

# 第35期-分库分表与连接字符串

2020年9月27日 9:14

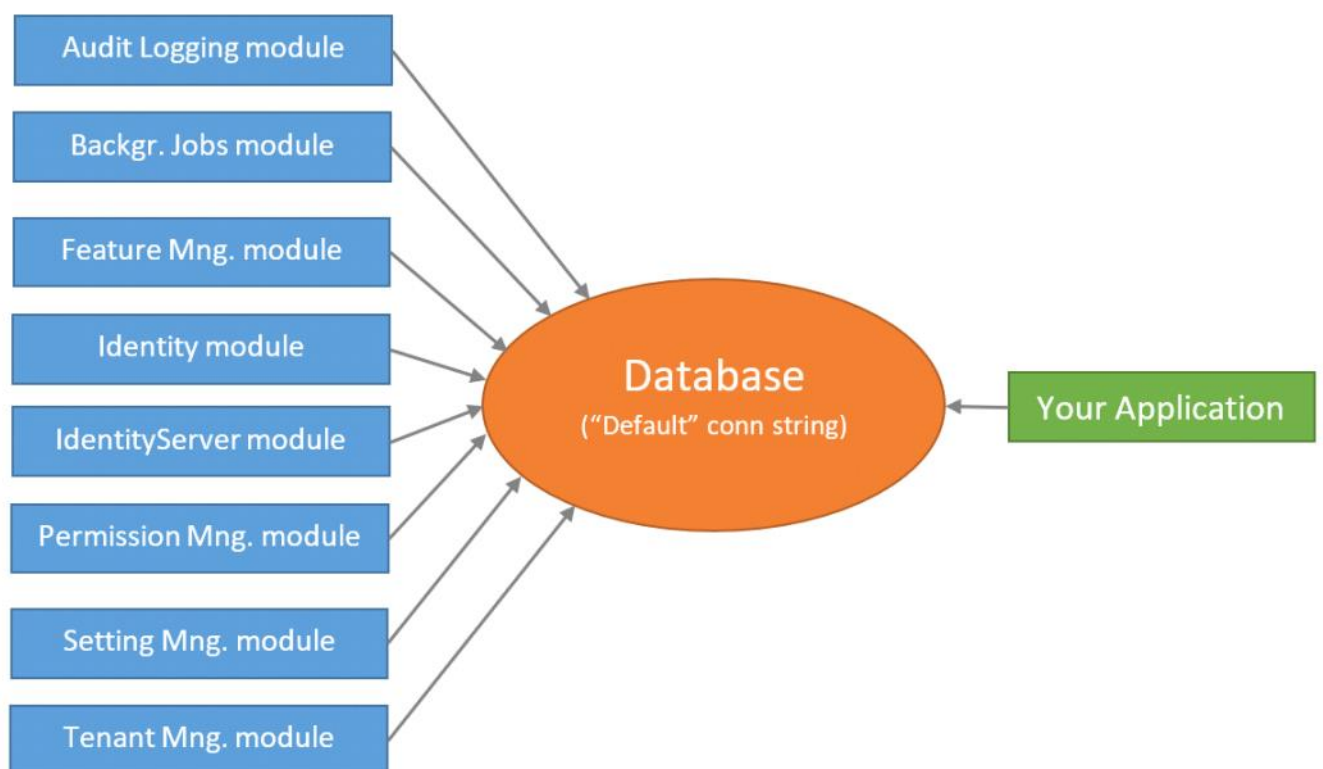
## 数据访问提供程序支持

ABP框架被设计为与数据库无关，借助仓储与工作单元进行抽象，可以处理任何类型的数据源，当前实现了以下提供程序：

- Entity Framework Core (DBMS and providers)
- MongoDB
- Dapper

## 连接字符串

为了灵活支持模块化、微服务和多租户的数据库分表分库部署方案，连接字符串应该根据不同租户不同模块甚至不同的微服务动态获取。



## 分离方案

- 不同模块使用不同的数据库
- 不同的租户使用不同的数据库

## 混合方案

- 不同的模块可以共享一个数据库（A模块和B模块使用数据库1，C和D模块使用数据库2，E模块单独使用数据库3，其它模块全用数据库4）
- 不同的租户可以共享数据库，类似于模块机制。
- 每个租户每个模块分离数据库，这会导致数据库太多，难以维护，但 ABP 支持。

## 配置连接字符串

ABP使用该IConfiguration服务来获取应用程序配置,虽然是将配置写入appsettings.json文件的最简单方法，但它不仅限于此文件。您可以使用环境变量，用户机密，Azure密钥保管库等

```
"ConnectionStrings": {
  "Default": "Server=localhost;Database=MyMainDb;Trusted_Connection=True;",
  "AbpIdentityServer":
"Server=localhost;Database=MyIdsDb;Trusted_Connection=True;",
  "AbpPermissionManagement": "Server=localhost;Database=MyPermissionDb;Trusted_Connection=True;"
}
```

```
public override void ConfigureServices(ServiceConfigurationContext context)
{
    Configure<AbpDbConnectionOptions>(options =>
    {
        options.ConnectionStrings.Default = "...";
        options.ConnectionStrings["AbpPermissionManagement"] = "...";
    });
}
```

**public static class <ModuleName>DbProperties**

AbpAuditLoggingDbProperties  
 FeatureManagementDbProperties  
 AbpPermissionManagementDbProperties  
 AbpSettingManagementDbProperties  
 AbpTenantManagementDbProperties

## 设置连接字符串名称

```
[ConnectionStringName(ConnectionString.DefaultConnectionStringName)]
public class HelloWorldDbContext : AbpDbContext<HelloWorldDbContext>
```

## 连接字符串解析器

具有两个预构建的实现：

## **ConnectionStringResolver->DefaultConnectionStringResolver->MultiTenantConnectionStringResolver**

- **DefaultConnectionStringResolver**: 使用 **AbpDbConnectionOptions** 来根据上面“配置连接字符串”部分中定义的规则选择连接字符串。
- **MultiTenantConnectionStringResolver**: 用于多租户应用程序, 并尝试获取当前租户的已配置连接字符串 (如果有)。它使用 **ITenantStore** 来查找连接字符串。如果没有为当前租户指定连接字符串, 则使用默认字符串。

如果需要自定义逻辑来确定连接字符串, 请实现 **ConnectionStringResolver** 接口 (可选地从现有实现派生) 并使用依赖项注入系统替换现有实现。