Prescriptive Models

- Prescriptive process models advocate an orderly approach to software engineering

That leads to a few questions …

- If prescriptive process models strive for structure and order, are they inappropriate for a software world that thrives on change?
- Yet, if we reject traditional process models (and the order they imply) and replace them with something less structured, do we make it impossible to achieve coordination and coherence in software work?
The Waterfall Model

- **Communication**
  - project initiation
  - requirement gathering

- **Planning**
  - estimating
  - scheduling
  - tracking

- **Modeling**
  - analysis
  - design

- **Construction**
  - code
  - test

- **Deployment**
  - delivery
  - support
  - feedback

The Incremental Model

- **Communication**

- **Planning**

- **Modeling**

- **Construction**

- **Deployment**

These courseware materials are to be used in conjunction with *Software Engineering: A Practitioner’s Approach*, 6/e and are provided with permission by R.S. Pressman & Associates, Inc., copyright © 1996, 2001, 2006.
The RAD Model

Communication

Planning

Modeling
- business modeling
- data modeling
- process modeling

Construction
- component reuse
- automatic code generation

Testing

Deployment
- integration delivery
- feedback

Team # 1

Team # 2

60 - 90 days

Evolutionary Models: Prototyping

Communication

Quick plan

Modeling
- Quick design

Deployment
- delivery & feedback

Construction of prototype

These courseware materials are to be used in conjunction with Software Engineering: A Practitioner's Approach, 6/e and are provided with permission by R.S. Pressman & Associates, Inc., copyright © 1996, 2001, 2005.
Evolutionary Models: The Spiral

Evolutionary Models: Concurrent
Still Other Process Models

- **Component based development**—the process to apply when reuse is a development objective
- **Formal methods**—emphasizes the mathematical specification of requirements
- **AOSD**—provides a process and methodological approach for defining, specifying, designing, and constructing aspects
- **Unified Process**—a "use-case driven, architecture-centric, iterative and incremental" software process closely aligned with the Unified Modeling Language (UML)

The Unified Process (UP)

These courseware materials are to be used in conjunction with *Software Engineering: A Practitioner’s Approach*, 6/e and are provided with permission by R.S. Pressman & Associates, Inc., copyright © 1999, 2001, 2005.
UP Phases

<table>
<thead>
<tr>
<th>Workflows</th>
<th>Inception</th>
<th>Elaboration</th>
<th>Construction</th>
<th>Transition</th>
<th>Production</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirements</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Analysis</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Design</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Implementation</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Test</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Support</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Iterations</td>
<td>#1</td>
<td>#2</td>
<td>#n-1</td>
<td>#n</td>
<td></td>
</tr>
</tbody>
</table>

UP Work Products

**Inception phase**
- Vision document
- Initial use-case model
- Initial project glossary
- Initial business case
- Initial risk assessment
- Project plan, phases and iterations, business model, if necessary
- One or more prototypes

**Elaboration phase**
- Use-case model
- Supplementary requirements
- Configuration management plan
- Detailed project plan
- Initial use-case model
- Software architecture
- Detailed planning document
- Detailed cost estimates
- Detailed schedule
- Revised risk list
- Project plan including
- Initial workflows
- Initial product
- Technical work products
- Preliminary user manual

**Construction phase**
- Design model
- Software components
- Integrated software increment
- Test plan and procedure
- Test cases
- Support documentation
- User manuals
- Installation manuals
- Description of current increment

**Transition phase**
- Delivered software increment
- Beta test reports
- General user feedback