



# Model-Driven Development Strategy

- **Model-driven development** یک استراتژی توسعه سیستم که بر رسم مدل‌های سیستم به منظور بصری‌سازی و تحلیل مشکلات، تعریف نیازمندی‌ها و طراحی سیستم‌ها، تاکید دارد.

- **Process modeling** – a process-centered technique popularized by the structured analysis and design methodology that used models of business process requirements to derive effective software designs for a system.
- **Data modeling** – a data-centered technique used to model business data requirements and design database systems that fulfill those requirements.
- **Object modeling** – a technique that attempts to merge the data and process concerns into singular constructs called objects. Object models are diagrams that document a system in terms of its objects and their interactions.

# Model-Driven Development Strategy

## Advantages

- بررسی بیشتر نیازمندی‌ها
- ساده‌تر بودن تحلیل راه‌حل‌های جایگزین
- مشخصات طراحی پایدارتر و انعطاف‌پذیرتر
- سیستم‌ها دفعه اول دقیق‌تر ساخته می‌شوند

## Disadvantages

- زمان‌بر
- مدل‌ها تنها به اندازه فهم کاربران از نیازمندی‌ها خوب هستند
- نقش کاربر کاهش می‌یابد زیرا تصاویر مثل نرم‌افزار نیستند
- می‌تواند غیر انعطاف‌پذیر باشد

# مدل‌های Prescriptive Models

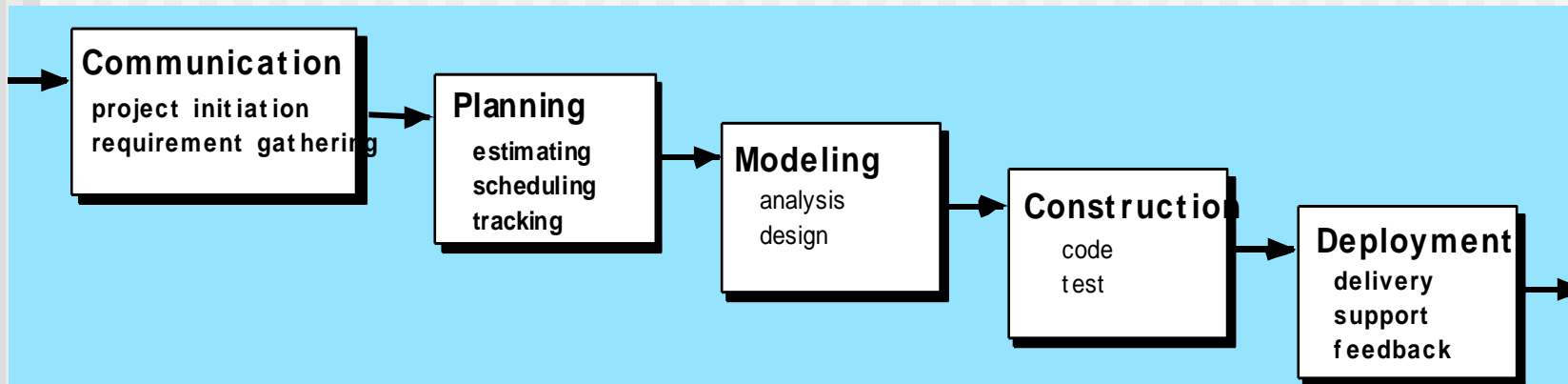
## تجویزی

- این روش‌ها طرفدار یک رویکرد منظم به مهندسی نرم‌افزار هستند.

