第一部分 基础篇

第一章 人机交互概述

1. 什么事人机交互（重要性 相关领域）

Answer: Human-computer Interaction(HCI) ACM给出的定义： 有关交互式计算机系统的设计，评估，实现以及与之相关现象的学科。

重要性：a. 市场角度（1.用户期望简单易用的系统 2.对设计低劣系统的容忍度越来越差）

b. 企业角度（1.提高员工的生产效率 2.降低产品的开发成本 3.降低产品的后续支持成本）

c.用户角度（1.获得较高的主观满意度 2.减少时间，金钱，生命损失）

相关领域：心理学，人机工程学，社会学，计算机科学，工程学，商务知识，图形设计，家算计支持的协作，人性因素，电影产业，工业设计，产品设计，科技写作）

1. 人机交互的发展历史（旧的交互形式作为特例保存下来）

A：1.新的界面变革包含了上一代界面（作为一种特例）

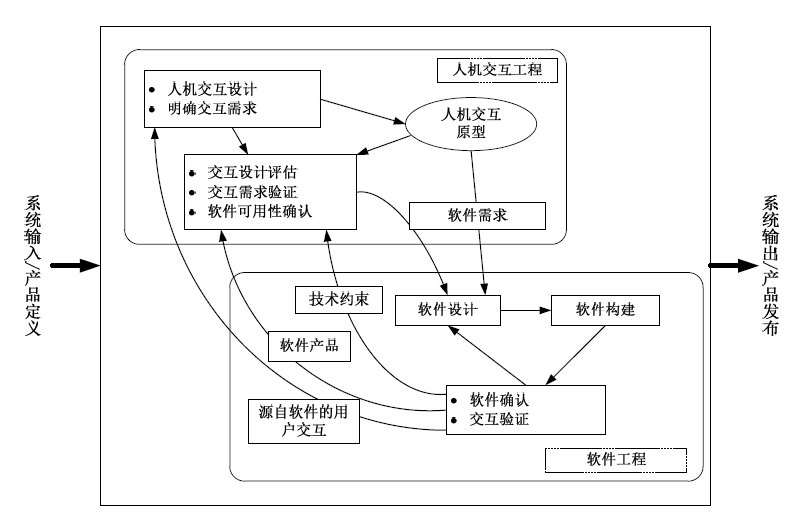
2.旧的交互方式仍有其存在的必要性（以前的用户从未消失）

3. 学习目的（利用原有技术实现新的交互手段）

1. 交互设计与软件工程（1.前者是对后者的促进和补充 2. 二者结合存在许多困难）

A： HCI对SE的促进作用（传统SE方法在实现交互式系统方面的缺陷）：没有提出明确地对用户界面及可用性需求进行描述的方法；不能够在系统开发过程进行中对用户界面进行终端测试；具有完善的系统功能 产品的可用性，有效性以及满意度并不高

二者在系统工程中的关系：



两者结合的困难：

1. 价值观不同 ：SE 实施策略和方法选择上常有一定的倾向性

HCI包含较多的主观性和灵活性

方法论存在差异：SE 形式化分析方法

HCI 非形式化方法

第二章 人机交互基础知识

1. 交互框架（执行评估/活动周期EEC）

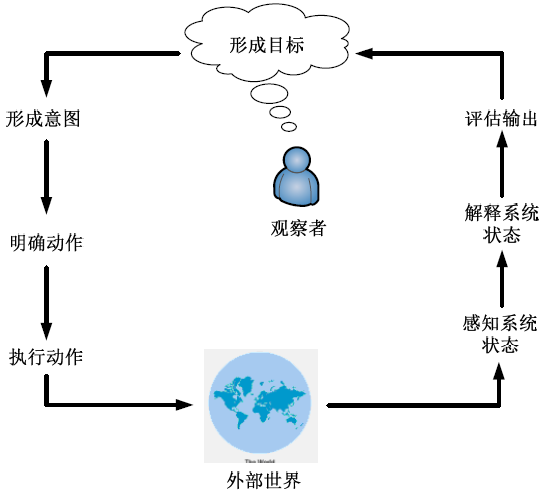
作用

* 1. 提供理解或定义某种事物的一种结构
  2. 能够帮助人们结构化设计过程
  3. 认识设计过程中的主要问题
  4. 还有助于定义问题所涉及的领域

执行/评估活动周期 EEC

* 1. 最有影响力的框架
  2. 定义了活动的四个组成部分
     1. 目标(Goal) ≠意图(Intention)
     2. 执行(Execution)
     3. 客观因素(World)
     4. 评估(Evaluation)

