**ORIENTSOFT**



**业务运行监控系统**

**安装配置手册**

**业务运行监控系统新平台**

**安装配置手册**



****

**拓讯科技有限公司**

**2015年12月**

文档信息及修订记录

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 项目名称 | 业务运行监控系统 | | | | |
| 项目编号 |  | | 文档密级 | 机密/秘密/**内部**/公开 | |
| 项目经理 |  | | 项目总监 |  | |
| 文档主送 |  | | | | |
| 文档抄送 |  | | | | |
| 修 订 人 | 修订日期 | 修订说明 | | | 版本号 |
| 袁强远 | 2015-12-15 | 创建 | | | V1.0 |
|  |  |  | | |  |
|  |  |  | | |  |
|  |  |  | | |  |
|  |  |  | | |  |

填写说明：

1.名称、项目编号、项目经理、项目总监按照本项目实际情况填写。

2.密级是该文档允许扩散的范围。对于兰州银行，机密文件必须由信息科技部经理室批准方可借阅；秘密文件必须由项目负责人批准方可借阅；内部文件经一般授权后可由在项目组内部传阅；公开文件不需经过授权，可自由进行阅读。对于欧润特公司，机密文件、秘密文件必须由银行产品事业部经理室批准方可借阅；内部文件经一般授权后可由在公司内部和项目组内部传阅；公开文件不需经过授权，可自由进行阅读。

3.主送是指该文档应该主送的对象，双方项目总监、项目经理是该文档必须主送的对象之一。

4.抄送是指该文档应该抄送的对象，项目管理组是该文档应该抄送的对象之一。

5.指该文档的版本次序号，该文档首次发布时可确定为1.0，如果在上一版的基础上有细微的调整和修改，则可在小数点后次版本号加1；如果该文档内容总体上有重大变化或增加/删除了重要章节，则小数点主版本号加1。

目录

[业务运行监控系统 1](#_Toc438041917)

[文档密级 1](#_Toc438041918)

[机密/秘密/内部/公开 1](#_Toc438041919)

[项目总监 1](#_Toc438041920)

[修 订 人 1](#_Toc438041921)

[修订日期 1](#_Toc438041922)

[修订说明 1](#_Toc438041923)

[版本号 1](#_Toc438041924)

[袁强远 1](#_Toc438041925)

[2015-12-15 1](#_Toc438041926)

[创建 1](#_Toc438041927)

[V1.0 1](#_Toc438041928)

[1前言 4](#_Toc438041929)

[1.1安装前准备 4](#_Toc438041930)

[1.1.1目录结构设定 4](#_Toc438041931)

[1.1.2系统参数设定 5](#_Toc438041932)

[1.1.3软件版本 5](#_Toc438041933)

[1.2检查列表 5](#_Toc438041934)

[1.2.1网络环境 5](#_Toc438041935)

[1.2.2硬件环境 5](#_Toc438041936)

[2平台组件的安装 6](#_Toc438041937)

[2.1 node-red安装 6](#_Toc438041938)

[2.2 Python 2.7 安装 7](#_Toc438041939)

[2.3 MongoDB安装 7](#_Toc438041940)

[2.4 redis安装 7](#_Toc438041941)

[2.5 orientboard安装 7](#_Toc438041942)

[2.6 pushserver 安装 8](#_Toc438041943)

[2.7 Node安装 8](#_Toc438041944)

[2.8 Nginx安装 8](#_Toc438041945)

[2.9 PaymentG2安装 9](#_Toc438041946)

[3系统配置 9](#_Toc438041947)

[3.1平台的组件的配置 9](#_Toc438041948)

[3.1.1 Orient Board配置 9](#_Toc438041949)

[3.1.2 pushserver配置 9](#_Toc438041950)

[3.1.3 node-red配置 10](#_Toc438041951)

[3.1.4 MongoDB配置 13](#_Toc438041952)

[3.1.5 nginx配置 15](#_Toc438041953)

[3.1.6 PaymentG2配置 15](#_Toc438041954)

[4运行和检查 16](#_Toc438041955)

[4.1启动组件 16](#_Toc438041956)

[4.2组件状态检查 16](#_Toc438041957)

[5常见问题及解决办法 16](#_Toc438041958)

[5.1 常见问题 16](#_Toc438041959)