*That leads to a few questions ...*

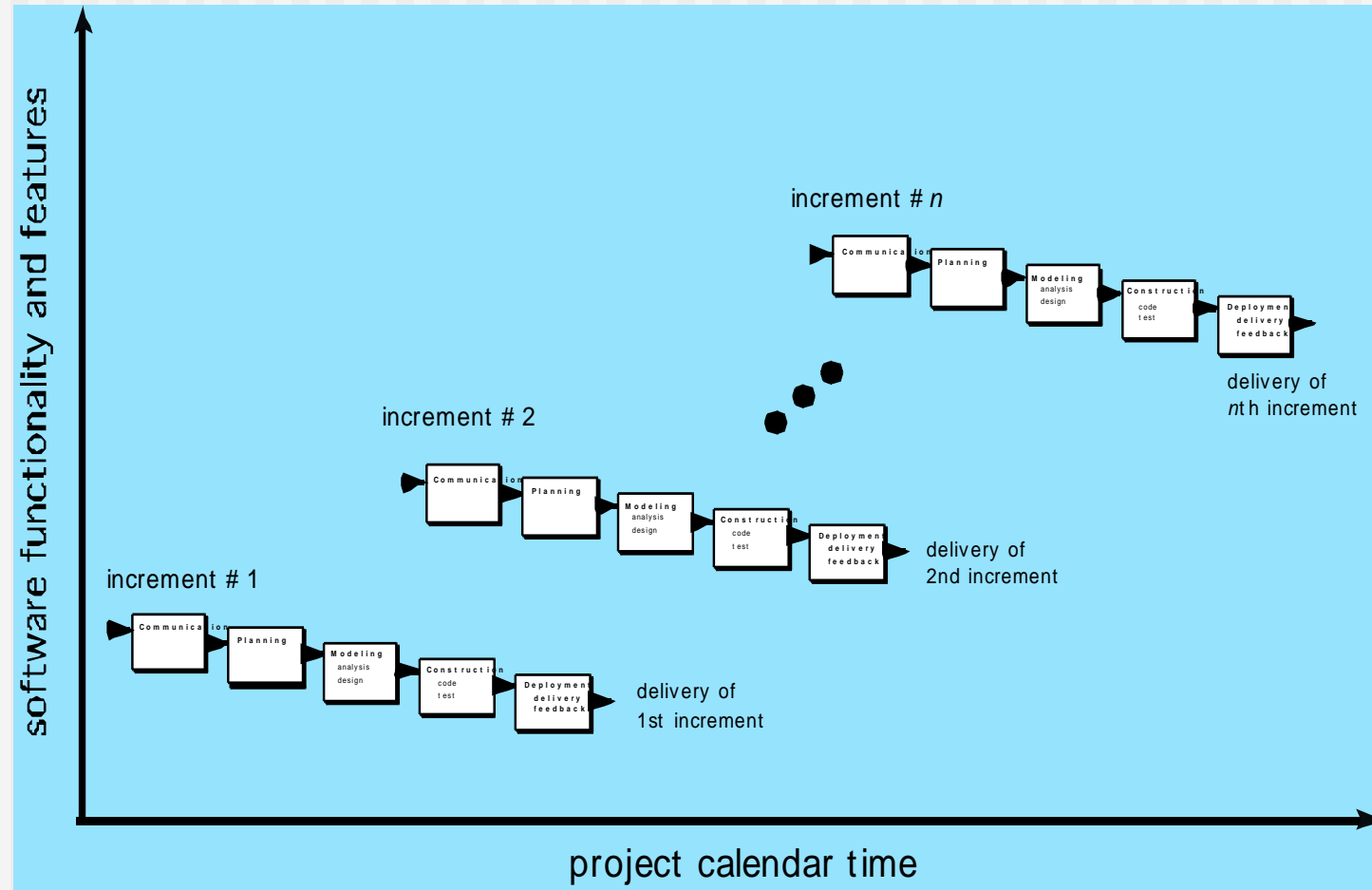
- اگر مدل‌های فرایند تجویزی نیازمند ساختار و ترتیب هستند، آیا برای دنیای نرم‌افزاری که مداوم در حال تغییر است مناسب هستند؟
- با این وجود اگر مدل‌های فرایند سنتی را نادیده بگیریم و آنها را با فرایندی کمتر ساخت‌یافته تعویض نماییم، آیا ایجاد هدایت و یکپارچگی را در فعالیت نرم‌افزاری غیر ممکن ساخته‌ایم؟

