作者在工作中遇到了类似流式数据实时接入的业务场景，所以对淘宝的实时数据仓库这一块做了一些调研和了解。本文从业务场景和设计上介绍了淘宝的TimeTunnel工具，文中的图片来自淘宝数据仓库团队交流过程中的sildes，也参考了一些相关文档。

**业务背景**

TimeTunnel（简称**TT**）是一个基于thrift通讯框架搭建的实时数据传输平台，具有高性能、实时性、顺序性、高可靠性、高可用性、可扩展性等特点（基于Hbase）。

目前TimeTunnel在阿里巴巴广泛的应用于日志收集、数据监控、广告反馈、量子统计、数据库同步等领域。

开源在：[TaoCode](http://code.taobao.org/p/TimeTunnel/src/)，代码是开源的。

业务保障：

保证为所有的报表按时准备好所需数据，延迟不超过一分钟；

保证为所有的实时应用提供实时数据，延迟不超过一秒钟。

在整个数据仓库中的角色（图中”TT”字样）：

在全量/批量数据导入部分使用的是DHW和DataX，其中DataX貌似是开源的，也是一个不落地的异构数据源传输工具。

而在增量，或者说是流式场景下，这部分数据的采集和导入就依赖了TimeTunnel。

TT对接的Storm和Galaxy流计算模块也简单提一下：

galaxy是一套支持SQL定义业务逻辑的流计算服务化平台

galaxy之于storm，就类似hive之于hadoop的关系

下图为Galaxy的一个架构图：

**组件**

Time Tunnel大概有几部分组成，TTmanager，Client，Router，Zookeeper，Broker。

TTManager: 负责对外提供队列申请、删除、查询和集群的管理接⼝口；对内故障发现，发起队列迁移

Client是一组访问timetunnel的api，主要有三部分组成：安全认证api，发布api，订阅api。目前client支持java，python，php三种语言。

Router:为客户端提供路由信息，找到为消息队列提供服务的Broker。Router是访问timetunnel的门户，主要负责路由、安全认证和负载均衡。Client访问timetunnel的第一步是向Router进行安全认证，如果认证通过，Router根据Client要发布或者订阅的topic对Client进行路由，使Client和正确的Broker建立连接，路由的过程包含负载均衡策略，Router保证让所有的Broker平均地接收Client访问。

Zookeeper是hadoop的开源项目，其主要功能是状态同步，Broker和Client的状态都存储在这里。

Broker是timetunnel的核心，负责消息的存储转发，承担实际的流量，进行消息队列的读写操作。

**其他**

对接数据

 数据库的日志（如mysql、oracle等）

 服务器产生的日志（如apache）

 app通过接口产生的数据

队列资源

 队列是一种资源，TimeTunnel提供队列。

 队列按需申请，订阅队列接口

持续服务

 故障发现，发起迁移

 Broker之间流量均衡

 上下线机器平稳扩容

WhyHBase？

 顺序scan速度快

 很好的扩展性

 强一致性

 高并发写

 底层数据存储基于HDFS

 开源，社区活跃

 国内顶级的运维和开发团队

HBase表设计

 一个Queue对应一个region

 一组Queue(256/512)对应一个表

 Rowkey：queueID + timestamp + seq +brokerID

 数据列族+属性列族

 按天分表，方便删除历史数据

 Pre-Sharding降低compact和split的发生

全文完 :)