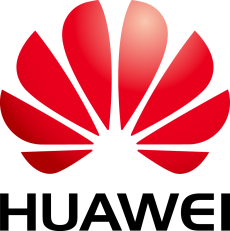
|  |  |  |
| --- | --- | --- |
| 华为技术有限公司 | 产品名称 | 密级 |
| NetworkOS | 内部公开 |
| 产品版本 | 共44页 |

OPS2.0 开放性增强Eclipse编程插件操作手册

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 拟制: | 黄少鹏 |  | 日期： | 2013-7-27 |
| 审核: |  |  | 日期： |  |
| 审核: |  |  | 日期： |  |
| 批准: |  |  | 日期： |  |



华为技术有限公司

版权所有 侵权必究

修订记录

| 日期 | 修订 版本 | CR号 | 修改 章节 | 修改描述 | 作者 |
| --- | --- | --- | --- | --- | --- |
| 2013-7-27 | 1.00 |  |  | 初稿完成 | 黄少鹏WX168991 |
| 2014-4-26 | 2.00 |  |  |  | 张鹏宇ZWX199308 |
|  |  |  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目 录

[OPS2.0 开放性增强Eclipse编程插件操作手册 1](#_Toc396916737)

[1 总体说明 5](#_Toc396916738)

[1.1 软件要求 5](#_Toc396916739)

[1.2 范围 5](#_Toc396916740)

[1.3 发布目录结构 6](#_Toc396916741)

[2 安装向导 7](#_Toc396916742)

[2.1 Eclipse安装要求 7](#_Toc396916743)

[2.2 安装OPSDEV for Python 7](#_Toc396916744)

[2.3 安装OPSDEV-Python 14](#_Toc396916745)

[2.4 导入OPSDEV-Python Release 16](#_Toc396916746)

[3 使用指南 20](#_Toc396916747)

[3.1 操作OpsDev-Python工程 20](#_Toc396916748)

[3.1.1 新增工程 20](#_Toc396916749)

[3.1.2修改OpsDev-Python工程属性 25](#_Toc396916750)

[3.2 配置数据源 28](#_Toc396916751)

[3.2.1 添加数据源 28](#_Toc396916752)

[3.2.2 配置数据源操作 30](#_Toc396916753)

[3.2.3 修改数据源 33](#_Toc396916754)

[3.2.4 测试数据源 33](#_Toc396916755)

[3.2.5 删除数据源信息 34](#_Toc396916756)

[3.3 操作OpsDev-Python Restful Api 35](#_Toc396916757)

[3.3.1 使用向导新建Api 35](#_Toc396916758)

[3.3.2 绑定数据源 37](#_Toc396916759)

[3.3.3 调试OpsDev-Python Restful Api 38](#_Toc396916760)

[3.3.4 在代码中进行动态生成body（add post body）操作 38](#_Toc396916761)

[3.4 编写例程 43](#_Toc396916762)

[3.4.1 启动OPS2.0 43](#_Toc396916763)

[3.4.2增加OPS2.0设备 44](#_Toc396916764)

[3.4.3运行与调试 44](#_Toc396916765)

[3.5 操作网络监视器 45](#_Toc396916766)

[3.5.1 打开网络监视器 45](#_Toc396916767)

[3.5.2 开启网络监视器 46](#_Toc396916768)

[3.5.3 监视内容 46](#_Toc396916769)

[3.5.4 删除监视内容 48](#_Toc396916770)

# 总体说明

## 软件要求

软件要求

|  |  |
| --- | --- |
| 软件名称 | 软件版本 |
| Microsoft Windows | XP SP2，SER2003及其他兼容 |
| Eclipse | Eclipse-sdk-3.7.1版本 |
| JRE | 1.6.0\_11-b03 |
| Python | Python2.7.3版本 |
| OpsDev | OpsDev for Eclipse 1.0.0.2014.52314574710 版本 |
| PyDev | PyDev 2.7.5 版本 OpsDEV-Python工程依赖于PyDev |

## 范围

VRP OPS2.0 开放性增强Eclipse编程插件, 提供给华为VRP网络业务编程者的编辑器环境。主要功能:

可视化的IDE编程环境，支持python编程环境。

进行OPS数据源管理

主要完成对数据源的添加，修改，删除，配置相关数据源。

进行数据接口加载。

对数据接口进行配置加载。

提供网络监视器：

提供对OPS中涉及网络访问的监视功能。

Python IDE辅助编程提示：

主要完成对python编辑，可视，调试，编译的功能的相关提示。

由ops规格schema文件，自动生成restfulApi文档。

主要完成接口代码的生成功能。

提供对api路径编程提示功能。

## 发布目录结构

OpsDev插件发布目录结构：

--

|---- com.huawei.networkos.ops\_1.0.0.2014060416140410.jar

|---- com.huawei.networkos.ops.core\_1.0.0.2014060416140410.jar

(OpsDev)

|---- com.huawei.networkos.ops.python\_1.0.0.2014060416140410.jar

(OpsDev\_Python)

# 安装向导

## Eclipse安装要求

下载安装Eclipse开发环境 ，推荐版本3.7.1， 可选择大于3.7.1版本以上。

## 安装OPSDEV for Python

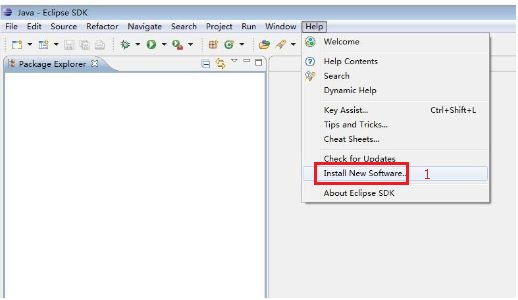
将PyDev 2.7.5.zip解压到eclipse的plugins目录中。

安装OPS2.0，打开OPS2.0.exe文件根据提示点击下一步即可，直至安装完成。

安装完成后，在生成目录plugins文件夹中找到com.huawei.networkos.ops.feature-1.0.0.2014042218392610.zip。

打开eclipse开发环境，开始安装插件。

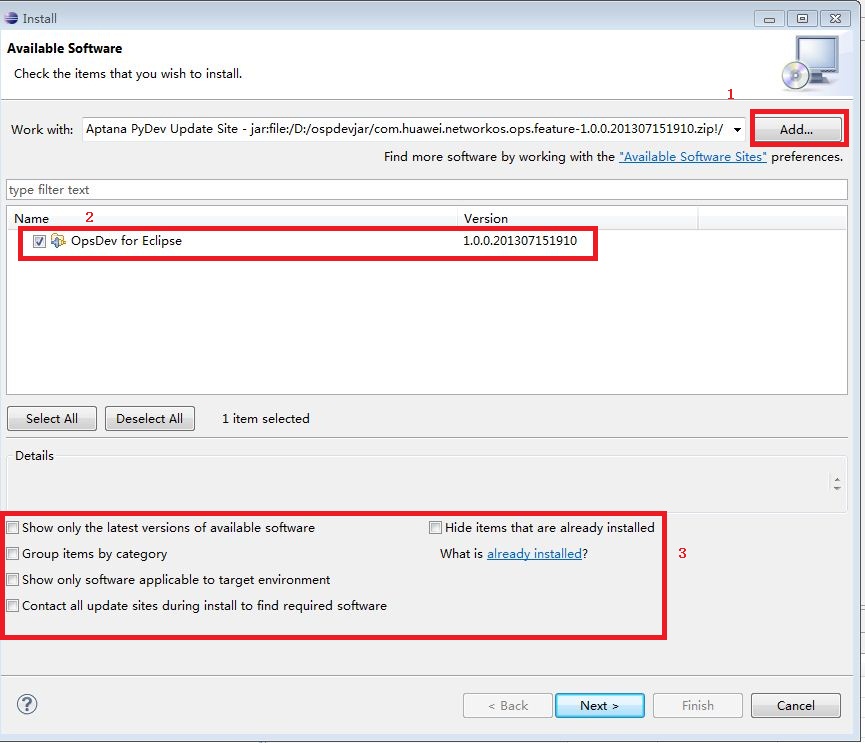
新增OpsDev插件示意图



如图2-1所示，菜单栏点击Help后选择Install New Software，如图2-1所示界面。

点击Install New Software后出现如图2-2所示界面。

增加OpsDev插件示意图



增加OpsDev插件步骤如下：

点击Add按钮，将com.huawei.networkos.ops.feature-1.0.0.2014042218392610.zip包加入到Work with路径框里。如图2-2所示红色框1。