# The Waterfall Model



قدیمی ترین نمونه مهندسی نرم افزار. وقتی نیازمندی ها کاملاً مشخص باشند و همچنین پایدار باشند. این مدل خطی است. مشکلات: نوعاً نیازمند تکرار هستیم. مشخص سازی تمامی نیازمندی ها به صورت صریح دشوار است. کد در اواخر کار تولید می شود.

# The Incremental Model



These slides are designed to accompany *Software Engineering: A Practitioner's Approach*, 7/e (McGraw-Hill, 2009). Slides copyright 2009 by Roger Pressman.

## مدل افزایشی

---

- وقتی نیازمندی‌ها به صورت معقول و مناسبی خوش تعریف هستند اما حوزه کلی توسعه مانع به کار گیری یک فرایند خطی می‌شود. نیازی وجود دارد که برخی عملکردهای سیستم را به نسخه‌های بعدی نرم‌افزار موکول کنیم.
- عناصر جریان‌های خطی و موازی فرایند را ترکیب می‌کند. هر دنباله خطی افزایش‌های قابل تحویل نرم‌افزار را تولید می‌کند.
- افزایش اول یک محصول هسته‌ای است. کاربران از آن استفاده نموده و آن را ارزیابی می‌کنند و تغییراتی داده می‌شود تا با نیازمندی‌ها سازگارتر باشد.

# مدل‌های تکاملی

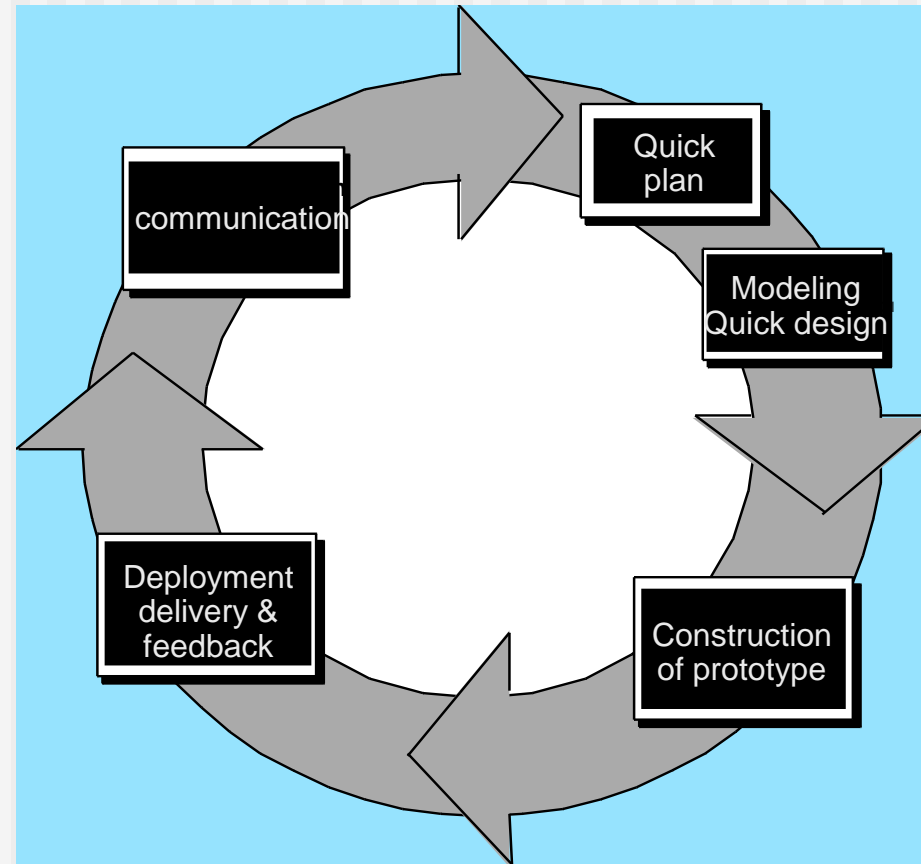
Two types: **Prototyping and Spiral models.**

