



UML

طراحی شی گرای سیستمها

دکتر قاسمی

دانشگاه آزاد تهران شمال

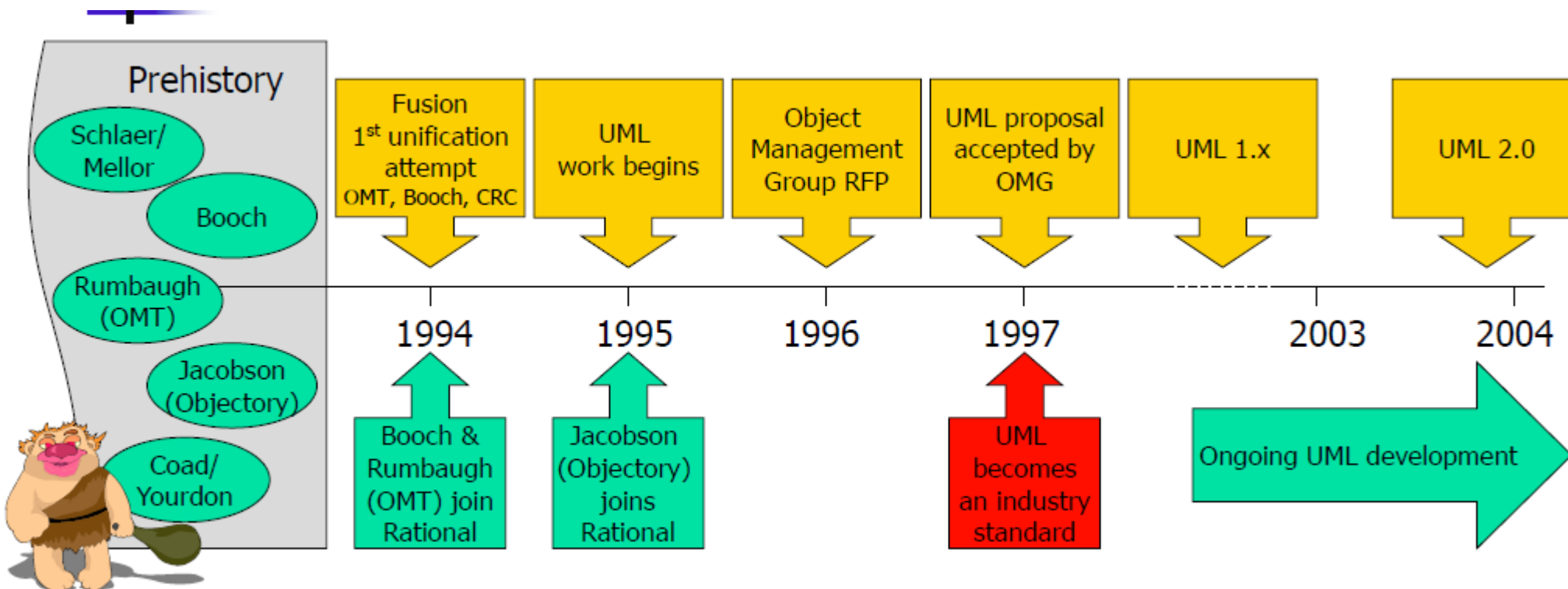
تاریخچه UML

❖ قوام بیشتر

❖ معناشناسی دقیق تر تعریف شده است

❖ انواع نمودارهای جدید

❖ سازگار با نسخ قبلی



UML – Unified Modeling Language

❖ زبان مدلسازی یکپارچه UML یک زبان استاندارد برای تعیین، تجسم، ساخت و مستندسازی مصنوعات artifacts یا مستندات طراحی سیستم های نرم افزاری است

❖ **Artifact** : مصنوعات : هر آنچه به عنوان خروجی طراحی مستند ورد طراحی نمودار و ... استخراج می شود.

❖ **UML** همچنین برای مدل سازی کسب و کار و سایر سیستم های غیر نرم افزاری نیز مناسب است.

❖ **UML** مجموعه ای از بهترین روش های مدل سازی را نشان می دهد که در مدل سازی

پیچیده موفق بوده اند.



❖ کسب و کارها به طور خاص نیاز به حل برخی مشکلات که به طور مکرر در زمینه معماری رخ می دهد مانند توزیع فیزیکی، هم زمانی، تکرار، امنیت، متعادل سازی بار و تحمل خطا نیاز داشتند. به علاوه توسعه برای وب در حین ساده تر ساختن، موجب تشدید این مشکلات می شد.