钩选OpsDev for Eclipse，如图2-2所示红色框2。

如图2-2所示红色框3中的五个选项均不钩选。最后点击Next进入下一步。

图2-2中红色框3中的内容解释如下：

Show only the latest versions of available software

显示最终版本。

Group items by category

插件项目类别总和。

Show only software applicable to target environment

显示可用的软件。

Contact all update sites during install to find required software

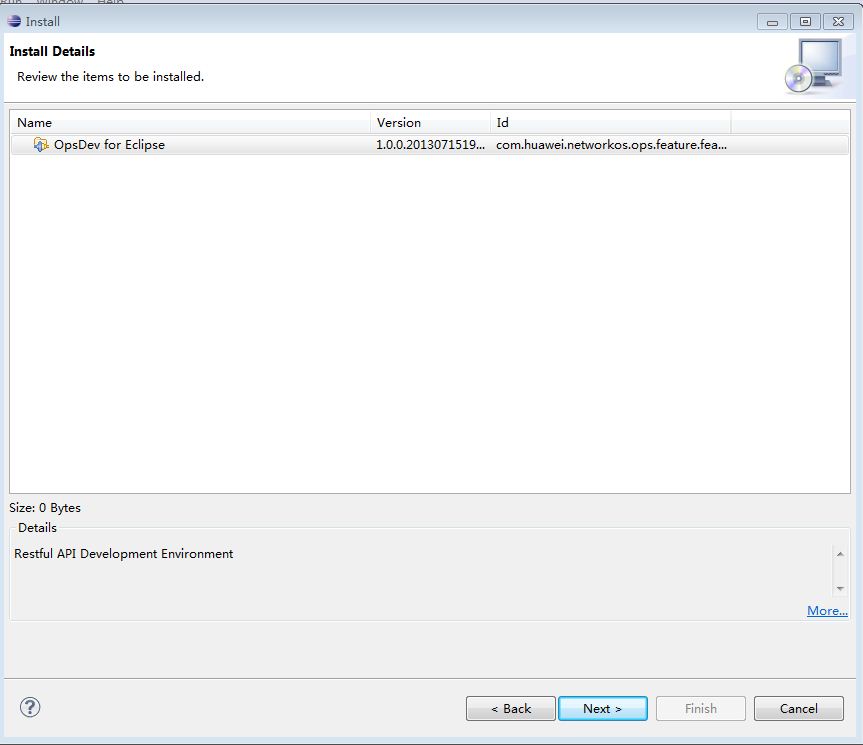
安装更新软件。

Hide items that are already installed

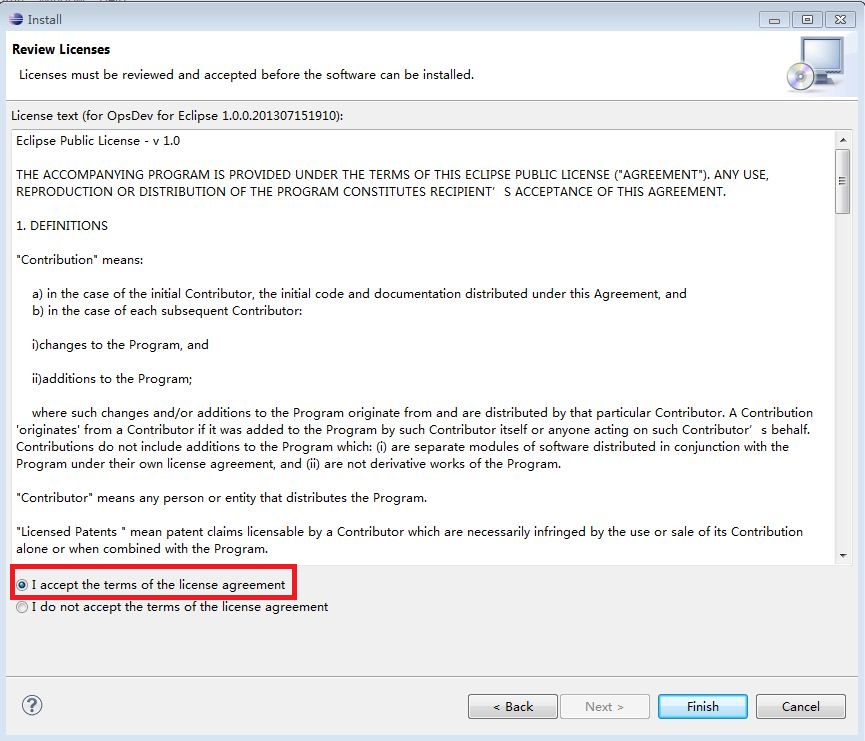
隐藏项目插件已安装。

进入安装信息界面。如图2-3所示，点击Next进入下一步。

OpsDev安装版本信息示意图



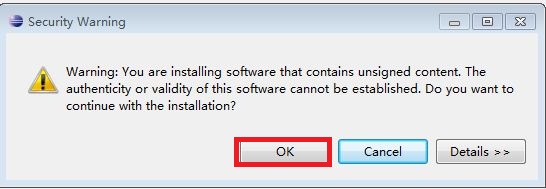
安装许可示意图



如图2-4所示，选择红色框，同意许可，点击Finish完成。

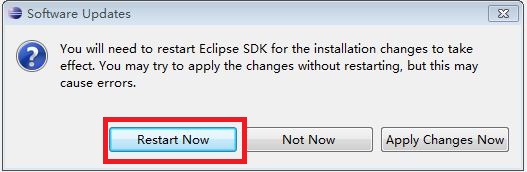
点击完成后，出现警告框，如图2-5所示，点击OK键，继续。

安全完成后告警示意图



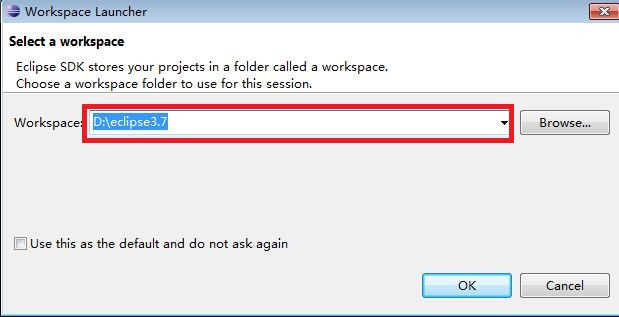
安装完成后，提示Eclipse需要重启。如图2-6所示，选择立即重启。

重启提示示意图



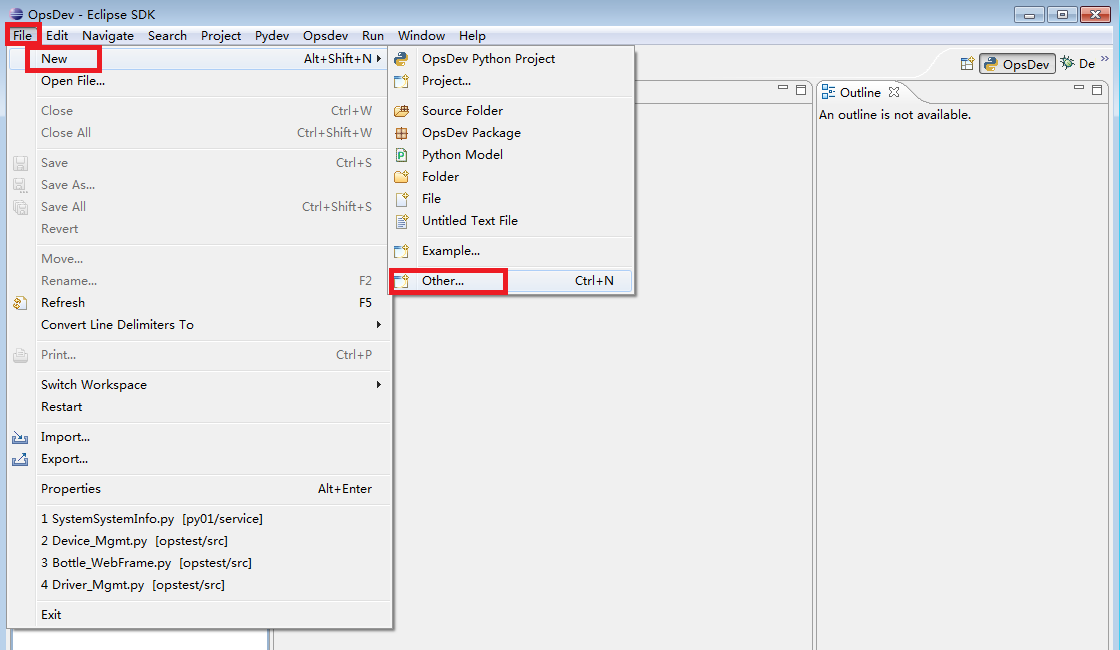
重新启动Eclipse后，如图2-7所示，选择工作路径。

选择工作路径示意图



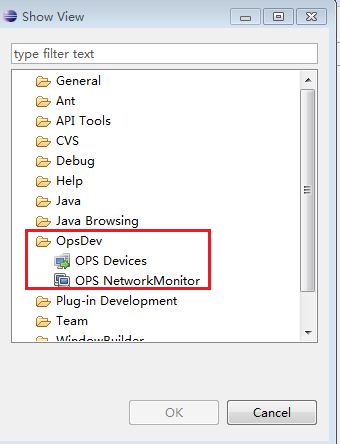
打开Eclipse，在菜单栏中选择File—>New—>Other进行插件安装，如图2-8所示。

OpsDev插件安装页面示意图



当页面出现了OpsDev选项，说明安装成功，如图2-9所示。

OpsDev界面示意图



## 安装OPSDEV-Python

说明：

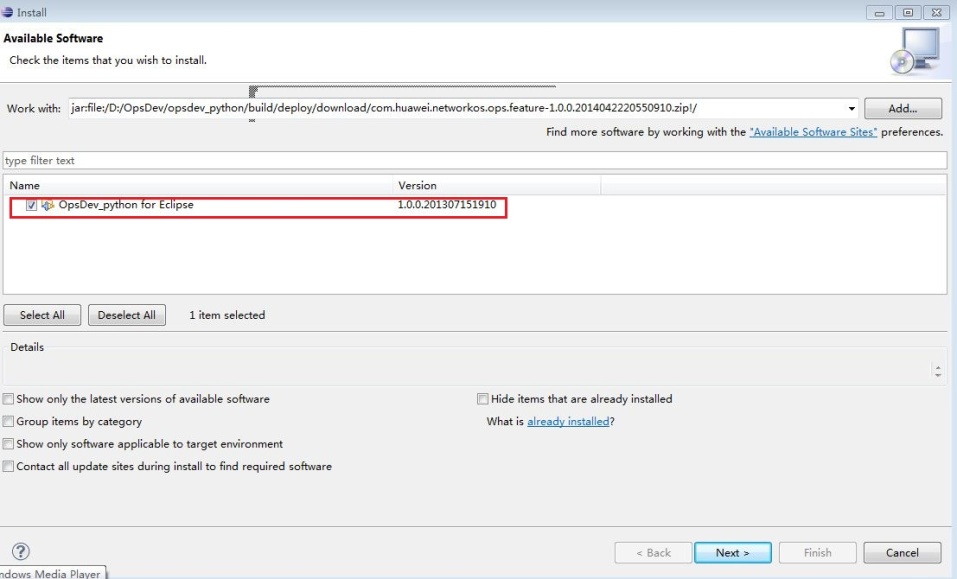
1.OpsDev-python的使用依赖于OPSDEV的安装。

2.OpsDev-python安装步骤和OPSDEV的安装步骤一致，在这不做重复说明。

3.OpsDev-python安装，以下需要注意的几点。

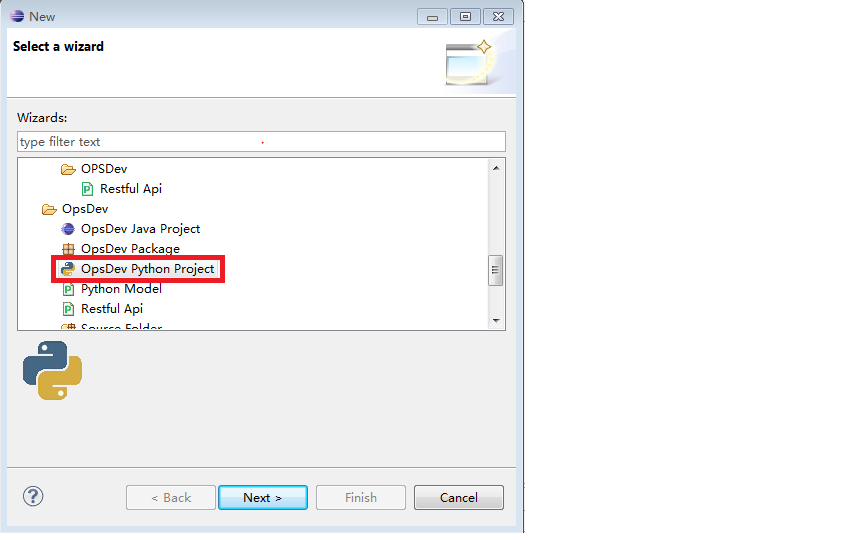
插件安装如图2-10所示。

OpsDev-Python安装示意图



如图2-11所示，OpsDev-Python安装成功。

OpsDev-Python安装成功示意图



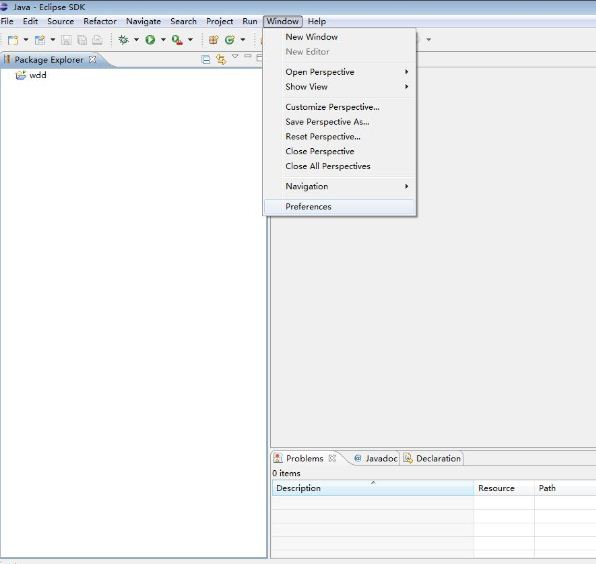
## 导入OPSDEV-Python Release

导入Python lib 包：因安装Python解释器，需导入相关Python lib包，以保证Python正常运行。可以通过两种方式进行安装：1）通过eclipse配置. 2) 在新建工程时，提示需要导入python lib包。

