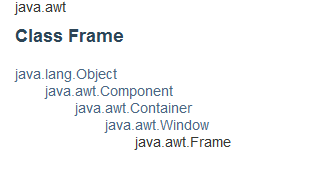
**坦克大战问题**

1. 客户端继承Frame类
2. 定义paint()方法绘画tank和客户端，绘画使用的是awt里面Graphics工具包来绘制方形和圆形的血调
3. 定义updata()方法来绘制背景
4. Frame继承于Compoent中有个repaint()方法,调用runable接口重写run()方法实现交互。



**Hadoop相关问题**

1.mapreduce，map任务分配是通过jobtrcker分配给各个虚拟机的tasktrcker进行实时监控

2.hadoop实现wordcount，在本地ecslipse新建项目，导入zookeeper等相关jar包还有hadoop-hdfs相关包，在hadoop官网找到word count代码调式

3. HDFS本地上传，链接的核心代码是

FileSystem fs = FileSystem.get(URI.create("hdfs://172.20.237.112:9000"),conf);

链接方式是TDH

同时要配置Maven中的pom.xml和hadoop上面的HDFS.site文件

//mapreduce工作流程

jobTracker和namenode一起启动， namenode存储元数据，

jps查看当前虚拟机内容

HDFS 默认 BlockSize 64MB，2.0 是128MB

重新格式化Namenode：hadoop namenode –format

SecondaryNameNode是帮助NameNode 合并编辑日志，减少 NameNode 启动时间

Cloudera提供了Hadoop的商业发行版本CDH

**Hive**不支持更改数据的操作，Hive基于数据仓库，提供静态数据的动态查询。其使用类SQL语言，底层经过编译转为MapReduce程序，在[Hadoop](http://lib.csdn.net/base/hadoop)上运行，数据存储在HDFS上。

**HDFS**是GFS的一种实现，他的完整名字是分布式文件系统，类似于FAT32，NTFS，是一种文件格式，是底层的。

**Hive**与[Hbase](http://lib.csdn.net/base/hbase)的数据一般都存储在HDFS上。Hadoop HDFS为他们提供了高可靠性的底层存储支持。

**Hbase**是Hadoop database，即Hadoop[数据库](http://lib.csdn.net/base/mysql)。它是一个适合于非结构化数据存储的数据库，HBase基于列的而不是基于行的模式。

**YARN**总体上仍然是master/slave（主从）结构

ResourceManager是Master上一个独立运行的进程，负责集群统一的资源管理、调度、分配等等；NodeManager是Slave上一个独立运行的进程，负责上报节点的状态；App Master和Container是运行在Slave上的组件，负责应用程序相关事务，比如任务调度、任务监控和容错等，Container是yarn中分配资源的一个单位，包涵内存、CPU等等资源，yarn以Container为单位分配资源。

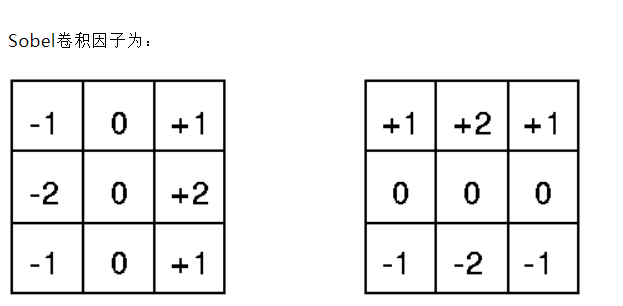
ApplicationMaster管理一个在YARN内运行的应用程序的每个实例。ApplicationMaster负责申请而获得的来自ResourceManager的资源，并通过NodeManager监视容器的执行和资源的使用（cpu、内存等资源分配）。

**数学建模问题**

PAC算法核心把所有属性的值看作一个向量做映射到低维向量上，并计算协方差，通过协方差值来选择剔除向量

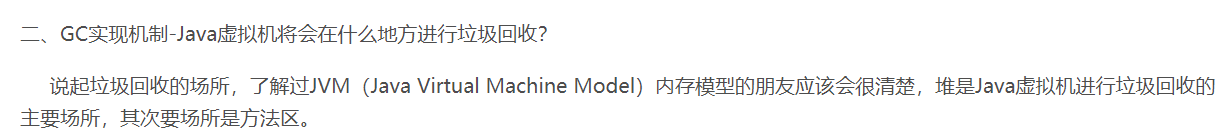
**卷积网络问题**

卷积用的3x3的卷积核，原因是经验，核心问题是28x28的数据，通过损失函数进行方向传播



JAVA相关问题

GC



404

200 - 服务器成功返回网页   
404 - 请求的网页不存在   
503 - 服务不可用

ETL将ip地址解析为地域名字，使用的是IPseek代码网上找的.java

设计模式

