UI Hall of Fame or Shame
lock Holmes was able, by winding up the dead man's watch, to prove that it had been wound up two hours before, and that therefore the deceased had gone to bed within that time—a deduction which was of the greatest importance in clearing up the case. All these I may sketch out at some future date, but none of them present such singular features as the strange train of circumstances which I have now taken up my pen to describe.

It was in the latter days of September, and the equinoctial gales had set in with exceptional violence. All day the wind had screamed and the rain had beaten against the windows, so that even here in the heart of great, handmade London we were forced to raise our minds for the instant from the routine of life and to recognise the presence of those great elemental forces which shriek at mankind through the bars of his civilisation, like untamed beasts in a cage. As evening drew in, the storm grew higher and louder, and the wind cried and sobbed like a child in the chimney. Sherlock Holmes sat moodyly at one side of the fireplace cross-indexing his records of crime, while I at the other was deep in one of Clark Russell's fine sea-stories until 101/294.
User Centered Design for Technology in the Developing World

6.831 Special Lecture - Clayton Sims
Technology in Developing Nations

• 85% of the world’s population lives in a developing* nation

• The ICTD Research Field is Dedicated to the Creation of Technology in these nations
  – OpenMRS
  – OLPC Project

*UN HDI Index < .9
The Wrong Question

• How is User Centered Design Fundamentally Different in the Developing World?
  — Well, it Isn’t.

• Principles are the Same
• Avoid Temptation to Ditch a Disciplined Approach
A Better Question

• How is User Centered Design *Practically* Different in the Developing World?
  – Radically Different User Populations
  – Resource Limitations
  – Different Expectations for Adoption
Challenges

• Can’t Rely on Prior Knowledge
  – Computer exposure is almost always limited

• You are not the User
  – Seriously this time
  – Neither are your friends
Challenges

• User Population Traits are Unique
  – Learning about one population won’t translate to learning about another

• Limited Access to Users
  – Often can’t schedule weekly formative testing
Advantages

• Old Paradigms Die Hard
  – Design isn’t weighed down by convention
  – We’re slow to adopt potentially better interfaces

• Barriers to Adoption are Different
  – Less existing infrastructure to integrate with
Advantages

• Scale of Impact
  – Good systems make a difference
• It’s Just More Fun
  – To learn how wrong you are
User Centered Design for ICTD

• Examining User Populations
• Reviewing Design Techniques
• Thinking about ‘The Interface’
• Avoiding Pitfalls
User Populations

• Avoid Generalizations
  – I can’t tell you about your users in particular

• Some Common Traits
  – Varying literacy levels
  – Linguistic Concerns
  – Little or no Exposure to Computers
  – Differences in Usability Priorities
Limited Literacy

• Avoid Excessively Verbose Labeling
• Consistency in Visual Displays is Valuable
• Alternative Display Modalities
  – Speech Systems
  – Iconographic UI
Language Concerns

• Limited Geographies Still Have Multiple Languages
  – There are 29 Different Languages in India with more than 1 Million Native Speakers
  – Always Design Systems with internationalized strings inherent
Limited Exposure to Computers

• Trajectories Should be Clear and Narrow
  – It should be clear what actions can be taken
  – Actions should be grouped visually
  – Tend towards avoiding “overloading” screens

• Focus on Imperative Language
  – “View Orders” v. “Orders” as a selection
Limited Exposure to Computers

• Beware of Seemingly Natural Groupings
  – Login Screens
  – High Level “Settings” Explorers
  – Popup Menus
Limited Exposure to Computers

• Be Flexible on Inputs
  – Especially with newer systems
  – If a lot of your users seem to something it should work, maybe it should
  – Redundancy is preferable
Difference in Priorities

• Users are more tolerant of lessened efficiency
  – Although there’s a bias in that notion
• Discoverability and Learnability are extremely important
  – Users are more hesitant to explore and find features
Difference in Priorities

• Error Recovery and Prevention
  – Users familiar to computers expect errors and bugs
  – Users with limited exposure to computers often find it extremely hard to recover from errors
  – This problem tends to alleviate with time
User Centered Design for ICTD

• Examining User Populations
• **Reviewing Design Techniques**
• Thinking about ‘The Interface’
• Avoiding Pitfalls
Task and Domain Analysis

• Computerized Tasks are Often Based on Real World Analogs
  – This consistency is valuable
  – But, don’t get caught up trying to emulate existing systems
  – Task and Domain Analysis doesn’t stop with replicating an existing process
User Inclusion

• Participatory Design and Formative Evaluation are Vital
  – Ongoing evaluation is a necessity
  – Participatory Design often generates unexpected and useful information and feedback
Feedback

