

扩展包开发指引

- [1 扩展包概述](#)
- [2 适用范围](#)
- [3 学习指南](#)
- [4 快速访问](#)

1 扩展包概述

随着客户对 BI 功能的要求越来越深入，Smartbi 的很多应用项目都有定制化需求。为满足这类需求，产品进一步增强“稳定内核+API扩展”的架构，支持扩展包扩展。每个项目的定制化的需求都可以做成单独的扩展包，这样，最大限度地控制项目化的需求对产品的影响，控制不同项目之间相互影响，项目需求响应的及时性也能得到保证。

扩展包从应用角度上看，它是一个可以加载到 Smartbi 中执行的功能插件；从技术角度上看，它是一个web项目，包含Html、Jsp、Java、Js和资源等文件。

2 适用范围

本文面向开发商以及外部开发人员，他们需要对 Smartbi 产品的默认功能进行扩展、修改或删除。

3 学习指南

1、请先查看 [扩展包简介](#)，了解扩展包是什么，扩展包加载原理、如何做。

2、接下来看 [扩展包开发环境部署](#)，快速搭建扩展包开发环境。

3、接下来看 [开发扩展包](#)，尝试创建一个HelloWorld程序。

4、看完入门文档，搭建了开发环境后，建议看下[Smartbi前端框架介绍](#)，了解Smartbi的前端组件框架、通信框架、关键组件，及拿到需求该如何入手，再看下[入门示例](#)，了解简单的扩展包示例。

5、[常用知识点](#)，里面有各种各样在工作中可能遇到的功能点示例，看完这个心里会有个案例库。

4 快速访问

您可以使用下表快速访问 Smartbi 扩展包开发指南的各个部分。

分类	主题	简要说明
扩展包基础	扩展包简介	介绍扩展包的概念、特点、应用场景等。
	扩展包开发环境部署	部署扩展包开发环境有两种方式，一种是使用插件开发IDE，其中包含了插件开发所需要的开发工具Eclipse，JDK，Tomcat，Smartbi的war包，适合于新手；另一种是手动配置插件开发环境，适合于已有WEB开发环境用户（推荐使用手动配置的方式）。
	开发扩展包	如何创建、部署、调试并打包一个扩展包。
常用知识点	替换Smartbi文件	Smartbi插件可以方便的替换Smartbi原有的文件，包括图片、样式、HTML、JSP和JS等通过URL直接访问的文件资源。
	修改Smartbi JS文件	在扩展包开发过程中经常遇到要修改Smartbi原有的JavaScript文件的情况，如重写JS的某个函数，以达到定制化执行逻辑的目的。
	新增系统选项	通过扩展包开发，可以自由添加“系统选项”的配置项。
	自定义对话框	在Smartbi中，有较多场景会用到对话框。本文介绍在Smartbi中创建自定义对话框的方法。
	自定义Module	Smartbi前后端交互的后端入口。
	知识库升级	扩展包中需要往知识库中添加库表、修改库表或自动插入初始化数据时，需要使用知识库升级（upgrade升级类）处理相关逻辑。

	创建知识库对象	扩展包中如果往知识库中添加了库表，可以同时创建对应的知识库对象，以便使用Smartbi所包含的Hibernate机制操作数据库。
	自动导入资源	扩展包中需要在系统启动过程中自动导入已经开发好的报表、数据源等Smartbi资源时，可以使用自动导入资源（postupgrade升级类）处理。
	调用服务端Java接口	当第三方系统需要远程调用Smartbi接口时，可使用Smartbi的服务端Java接口。 服务器端Java接口通过Java API提供七项服务接口。
	Java查询介绍	扩展包中提供给客户实现自定义数据结构的一种扩展方法，可以通过Java类将任意一个含有数据的文件或报表通过解析后，在Smartbi中展现分析。
通用技能	扩展包开发规范	扩展包最终是作为产品代码的一部分运行的，所以对质量同样需要有高度的要求。本文所列举的规范需严格遵守，否则可能造成严重事故。
	调试&定位技巧	定制开发，首先需要找到切入点。本文主要介绍常用的调试&定位技巧，以便快速定位到产品功能对应的代码，以作为定制切入点。
附录	扩展包项目介绍	扩展包项目的完整结构，了解扩展包的结构有利于开发者更加顺利的进行开发。
	部署扩展包	通过smartbi部署扩展包的操作步骤。
	扩展包开发入门示例	一个简单的扩展包开发示例。
	Smartbi前端框架介绍	Smartbi 是典型的基于JavaScript的面向对象框架，整个系统只有几个入口.jsp（譬如index.jsp、login.jsp），剩下的业务或操作逻辑会按需动态加载或注销组件。本文主要介绍Smartbi 系统前端框架、前后端通信框架、以及关键组件。
☆☆☆	☆☆☆☆☆ ☆☆☆☆☆☆	