❖ اهداف اصلی طراحی UML به صورت زیر خلاصه شده است:

- به کاربران یک زبان با استفاده آسان و مدل سازی از نظر بصری گویا ارائه کند تا بتوانند مدل های معناداری را تهیه و مبادله کنند.
- سازوکار بسط پذیر و تخصصی سازی برای گسترش مفاهیم پایه ارائه کند.
- از زبان های برنامه نویسی و فرایندهای توسعه خاص مستقل باشد.
- یک مبنای رسمی برای درک زبان مدل سازی ارائه کند.
- رشد بازار ابزارهای شیء گرایی را تشویق کند.
- از مفاهیم توسعه سطح بالا مانند همکاری ها، فریمورک ها، الگوها و کامپوننت ها پشتیبانی کند.
- بهترین رویه ها را یکپارچه سازی کند.

❖ مدل یک تجرید از چیزی است که قبل از ساختن آن چیز باید درک کنیم. از آنجا که مدل سازی جزئیات غیر ضروری را نادیده می گیرد، دست کاری آن نسبت به موجودیت اصلی آسان تر است.

❖ منظور از مدل، سازمان دهی چیزی thing برای مقصود خاص است. در واقع مدل یک ساده سازی از یک واقعیت است. مدل امکانات زیر را ارائه می کند:

- نقشه اولیه سیستم
- سازمان دهی سیستم
- ارتباطات سیستم

مدل ها و اشیا در UML

UML سیستم‌ها را به عنوان مجموعه‌ای از اشیا مدل سازی می کند که برای ارائه منافع به کاربران خارجی با هم تعامل دارند

