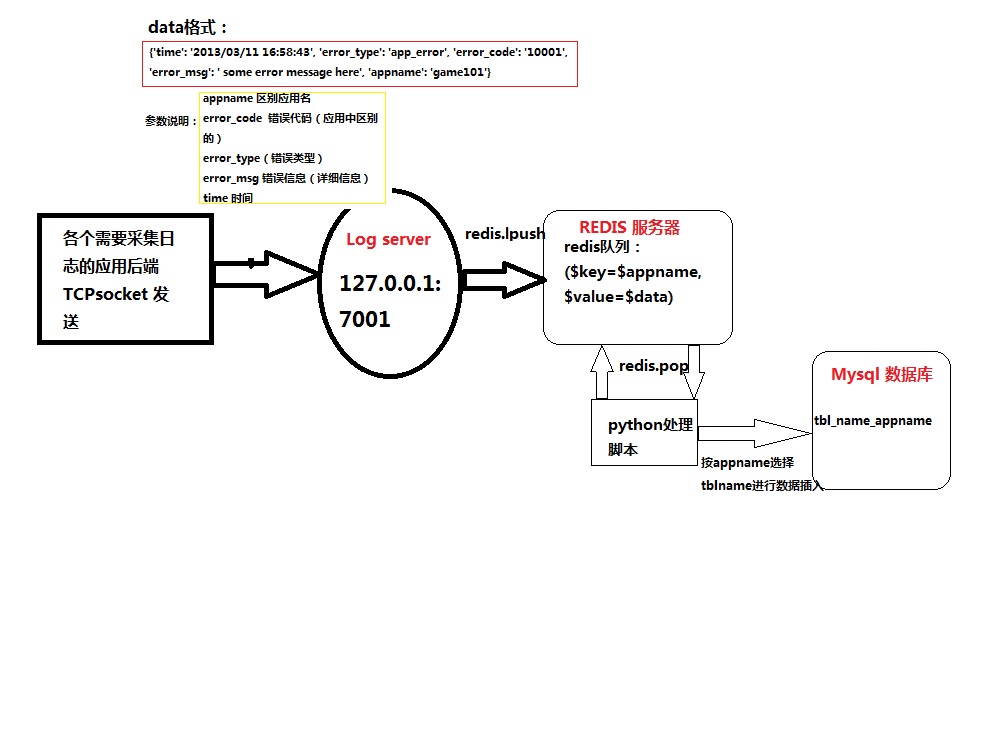
# 日志采集服务端设计

一、主体设计思路：



主要采用redis队列的方式，将各应用传递的data同过l\_push() 推送进redis队列，后端使用python脚本进行redis队列数据的pop采集和mysql-insert操作

二、参数设计

Socket 发送方式data格式为：

{'appname’:'game101', 'err\_code': '10001','error\_type': 'app\_error', 'error\_msg': ' some error message here','time':'2013/03/11 16:58:43'}

各参数说明：

appname 区别应用名称如(game101等)  
error\_code  错误代码（应用中区别的）  
error\_type 错误类型 （由应用自行设定）  
error\_msg 错误信息（详细信息）  
time 时间

对应的appname、error\_code、error\_type 等会在数据采集的后台数据库设计中进行配置设计，如gs\_error\_log\_type\_game

gs\_error\_log\_type\_gplatform等

其中中文字符一律使用编码格式utf8进行编码

三、表结构设计

gs\_log\_err\_appname\_cfg appname配置表，在表中的

才会被python写入脚本执行插入mysql数据操作

CREATE TABLE IF NOT EXISTS `gs\_log\_err\_appname\_cfg` (

`id` int(10) NOT NULL AUTO\_INCREMENT COMMENT 'id',

`appname` varchar(20) NOT NULL COMMENT 'appname 不能重复',

`app\_desc` varchar(50) NOT NULL COMMENT 'app描述',

`create\_time` timestamp NOT NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP COMMENT '创建时间',

`app\_type` int(20) NOT NULL COMMENT 'app类型1游戏2平台',

PRIMARY KEY (`id`),

UNIQUE KEY `appname` (`appname`),

KEY `create\_time` (`create\_time`,`app\_type`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='gs\_err\_log appname配置表' AUTO\_INCREMENT=1 ;