

Hadoop命令路径配置



什么是BC范式，它和第三范式有什么关系？



BC范式（Boyce-Codd Normal Form，BCNF）

BC范式是关系数据库设计中的一种更为严格的规范化标准。它是在 **第三范式（3NF）** 的基础上进一步强化的，要求**每个决定因素都是候选键**。

BC范式的定义

一个关系满足 **BC范式**，需要同时满足以下条件：

1. **满足第三范式（3NF）**：表中不存在非主属性对主键的传递依赖。
2. **每个函数依赖中的决定因素（determinant）必须是候选键**：
 - 决定因素：在函数依赖中，用来决定其他属性值的属性或属性组合。
 - 候选键：能够唯一标识表中每一行数据的属性或属性组合。

BCNF 与 3NF 的区别

虽然 3NF 和 BCNF 非常相似，但 BCNF 更严格。**3NF 关注的是消除非主属性之间的传递依赖，而 BCNF 关注的是候选键是否决定所有其他属性。**

区别的核心点：

- **3NF 的放宽条件**：允许主属性通过非候选键的决定因素产生传递依赖。
- **BCNF 的严格条件**：任何决定因素都必须是候选键，无论是主属性还是非主属性。

举例：3NF 满足但 BCNF 不满足的情况

未满足 BCNF 的表

考虑一个大学的课程表 `Course`：

| CourseID | Professor | Department |
|----------|-----------|------------|
| CS101 | Dr. Smith | CS |
| CS102 | Dr. Brown | CS |
| MATH101 | Dr. Smith | Math |

主键：

- `CourseID` 是主键。

问题：

1. 存在函数依赖：`Professor → Department`。
 - 一个教授只能属于一个特定的系。
2. 决定因素 `Professor` 不是候选键，因为 `Professor` 不能唯一标识一行记录。

转换为满足 BCNF 的表

为了满足 BCNF，将表分解为两张表：

1. **Professor_Department** 表（教授和系之间的关系）：

| Professor | Department |
|-----------|------------|
| Dr. Smith | CS |
| Dr. Brown | CS |

2. **Course** 表（课程和教授之间的关系）：

| CourseID | Professor |
|----------|-----------|
| CS101 | Dr. Smith |
| CS102 | Dr. Brown |
| MATH101 | Dr. Smith |

什么时候需要 BCNF？

1. 如果表满足 3NF，但仍然存在主属性的部分依赖或传递依赖，需要考虑使用 BCNF。
2. 在有多个候选键且候选键之间存在依赖关系的情况下，3NF 可能不足以消除所有异常。

总结：BCNF 和 3NF 的关系

1. **包含关系**：BCNF 是 3NF 的严格形式。满足 BCNF 的表必然满足 3NF，但满足 3NF 的表不一定满足 BCNF。
2. **设计原则**：
 - **3NF** 主要用于消除非主属性的依赖问题，适用于大多数数据库设计。
 - **BCNF** 在候选键之间存在依赖时更适用，确保所有决定因素都是真正的候选键。
3. **选择依据**：实际设计中，如果 3NF 已经足以避免冗余和更新异常，不需要强制使用 BCNF。但如果遇到多候选键复杂依赖关系，BCNF 更合适。

示意图比较

- **3NF**：主键 → 非主属性；传递依赖消除。
- **BCNF**：候选键 → 非候选键；所有决定因素必须是候选键。