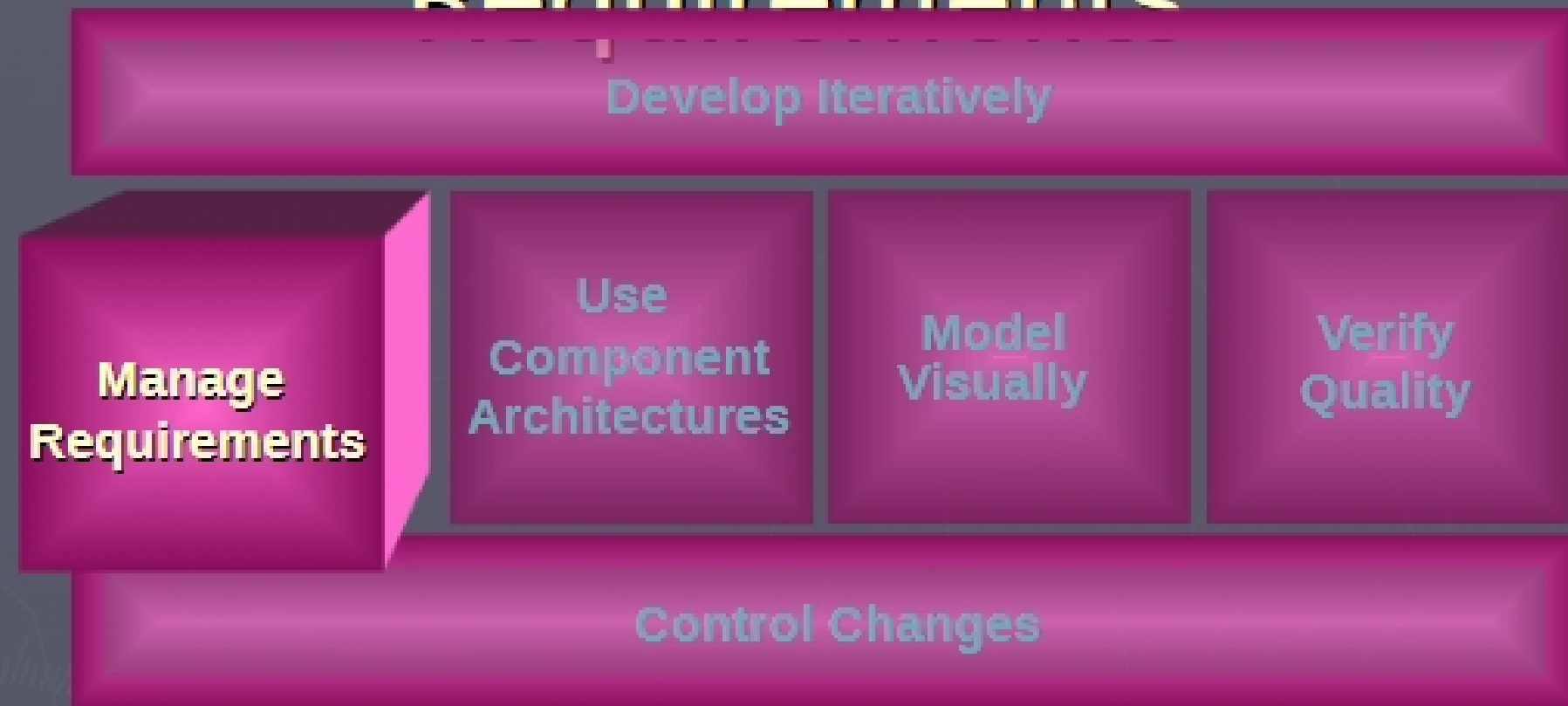


Continuing Best Practices

- ▶ *Slide information taken in large part from former Rational Corporation slides and the RUP textbook - considerably modified and supplemented for classroom use.*
- ▶ *Additional information taken from other sources.*

Practice 2: Manage Requirements



A report by Standish Group cites **forty percent of software projects fail.** According to them, the main causes of failure are: (note: there are others...)

Poor Requirements Management; (eliciting, capturing, documenting, modeling, verification; prototyping, throughout life cycle)
(handling Change!! Addressing Risk)

Incorrect Definition of Requirements
from the start of the project; and (feedback, verification...)

Practice 2: Manage Requirements

- ▶ Elicit, organize, and document required functionality and constraints
- ▶ Evaluate changes and determine impacts
- ▶ Track and document tradeoffs and decisions
- ▶ → Getting comprehensive system requirements is a continuous process
- ▶ → Obtaining a complete, exhaustive set of requirements prior to development is impossible! (except for trivial applications)



**Requirements are dynamic --
Expect them to change during software development**

Change cannot be stopped, but it can be managed!!