EEC模型：

* 从用户视角探讨人机界面问题
* 共有七个阶段
  + 1-4：执行阶段
  + 5-7：评估阶段
* 每个循环代表一个动作
* 夜晚看书的例子

1. 交互形式（直接操纵、隐喻、问答）

直接操纵：

Ben Shneiderman，1982

* 1. 用户通过在可视化对象上面进行某些操作来达到执行任务的目的
  2. 展现了真实世界的一种扩展
  3. 对象和操作一直可见
  4. 迅速且伴有直观的显示结果的增量操作
  5. 增量操作可以方便地逆转

直接操纵的三个阶段：

自由阶段——指用户执行操作前的屏幕视图；

捕获阶段——在用户动作（点击、点击拖拽等）执行过程中屏幕的显示情况；

终止阶段——用户动作执行后屏幕的显示情况。

优点

* + 将任务概念可视化，用户可以非常方便地辨别他们；
  + 容易学习，适合新手用户；
  + 基于识别，对记忆的要求不高，可减少错误发生；
  + 支持空间线索，鼓励用户对界面进行探索；
  + 可实现对用户操作的快速反馈，具有较高的用户主观满意度。

缺点

* + 实现起来比较困难；
  + 对专家用户而言效率不高；
  + 不适合小屏幕显示设备；
  + 对图形显示性能的需求较高；
  + 不具备自解释性，可能误导用户

问答界面：通过询问用户一系列问题实现人与计算机的交互 (Web问卷是典型的采用问答方式进行组织的应用 ;应允许用户方便地取消其中一个界面的选项)

优点

* + 对记忆的要求较低；
  + 每个界面具有自解释性；
  + 将任务流程以简单的线性表示；
  + 适合新手用户。

缺点

* + 要求从用户端获得有效输入；
  + 要求用户熟悉界面控制；
  + 纠错过程可能比较乏味。

隐喻界面：

本质

* + 在用户已有知识的基础上建立一组新的知识，实现界面视觉提示和系统功能之间的知觉联系，进而帮助用户从新手用户转变为专家用户

优点

* + 直观生动
  + 无需学习

局限性

* + 不具有可扩展性
  + 不同用户对同一事物可能产生不同的联想
  + 紧紧地将我们的理念和物理世界束缚在一起
  + 寻找恰当的隐喻可能存在困难

Eg 回收站

与直接操纵比较： 直接操纵 相对动态 操作对象常以隐喻存在

隐喻：相对静态

1. 信息处理模型(人类处理机模型，格式塔心理学,记忆特性)

人类处理机：

最著名的信息处理模型

Card等，1983

包含三个交互式组件

感知处理器

* + 信息将被输出到声音存储和视觉存储区域

认知处理器

* + 输入将被输出到工作记忆

动作处理器

* + 执行动作

格式塔心理学：研究人是如何感知一个良好组织的模式的，而不是将其视为一系列相互独立的部分 ，

相近性原则、相似性原则、连续性原则、完整性和闭合性原则

认知特性:

感觉记忆

* + 又称瞬时记忆
  + 在人脑中持续约为1秒钟
  + 帮助我们把相继出现的一组图片组合成一个连续的图像序列，产生动态的影像信息

短时记忆

* + 感觉记忆经编码后形成
  + 又称工作记忆，约保持30秒
  + 储存的是当前正在使用的信息，是信息加工系统的核心，可理解为计算机的内存
  + 短时记忆的存储能力约为7±2个信息单元