* + - 1. 通过eclipse配置

点击Window->Preferences，如所示。

OpsDev-pythonRelease安装示意图

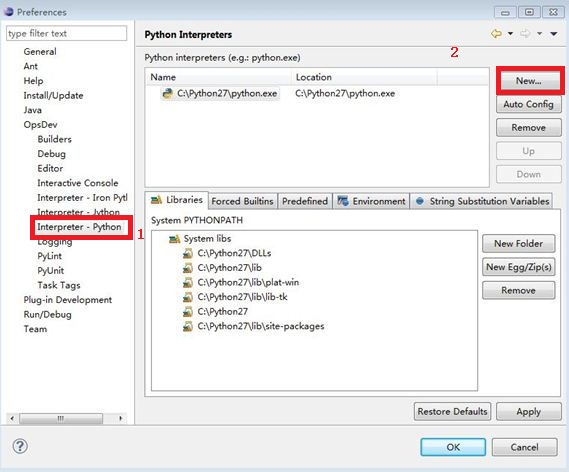


进入Preferences界面后，如图2-13所示进行Python.lib包导入，具体步骤如下。

选择OpsDev选项，点击Python。

点击new按键，选择之前安装的Python解释器，然后点击ok键即可。

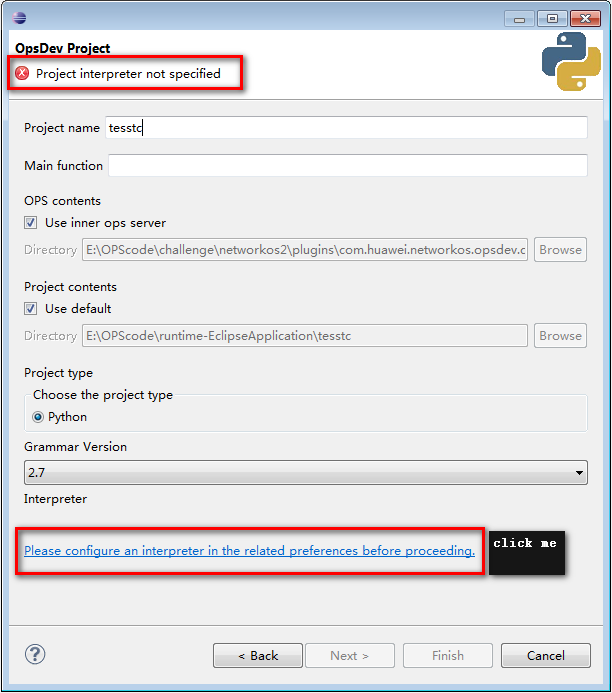
Python架包安装示意图



* + - 1. 在新建工程时，提示需要导入python lib包

新建工程时，如果出现Project interpreter not specified，如图2-14所示，请点击please configure an interoreter in the related preferences before proceeding， 进行python解析器的加载。

处理新建工程时导入python lib示意图

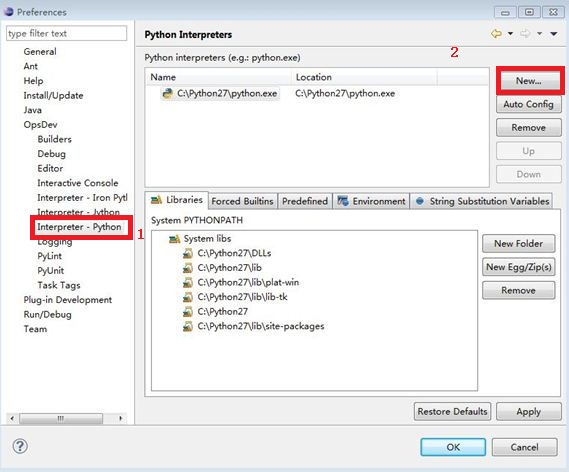


Python.lib包导入，如图2-15所示，具体步骤如下。

选择OpsDev选项，点击Python（红框1）。

点击new按键，选择之前安装的Python解释器，最后点击ok键即可。

Python架包安装示意图



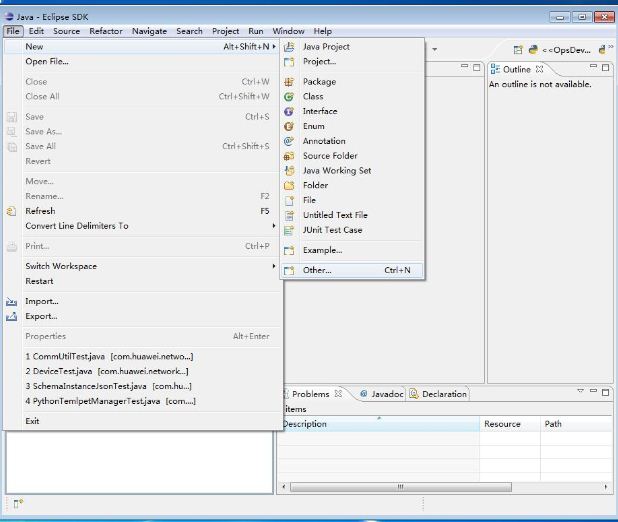
# 使用指南

## 操作OpsDev-Python工程

### 新增工程

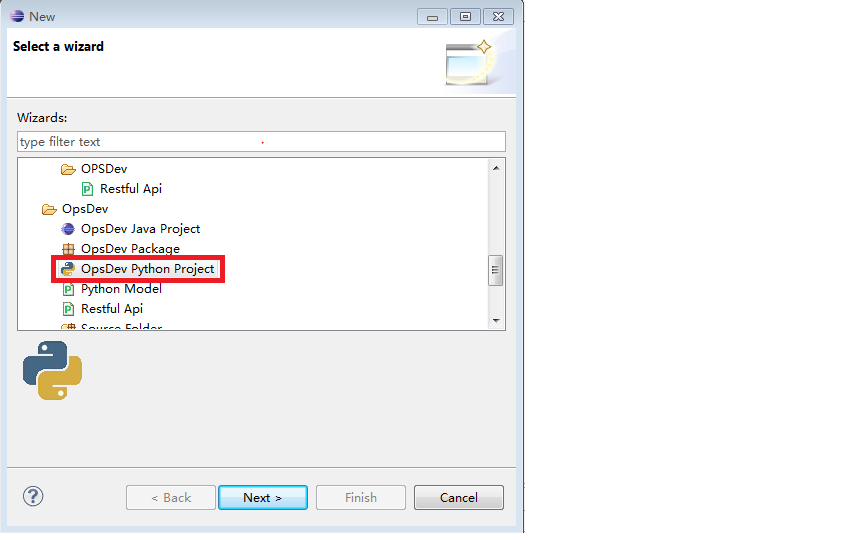
新增一个工程，选择OpsDev\_python Project。具体操作为打开Eclipse，如图3-1所示，选择File-->New-->Other。

创建OpsDev-python Project示意图



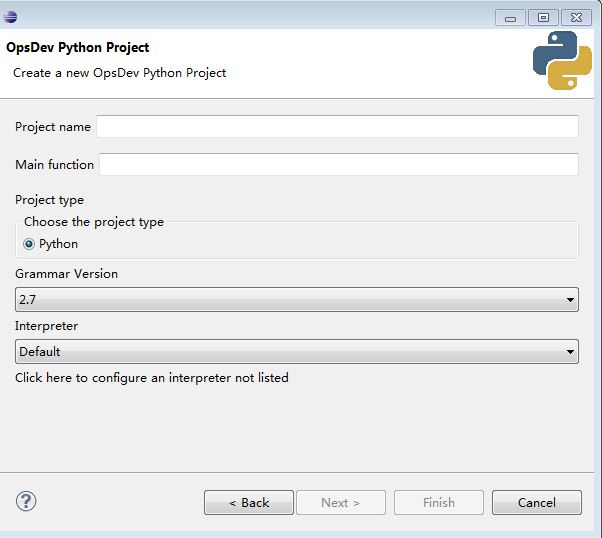
点击Other后，选择OpsDev-python Project，点击Next，如图3-2所示。

OpsDev-python Project选项示意图



设置OpsDev-Python Project项目，按需要进行以下设置，如图3-3所示。

新建OpsDev-python项目示意图



Project Name: 工程名称。

Main function ：设定主函数的名称, 空为默认main函数。

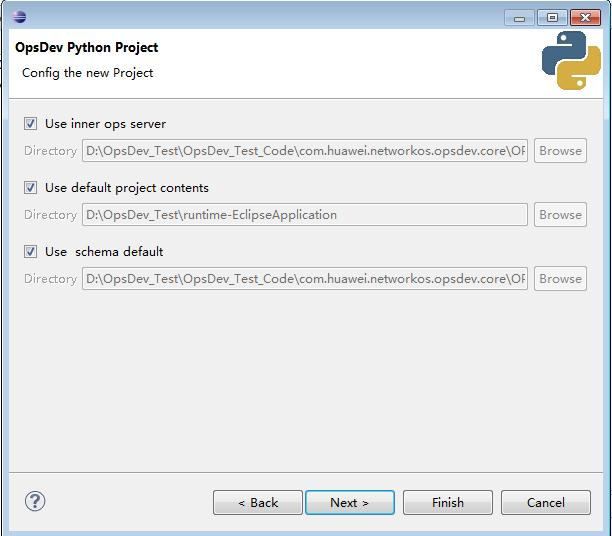
Project Type: 工程类型为Python。

Grammar Version:版本为2.7。

Interpreter：编辑器为默认版本。

设置OpsDev-Python工程路径，如图3-4所示。

设置OpsDev-Python工程路径示意图



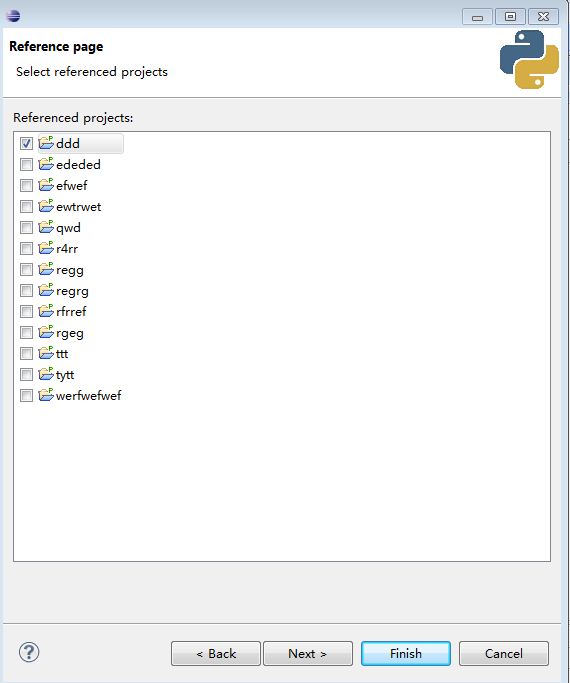
Use inner ops server: OpsDev服务器路径。

Use default project contents：OpsDev-Python 工程路径。

Use schema defalut：Schema路径。

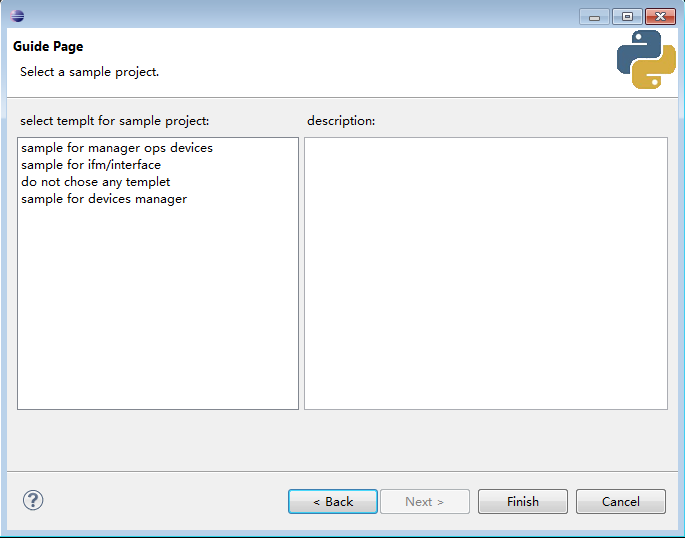
选择当前项目所关联工程, 当关联工程时，其它工程中代码与lib，都可以在本工程中访问到 。点Next进入下一步，如图3-5所示。