❖ مدل های از نوع ساختار استاتیک Static structure

- چه نوع اشیایی مهم هستند
- روابط آنها چگونه است

❖ مدل های از نوع رفتار پویا Dynamic behaviour

- چرخه زندگی اشیا چگونه است
- تعاملات اشیا برای دستیابی به اهداف چگونه است

نمودارهای UML

❖ نمودار کلاس – Class

❖ نمودار شی – Object

❖ نمودار مورد کاربرد – Use case diagram

❖ نمودار توالی – Sequence

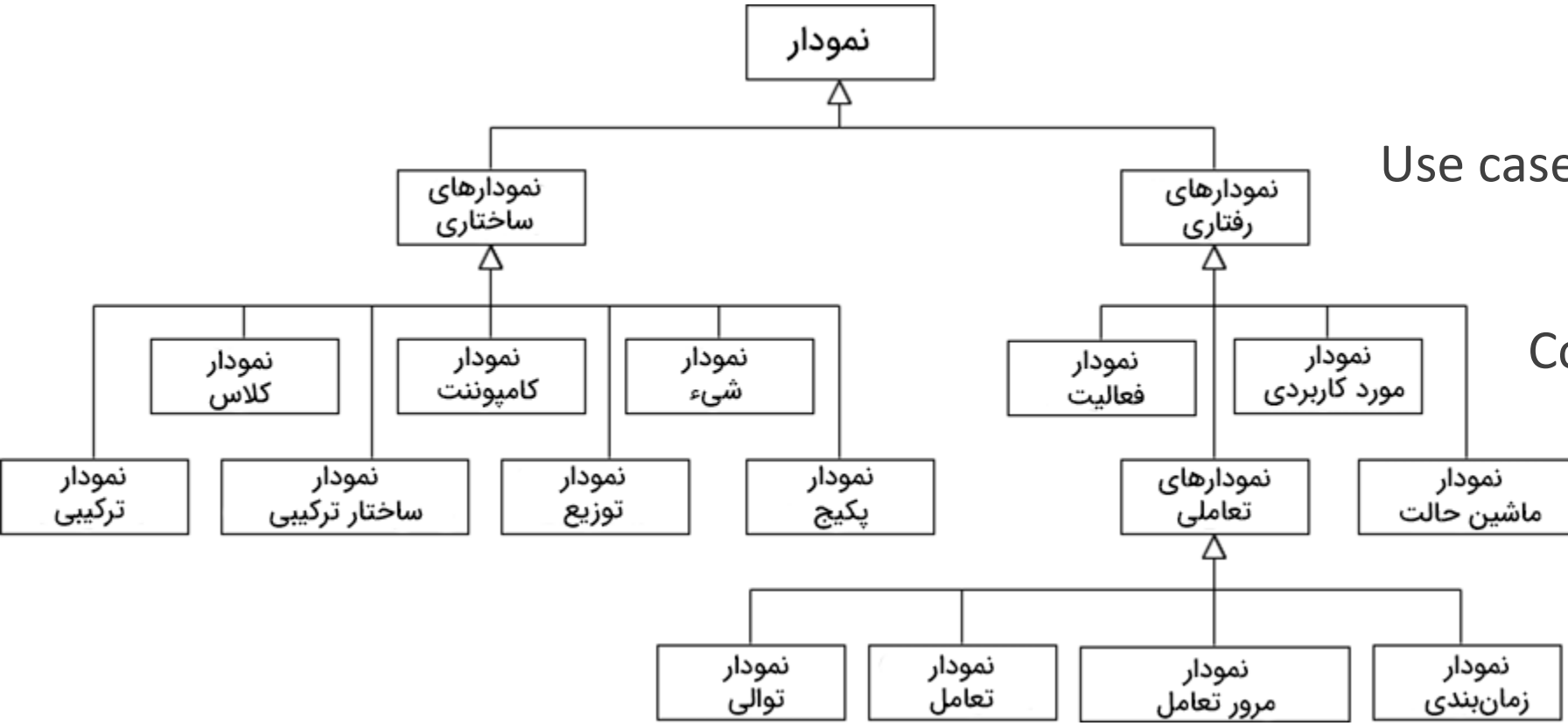
❖ نمودار همکاری – Collaboration