短时记忆->长时记忆

* + 短时记忆中的信息经进一步加工后会变为长时记忆
  + 只有与长时记忆区的信息具有某种联系的新信息才能够进入长时记忆

长时记忆的信息容量几乎是无限的

启发

* + 注意使用线索来引导用户完成特定任务
  + 在追求独特的创新设计时也应注重结合优秀的交互范型

1. 交互设备

文本输入设备（键盘；和弦键盘；投影键盘；手写输入；语音输入；光学字符识别）

定位设备（WIMP;鼠标;触摸板；指点杆;触摸屏;光笔）

图像输入设备(扫描仪;数码相机;传真机)

显示设备(光栅扫描阴极射线管;液晶显示器;等离子体显示器;电子墨水;发光二极管;点字显示器)

虚拟环境下的交互设备

第三章 交互设计目标与原则

1. 可用性目标（5个属性）

易学性；易记性；少出错；高效率；主观满意度

1. 用户体验目标

问题

* 1. 随着新技术渗透到人们的日常生活中，人们对产品有了更多的要求
  2. 到底什么样的产品才是用户愿意使用和购买的？
     1. 让用户感到满意并留下愉快主观感受的产品更可能被多次使用

什么是用户体验

* 1. 用户在与系统交互时的感觉
  2. 为儿童创建的网站应该要有趣并且引人入胜
  3. 面向年轻人的网站则应该更注重时尚感和趣味性
  4. 较可用性目标主观
  5. 可用性可能对用户体验带来阻碍

1. 简易可用性工程（可用性度量,四种主要技术）

可用性度量

常用方法

* + 选择一些能够代表目标用户群体的测试用户
  + 让这些用户使用系统执行一组预定的任务
  + 比较任务的执行情况
  + 针对多维属性
    - 取每个可用性属性的平均值
    - 查看整体分布情况

主观满意度度量举例

* + 在1~5分的5分制情况下平均值至少为4
  + 或至少50%的用户给系统打5分
    - 或给系统打1分的用户不超过5%

四种主要技术：

* + 用户和任务观察
  + 场景（scenario）
  + 简化的边做边说（thinking aloud）
  + 启发式评估

1. 交互设计原则(启发式原则,黄金规则,七项原理)

黄金规则：

1. 尽可能保证一致

2. 符合普遍可用性

3. 提供信息丰富的反馈

4. 设计说明对话框以生成结束信息

5. 预防并处理错误

6. 让操作容易撤销

7. 支持内部控制点

8. 减轻短时记忆负担

七项原理：

1. 应用现实世界和头脑中的知识
2. 简化任务结构
3. 使事情变得明显
4. 获得正确的映射
5. 利用自然和人为的限制力量
6. 容错设计
7. 当所有都不成功时进行标准化

第四章 交互设计过程

1. 设计过程的基本活动和特征

1.1基本活动：

标识用户需要并建立需求

* + 必须了解谁是目标用户
  + 交互式产品应提供哪些支持
  + 最基本的

开发满足需求的候选设计方案

* + 设计的核心活动
  + 概念设计和物理设计

构建设计的交互式版本

* + 评价设计的最佳方法就是让用户与产品交互
  + 不一定是可运行的软件版本

评估设计

* + 评估它的可用性和可接受性
  + 制定各种评估标准

1.2特征：

以用户为中心

* + 人机交互领域的一个核心观点

稳定的可用性标准

* + 有助于设计人员选择不同的候选方案
  + 并在产品开发过程中随时检查

迭代

* + 设计人员不可能一次就找出正确的解决方案
  + 利用反馈来改进设计

1. 主要问题
   * 用户、需求、候选方案、决策

2.1 用户的选取

* + 用户：直接与产品交互、以期完成某个任务的人？
  + 涉众（stakeholders）：被系统影响的、而且对系统需求有直接或间接影响的个人或机构
    - 举例：电子日历
    - 你，与你约会的人；你要记住生日的亲友；甚至是生产记事本的公司
  + 不建议让所有的当事人都参与开发
    - 应该了解产品会造成多大范围的冲击
    - 明确让什么人在何种程度上参与开发