---

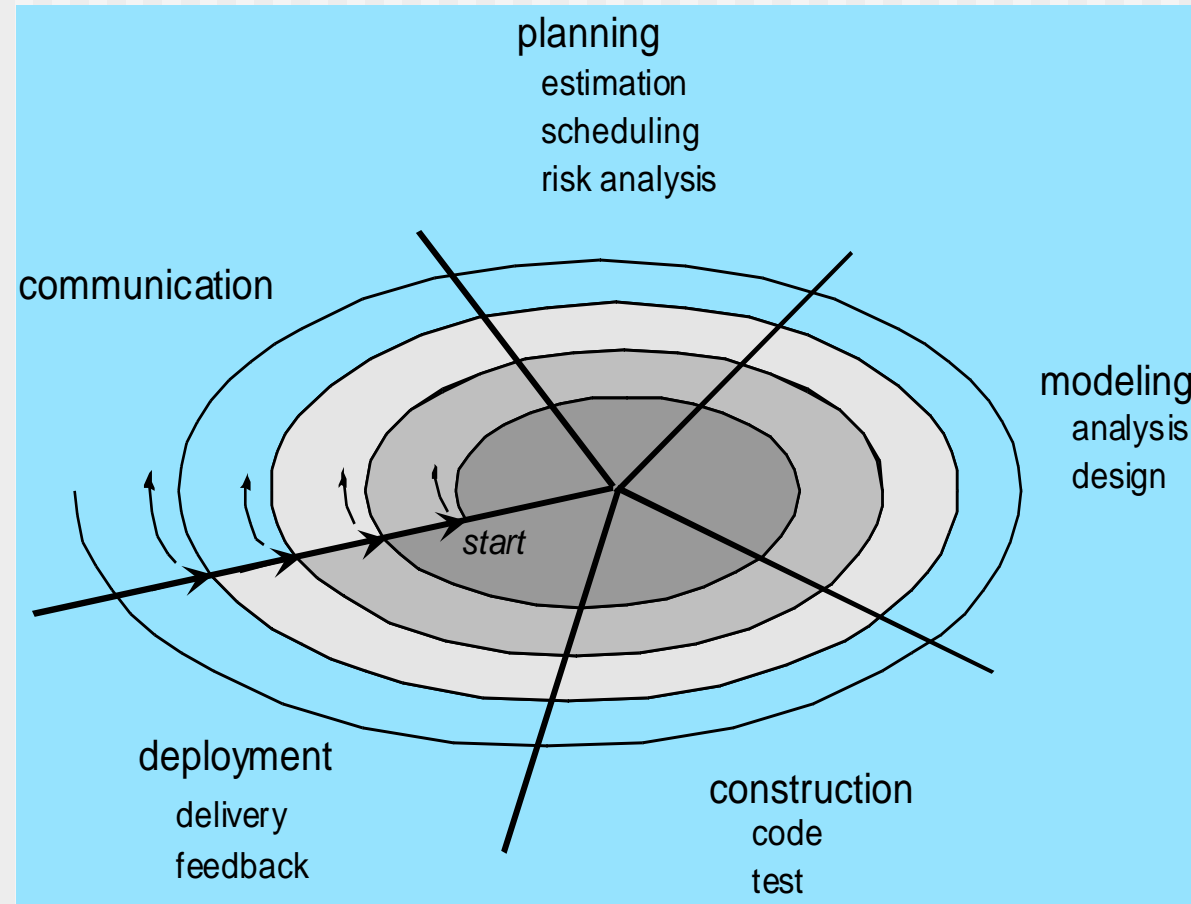
- سیستم نرم‌افزاری در طول زمان تکامل می‌یابد. این امر به دلیل تغییر نیازمندی‌ها در طول توسعه نرم‌افزار است. بنابراین یک خط مستقیم به محصول نهایی ناممکن خواهد بود. با این وجود، یک نسخه محدود بایستی تحویل داده شود تا فشار رقابتی ارضا گردد.
- معمولاً مجموعه‌ای از محصولات هسته‌ای یا نیازمندی‌های سیستم به خوبی فهم شده‌اند، اما جزئیات و توسعه‌ها بایستی تعریف شوند.
- به مدل فرایندی نیاز داریم که به صورت صریح برای یک محصول تکاملی طراحی شده است.
- همچنین تکراری است به این معنا که امکان تولید نسخه‌های کامل‌تر نرم‌افزار را فراهم می‌کند.