❖ نمودار فعالیت – Activity

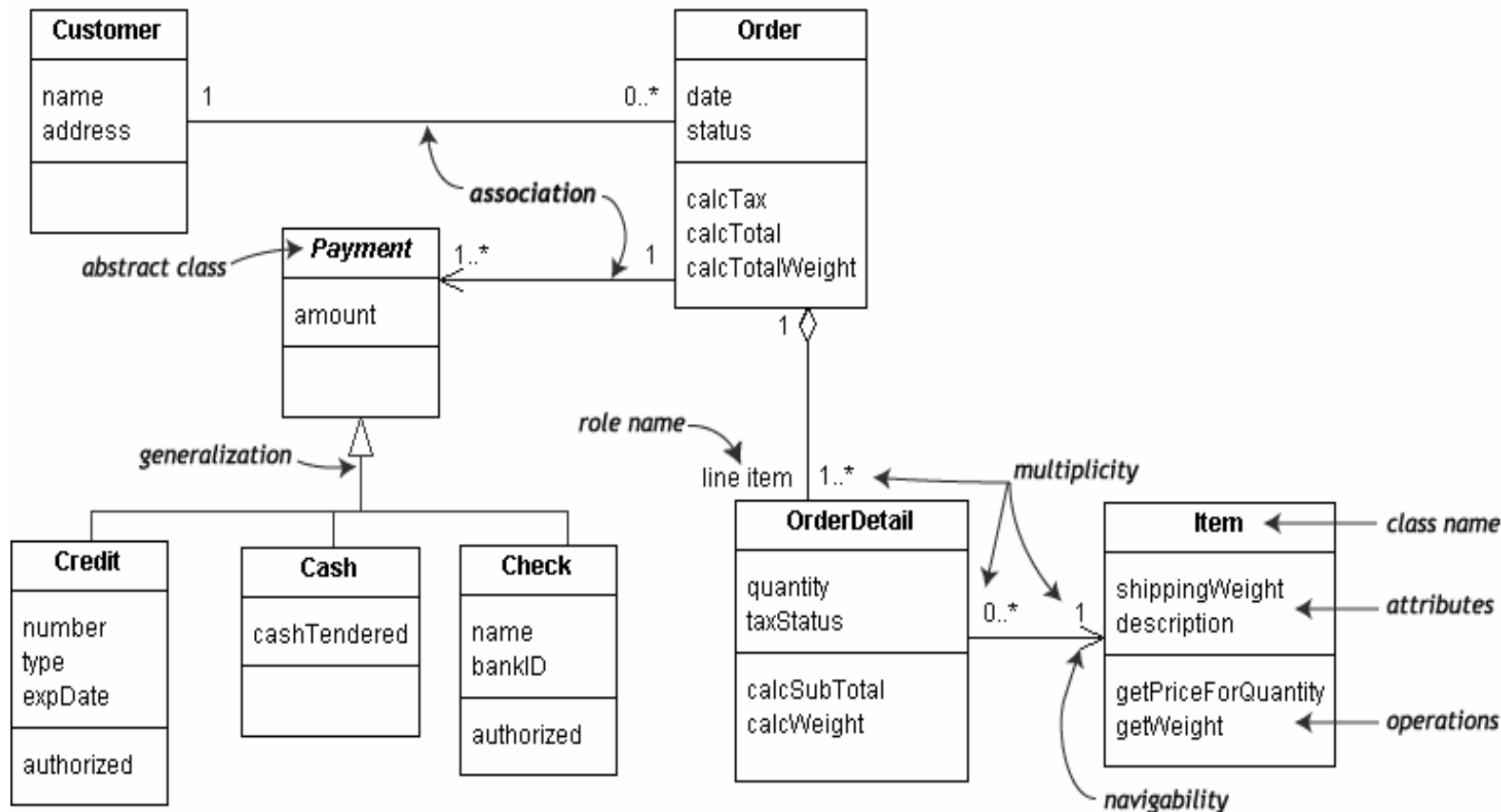
❖ نمودار وضعیت – statechart

❖ نمودار استقرار – Deployment

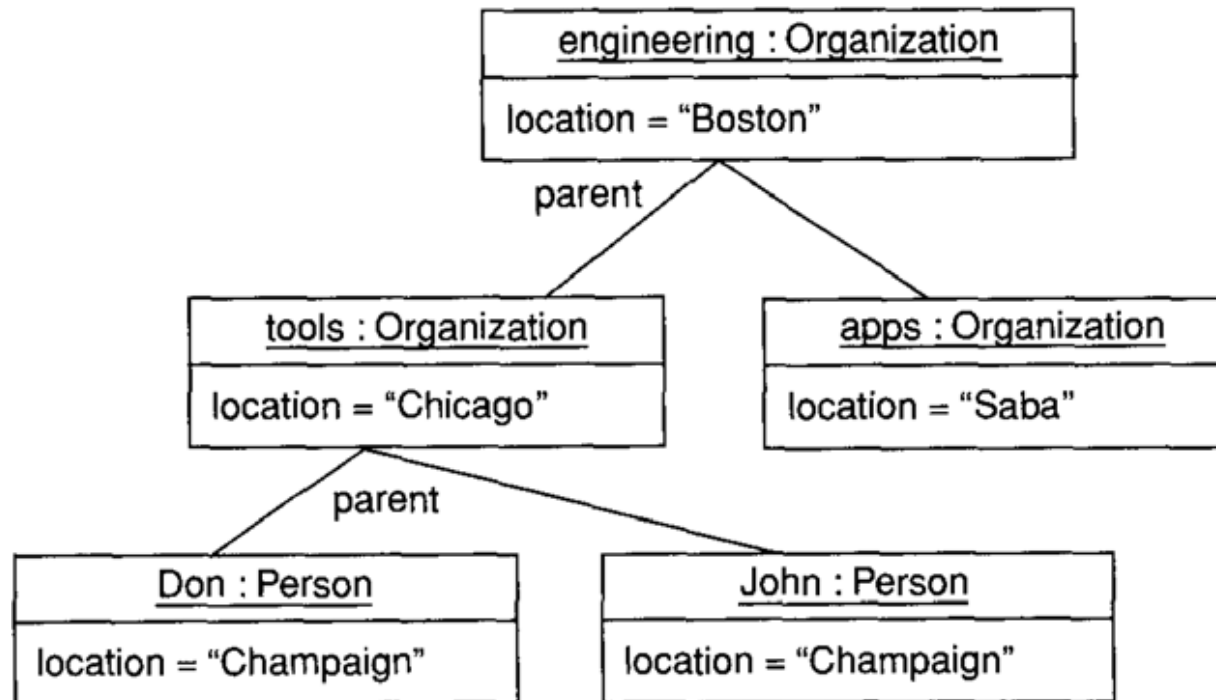
❖ نمودار اجزا – Component



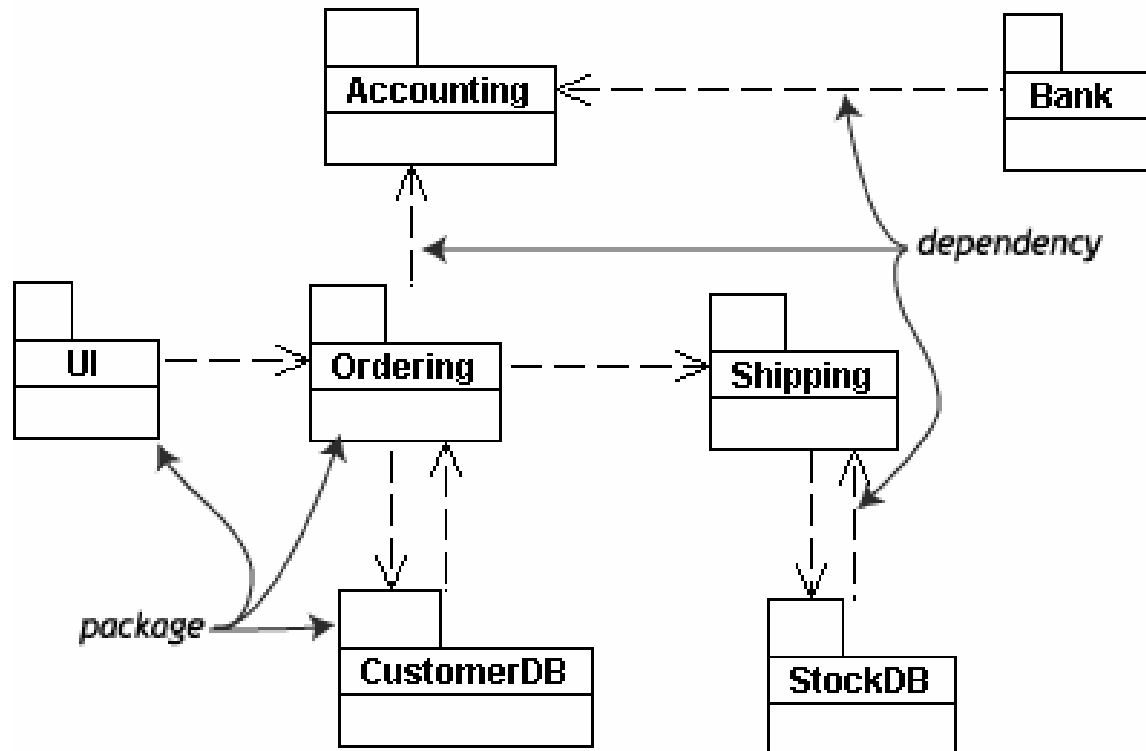
❖ «نمودار کلاس» Class Diagram یک تکنیک مدل سازی مرکزی است که در تقریباً همه روش های شیء گرای اجرا می شود. این نمودار انواع مختلف اشیا در سیستم و انواع مختلف رابطه هایی که بین آنها وجود دارد را توصیف می کند