2.2需求的确定

* + 错误：简单地问：“你需要什么” ？
  + 正确：必须理解用户的特征和能力，了解他们想要达到什么目标，以及可满足的支持
    - 用户的能力、特征将影响产品的设计
    - 新产品的用户和有代表性的任务较难确定
  + 现状
    - 开发人员往往倾向于创建自己想要的产品或者是过去开发过的类似产品
  + 当前或过去的行为是对未来行为的很好启示

2.3候选设计方案

* + 个别设计人员的才干、创造力 ？
    - 有时候是这样
  + 正解：候选方案来自考虑其他相似的设计
    - 竞争对手的产品
    - 类似系统的早期版本
    - 甚至是完全不同的东西
    - “所谓‘专家’，指的是那些能够从先前的经验中找到正确灵感并应用于当前工作的人”
  + 注意：不应局限于已有系统的限制

2.4最终方案的决策

* + 用户与产品的交互方式是设计的推动力
    - 把注意力集中在可见、可测量的外部行为上
    - 例如，检索数据库（或网页）时，响应时间是4秒
    - 产品内部的工作详情只有在牵涉外部行为时才是重要的
  + 正解（以用户为中心）
    - 让用户和当事人与各种方案相交互
      * 听取他们的体验、偏好和改进建议等
    - 制定明确、无歧义的质量基准
      * 不明确：系统应能够快速响应
      * 明确：系统响应时间不应超过1秒

1. 交互设计生命周期

3.1星型模型

3.1.1分析模式

自顶向下、组织化、判定和正式化，它是从系统到用户的方法

3.1.2合成模式

自底向上、自由思考、创造性，这是由用户至系统的方法

3.1.3特点

没有指定任何活动次序

“评估”是这个模型的核心

源于开发人员的实际经验

3.2可用性工程生命周期模型

Deborah Mayhew，1999

3.2.1体现了可用性工程的总体概念

3.2.2详细描述了如何执行可用性任务  
3.2.3说明了如何把可用性任务集成到传统的软件开发生命周期中

第二部分 设计篇

第五章 交互式系统的需求

1. 产品特性
   1. 功能不同

智能冰箱：应能够提示黄油已用完

字处理器：系统应支持多种格式

* 1. 物理条件不同

移动设备运行的系统应尽可能小，屏幕显示限制

* 1. 使用环境不同

物理环境：如操作环境中的采光、噪音和尘土状况

社会环境：是否要共享数据，同步还是异步？

组织环境：用户支持的质量、响应速度如何？是否提供培训资源或设施？

技术环境：产品应能运行于何种平台上？应与何种技术兼容？

1. 用户特性

2.1心理学原理部分，假设每个人都有相似的能力和局限性

合理的，心理学原理可以适用于大多数人

2.2交互产品设计人员应该意识到个性的差异

用户并不是完全相同的

在设计中尽可能地体现这些差异

2.3用户差异

体验水平

年龄

文化

健康

1. 用户建模
   1. 人物角色

人物角色（Personas）

不是真实的人

是基于观察到的那些真实人的行为和动机，并且在整个设计过程中代表真实的人

是在人口统计学调查收集到的实际用户的行为数据的基础上形成的综合原型

概念简单，但使用起来相当复杂

拼凑出几个用户档案是不行的

1. 需求获取、分析和验证
   1. 场景和观察

场景

1. 场景是表示任务和工作结构的“非正式的叙述性描述”

以叙述的方式描述人的行为或任务，从中可以发掘出任务的上下文环境、用户的需要、需求

形式可以类似于一篇故事、一个小品或者在给定环境下按照时间顺序的一段情节

1. “讲故事”是人们解释自己做什么或者希望执行某个任务的最自然方式

故事的焦点就是用户希望达到的目标

若场景说明不断提到某个特定形式、行为或者地点，就表明它是这个活动的核心内容

1. 来源

场景说明通常来自专题讨论或者访谈，目的是解释或讨论有关用户目标的一些问题

观察

1. 设计的最初，可能不知道问什么问题或由谁来回答这些问题
   * 没有与参与者实际的讨论和观察是不可能得到完整的工作流程画面的
   * 有用的信息可以通过观察人们在工作环境中完成他们的活动来获得
2. 直接观察
   * 陪同他们工作而直接获得信息
   * 可能影响被观察者的日常活动
     + 可提问：这是你通常完成任务的方式吗？
3. 间接观察
   * 用视频/录音获得信息
   * 观察者更舒适
   1. HTA
      1. 文字和图形描述