# 1前言

## 1.1安装前准备

### 1.1.1目录结构设定

在部署实施的时候我们软件程序文件，程序配置文件，程序启动文件等都放在/opt/voyager目录下，具体如下：

/opt/voyager/apps /\*程序的目录\*/

/opt/voyager/bin /\*程序启动脚本的目录\*/

/opt/voyager/conf /\*程序配置文件的目录\*/

/opt/voyager/init /\*程序初始化脚本的目录\*/

/opt/voyager/logs /\*程序日志文件的目录\*/

/opt/voyager/status /\*程序运行状态的目录\*/

/opt/voyager/tools /\*相关工具的目录\*/

具体程序，脚本，配置文件所在目录如下：

|  |  |
| --- | --- |
| 程序 | 目录 |
| apibus | /opt/voyager/apps/APIBus |
| orientboard | /opt/voyager/apps/orientboard |
| psEX | /opt/voyager/apps/psEX |
| nodered | /opt/voyager/apps/nodered |
| Payment2 | /opt/voyager/apps/Payment2 |
| portal | /opt/voyager/apps/portal |
| Nodered\_proj | /opt/voyager/nodered\_proj |
| 配置文件 | 位置 |
| orientboard | /opt/voyager/conf/orientboard\_config.json |
| psEX | /opt/voyager/conf/config-psEX.json, |
| /opt/voyager/conf/config-PsWeb.json |
| Payment2 | /opt/voyager/conf/PaymentG2\_config.json |
| portal | /opt/voyager/conf/portal\_config.json |
| nginx | /opt/voyager/conf/voyager.conf |
| /opt/voyager/conf/nginx.conf |
| 启动脚本 | 位置 |
| apibus | /opt/voyager/bin/startAPIBusCenter.sh |
| /opt/voyager/bin/startAPIBusGateway.sh |
| orientboard | /opt/voyager/bin/start-orientoard.sh |
| psEX | /opt/voyager/bin/start-PsWeb.sh |
| /opt/voyager/bin/start-psex.sh |
| nodered | /opt/voyager/apps/nodered/nodered.sh |
| Payment2 | /opt/voyager/bin/startPaymentG2Pusher.sh |
| /opt/voyager/bin/startPaymentG2Web.sh |
| portal | /opt/voyager/bin/startPortal.sh |

设置环境变量：

在/etc/bash.bashrc中添加：

export VOYAGER\_HOME=/opt/voyager

### 1.1.2系统参数设定

### 1.1.3软件版本

Python-2.7.3

Mongodb：2.6.10

nodejs v0.10.39

pushserver v1.0

orientboard v1.0

node-red v0.9.1

redis v2.8.21

## 1.2检查列表

### 1.2.1网络环境

为三台服务器分配三个IP地址，并确保局域网内的机器能访问服务器的20,21,22,8888,10000,51008,1880,2880,3880,4880,5880,2888,1888,15000,7767,51005,6379,11883,51007,80,51051端口。

### 1.2.2硬件环境

**（1）硬件环境**

**监控服务器硬件配置表**

|  |  |
| --- | --- |
| **名称** | **规格和数量** |
| **CPU** | 2颗 英特尔®至强™处理器E5-2400系列 |
| **内存** | **16条32G DDR3 内存** |
| **硬盘** | 12块热插拔3.5寸SATA硬盘3T容量，2块机箱内置2.5寸SSD硬盘 480G容量 |
| **阵列卡** | **支持Raid0、Raid1、Raid5、Raid10** |
| **网络** | **2个千兆网卡** |

**（2)软件环境**

**监控服务器软件配置表**

|  |  |  |
| --- | --- | --- |
| **机器名** | **名称** | **版本** |
| **Core01**  **（98.11.1.113）** | **操作系统** | **Ubuntu 14.10** |
| **数据库（active）** | **Redis 2.8.19** |
| **Orient Board** | **V1.0** |
| **PushServer** | **V3.9.3** |
| **Node-red** | **0.10.6** |
| **Core02**  **（98.11.1.114）** | **操作系统** | **Ubuntu 14.10** |
| **数据库（active）** | **Redis 2.8.19** |
| **Orient Board** | **V1.0** |
| **Node-red** | **0.10.6** |
| **Pushserver** | **V3.9.3** |
| **Db01**  **（98.11.1.115）** | **操作系统** | **Ubuntu 14.10** |
| **数据库** | **mongoDB 2.4.6** |
| **shinken** | **1.2.4** |
| **Nrpe** | **2.12** |
| **GeneralParser** | **V1.0** |
| **Db02**  **（98.11.1.116）** | **操作系统** | **Ubuntu 14.10** |
| **数据库** | **MongoDB 2.4.6** |
| **GeneralParser** | **V1.0** |

**（3)报表展示端的主机硬件配置要求**

|  |  |
| --- | --- |
| **名称** | **规格** |
| **CPU** | **intel 奔腾4 以上** |
| **内存** | **4G以上** |
| **浏览器** | **Google Chrome** |

# 2平台组件的安装

## 2.1 node-red安装

拷贝node-red压缩包到/opt/voyager/apps目录下，然后用下面的命令解压即可：

[root@voyager\_linux]#tar -xvf node-red.tgz

拷贝nodered-proj压缩包到/opt/voyager目录下，然后用下面的命令解压即可：

[root@voyager\_linux]#tar -xvf nodered-proj.tgz

## Python 2.7 安装

安装python 2.7

下载Python-2.7.3.tar.bz2, 解压到/opt/voyager/目录下，执行下面的命令安装

[root@voyager\_linux]#cd python-2.7.3

[root@voyager\_linux]#./configure

[root@voyager\_linux]#make

[root@voyager\_linux]#make install

## MongoDB安装

安装Mongo DB(2.4.x stable):