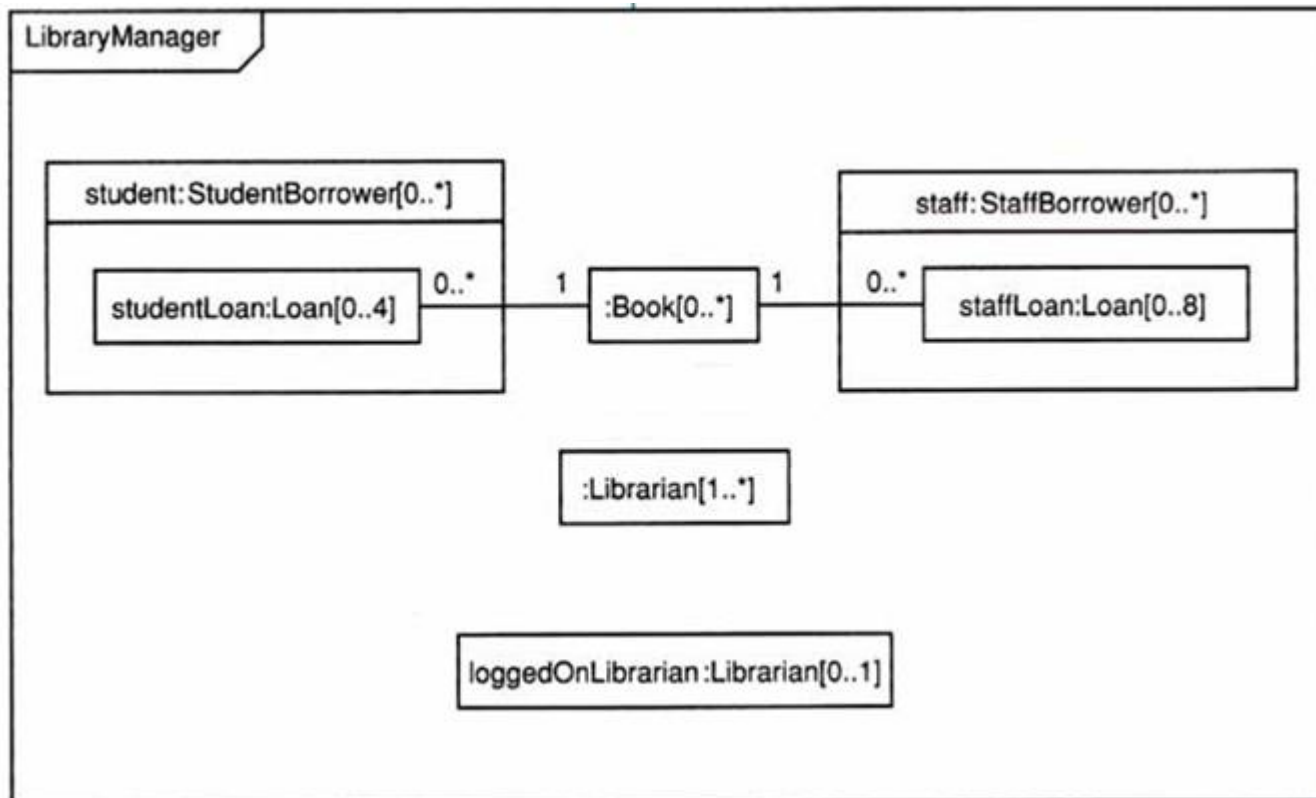
❖ با نمونه سازی عناصر مدل در یک نمودار کلاس، می توانید رفتار یک سیستم را در یک نقطه از زمان کشف کنید. نمودار شی یک نمودار ساختاری UML است که نمونه هایی از طبقه بندی کننده ها را در مدل ها نشان می دهد. نمودارهای شی از نمادهایی استفاده می کنند که مشابه آنچه در نمودارهای کلاس استفاده می شود