层次化任务分析（HTA）是应用最广的任务分析技术

把任务分解为若干子任务，再把子任务进一步分解为更细致的子任务。之后，把他们组织成一个“执行次序”，说明在实际情形下如何执行各项任务

执行次序0：执行1-3-4；若图书不在期望的书架上，则执行2-3-4。

执行次序2：执行2.1-2.4-2.5；若未查到此书，则执行2.2-2.3-2.4-2.5

编号说明了对应步骤编号。

* 0.借书
  + 1.前往图书馆
  + 2.检索需要的图书
    - 2.1 访问图书馆目录
    - 2.2 使用检索屏
    - 2.3 输入检索准则
    - 2.4 找出需要的图书
    - 2.5 记录图书位置
  + 3.找到书架并取书
  + 4.到柜台办理借阅手续

HTA的图形描述

* 1. 原型

4.3.1 原型的重要性

用户往往不能准确描述自己的需要

用户在看到或尝试某些事物后，就能立即知道自己不需要什么

4.3.2 原型

在某一方面和真正产品比较接近、以便人们能对这一方面的各种技术方案进行不断评估和改进的一种接近于实际产品的模型

如房屋、桥梁的缩微模型

借助于原型，当事人就能与未来的产品交互，从中获得一些实际的使用体验，并发掘新思路

低保真原型

* + 与最终产品不太相似的原型
  + 使用与最终产品不同的材料，如纸张、纸板

如PalmPilot掌上电脑的木雕原型

* + 优点是简单、便宜、易于制作和修改

高保真原型

* + 与最终产品更为接近，使用相同的材料

如使用Visual Basic开发的软件系统原型

* + 风险

用户会认为原型就是系统

开发人员可能认为已找到了一个用户满意的设计

第六章 交互式系统的设计

1. 设计框架（定义高层次上的屏幕布局；定义产品的工作流、行为和组织 ）
   1. 定义外形因素和输入方法
   2. 定义功能和数据元素
   3. 决定功能组合层次
   4. 勾画大致的设计框架
   5. 构建关键情景场景剧本
   6. 通过验证性的场景剧本来检查设计
2. 简化交互设计策略
   * 删除、组织、隐藏、转移
3. 设计中的折中
4. 细节
   * 让软件体贴
   * 加快响应时间
   * 减轻记忆负担
   * 减少等待感
   * 设计模式（金字塔模型；序列地图模型；面包屑层级结构）

第七章 可视化设计

1. 窗口和菜单
   * 菜单的标准

所有窗口必备的基本组件

菜单栏是代表下拉式菜单的菜单

菜单选项的标签、位置、归类等均已标准化

* + - Q：标准菜单选项有哪些？它们的摆放位置是怎样的？
    - 从左向右：文件、编辑、窗口和帮助，帮助位于窗口的最右边
    - 可选菜单常出现在编辑和窗口菜单之间
  + 菜单的目的

1. 对话框
   1. 类型和用途

类型

模态对话框 vs. 非模态对话框

模态对话框

冻结了它属于的应用，禁止用户做其他操作，直到处理了对话框中出现的问题

可以切换到其他程序进行操作

用户最容易理解，操作非常清晰

* “应用模态”
  + 只停止其所属的应用程序
* “系统模态”
  + 使系统中的所有程序都停止

大多数情况下，应用程序不应该有系统模态对话框

非模态对话框

* + 打开后无须停止进度，应用程序也不会冻结
  + 由于其操作范围不确定而难以使用和理解
  + 举例
    - Word的查找和替换对话框
    - 画图程序
      * 可以在主窗口和非模态对话框之间拖动对象
  + 存在的问题
    - 缺乏一致的终止命令，如取消、应用、关闭等