# Evolutionary Models: Prototyping



# Evolutionary Models: The Spiral



# سه دغدغه در باب مدل‌های تکاملی

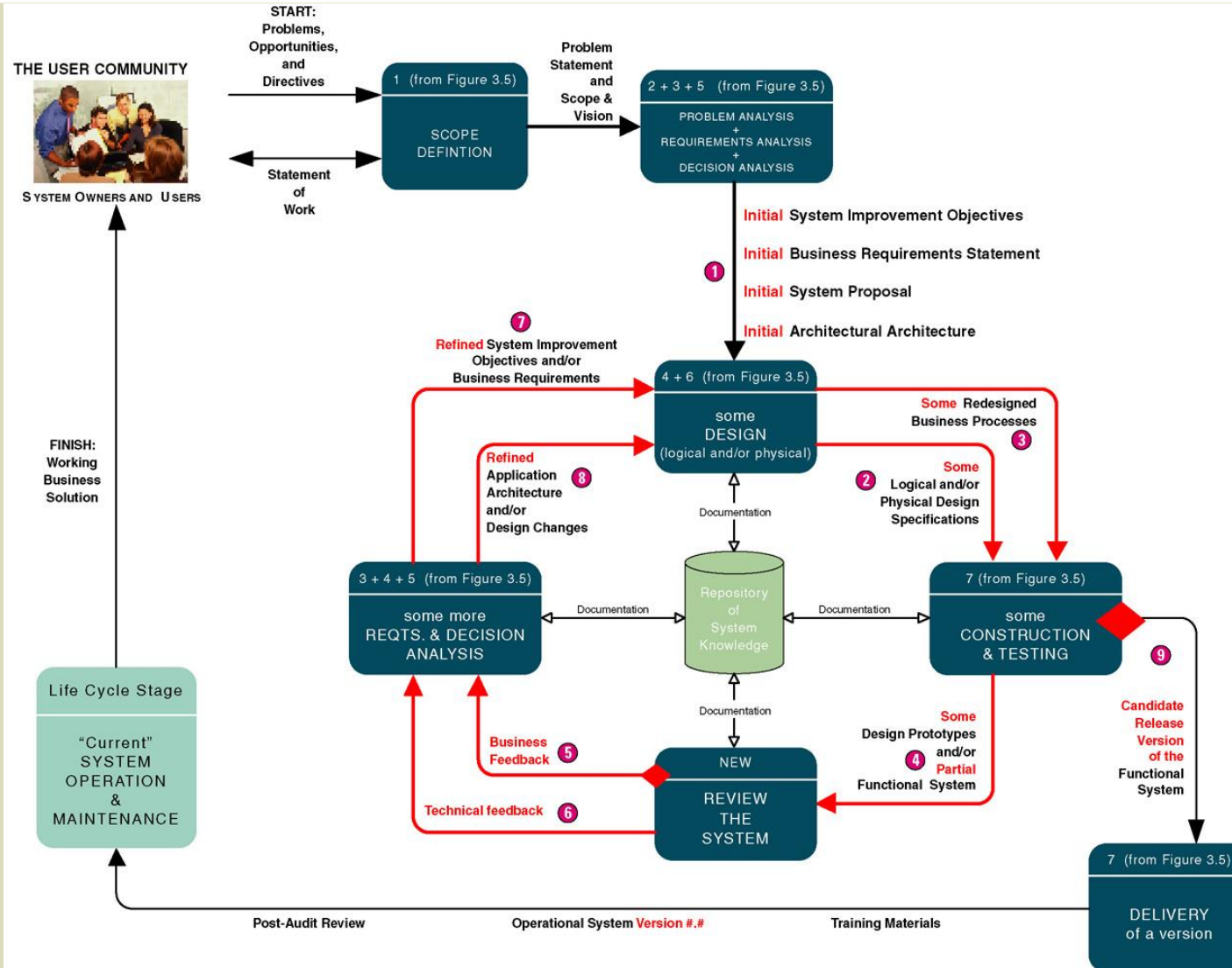
- نمونه‌سازی مشکلی برای مدیریت نرم‌افزار ایجاد می‌کند. این امر به دلیل تعداد نامشخص حلقه‌ها (تکرارها) برای ایجاد محصول است.
- دوم اینکه حداکثر سرعت تکامل بدست نمی‌آید. اگر تکامل خیلی سریع رخ بدهد، بدون یک بازه استراحت، فرایند دچار آشفتگی می‌شوند. اگر سرعت خیلی کند باشد، کارایی تولید تحت تاثیر قرار می‌گیرد.
- سوم اینکه، فرایند نرم‌افزار بایستی بر انعطاف‌پذیری و گسترش‌پذیری به جای کیفیت بالا تمرکز کند. بایستی به سرعت توسعه اولویت بیشتری نسبت به خطاهای کمتر بدهیم. گسترش توسعه برای دستیابی به کیفیت بالا، ممکن است منجر به تحویل دیر هنگام محصول شود.