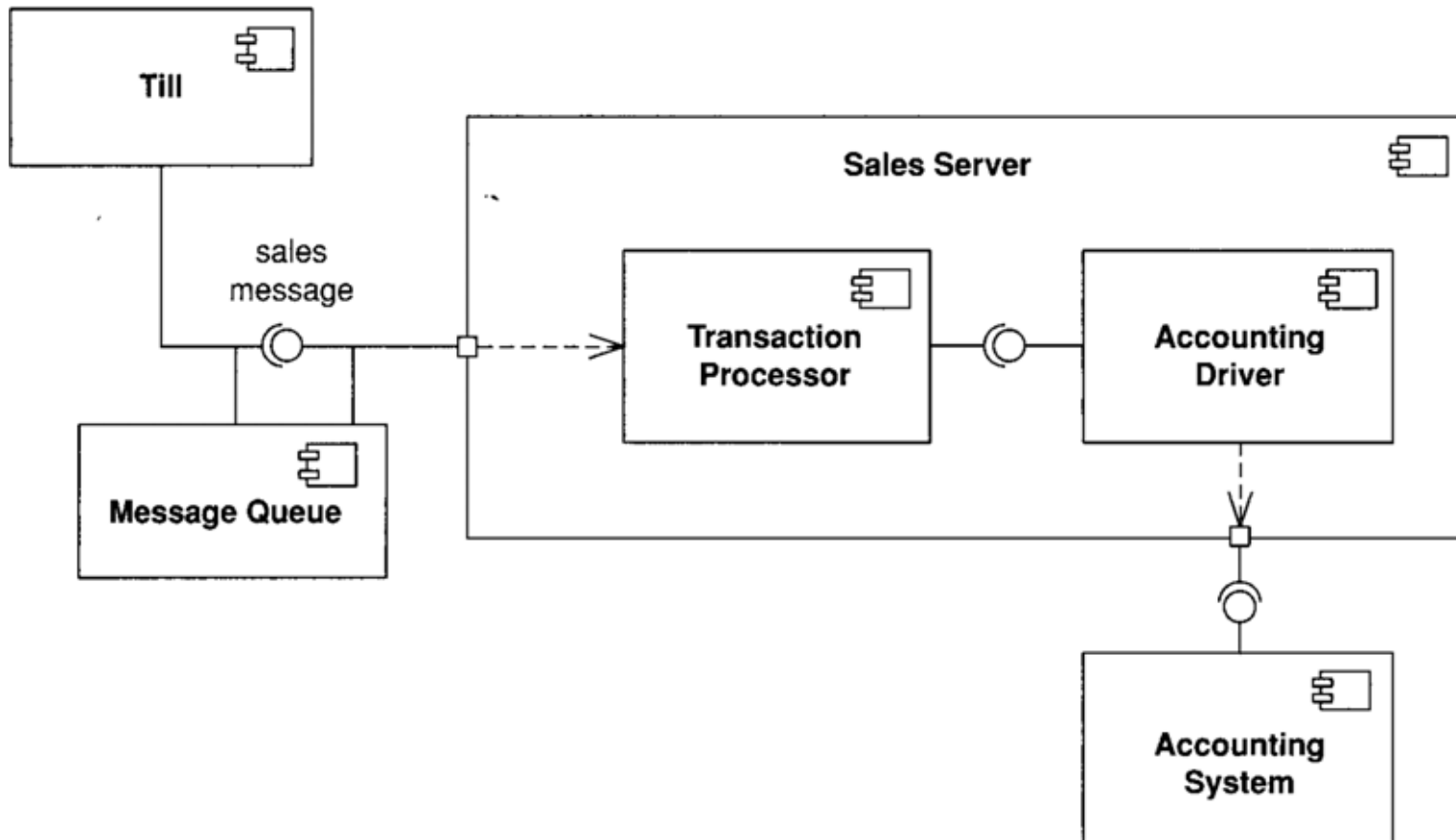
❖ «نمودار پکیج» Package Diagram یک نمودار ساختاری UML است که پکیج‌ها و وابستگی‌های بین پکیج‌ها را نمایش می‌دهد. نمودارهای پکیج به کاربر امکان می‌دهند که نماهای مختلف یک سیستم را نمایش دهد. برای نمونه در یک اپلیکیشن چندلایه می‌توان مدل چندلایه را نمایش داد.