用途：属性对话框；功能对话框；进度对话框；公告对话框；特殊对话框

* 1. 如何管理

标签对话框；扩展对话框；级联对话框

1. 控件
2. 工具栏
   * 工具栏图标

工具栏上的图标与文本

文本标签精确，但阅读与识别速度较慢

工具栏和菜单的用途不同

* + - 工具栏主要为常用功能提供快速访问
    - 图形的表意特征较文本更适合担当这种角色

图标图像

* + - 找代表事物的图像比寻找代表动作或关系的图像容易得多，如垃圾桶、打印机等

Q:为什么不能同时使用文本和图像？

* + - 屏幕空间非常宝贵
    - 解决：工具提示

第八章 交互设计模型与理论

1. 预测模型
   * GOMS

最著名的预测模型

1983年由Card, Morgan和Newell提出

基于人类处理机模型

泛指整个GOMS模型体系

是关于人类如何执行认知—动作型任务以及如何与系统交互的理论模型

采用“分而治之”的思想，将一个任务进行多层次的细化

把每个操作的时间相加就可以得到一项任务的时间

* + - 操作指用户的目光从屏幕的一处移到另一处、识别出某个图标、手移到鼠标上
  + KLM

Card等1983

对用户执行情况进行量化预测

仅涉及任务性能的一个方面：时间

用途

预测无错误情况下专家用户在下列输入前提下完成任务的时间

便于比较不同系统

确定何种方案能最有效地支持特定任务

* + Fitts

能够预测使用某种定位设备指向某个目标的时间

人机交互中，根据目标大小及至目标的距离，计算指向该目标的时间

可指导设计人员设计按钮的位置、大小和密集程度

对图形用户界面设计有明显的意义

“最健壮并被广泛采用的人类运动模型之一”

大目标、小距离具有优势

* + 对选择任务而言，其移动时间随到目标距离的增加而增加，随目标的大小减小而增加

屏幕元素应该尽可能多的占据屏幕空间

最好的像素是光标所处的像素

屏幕元素应尽可能利用屏幕边缘的优势

大菜单，如饼型菜单，比其他类型的菜单使用简单

* + 各自的适用场合

1. 状态转移网络

用于描述用户和系统之间的对话

* + 自20世纪60年代后期被使用
  + 可被用于探讨菜单、图标和工具条等屏幕元素，还可以展示对外围设备的操作
  + 适合表达顺序操作和循环操作

状态转移图

* + 最常用的状态转移网的形式
  + 有向图
  + 图中的结点表示系统的各种状态
  + 图中的边表示状态之间可能的转移

状态之间通过转移（用带方向箭头的线段表示）互相连接

转移被事件（转移线段上的标记）触发

伪状态——初始状态和终止状态

* + - 是STNs的起始和终止
    - 可以与系统的其他部分相连接。



1. 语言模型
   * BNF表达式
2. 系统模型
   * Z标记法

4.1．基于集合和函数

最简单的集合对应于编程语言中的标准类型

如实数R、整数Z和自然数N等

4.2．非标准类型定义举例

显式列出集合中有限的可能值就可以定义新集合

举例：包含所有图形包中使用的集合形状(线、椭圆和矩形)的集合

* + - *Shape\_type* ::= *Line* | *Ellipse* | *Rectangle*

举例：一个所有可能按键的集合

* + - *Keystroke* ::= *a* | *b* |…| z | A | … | 9 | *Cursor\_left…*

第九章 以用户为中心的设计

1. 用户参与设计
   1. 重要性

项目初期调研对用户的了解程度仍不足以解决设计过程中出现的所有问题

选择哪些用户？

* + 将来真正使用系统的人，而非经理或工会代表

为什么让用户参与设计？

* + 期望管理
    - 保证产品不会出乎用户的意料 ，避免失望
    - 对用户进行培训也有助于改进期望管理
  + 拥有权
    - 人们总渴望自己的意见得到重视
    - 更容易接受有”拥有权”的最终产品
  1. 形式  
     作为设计组成员