选择关联项目目录示意图



向导页面，选择一个工程所带的示例模板。通过选择的工程有对应的描述，点击Finish完成工程，如图3-6所示。

向导页面示意图



生成OpsDev-Python工程路径，如图3-7所示。

OpsDev-Python目录结构示意图

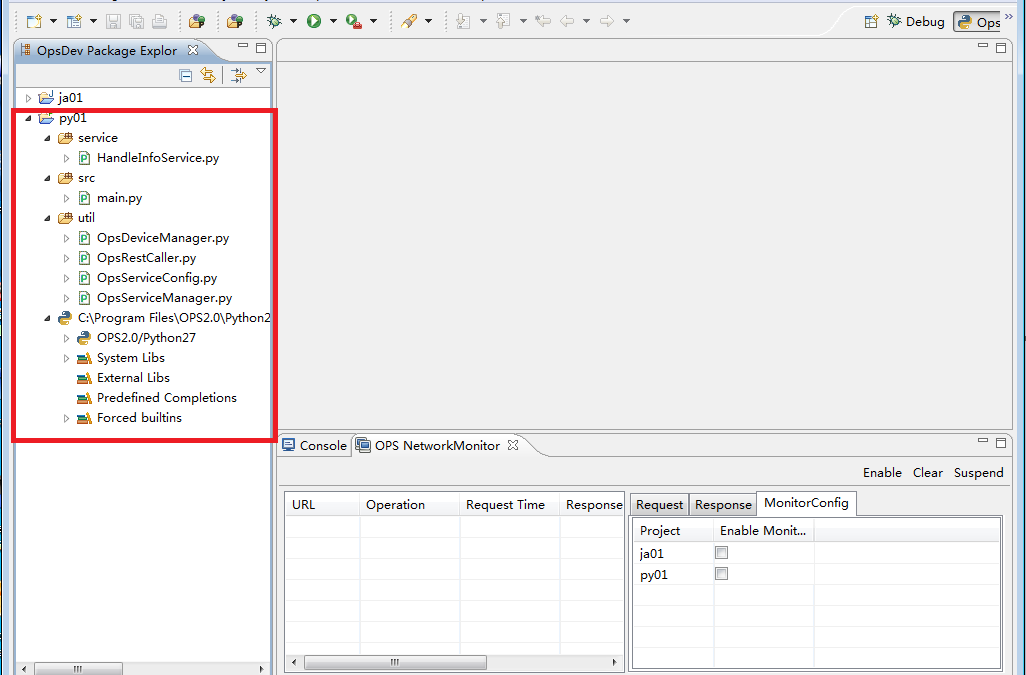


图3-7为新建的工程。目录结构由工程主目录py01，service,src和util三个文件夹组成。

HandleInfoService.py: 配置OPS数据源相关信息。

main.py：main.python文件。

Ifmterface.py：工程自动生成的示例模板。

OpsRestCaller.py: 访问OPS客户端类。

OpsServiceConfig.py: 配置OPS连接信息。

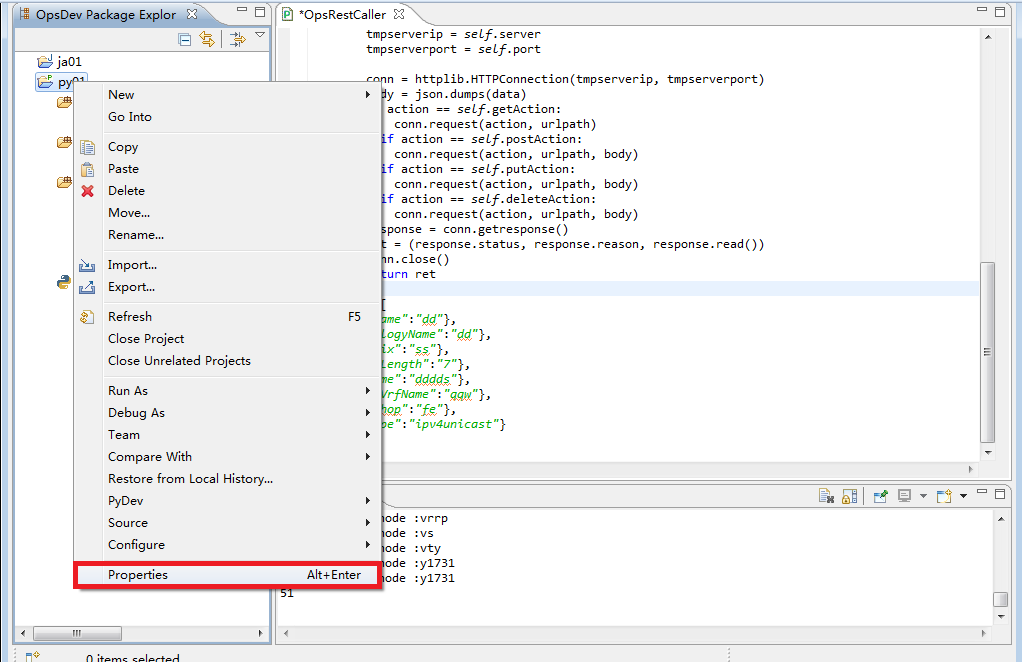
OpsServiceManager.py: OPS访问相关服务类。

C:\Python27\python.exe：python架包。

### 修改OpsDev-Python工程属性

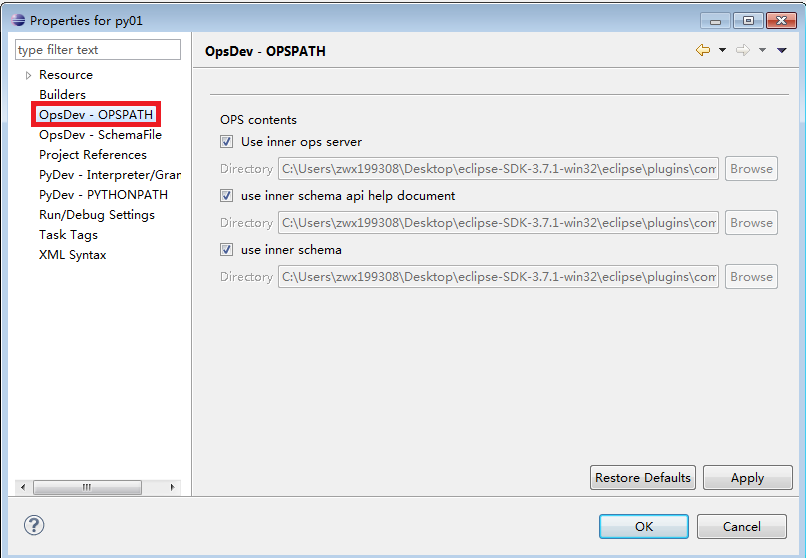
在新建的python工程上点击右击，选择Properties，如图3-8所示。

Python选择Properties示意图



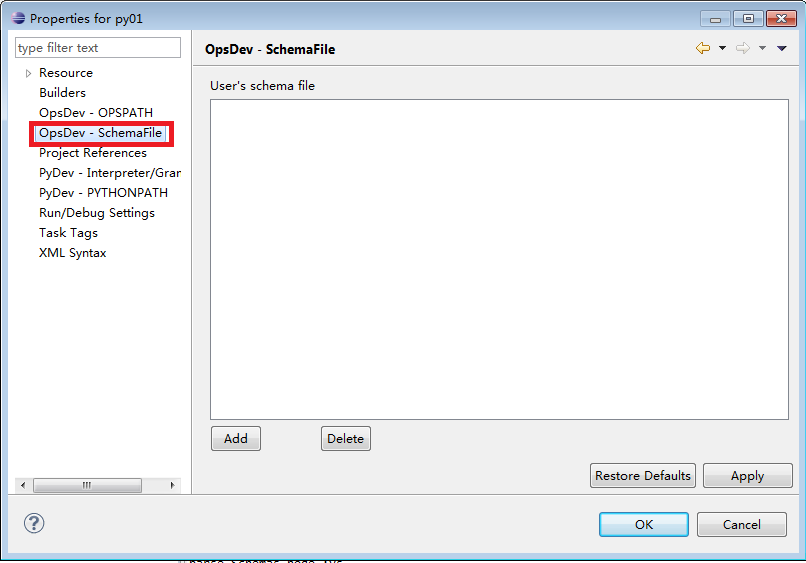
改变工程的配置路径，如图3-9所示。

改变工程配置路径示意图



可以添加、删除外部user schema文件，并可以通过Restful Api接口调用，如图3-10所示。

添加、删除外部user schema文件示意图



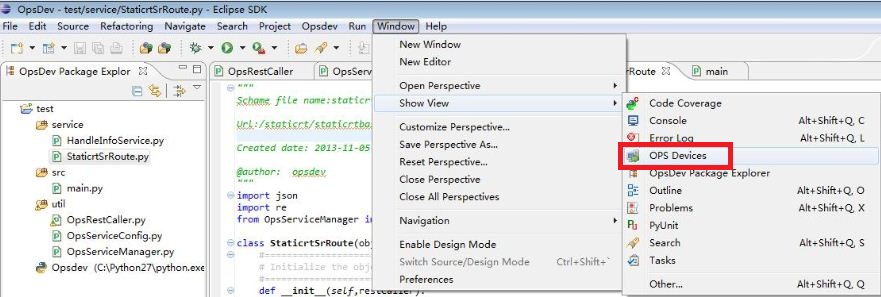
## 配置数据源

数据源是工具与OPS连接的抽象， 每一个数据源，可以认为是ops2.0中的一台设备。而数据源中的操作对象，认为是对当前设备进行的操作，如get 、post、put、delete 操作。

### 添加数据源

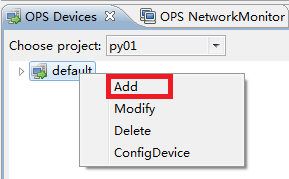
打开数据源窗口，选择Window-->Show View-->OPS Devices，如图3-11所示。