❖ نمودار ساختار ترکیبی یکی از مصنوعات جدیدی است که به UML 2.0 اضافه شده است. نمودار ساختار ترکیبی یک نمودار ساختاری UML است که شامل کلاس‌ها، رابط‌ها، بسته‌ها و روابط آنهاست و نمای منطقی از همه یا بخشی از یک سیستم نرم‌افزاری را ارائه می‌دهد. ساختار داخلی (شامل قطعات و اتصال دهنده‌ها) یک طبقه بندی یا همکاری ساختار یافته را نشان می‌دهد.



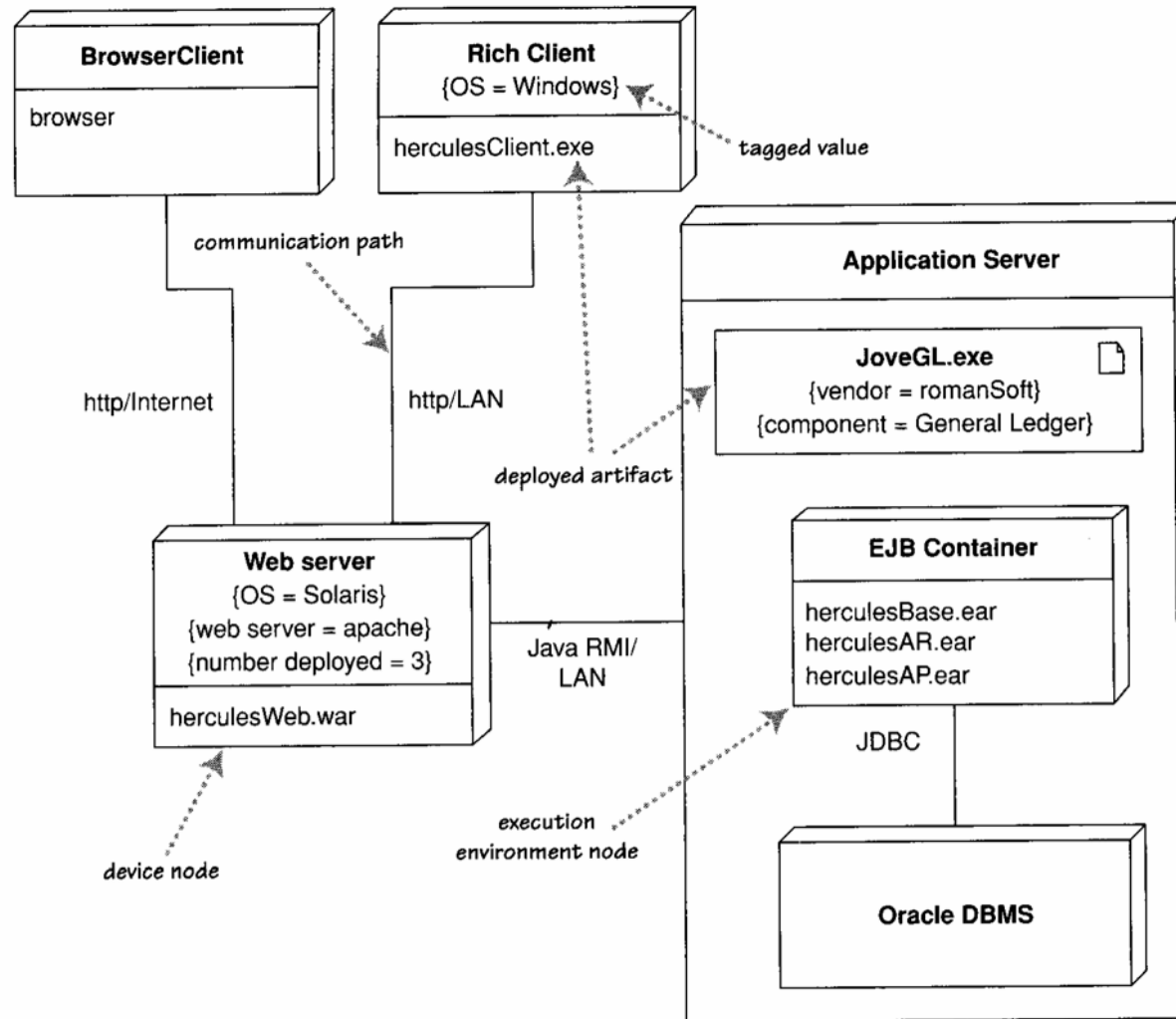
❖ در زبان مدل سازی یکپارچه منظور از «نمودار کامپوننت» Component Diagram شیوه اتصال کامپوننت ها به همدیگر و به کامپوننت های بزرگ یا سیستم های نرم افزاری است.



