为 0.1℃。
2. 创建数据桶表和数据块表，将经过空间网格划分和编码的数据存储到表中。
3. 给定时间范围 T（3，5）（以月为单位，包括第三个月和第五个月），根据 T 查询数据桶表，输出每个数据桶的 BlockCode。
4. 给定时间范围 T（3，5），位置范围 S（(90，30，0),(91，31，10)），括号内分别为经度、纬度、高度，高度单位为千米，查询该范围的数据块元素。