

# 关于将已存在时间序列的前缀序列创建为时间序列的设计

## 一.背景

目前，在IoTDB系统中若将已存在时间序列的前缀序列创建为时间序列，则会创建失败。

例子：

- (1) 通过create语句创建时间序列：（语句执行顺序如下）

```
create timeseries root.a.b.c.d with datatype=float
```

```
create timeseries root.a.b.c with datatype=float
```

则时间序列root.a.b.c将创建失败

- (2) 通过insert语句创建时间序列：（语句执行顺序如下）

```
insert into root.a.b.c(timestamp, d) values(1, 1.0)
```

```
insert into root.a.b(timestamp, c) values(1, 1.0)
```

则往时间序列root.a.b.c中插入数据失败（因为时间序列root.a.b.c不存在，需要先创建root.a.b.c，而创建该时间序列将会失败）

## 二.设计

### 1.背景

目前系统中表示路径节点的类型有三种，分别是MNode, StorageGroupMNode, MeasurementMNode

StorageGroupMNode, MeasurementMNode 继承自 MNode 分别表示 存储节点 测量节点，剩余的节点（包括root节点）由MNode直接实现。

### 2.修改思路

2.1修改后将支持通过以下方式将已存在时间序列的前缀路径修改为时间序列：

前提：时间序列root.a.b.c.d已存在

方式1：

```
create timeseries root.a.b.c with datatype=float
```

方式2：

```
insert into root.a.b(timestamp, c) values(100, 1.0)
```

### 2.2

1.方式1创建时间序列的过程：

(1) 首先，先判断c是否已经是MeasurementMNode，若是，则抛出 PathAlreadyExistException 异常；（新增）

(2) 若不是，则创建c为MeasurementMNode，且创建完后将 root.a.b.c.d 时间序列中的 c（MNode类型）替换为c（MeasurementMNode）；（新增）

(3) 将该创建时间序列的plan序列化到meta log中，作为系统重启后恢复的信息。（该功能系统中已有，不用修改）

2.方式2创建时间序列的过程:

(1) 首先，先判断c是否已经是 MeasurementMNode ，若是，则直接返回；

(2) 若不是，则创建c为MeasurementMNode ，且创建完后将root.a.b.c.d时间序列中的 c（MNode类型），替换为c（MeasurementMNode）；（新增）

(3) 将替换的信息写入meta log中，作为系统重启后恢复的信息（新增）；

(4) 基于步骤(2)，需要新增 AlterTimeSeriesBasicInfoPlan 方法，其主要作用为将修改信息序列化入磁盘以及从磁盘中反序列化信息，恢复元数据在内存中的信息。

### 3.测试

(1) 测试是否能通过以上描述的两种方式创建时间序列；

(2) 创建完对应时间序列后，重启系统，看是否创建的序列还存在，且系统未出现异常。