1. 下载

mongodb-org-2.6.10-1.x86\_64.rpm

mongodb-org-shell-2.6.10-1.x86\_64.rpm

mongodb-org-mongos-2.6.10-1.x86\_64.rpm

mongodb-org-tools-2.6.10-1.x86\_64.rpm

mongodb-org-server-2.6.10-1.x86\_64.rpm

执行下面命令安装

[root@voyager\_linux]# rpm -ivh mongodb\*

## redis安装

1. 下载

jemalloc-3.6.0-1.el6.x86\_64.rpm

redis-2.8.21-4.el6.art.x86\_64.rpm

1. 执行下面命令安装，

[root@voyager\_linux]# rpm -ivh jemalloc-3.6.0-1.el6.x86\_64.rpm

[root@voyager\_linux]# rpm -ivh redis-2.8.21-4.el6.art.x86\_64.rpm

## orientboard安装

下载以下依赖包并依次进行解压安装：

pymongo-2.7.2.tar.gz

python-termstyle-0.1.10.tar.gz

colorama-0.3.3.tar.gz

rednose-0.4.3.tar.gz

nose-1.0.0.tar.gz

mongoengine-0.10.0.tar.gz

MarkupSafe-0.23.tar.gz

Jinja2-2.7.3.tar.gz

itsdangerous-0.24.tar.gz

Werkzeug-0.10.4.tar.gz

Flask-0.10.1 (1).tar.gz

WTForms-2.0.2.zip

Flask-WTF-0.12.tar.gz

flask-mongoengine-0.7.1.tar.gz

python-bcrypt-0.3.1.zip

Flask-Bcrypt-0.6.2.tar.gz

Flask-Login-0.2.11.tar.gz

six-1.9.0.tar.gz

Flask-Cors-2.0.1.tar.gz

安装命令如下：（**root权限下**）

（以pymongo-2.7.2.tar.gz作为列子说明，其他包安装方式完全一致）：

tar –xzvf pymongo-2.7.2.tar.gz

cd pymongo-2.7.2

python setup.py install

（WTForms-2.0.2.zip只是解压方式不同，安装完全一样，命令如下）

unzip WTForms-2.0.2.zip

cd WTForms-2.0.2

python setup.py install

orientboard程序包在/opt/voyager/apps目录下，不用另外进行安装。

## pushserver 安装

pushserver程序包在/opt/voyager/apps目录下，不用另外进行安装

## Node安装

下载获取安装包：nodejs-0.10.39-1nodesource.el6.x86\_64.rpm

。执行下面的命令安装Node

rpm –ivh nodejs-0.10.39-1nodesource.el6.x86\_64.rpm

执行下面的命令查看是否安装成功：（得到如下图结果表示安装成功）



## Nginx安装

下载获取安装包：nginx-1.2.8.tar.gz

执行下面命令安装nginx

tar –xzvf nginx-1.2.8.tar.gz

cd nginx-1.2.8

./configure

make

make install

## PaymentG2安装

PaymentG2程序包都在/opt/voyager/apps目录下，不用另外进行安装。

# 3系统配置

## 3.1平台的组件的配置

### Orient Board配置

Orientboard配置文件为：/opt/voyager/conf/orientboard\_config.json

内容如下：

{

"root\_url":"/orientboard",

"puship":"192.168.0.181:51005",

"db\_address":"127.0.0.1",

"db\_port":"27017",

"secret\_key":"55eae02566fdd7adb43",

"mongodb\_name" : "OrientBoard"

}

需要修改三项：

1：puship：指定pushserver程序的ip与端口。

2：db\_address：指定mongo数据库的地址

3：db\_port：指定mongo的端口

### pushserver配置

pushserver的配置文件为：/opt/voyager/conf/config-psEX.json与/opt/voyager/conf/config-Psweb.json两个配置文件。

内容如下：

1：/opt/voyager/conf/config-Psweb.json（用于web页面）

{

"cookieSecret": "psweborientsoft",

"db": "psweb",

"host": "localhost",

"mongohost": "192.168.0.181",

"auth": "psweb:welcome1",

"server": {

"host": "127.0.0.1",

"port": "51008"

},

"psServer":{

"phost":"http://192.168.0.181",

"pport":"51007"