# Rapid Application Development Strategy

- **Rapid application development (RAD)** یک استراتژی توسعه سیستم که بر سرعت توسعه یا درگیر کردن زیاد کاربر تاکید دارد. این روش بر ساخت یک سری نمونه‌های سیستم که کار می‌کنند به شیوه سریع و افزایشی تاکید دارد و این نمونه‌ها به تدریج به سیستم نهایی تبدیل می‌شوند.

- **Prototype** – a small-scale, representative, or working model of the users' requirements or a proposed design for an information system.
- **Time box** – the imposition of a non-extendable period of time, usually 60-90 days, by which the first (or next) version of a system must be delivered into operation.

# Rapid Application Development Strategy



# Rapid Application Development Strategy

## Advantages

- نیازمندی‌ها کاربر معمولاً نامطمئن و غیر دقیق است
- تشویق به مشارکت فعالانه کاربر و مدیریت
- پروژه‌ها عینیت و پشتیبانی بیشتری پیدا می‌کند
- ذیفعان راهکارهایی که کار می‌کنند را زودتر می‌بینند
- خطاها زودتر کشف می‌شوند
- آزمون و آموزش به صورت طبیعی اتفاق می‌افتد
- فرایند طبیعی‌تری است زیرا تغییر مورد انتظار است

## Disadvantages

- ممکن است مشوق ذهنیت «کد، پیاده‌سازی، تعمیر» شود
- می‌تواند مشکل اشتباهی را حل کند زیرا تحلیل خلاصه شده است
- ممکن است تحلیل گران را از بررسی راه‌کارهای جایگزین منصرف کند
- ذیفعان در مقابل دورریختن نمونه اولیه مقاومت می‌کنند
- تاکید بر سرعت ممکن است کیفیت را تحت تاثیر قرار دهد

# A System Maintenance Perspective

