

第一章 物联网概述

物联网概念

物联网是一个基于互联网、传统电信网等信息承载体，让所有能够被独立寻址的普通物理对象实现互联互通的网络

物联网层次

- 感知识别层 物理
- 网络构建层 链路
- 管理服务层 网络
- 综合应用层 应用

第二章 自动识别 RFID

RFID

射频识别技术利用射频信号通过空间耦合，实现无接触信息传递并通过所传递的信息达到自动识别的目的

三大组件

- 阅读器
- 天线
- 标签

标签频率

- 125K 133K
- 13.56M
- 433M

标签

- 被动 无源标签
- 主动 有源标签
- 半主动 内带电池

优点

- 体积小 形状多
- 环境适应
- 穿透性
- 重复使用
- 安全性 加密

第三章 无线传感网

无线传感节点组成

- 传感器
- 微处理器
- 通讯芯片
- 功能装置

不符合摩尔定律原因

1. 技术发展不平衡
2. 功耗制约
3. 价格、体积制约

所以设计时需要考虑

1. 低成本 微型化
2. 低功耗
3. 灵活性 拓展性
4. 鲁棒性

第四章 定位系统

GPS组成

- 宇宙空间部分
- 地面监控部分
- 用户设备部分

AGPS

- 辅助GPS

利用基站快速确定当前所处大致位置，利用基站连入网络，查询上空可用卫星

- 优点

GPS定位启动慢，A-GPS可节省搜索卫星的时间

基站定位方法

- 基于距离定位
- 基于距离差定位
- 基于信号特征定位

ToA (Time of Arrival)

- 利用信号发送和到达基站的时间来求位置（要求时钟同步）

TDoA (Time Difference of Arrival)

- 通过信号到达不同基站的时间差建立方程求解位置

AoA (Angle of Arrival)

- 通过天线阵列测量目标与两基站间连线的方位

C00

- 通过一个基站位置大致定位，精准度低

RSS

- 利用信号强度定位

第五章 智能信息设备

参见课程报告

第六章 无线网络

无线网络组成元素

- 无线网络用户
- 无线连接
- 基站

无线网络难点

- 隐藏终端

基站在A,B的信号范围内。但是由于信号衰减，B不在A的信号范围内，因此A无法侦听B是否与基站通讯。A,B可能同时向基站传输数据，造成冲突。

几种无线网络类型

- IEEE 802.11 (Wifi)
- IEEE 802.16 (WiMAX)

不用CSMA/CD 而用CSMA/CA的原因

1. 冲突检测需要全双工的信道
2. 由于无线信号衰减特征和隐藏终端问题，硬件不能监听到全部可能的冲突

消除隐藏终端 怎样实现

- RTS/CTS控制帧

当传输端有数据帧要发送时，它先向接入点发送RTS(Request To Send)帧

当接入点收到传输端的RTS帧后，它等待SIFS后广播一个CTS(Clear To Send)帧作为回应

CTS为传输端提供信道使用权，并防止其他用户在传输端发送数据和接受确认这段时间内进行传输

第七章 无线低速网络

通讯特征

- 红外

距离1m左右，耗电极低

遥控器、摄像头、红外功能手机

- 蓝牙

距离几米至百米，耗电一般

蓝牙耳机、鼠标、键盘、游戏手柄、手环

- ZigBee

几十米到百米，耗电极低

自组织短距离无线通讯网络

低功耗、低成本、时延短、网络容量大、可靠、安全

- WiFi

百米，耗电高

无线节点耗电问题

耗电大的模块

- 无线收发模块

三种状态：发送、侦听、空闲

空闲侦听（没有数据发送接受时的侦听）占据能耗主要部分

解决方法 - 低功率侦听(LPL)协议

- 采样侦听

无线收发模块在无数据发送接收时，用采样的方法获取信道信息
在非采样时间，模块处于空闲状态

- 链路层调度

在节点的每个数据帧中加入采样周期等信息，从而与帧发送者进行同步

第八章 移动通讯网络

参见课程报告

3G通讯技术标准

1. TD-SCDMA

时分-同步码分多址

2. W-CDMA

宽带码分多址

3. CDMA 2000

高通提出，广泛使用

第九章 大数据与海量存储

网络化存储结构

1. 直接附加存储(DAS)

组成：存储设备

将存储系统通过缆线直接与服务器或者工作站相连

对已有服务器的简单拓展（**信息孤岛**）

2. 网络附加存储(NAS)

组成：存储设备 + **服务器**

一种文件级的计算机存储架构

计算机连接到一个仅为其他设备提供数据存储服务的网络

实现了**网络互连**

文件级别的访问 - 共享

3. 存储区域网络(SAN)

组成：存储设备 + 服务器 + **连接设备** (集线器、交换机)

一种通过网络方式连接**存储设备**和**应用服务器**的存储架构

为实现大量原始数据的传输进行了专门的优化

SAN文件系统(对文件的抽象) - 存储共享

数据中心

拥有大量服务器的网络存储实体

组成：N台服务器 + 通信设备 + 能源系统 + 环境控制设备 + 安全设备 + ...