添加数据源示意图



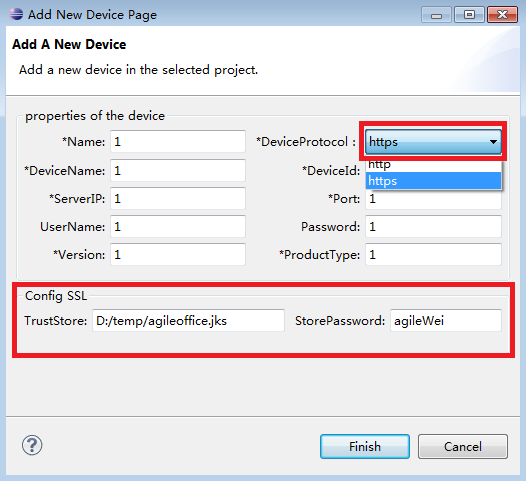
选择OPS Devices后，出现默认数据源，右键点击default-->Add，增加新数据源，如图3-12所示。

增加新数据源示意图



选择Add后出现新数据源相关信息，如图3-13所示。

新数据源信息示意图



Name:数据源名称。

DeviceProtocal:设备所用的协议。

DeviceName:设备名称。

DeviceId: 设备ID。

ServerIP：服务器IP地址。

Port：端口号。

UserName:用户名。

Password：密码。

Version：版本号。

ProductType：产品号。

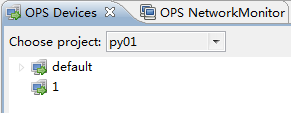
TrustStone:当协议为https时证书的路径。

StorePassword:当协议为https时的密码。

注意：加\*选项为必填项，只有当DeviceProtocal选择为https时下方的Config SSL一栏才会显示，否则不会显示。

点击Finish后完成新数据源相关信息配置。Default下方为选项新增数据源，如图3-14所示。

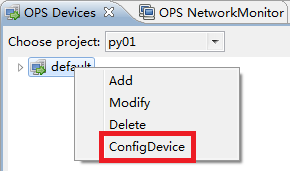
数据元提交后示意图



### 配置数据源操作

右键点击数据源出现选择项，点击ConfigDevice选项，进入配置数据源操作界面，如所示。

进入配置数据源操作界面示意图



点击ConfigDevice后出现配置数据源信息界面，如图3-16所示。

数据源信息界面示意图

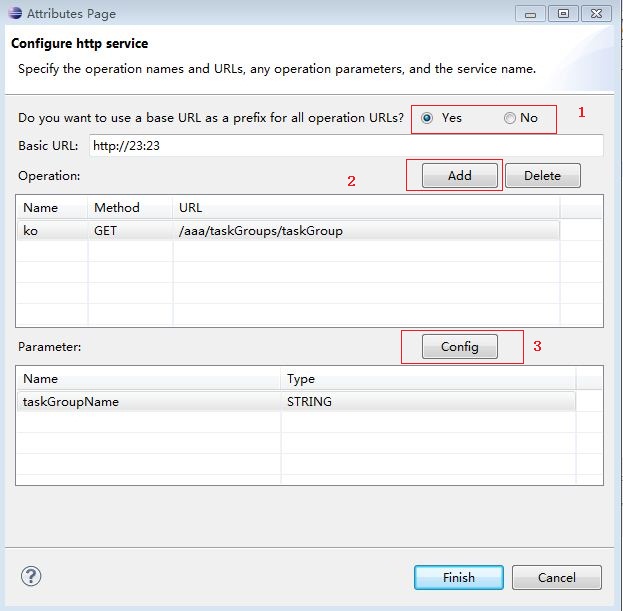


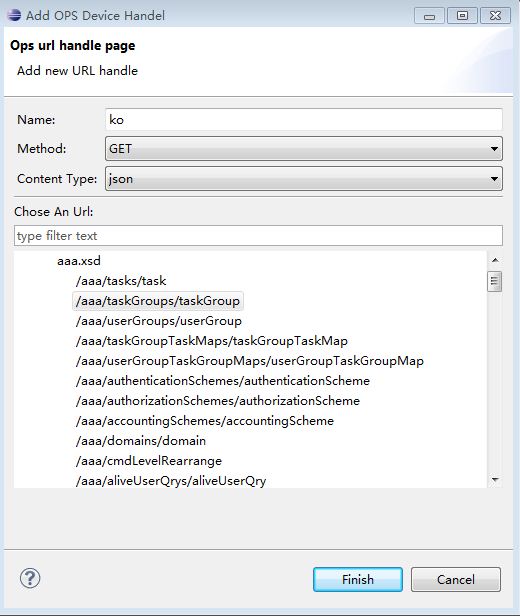
图3-16中红框1为基本URL选择默认选项。

图3-16中红框2为Add选择项为配置操作项。

图3-16中红框3为操作项相关属性。

点击图3-16中Add选项后出现操作项配置信息，如图3-17所示。

数据源配置操作项示意图



Name:操作项名称。

Method:操作项方法。

Content Type:文本显示格式。Json和XML供用户预览。

Chose An Url: url选择项。

点击图3-16中的Config按钮，出现操作项URL相关属性界面，如图3-18所示。

配置数据源操作中URL相关属性界面示意图

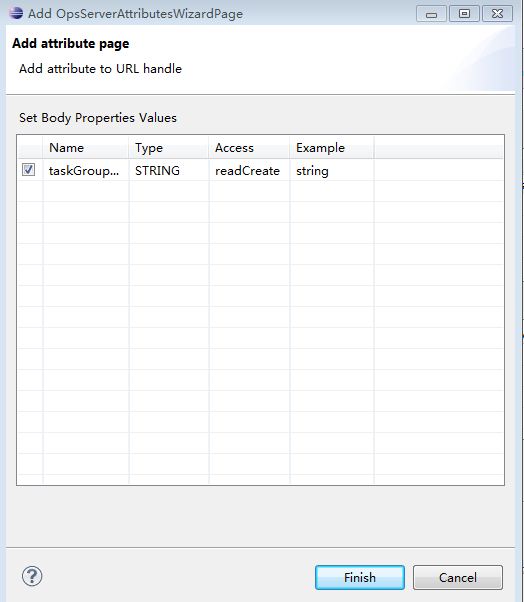
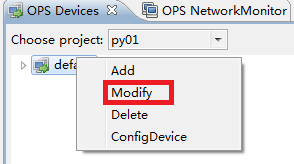