}

}

需要修改的为：

1：db：mongo数据库中pushserver对应的库名

2：mongohost：mongo数据库的地址

3：psServer的phost指定psserver的主机地址。

2：/opt/voyager/conf/config-psEX.json（用于数据推送）

{

"redishost": "127.0.0.1",

"redisport": 6379,

"prefix": "PSEX",

"port": 51005,

"debug": false,

"managerClientChannel": "PSEXManagerClient",

"managerServerChannel": "PSEXManagerServer",

"userhost": "192.168.0.181",

"userport": 8888,

"restport": 51007,

"mongohost": "127.0.0.1",

"mongoport": 27017,

"statusinterval": 1000

}

需要修改的为：

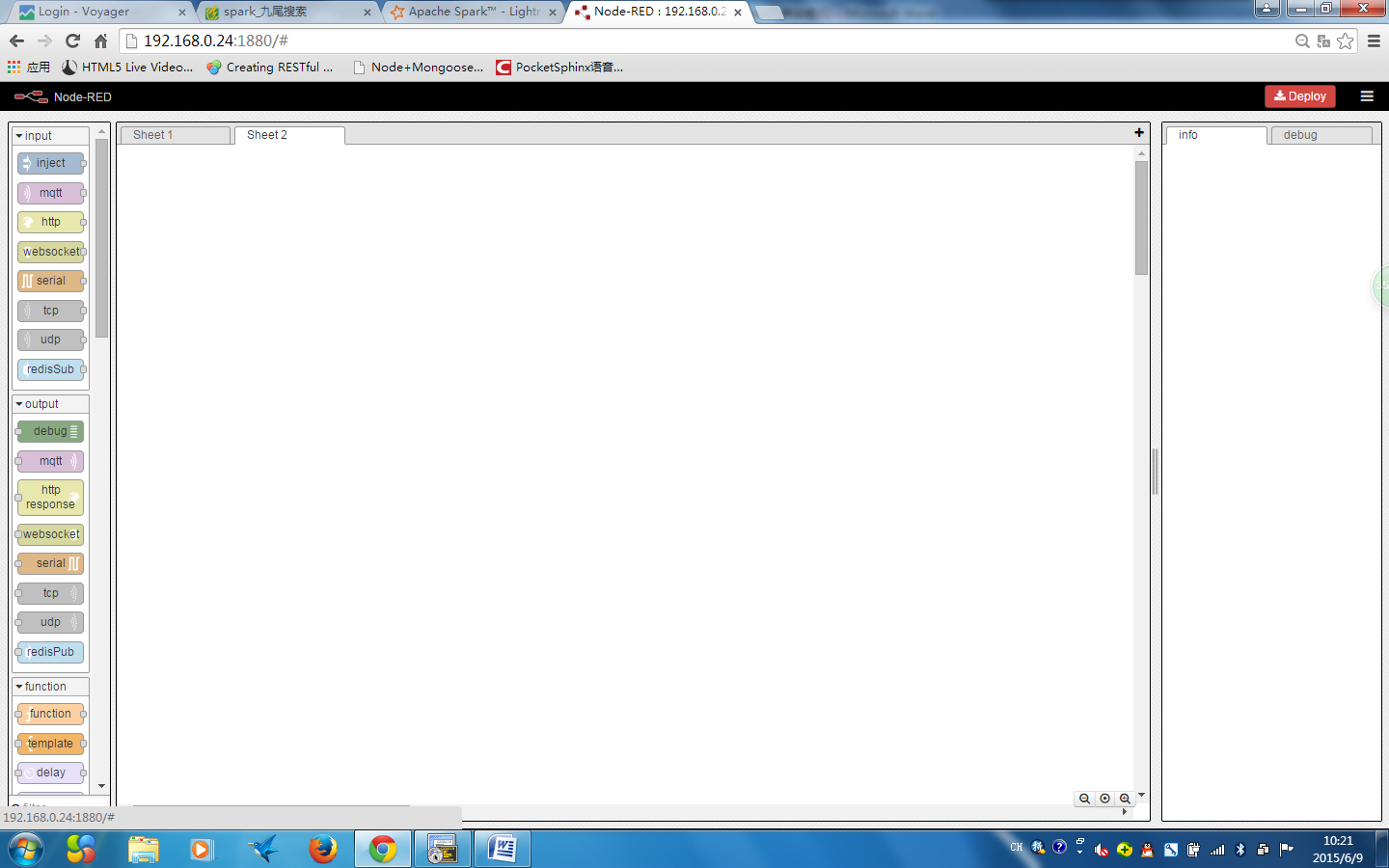
1：redishot：指定redis的地址

2：userhost：指定主机地址。

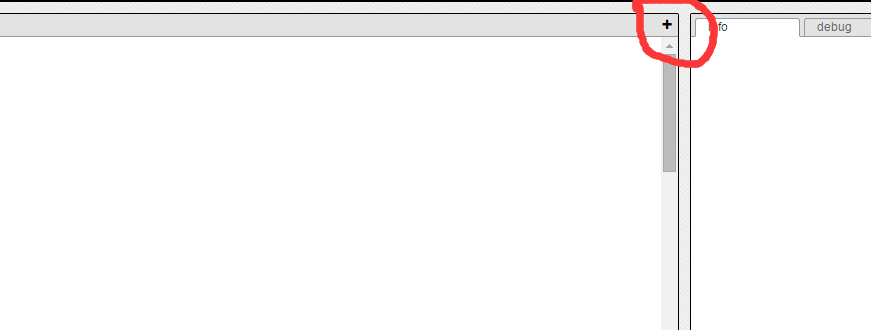
