Information Gathering
(Requirements Elicitation)

• Remember: distinguish “requirements” from “design” (big issue here?)

• Requirements are about “black box” external behavior of the proposed system
  – black box vs white box concepts
  – *software as transform of input to output*
    • what are these things?
  – precision of these observables
Steve McConnell says,
“The most difficult part of requirements gathering is not the act of recording what the users want; it is the exploratory, developmental activity of helping users figure out what they want.”
Finding Requirements (a process)

- Do outside research in domains of concern
  - what resources do we have?
  - how well have you researched the domains?
- Only speak user’s terms and definitions
- Ask questions (first few rounds)
  - after initial rounds, ask questions with binary answers
- Analyze, Follow up - Repeat
Ask Questions: Written

• Do basic research first!
  – do NOT ask questions that have been answered
  – show you followed up to last sessions

• Focus your questions
  – do NOT ask wide open questions (exceptions)
    • short, simple, answerable: yes/no preferred
  – if complex, ask multi-part questions
  – use models / documents as points of reference
Ask Questions - Written

• Give feedback on the answers
  – offer an example, “is this what you mean?”
  – narrow the question if you must
  – do not move on until you understand
    • or agree to look further
  – think like a customer who’ll have to live with this thing you’re going to describe
  – think like a coder who’ll have to build it!
Ask Questions - Written

• Have ALL critical written documents in your possession.
  – make sure your customer has copies when you discuss them

• Build models (documents) to discuss!
  – does our “context diagram” raise issues for you?
Analyze Customer’s Written Answers

• Give them an identification number, file them with all relevant information (dates...)
• Carefully parse answers for critical info:
  – priorities (“nice” “must” - clarify what these mean with your customer!)
  – get definitions of any new domain terms
  – generate new questions if necessary
Follow Up to Questions/Answers

• Careful review of all customer provided information - make connections
  – old things look different after new information
• Recognize customer’s time and effort
  – let them know how it helps you
  – always find some positive contribution, even when it looks bleak (then continue to ask...)
• Ask new questions only when done with old
Teleconferences

• Appoint a note-taker, the job is to RECORD
  – must be precise and complete

• Prepare:
  • current docs,
  • prepared question lists (with space for notes underneath)
  • pens, paper, tape recorder (with permission only!)
  • possibly: web access
  • FAX documents for discussion ahead
Teleconferences

• Be on time (respect customer’s schedule)
• Have customer and others present “sign off” on the notes after they are complete
  – make a full report of the session
    • who, what happened, when, where
  – send the report to all concerned parties and ask them for any corrections or comments
    • archive the info
In Person Meetings

• Have a note taker, recorder present
• Prepare a meeting report for sign off as before
• Have physical copies of all relevant documents
  – enough to go around to all parties at the meeting
In Person Meetings

• Be aware of body language / personal issues
  – if someone on your team doesn’t get along, they should be reassigned :-)
  – trust your intuition
  – be kind, gentle and understanding
  – be aggressive only when appropriate (when challenged or friendly banter)
Main Themes

• We’re writing Requirements
• Our job: serve the customer
  – be prepared
  • make the customer’s job as easy as possible
• Customer’s job: help us serve them
  – this is implicit in the customer / developer bill of rights in Wiegers
  • realistically this is a problem, refer to above (our job!)
• Be professional at all times