图3-18中勾选部分为操作项URL相关属性信息，最后点击Finish按钮。

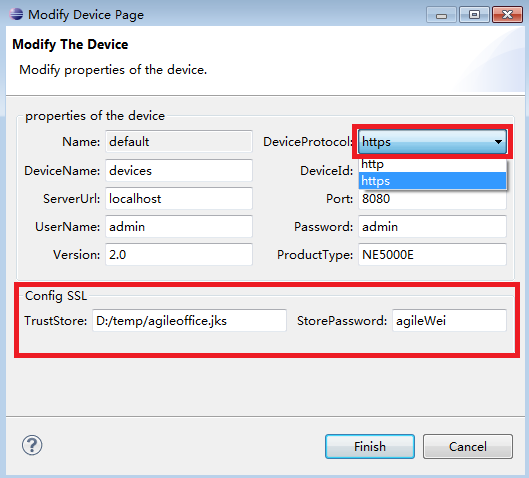
### 修改数据源

修改数据源，如所示。除Name唯一性标识外均可修改。

修改数据源信息示意图



点击Modify后出现如下图的修改项页面

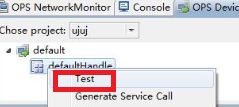


当修改DeviceProtocal时，若为https，则会显示Config SSL一栏，进行相应配置，参数的意义在前面添加数据源操作时已经说明，此处不再说明。

### 测试数据源

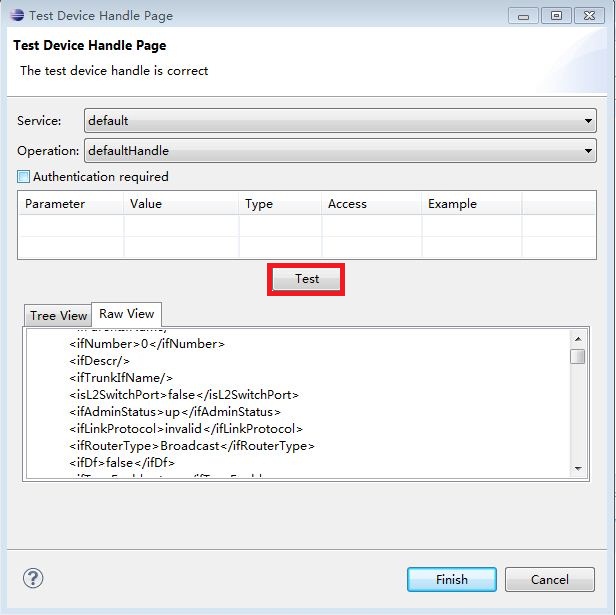
测试数据源，选择默认方法，右键点击Test按钮，如图3-20所示。

选择测试数据源示意图



点击Test按钮后，出现获取的属性界面，如图3-21所示，点击Test完成后，点击Finish按钮。

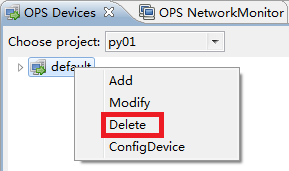
属性界面示意图



### 删除数据源信息

右击数据源图标，选择红色选框部分Delete，删除对应数据源，如图3-22所示。

删除数据源示意图



Add:增加数据源。

Modyfy：修改数据源。

Delete：删除数据源。

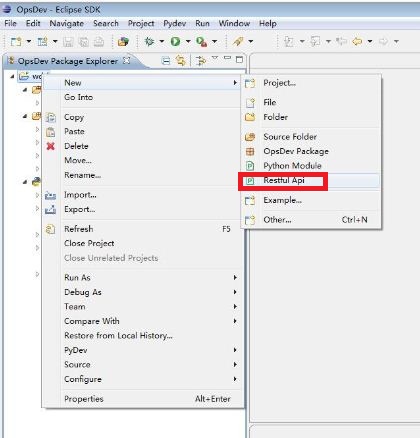
ConfigDevice：配置数据源信息。

## 操作OpsDev-Python Restful Api

### 使用向导新建Api

右键点击工程名，在弹出的右键菜单中选择New--> Restful api，如图3-23所示。

新建OpsDev-Python Restful Api示意图

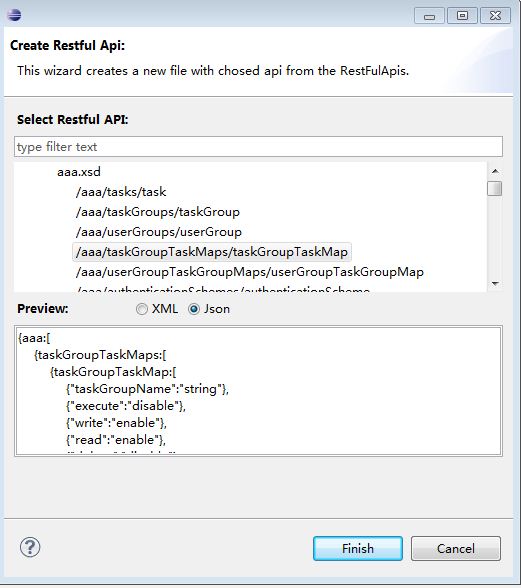


进入Restful Api页面，选择 /aaa/tasks/task为例，如所示。

Select Restful Api为选择具体的Api。

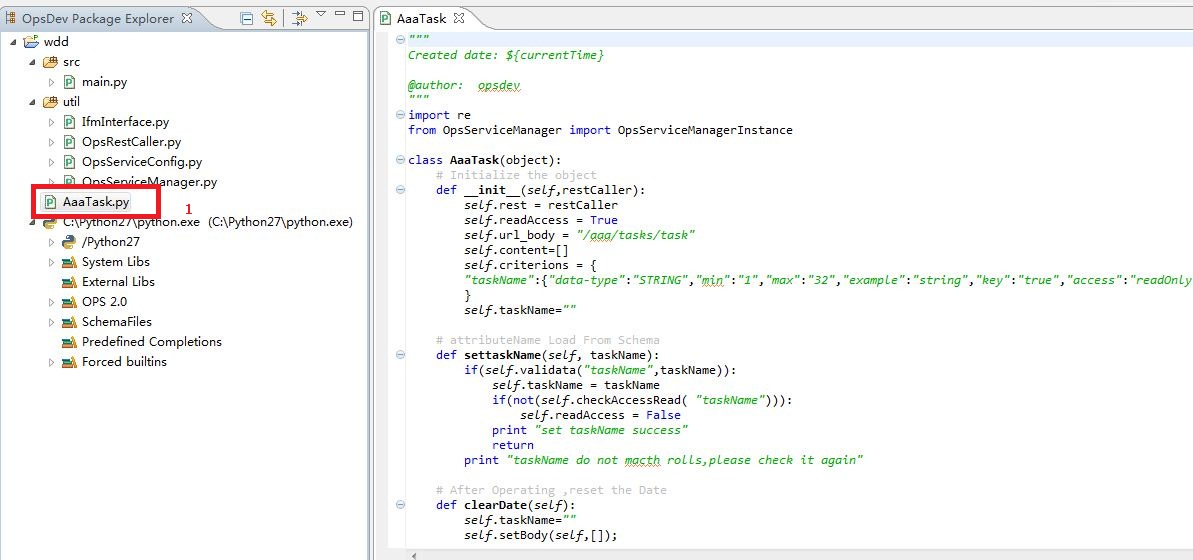
Preview为所选API预览方式以XML和Json两种文件存在，以供用户查看。

OpsDev-Python Restful Api 页面示意图



点击Finish按钮后，自动返回工程，工程自动打开了新生成的AaaTask.py文件，如图3-25所示中的红框1。AaaTask.py：RestfulApi生成的模板。

生成的RestfulApi示意图



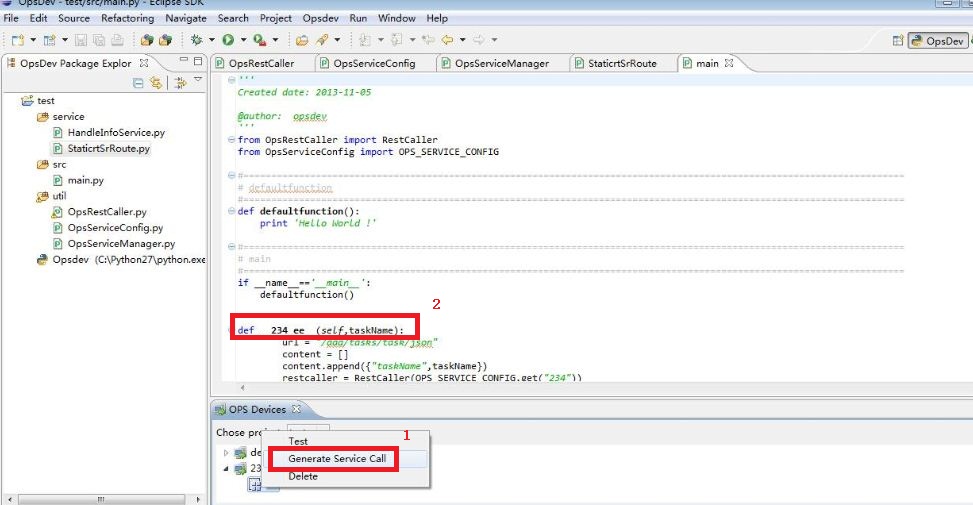
自动生成的代码包含具体的set与get方法，包含creat创建方法， modify修改方法，delete修改方法，get具体方法等。

### 绑定数据源

选择新增数据源，右键点击出现Generate Service Call选择项，如图3-26所示红框1，点击绑定。

在main.py中修改，绑定数据源，如图3-26所示红框2所示为新增的数据源。

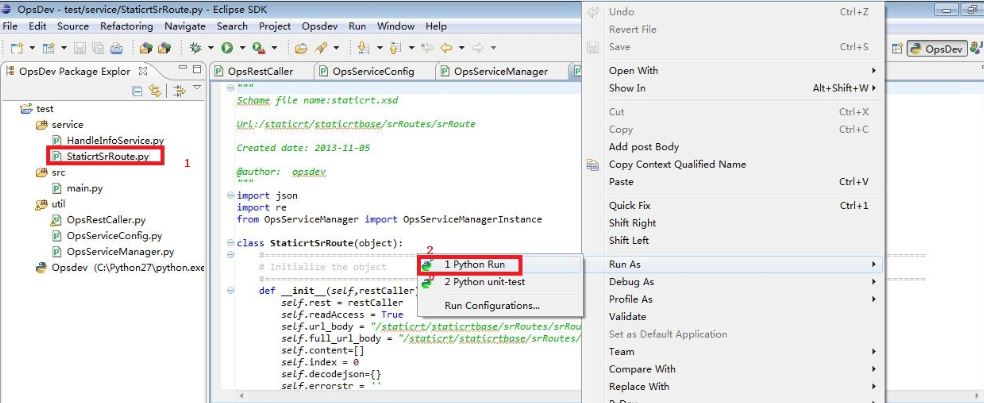
绑定数据源示意图



### 调试OpsDev-Python Restful Api

对生成api进行调试。如所示，选择生成的api，右键点击出现Python Run，运行。

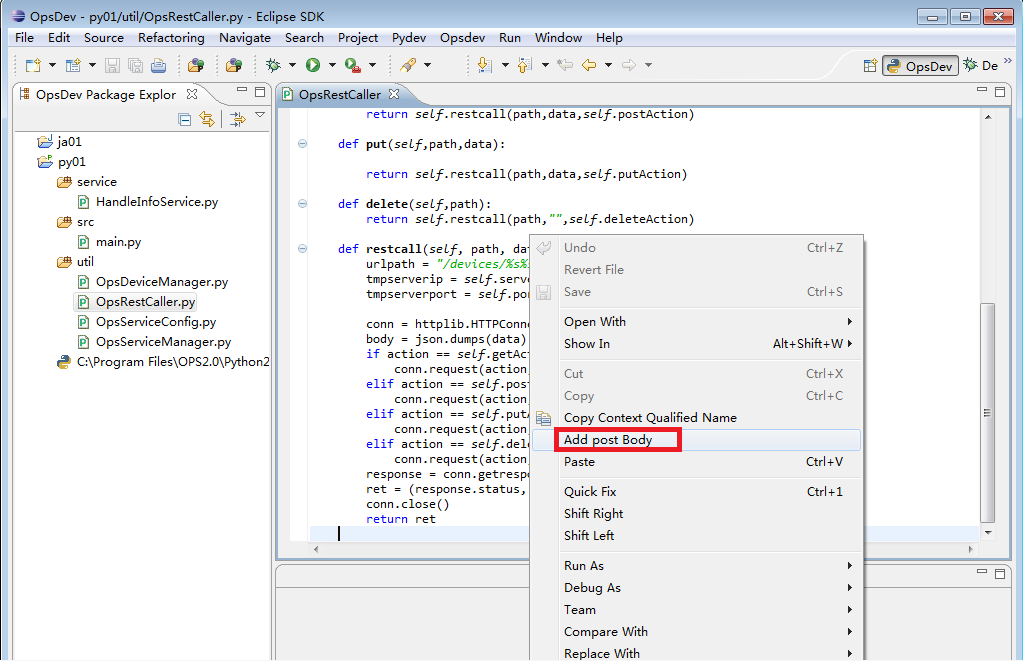
调试OpsDev-Python Restful Api示意图



### 在代码中进行动态生成body（add post body）操作

在代码适当位置右击，选择Add Post Body菜单，如图3-28所示。

在代码中进行动态生成body示意图



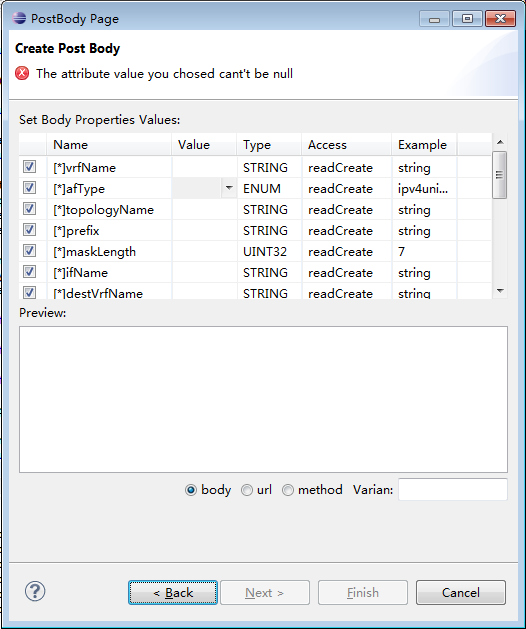
显示inner和user的Schema文件树，并选择一个Schema文件，如图3-29所示。

