Article Summary Outline

Introduction:

* What RE is
  + Definition
  + Components of RE
* Why we need it
* Its evolution
  + RE for the 21st century
    - New Approach to system development
    - Need for rapid software delivery
    - Increasing rate of requirements change
    - Need for improved Return on Investment (ROI) on software assets
* Usefulness
  + Other than Software Engineering
* History of RE

Concurrent:

* Documentation
  + Concern about lack of rigorous documentation
* Three main benefits
  + Lower process overheads
  + Early of identification and implementation of important stuff
  + Quick Response time to requirement changes

Supporting Requirements/ Trade-Offs in Design:

* Simple things crashing projects
  + Natural Language Interface crashed project
* Complicators/Simplifiers
  + For both customer and developer
  + Allows you to see both sides of process

Commercial-Off the Shelf Acquisition

* How to decide what commercial software that is available is appropriate for your business
* Selection: The Core Process
  + Procurement oriented requirements gathering (4 step process)
    - Acquire Customer Requirements
    - Check compliance with COTS software
    - Reject non-complaints systems
    - Explore remaining software candidates
* Integration:
  + Five Guidelines for Interoperability