

# 第七课：系统扩展点

Smartbi 系统扩展点包括服务端和客户端两种类型，主要用于通过扩展点的配置，方便的把新功能无缝集成到系统中，无需修改 Smartbi 产品源代码。

## 1、服务端

### 1.1 CatalogTreeModule

#### 1.1.1 registerCascadeDeleteDAO

##### 扩展点基本信息

- 原型: `public void registerCascadeDeleteDAO(String type, AbstractDAO dao)`
- 描述: 主要在增强Smartbi原有资源时使用，注册相应的DAO。在注册后，当资源目录树删除指定类型的资源时，除了执行原有的删除操作外还会调用dao中的 `deleteByResourceId` 方法删除相关资源。
- 参数:
  - type: 参照CatalogElementType;
  - dao: 继承AbstractDAO基类;
- 实例: `catalogTreeModule.registerCascadeDeleteDAO("BASETABLE", CMBDAOFactory.getCopyBookBeanDAO());`

##### 应用示例

- 暂无。

#### 1.1.2 registerDAO

##### 扩展点基本信息


- 原型: `public void registerDAO(String type, AbstractDAO dao)`
- 描述: 主要用于添加Smartbi资源类型时使用。注册相应的DAO。在注册后当资源目录树删除此种类型节点的时候将会调用dao中的delete方法删除相关资源。
- 参数:
  - type: 参照CatalogElementType;
  - dao: 继承AbstractDAO基类;
- 实例: `catalogTreeModule.registerDAO("TXTDATASOURCE", CMBDAOFactory.getTxtDataSourceDAO());`

##### 应用示例

- 暂无。

#### 1.1.3 registerNonChildElementType

##### 扩展点基本信息

- 原型: `public void registerNonChildElementType(String type)`
- 描述: 默认情况下新增加的资源类型在树上显示时都会显示 ，表示该节点可以展开。可以通过该方法注册某类型不会出现子节点。
- 参数:
  - type: 参照CatalogElementType;
- 实例: `catalogTreeModule.registerNonChildElementType("TXTQUERYOBJECT");`

##### 应用示例

- 暂无。

#### 1.1.4 registerFolderType

##### 扩展点基本信息

- 原型: `public void registerFolderType(String type)`
- 描述: Smartbi中有目录和文件二个类型，该方法可以把某类型注册为目录类型。
- 参数:
  - type: 参照CatalogElementType;
- 实例: `catalogTreeModule.registerFolderType("TXTDATASOURCE");`

##### 应用示例

- 暂无。

##### 文档目录:

- 1、服务端
  - 1.1 CatalogTreeModule
    - 1.1.1 registerCascadeDeleteDAO
    - 1.1.2 registerDAO
    - 1.1.3 registerNonChildElementType
    - 1.1.4 registerFolderType
    - 1.1.5 registerExporter
- 2、客户端
  - 2.1 资源类型
  - 2.2 系统选项

### 1.1.5 registerExporter

#### 扩展点基本信息

- 原型: `public void registerExporter(Importer importer, Exporter exporter)`
- 描述: Smartbi提供了资源的导入、导出功能, 允许在不同的服务器上进行迁移, 当扩展插件增加了资源类型后, 则可以添加导入、导出的处理类。
- 参数:
  - importer: 继承`bof.catalogtree.Importer`;
  - exporter: 继承`bof.catalogtree.Exporter`;
- 实例: `catalogTreeModule.registerExporter(importer, exporter);`

#### 应用示例

- 暂无。

## 2、客户端

### 2.1 资源类型

通过扩展插件添加新的资源类型后, 我们需要把该资源的类型信息在`catalogElementTypes`中进行声明。

扩展点	描述
<code>catalogElementTypes</code>	系统资源类型

#### 参考示例

- [资源类型示例](#)。

### 2.2 系统选项

通过扩展插件可以给系统选项添加自定义选项。

扩展点	描述
<code>SystemConfig.configitem</code>	管理 → 系统管理 → 系统选项

#### 参考示例

- [系统选项示例](#)。