选择Schema文件示意图



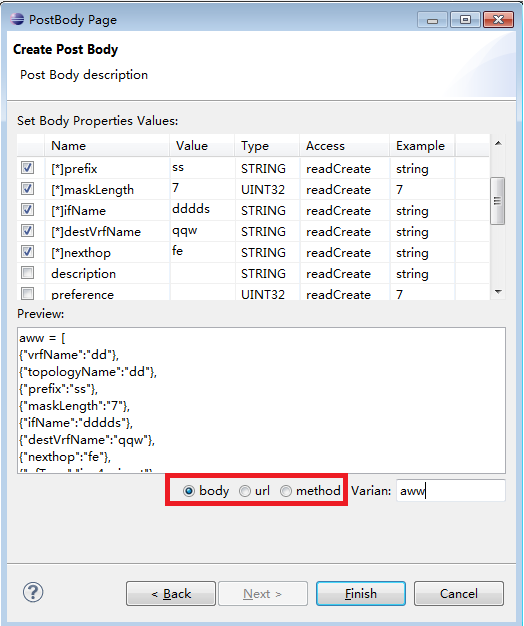
填写必填的参数，带\*号的为必填参数，如图3-30所示。

动态生成body填写参数示意图



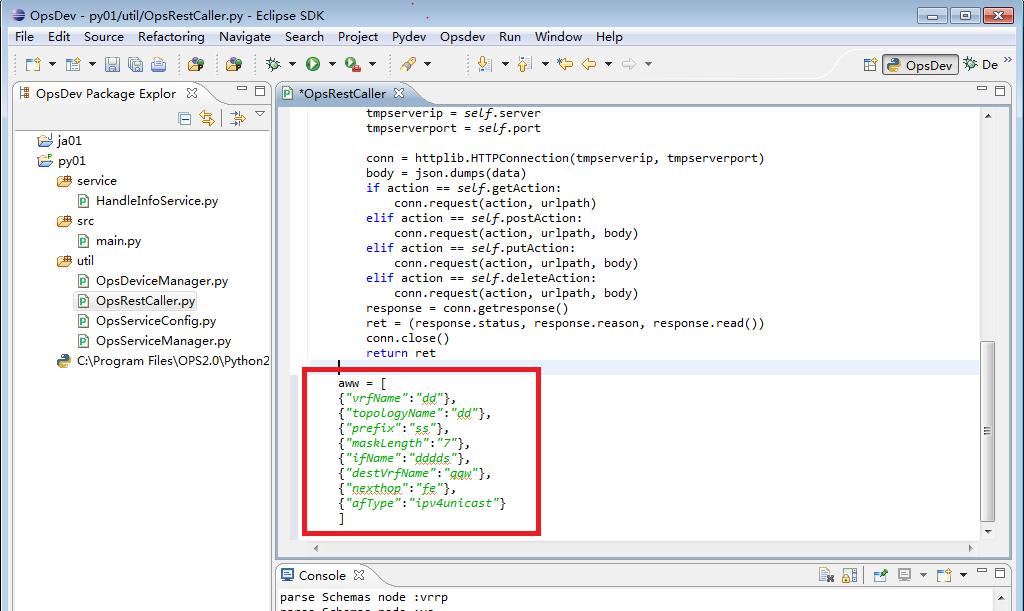
选择需要生成的报文格式，分别有body、url和method，如图3-31所示。

选择需要生成的报文格式示意图



生成相应的报文格式，如图3-32所示。

生成报文格式示意图



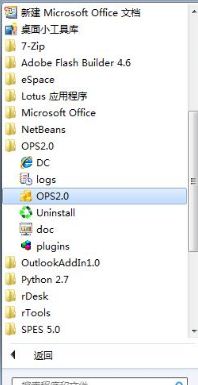
## 编写例程

### 3.4.1 启动OPS2.0

启动OPS2.0，打开OPS2.0.exe文件根据提示点击下一步即可直至安装完成，

如图3-33所示。

启动OPS2.0示意图



### 增加OPS2.0设备

增加OPS2.0设备，如图3-34所示

打开火狐浏览器。

设置POST方法，加入新设备。

设置URL，按SEND键加入设备。

增加OPS2.0设备示意图

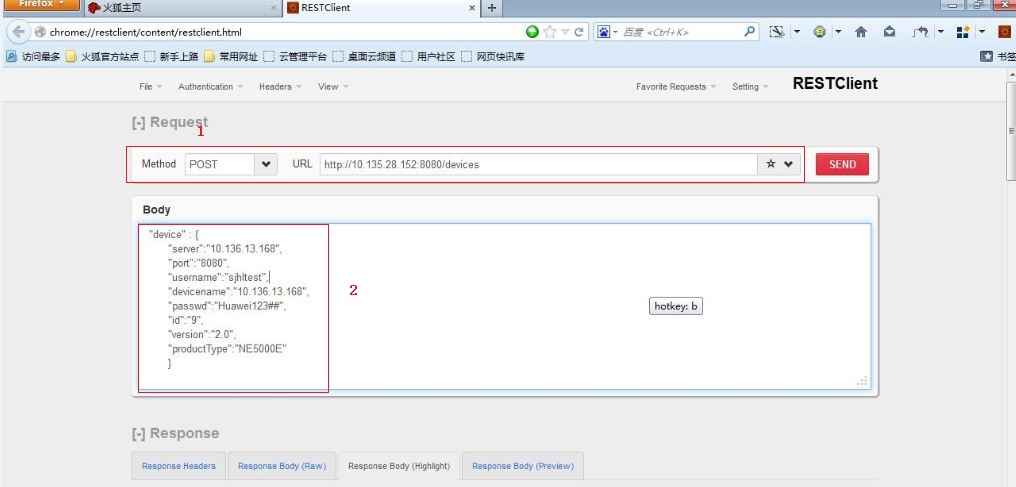


图3-34所示红框1为POST方法加入设备。

图3-34所示红框2为新设备的具体内容。

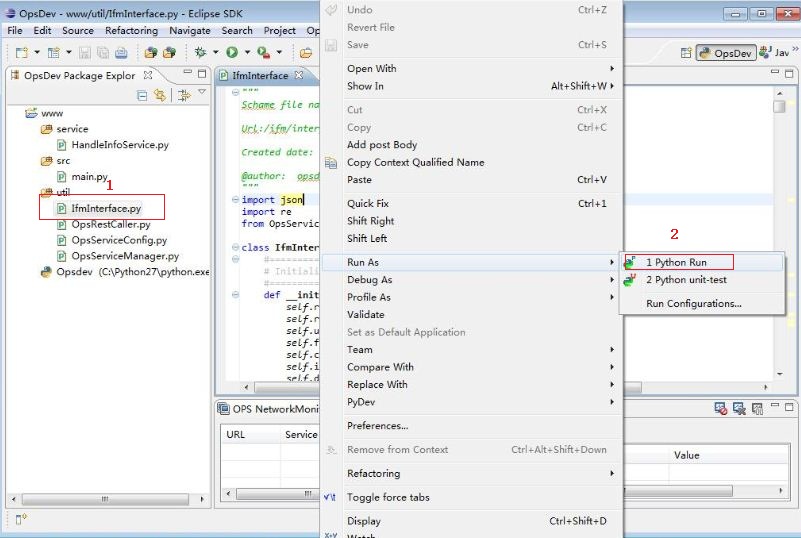
### 运行与调试

* + - 1. 运行示例模板。

选择生成的示例模板，如图3-35所示。

右键点击模板，出现Python Run选择项，点击Python Run运行。

运行示例示意图

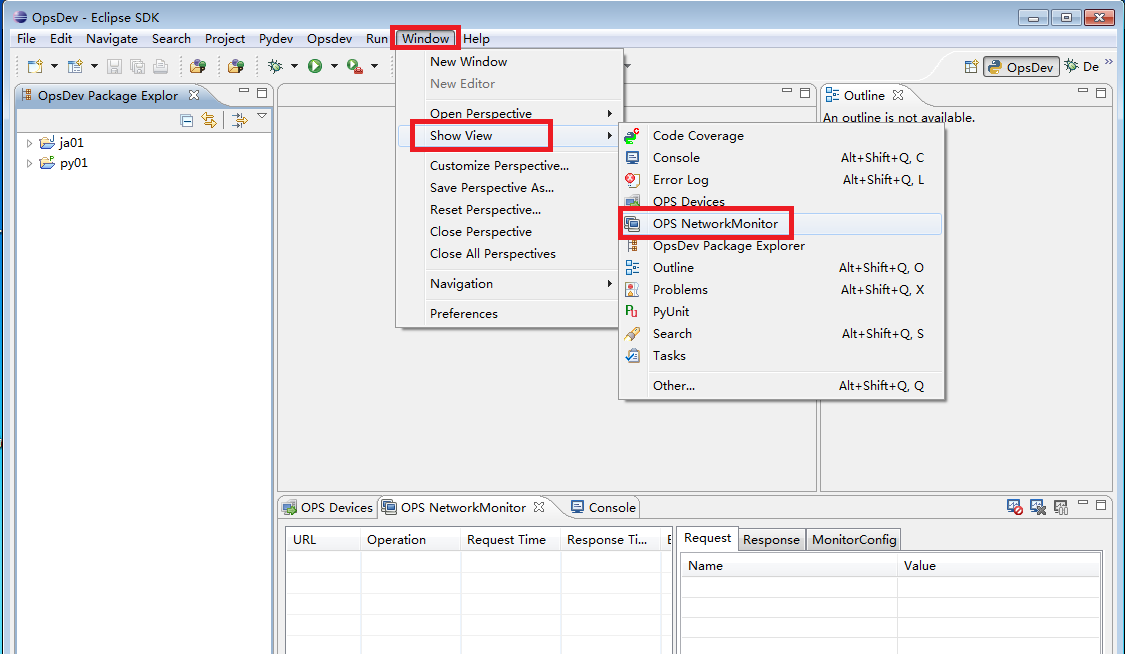


## 操作网络监视器

### 打开网络监视器

打开监视器界面。选择Window-->Show View-->OPS NetworkMonitor，如图3-36所示。

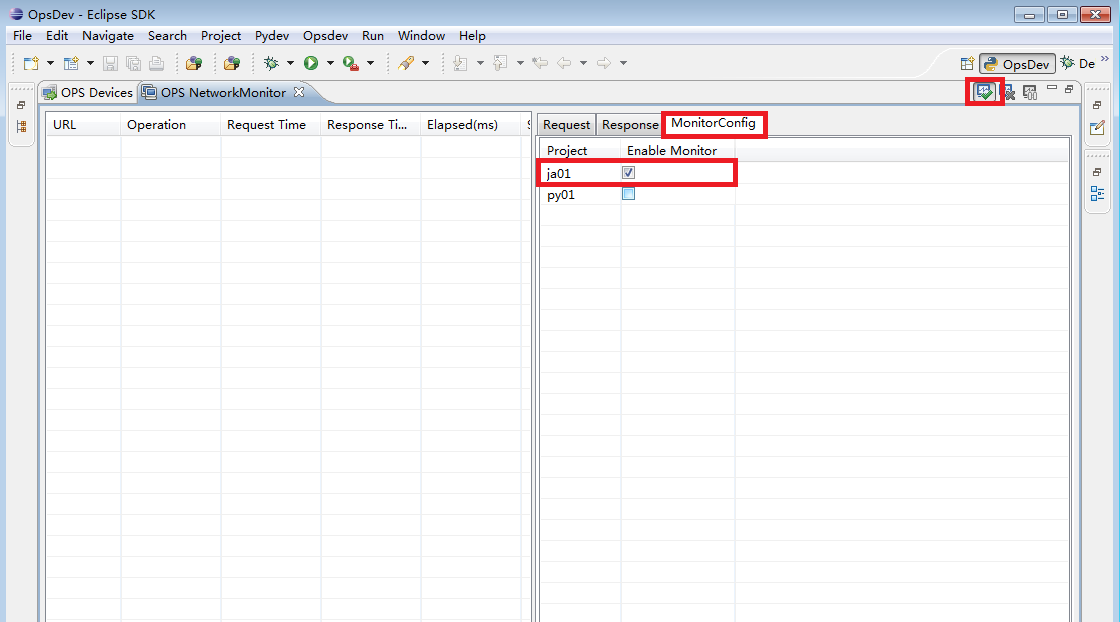
打开网络监视器示意图



### 开启网络监视器

开启监视器状态。由监视器的开关和MonitorConfig进行控制，当打开开关时，勾选Enable Monitor中的勾选框可以选择该工程是否被监听，钩选则进行监听，不钩选则取消监听，如图3-37所示。