全职或兼职参与整个项目或部分项目开发

优点：用户能透彻理解系统及其原理

缺点：用户会逐渐变得不具有普通用户的代表性

对大型项目应注意用户更新

以邮件、专题讨论或类似会议的形式参与

定期接收有关项目进展的信息

存在大量的用户时的折衷方案

各个用户组派代表以全职的形式加入设计组

其他人员通过专题讨论等形式参与

* 1. 上下文询问法

观察并与用户交流会比仅仅观察的效果要好

强调到用户工作的地方，在用户工作时观察，并和用户讨论他的工作

基于“学徒模型”

用户是师傅，研究人员是学徒

与观察法的区别

用户知道研究人员的存在

也知道他们是研究的一部分

四个原则：上下文环境；伙伴关系；解释；焦点

1. 正确看待“以用户为中心”

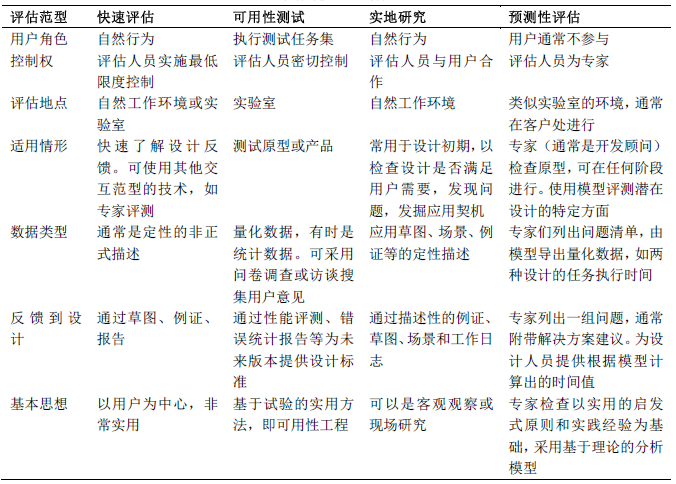
ACD思想是对UCD思想的一种反思

* + 早期的设计是以技术为中心
  + 直到出现以人为中心，这是一次飞跃
  + ACD把人与技术综合起来进行考虑，不单纯考虑人或者技术，而是关注事情本身的活动目标
  + ACD同样需要对用户进行研究或调

第三部分 评估篇

第十章 评估的基础知识

1. 评估范型和方法
   * 快速评估、可用性测试、实地研究、预测性评估



1. 评估方法的选择
   * 不同方法的适用阶段

评估在周期中的位置

设计早期阶段的评估更快速、便宜

评估的形式

实验室环境or工作环境

技术的主客观程度

技术越主观，受评估人员知识的影响越大

* + - 如认知走查等

测量的类型

与技术的主客观性有关

* + - 主观技术：定性数据
    - 客观技术：定量数据
  + 评估方法的组合

评估方法的组合取决于项目待评估的具体特性

常用组合

启发式评估+边做边说等用户测试技术

* + - 专家可通过启发性评估排除显而易见的可用性问题
    - 重新设计后，经用户测试，反复检查设计的效果

访谈+问卷调查

* + - 先对小部分用户进行访谈，确定问卷中的具体问题

启发式评估vs.用户测试

前者不需要用户参与

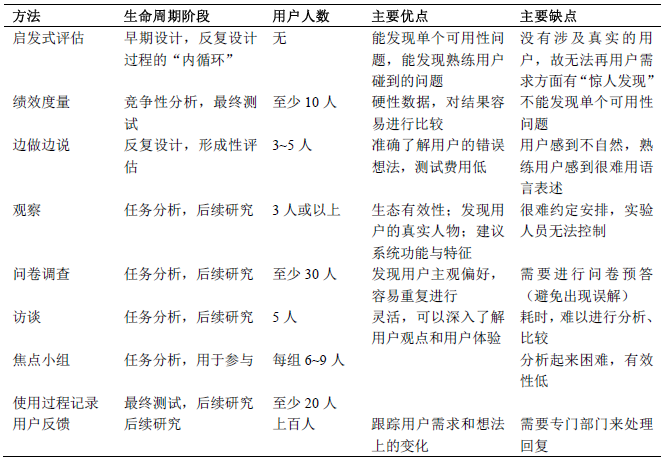
二者发现的可用性问题不同，可以互补

1. DECIDE评估框架

六个步骤

* + 决定评估需要完成的总体目标
  + 发掘需要回答的具体问题
  + 选择用于回答具体问题的评估范型和技术
  + 标识必须解决的实际问题，如测试用户的选择
  + 决定如何处理有关道德的问题
  + 评估解释并表示数据