• Users are hesitant to provide negative feedback
  – Make Alternatives Clear
  – Elicit Negative Opinions Explicitly and With Groups
  – Observe and Query

• Especially for Professional Users
  – Looking to be successful with existing system, not ‘complain’ about it
Prototyping

• Avoid Computer Prototypes as Long as Possible
  – Many confounding variables to any Computer Prototype if users aren’t comfortable with computers

• Establish the Idea of Non-Permanence
  – “Once it’s on the Screen, it’s finished”
Prototyping

• Environment Can be Hard to Control
  – Lighting Conditions
  – Weather
  – Space Concerns

• Better to be Flexible than Exact
Prototyping

• Maximize Team
  – Guide
  – Communicator
  – Emulator
  – Multiple Observers

• Prototype on Groups of Users
  – More willing to explore
  – More willing to explain
Prototyping

• Users Might need More Guidance
  – Exploring might not be intuitive
  – Try to provide as much up front as possible

• Have a Clear Goal for Prototyping
  – High Noise Level in Data
  – Formative evaluation is also useful
User Centered Design for ICTD

• Examining User Populations
• Reviewing Design Techniques
• Thinking about ‘The Interface’
• Avoiding Pitfalls
What is ‘The Interface’

• Everything
  – Operating System
  – Computer Power Button
  – Keyboard/Mouse

• You’re on the hook for all of them
  – Not just for learnability/understanding
  – Environmental Factors can make your systems unusable
What is ‘The Interface’

• Extraneous Layers to Interface are Damaging
  – Confusing for new users
  – Damaging to Efficiency and Error Prevention

• Lock it Up
  – Controlled Sandbox
  – Provide Easy Reset to a Central Location
  – How to Do This should be a concern early in the design process
What is ‘The Interface’

• All-in-one systems are the most desirable
  – Laptops have become a much more popular choice for deploying technology than desktops
  – Custom all-in-ones (OLPC) are a far-extreme of this solution
Touch Screens

• Extraordinarily Intuitive
  – Viewport and Input all in one place
• Very good for larger systems
• Simplifies Input Needs
  – No mouse
  – No keyboard (if you can avoid it)
Touch Screens

• Concerns
  – Even Further Extracts External Components
  – Touchscreens software environments have a lot of extraneous inputs possible
  – Avoid flash and hidden input modes and features, that based on pre-existing mental models that experienced computer users have
Voice Interfaces

• Voice Interfaces Greatly Increase Accessibility
  – Literacy is far less important
  – Voice Interfaces are linear and imperative by default

• Scale well in Breadth, but not Depth
  – Many users isn’t much harder
  – More complicated tasks make system polynomially harder to use
Voice Interfaces

• Pseudo-Human Interfaces are Tricky
  – Need to filter extraneous input
  – Potentially unrealistic expectations of capability
  – Machine learning with every dialect/accent/language probably isn’t realistic
Voice Interfaces

• Set Realistic Expectations
  – System should be forthcoming about inadequacies
  – Computerized voices are less subjectively satisfying, but have less errors
  – Keypads perform better in experiments
Cell Phones

• Cellular Phones are the Most Ubiquitous Form of Technology in the Developing World
  – If users have been exposed to computing, odds are it is with a cell phone
  – Many users are familiar with high level interface
    • Keypad
    • Screen
    • Some similarity of software
Cell Phones

• Leverage Consistency
  – Emulate Phone UI whenever possible
Cell Phones

• Challenges
  – Extremely difficult to sandbox
  – UI Capabilities are Extremely Limited
  – Hard to find a reasonable platform
  – Technology changes fast, your platform/app could be obsolete in 6 months
User Centered Design for ICTD

• Examining User Populations
• Reviewing Design Techniques
• Thinking about ‘The Interface’
• Avoiding Pitfalls
Avoiding Pitfalls

• Making every interaction a single decision
  – There is a place for guiding the user, it is not every screen
  – Wizards are a dangerously tempting paradigm
Avoiding Pitfalls

• Ignoring Efficiency and Expert Users
  – Lack of expertise is limited only by time and exposure
  – If your system is good, eventually it will have expert users and you still want it to be usable
Avoiding Pitfalls

• Ignoring User Roles and Gradations
  – Many features of a system don’t need to be available to every user
  – If some users are comfortable with computers and others aren’t, establish what tasks are simply out of scope
Avoiding Pitfalls

• Biting off more than you can chew
  – Iterative design and testing can show early on what features are impractical
Avoiding Pitfalls

• Ignoring What Real Users Need
  – An Awesome UI isn’t worth much if it isn’t solving a real problem