开始网络监视器示意图



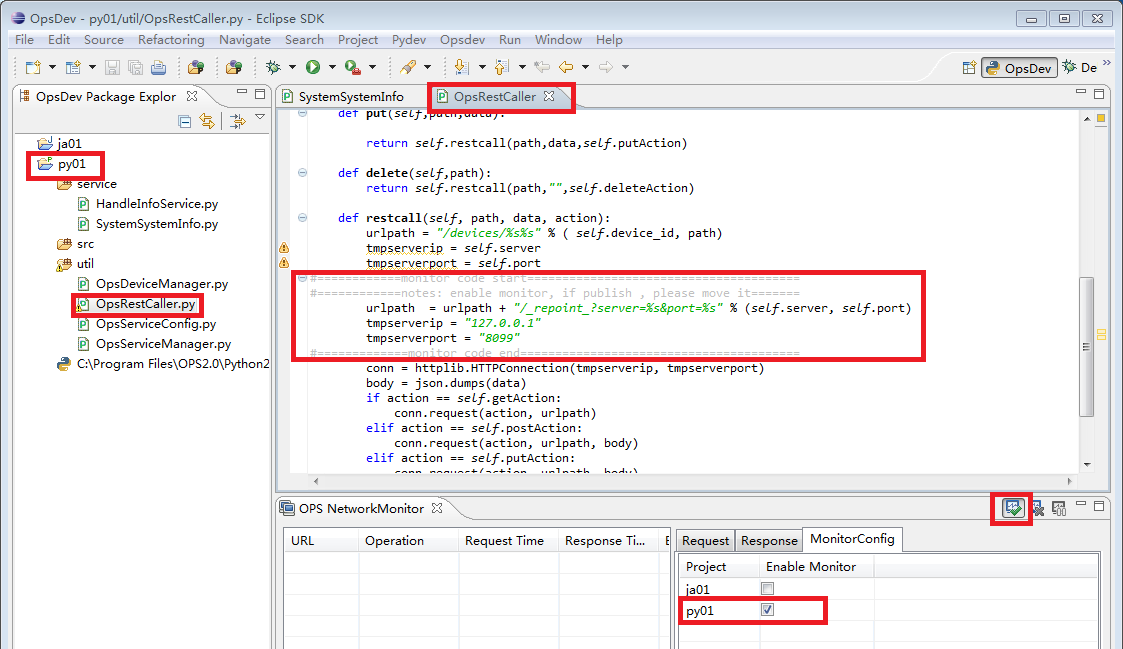
### 监视内容

打开SystemSystemInfo.py 。

开启监视器。

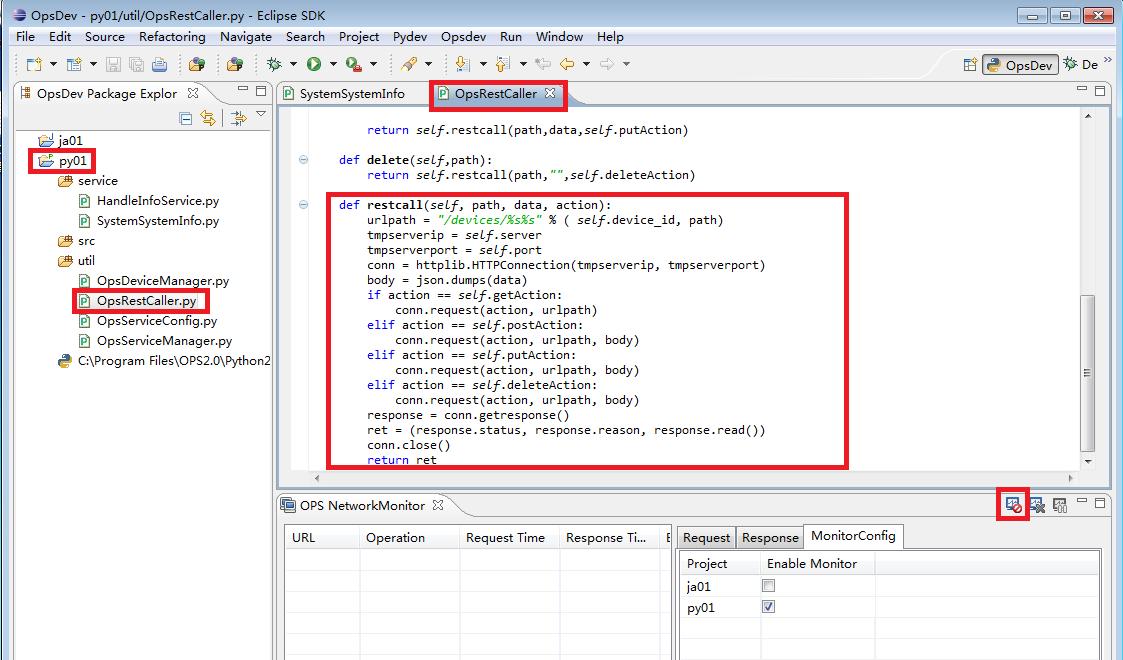
在MonitorConfig中勾选工程后的Enable Monitor选择框，则会打开监听的IP以及端口，如图3-38所示。

监视内容示意图



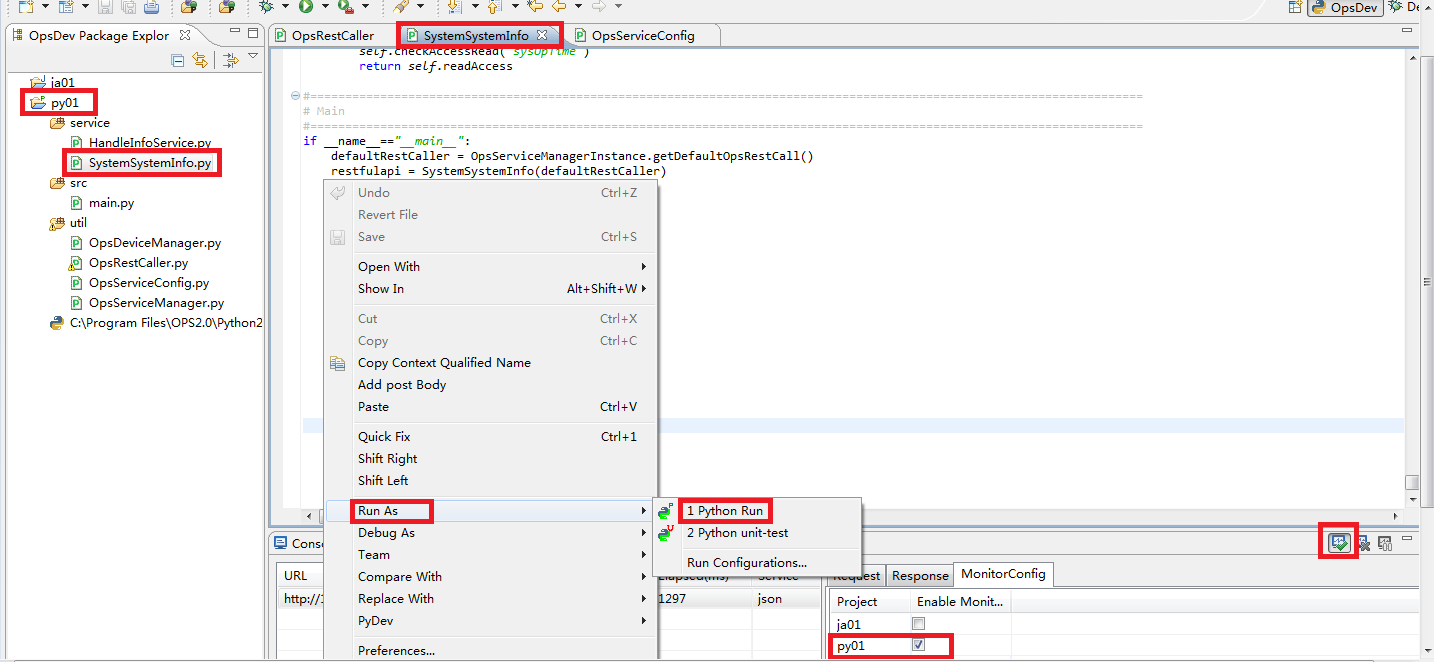
如果监视器的开关或者工程的Enable Monitor勾选框没有选中，则不会打开监听IP以及端口，如图3-39所示。

监视内容示意图



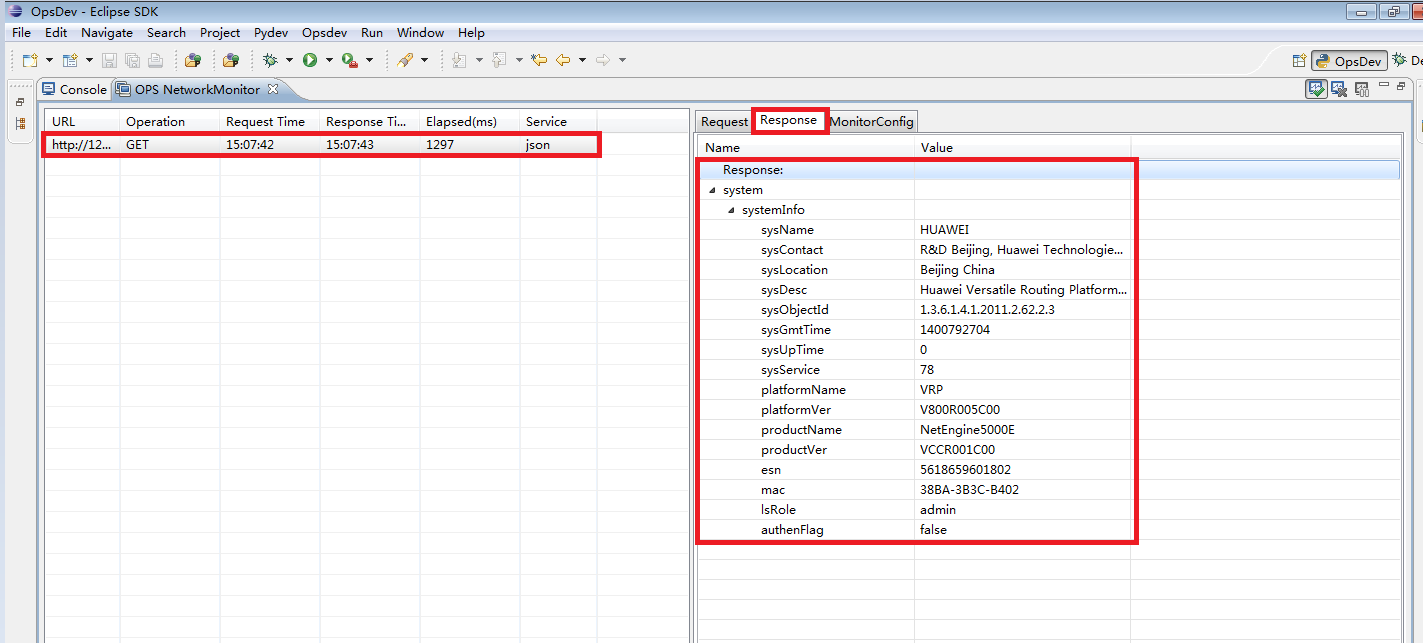
将监听IP以及端口打开（即打开监视器开关并且勾选工程的Enable Monitor勾选框），右击生成的api，出现Python Run 选项，点击运行，如图3-40所示。

监视内容运行示意图



出现监视内容，如图3-41所示。

监视内容结果示意图



### 删除监视内容

点击红框按钮删除监视内容。如图3-42所示。

删除监视内容示意图