### node-red配置

node-red的配置主要是已有的流程导入,

首先把已经导出的库（json格式的文件）拷贝到/opt/nodered\_proj目录下的lib/flows目录下，然后启动node-red,按如下方法导入：

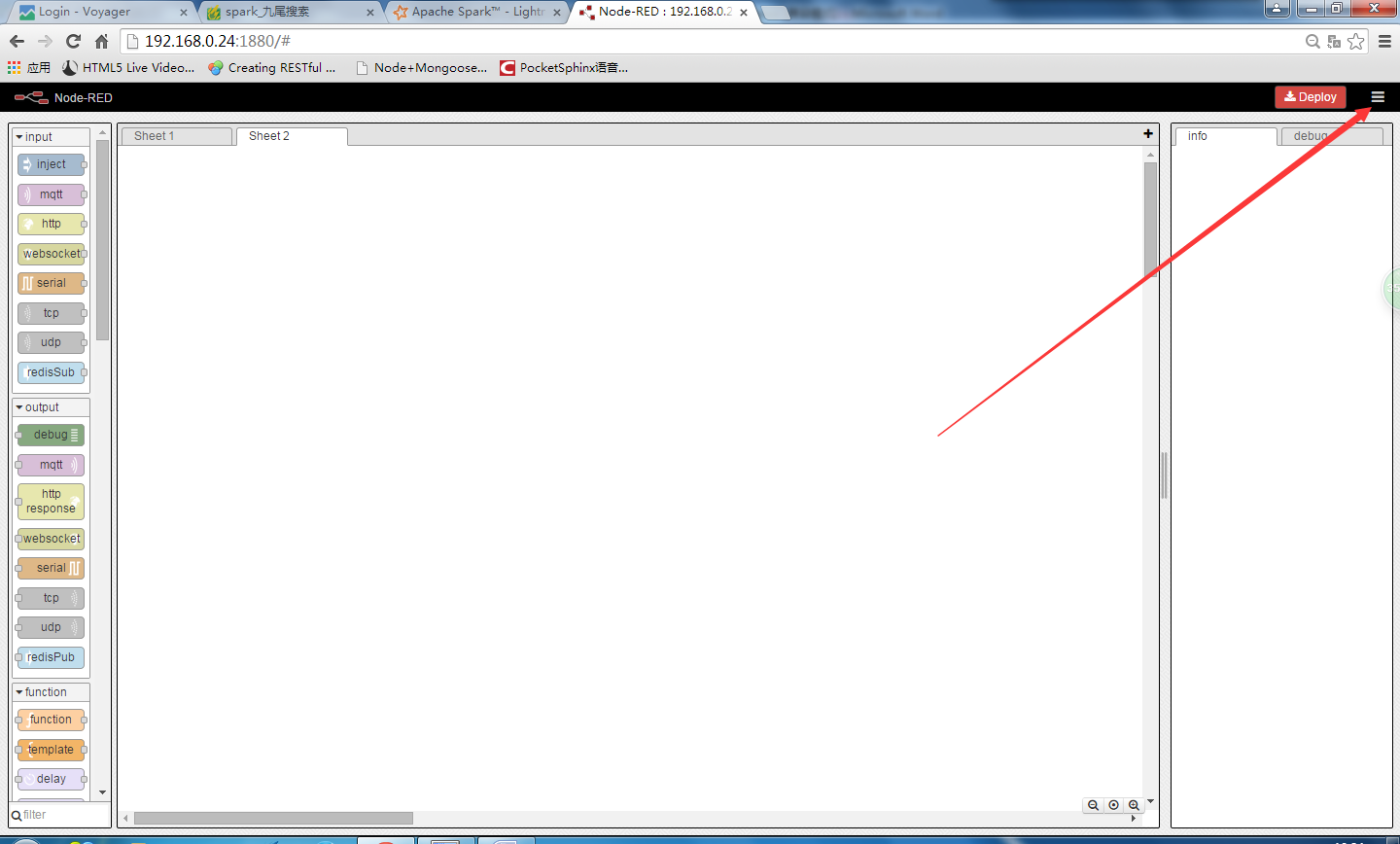


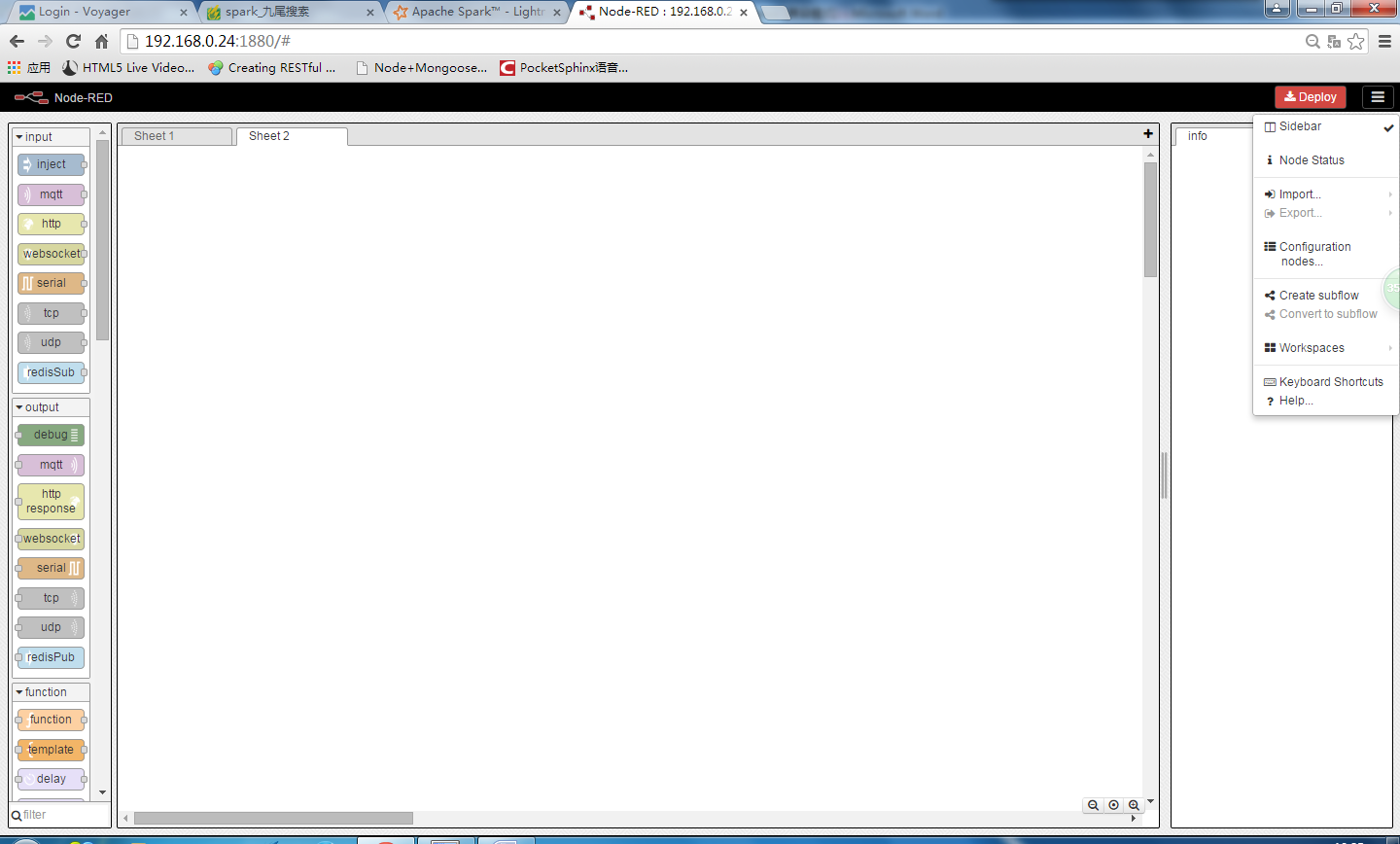
点击下图中圆圈里的加号：



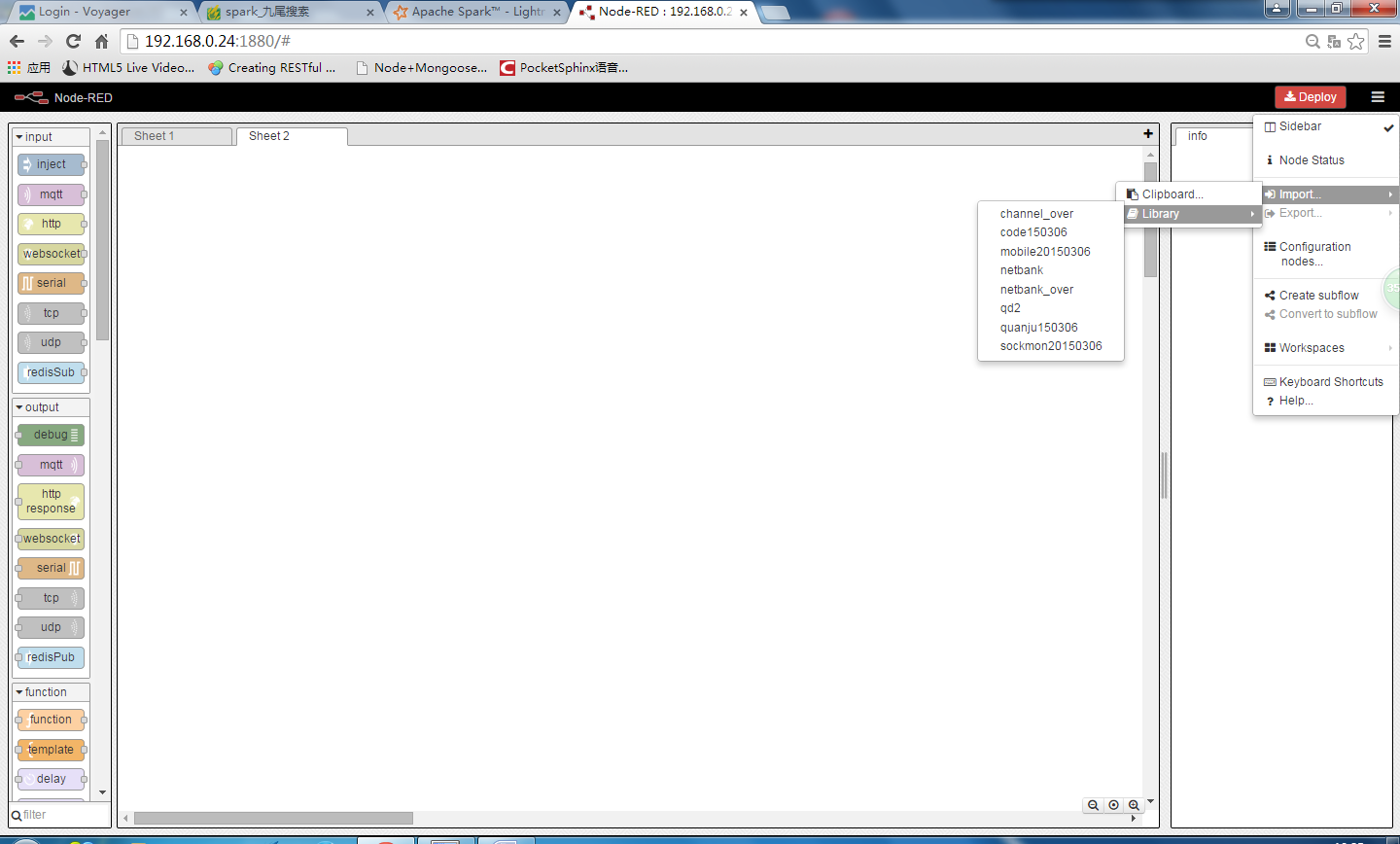
可以创建一个新的sheet，双击sheet的tab标签可以修改sheet的名字，