1. 小规模试验的重要性
   1. 对评估计划进行小范围测试
   * 以确保评估计划的可行性
   * 如检查设备及使用说明
   * 练习访谈技巧
   * 检查问卷中的问题是否明确
   1. 小规模试验可进行多次
   * 类似迭代设计
   * 测试——反馈——修改——再测试
   * 快速、成本低



第十一章 评估之观察用户

1. 观察方式
   * 实验室vs实地
2. 观察框架
   * 减少干扰、完成目标
3. 数据记录方式
   * 纸笔
   * 音视频
   * 日志

第十二章 评估之询问用户和专家

1. 访谈
   * 分类和技巧

分类：非结构化访谈；结构化访谈；半结构化访谈；集体访谈

* + 焦点小组

1. 问卷调查
   * 问卷设计与组织
2. 问卷调查是用于搜集统计数据和用户意见的常用方法

可单独使用

也可与其他技术结合使用

1. 问卷设计原则

应确保问题明确，具体

在可能时，采用封闭式问题并提供充分的答案选项

对于征求用户意见的问题，应提供一个“无看法”的答案选项

注意提问次序，先提出一般化问题，再提出具体问题

1. 认知走查
2. 启发式评估
   * 评估原则

复习：Nielsen的十条启发式规则

系统状态的可视性

系统应与真实世界相符合

用户的控制权及自主权

一致性和标准化

帮助用户识别、诊断和修复错误

预防错误

依赖识别而非记忆

使用的灵活性及有效性

最小化设计

帮助及文档

* + 评估步骤

|  |  |
| --- | --- |
| **阶段** | **步骤** |
| 准备（项目指导） | a. 确定可用性准则；  b. 确定由3-5个可用性专家组成的评估组；  c. 计划地点、日期和每个可用性专家评估的时间；  d. 准备或收集材料，让评估者熟悉系统的目标和用户。将用户分析、系统规格说明、用户任务和用例情景等材料分发给评估者；  e. 设定评估和记录的策略。是基于个人，还是小组来评估系统？指派一个共同的记录员还是每个人自己记录？ |
| 评估（评估者活动） | a. 尝试并建立对系统概况的感知；  b. 温习提供的材料以熟悉系统设计。按评估者认为完成用户任务时所需的操作进行实际操作；  c. 发现并列出系统中违背可用性原则之处。列出评估注意到的所有问题，包括可能重复之处。确保清楚地描述发现了什么？在何处发现？ |
| 结果分析（组内活动） | a. 回顾每个评估者记录的每个问题。确保每个问题能让所有评估者理解；  b. 建立一个亲和图（affinity diagram，又称KJ法、A型图解法），把相似的问题分组；  c. 根据定义的准则评估并判定每个问题；  d. 基于对用户的影响，判断每组问题的严重程度；  e. 确定解决问题的建议，确保每个建议基于评估准则和设计原则。 |
| 报告汇总 | a. 汇总评估组会议的结果。每个问题有一个严重性等级，可用性观点的解释和修改建议；  b. 用一个容易阅读和理解的报告格式，列出所有出处、目标、技术、过程和发现。评估者可根据评估原则（heuristics）来组织发现的问题。一定要记录系统或界面的正面特性；  c. 确保报告包括了向项目组指导反馈的机制，以了解开发团队是如何使用这些信息的；  d. 让项目组的另一个成员审查报告，并由项目领导审定。 |

第十三章 评估之用户测试

1. 复习DECIDE评估框架
   * 定义目标和问题
   * 选择参与者
   * 设计测试任务
   * 明确测试步骤
   * 数据搜集与分析