数据中心技术

1. GFS(Google File System)

Google设计的用来处理大规模数据密集型应用的分布式文件系统

2. MapReduce

一种针对超大规模数据集的编程模型和系统

3. BigTable

一种在**海量数据**规模下管理结构化数据的分布式存储系统

Google地球、RSS阅读器

4. Hadoop

Apache的分布式计算**开源**框架

MapReduce原理

1. Map

从输入文件中读取数据集合
执行所需的过滤和转换
以键值对(Key-Value)的形式输出数据集合

2. Reduce

按照用户定义的规则对Map输出的结果进行合并

MapReduce实现

1. 将输入文件分割成16~64MB的文件片段，然后在集群中的多个服务器上开始执行多个程序的副本

程序副本中，一个为master，其余的为worker

2. master选择空闲的worker为其分配map或者reduce任务

总共需要分配 M 个map任务， R 个reduce任务

3. 被分配到map任务的worker i 读取对应的文件片段->解析出键值对->交给**map函数**->将产生的键值对存入**缓存区**

map函数用用户定义

4. 缓存的键值对被**周期性的**写入到**本地磁盘**，并被分为 R 个区域。缓存数据在磁盘中地址返回给master，master将地址发给执行reduce任务的worker

R 个区域分别由 R 个负责reduce的worker处理

5. 执行reduce任务的worker收到地址后，使用远程调用从本地磁盘读取缓存数据。然后将所有读取的数据按照**key（键）排序**，使具有相同key的数据排到一起

6. 对每个唯一的key，执行reduce的worker将对应的数据交给**reduce函数**。函数的输出结果作为此reduce分区的结果添加输出文件中

reduce函数由用户定义

7. 当所有map任务和Reduce任务执行完后，master唤醒用户程序，并向用户返回结果

第十章 数据库系统

1. 导航式数据库

需遍历数据，查找数据缓慢

2. 关系式数据库

数据库逻辑组成与物理存储结构分离

3. 非关系式数据库

弥补关系数据库不足 Redis

关系数据库

运算

- 集合运算

并运算

```
(SELECT R.sid FROM R) UNION (SELECT T.sid FROM T);
```

交运算

```
(SELECT R.sid FROM R) INTERSECT (SELECT T.sid FROM T);
```

差运算

```
(SELECT R.sid FROM R) EXCEPT (SELECT T.sid FROM T);
```

- 关系运算

投影 选择

```
SELECT * FROM R WHERE C;
```

```
SELECT A1,A2,...,Ak FROM R WHERE sid = 3;
```

连接

```
SELECT R.sid, R.a, T.b, T.c, T.date FROM R, T  
WHERE R.sid = T.sid AND T.date > "2016-12-12";
```

- 比较运算

大于 小于 等于 不等于

```
SELECT sid FROM R WHERE sid > 3;
```

- 逻辑运算

与 或非

数据存储模式

- 分布式存储

传感器采集数据后，将原始数据传至存储节点或汇聚点
查询请求分发到网络中，查询在存储节点进行，仅将查询结果返回

- 集中式存储

所有数据存在数据汇聚点
查询在汇聚点进行，不被分发到网络中

分布式存储优缺点

- 优点

用户只对一部分数据感兴趣，查询时可减少不必要的数据传输

- 缺点

节点电量用尽时，数据丢失
网关对数据分布不知情，查询时通信开销大
若部分节点为查询热点，则其电量将快速消耗

集中式存储优缺点

- 优点

所有数据都能被永久存储，不会发生数据丢失
查询时速度快

- 缺点

数据包可能在传输过程中丢失，带来数据不完整
传输数据时，链路上所有传感器能量都会耗损

第十一章 物联网中的信息安全与隐私保护

信息安全

- 可靠性
- 可用性
- 保密性
- 完整性
- 不可抵赖性
- 可控性

RFID隐患

主要攻击方式 没啥意思

Hash-Lock

- 锁定过程

```
[RDR] key = rand(), metaID = hash(key)
[RDR] metaID -> TAG metaID写入标签
[TAG] TAG.locked 标签被锁定
[RDR] (metaID, key) -> Database 将键值对存入数据库
```

- 解锁过程

```
[TAG] (进入RDR信号范围) metaID -> RDR
[RDR] query(metaID) -> Database -> key 查询数据库返回key
[RDR] key -> TAG
[TAG] hash(key) == metaID? TAG.unlock; ID -> RDR: err;
计算hash(key)等于metaID则解锁，发送真实ID，否则认证不成功
```

位置隐私

用户对自己位置信息的掌控能力

k匿名技术

- 位置信息
 - 时间、地点、人物
- 将位置信息中的身份信息替换为匿名代号
- k匿名让第k个代号与其余k-1个代号不可分辨

At last

```
__author__ = "TaQini"
if Q == "填空题":
    print "答案必须一致"
elif Q == "选择题":
    print "都是单选"
elif Q == "简答题":
    print "答到要点即可"
elif Q == "论述题":
    print "要有理有据 多写"
print "祝大家考个好成绩~"
```