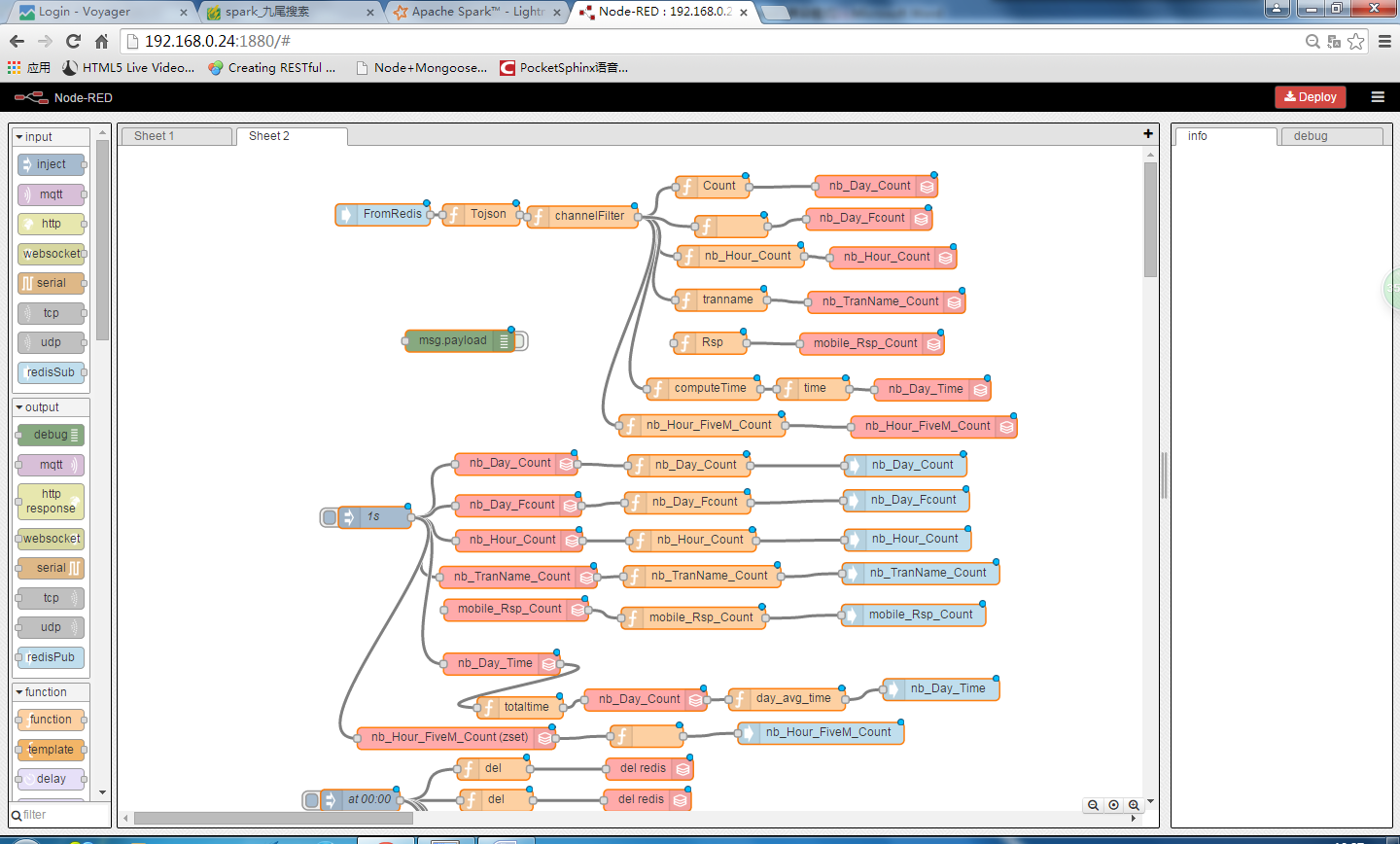
点击箭头所指的地方打开菜单





选择import->library可以看到要导入的流程，选择你要导入的流程。





可以看到已开发的流程，拖动到合适位置点击左键确定。然后点击右上角的Deploy按钮让导入的流程生效。

检查类似redisSub等节点修改其中的一些ip地址。

### MongoDB配置

使用ulimit配置Mongo DB所需系统资源:

1）先确认系统资源情况,可以先启动一下mongo db然后使用return-limits mongod

[root@voyager\_linux]# ulimit–a

core file size (blocks, -c) 0

dataseg size (kbytes, -d) unlimited

scheduling priority (-e) 0

file size (blocks, -f) unlimited

pending signals (-i) 257810

max locked memory (kbytes, -l) 64

max memory size (kbytes, -m) unlimited

open files (-n) 4096

pipe size (512 bytes, -p) 8

POSIX message queues (bytes, -q) 819200

real-time priority (-r) 0

stack size (kbytes, -s) 10240

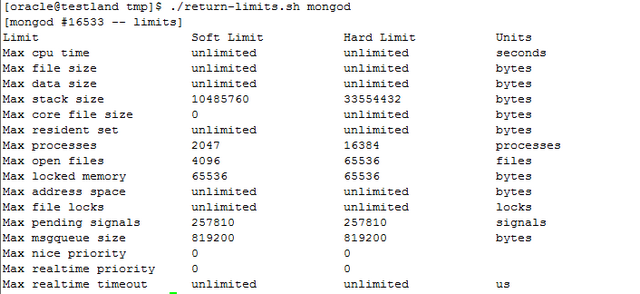
cpu time (seconds, -t) unlimited

max user processes (-u) 2047

virtual memory (kbytes, -v) unlimited

file locks (-x) unlimited

[root@voyager\_linux]# return-limits mongod



2）根据步骤一的具体情况

调整为建议参数

open file size 65536

process 32000

**库的导出/导入：**

需要导入的库：

OrientBoard：包含了BOARD中所有用户，面板，CHARTS的信息。

导出：mongodump –d OrientBoard –o /opt/voyager/OrientBoard(导出后的路径)

导入：mongorestore –d OrientBoard /opt/voyager/OrientBoard(库的路径)

PSEX：包含了PUSHSERVER张所有通道，用户信息。

导出：mongodump –d PSEX –o /opt/voyager/PSEX(导出后的路径)

导入：mongorestore –d PSEX /opt/voyager/PSEX(库的路径)

psweb：包含PUSHSERVER管理页面的信息。

导出：mongodump –d psweb –o /opt/voyager/psweb(导出后的路径)

导入：mongorestore –d psweb /opt/voyager/psweb(库的路径)

### nginx配置

nginx配置文件为：/opt/voyager/conf/nginx.conf与/opt/voyager/conf/voyager.conf

1：将/usr/local/nginx/conf/nginx.conf替换为/opt/voyager/conf/nginx.conf

2：在/usr/local/nginx/conf下新建一个sites-enabled目录

将/opt/voyager/conf/voyager.conf配置文件放到sites-enabled中。

### PaymentG2配置

PaymentG2配置文件为：/opt/voyager/conf/PaymentG2\_config.json

内容为：

{

"rootPrefix" : "/",

"webPort" : 10000,

"mongo" : {"URL": "mongodb://127.0.0.1:27017", "RS": "", "USER": "voyager", "PASSWORD": "welcome1","AUTH": "false"},

"redisHost": "127.0.0.1",

"redisPort": 6379,

"pushPort": 50051

}

需要修改的为：

1：webPort：页面端口

2：mongo：指定mongodb的地址，端口，认证信息

3：redisHost：指定redis的地址

# 4运行和检查

## 4.1启动组件

系统启动顺序以及命令：

1：mongo

mongod -f /etc/mongod.conf

2：redis

redis-server /etc/redis.conf

3：orientboard

/opt/voyager/bin/start-orientboard.sh

4：pushserver

/opt/voyager/bin/start-PsWeb.sh start

/opt/voyager/bin/start-psEX.sh start

5：apibus

/opt/voyager/bin/startAPIBusCenter.sh

/opt/voyager/bin/startAPIBusGateway.sh gw01

/opt/voyager/bin/startAPIBusGateway.sh gw02

/opt/voyager/ node-red/nodered.sh

6：PaymentG2

启动PaymentG2前需要先向redis导入相应的码表。

导入脚本为：/opt/voyager/tools/paymentG2data/importCodeTabletoRedis.sh

导入后在启动paymentG2

/opt/voyager/bin/startPaymentG2web.sh

/opt/voyager/bin/startPaymentG2Pusher.sh

7：nginx

启动nginx

/usr/local/nginx/sbin/nginx -c /usr/local/nginx/conf/nginx.conf

## 4.2组件状态检查

所有程序的日志保存在：/opt/voyager/logs

所有程序运行PID文件保存在：/opt/voyager/status

# 5常见问题及解决办法

## 常见问题

待续！