❖ این نوع نمودار معماری کامپوننت های نرم افزاری و وابستگی های بین آنها را به تصویر می کشد. این کامپوننت های نرم افزاری شامل کامپوننت های زمان اجرا، کامپوننت های اجرایی و کامپوننت های سورس کد هستند.

Structure Diagrams

Deployment Diagram



❖ «نمودار توزیع» Deployment Diagram به

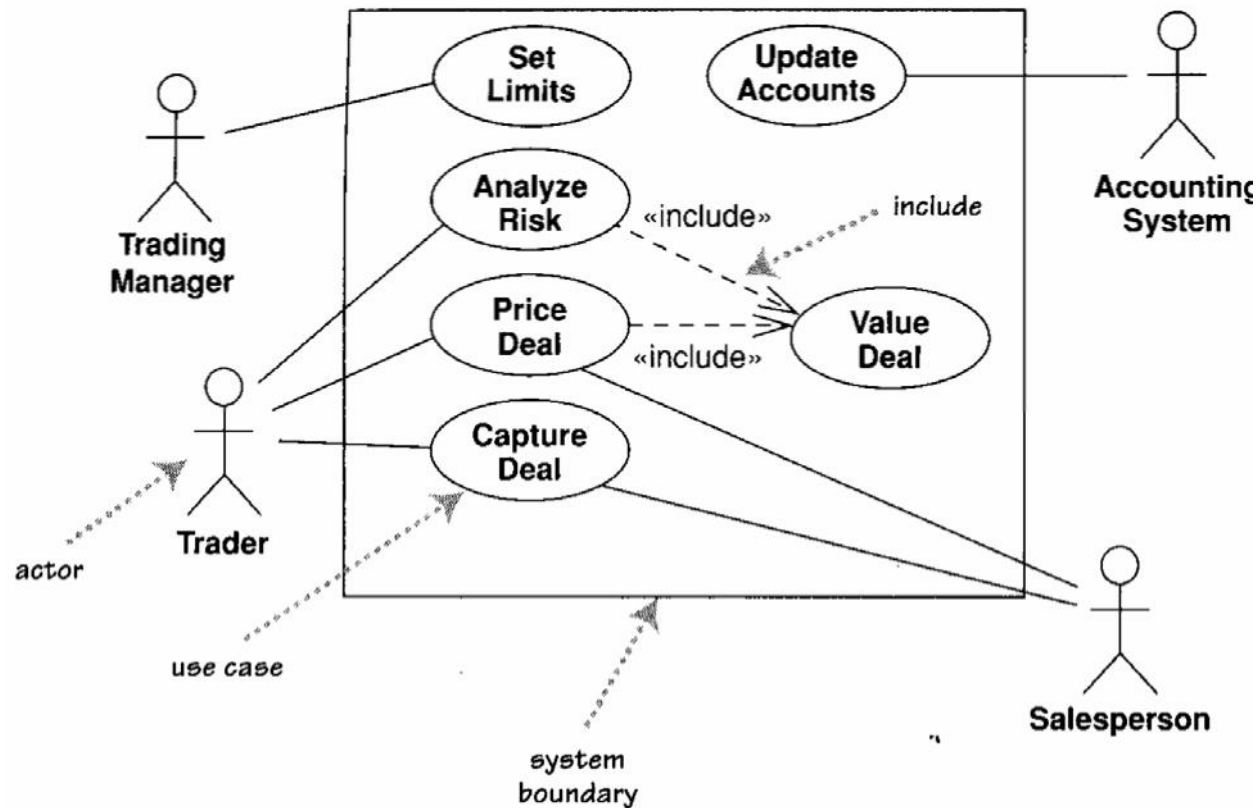
مدل سازی جنبه فیزیکی یک سیستم نرم افزاری
شیء گرا کمک می کند. این یک نمودار ساختاری
است که معماری سیستم را در زمان توزیع نرم افزار
روی هدف های توزیع نشان می دهد. -Artifactها
نشان دهنده عناصر مستحکمی در دنیای واقعی
هستند که نتیجه فرایند توسعه هستند.

❖ این نمودار پیکربندی زمان اجرای یک نمای

استاتیک را مدل سازی کرده و توزیع -artifactها
را در یک اپلیکیشن به تصویر می کشد. در اغلب
موارد این نمودار شامل مدل سازی پیکربندی
سخت افزاری به همراه کامپوننت های نرم افزاری
درون آن است.

❖ مورد کاربرد :

فهرستی از اقدامات یا مراحل رویداد است که معمولاً تعاملات بین یک نقش یا بازیگر و یک سیستم برای دستیابی به یک هدف را تعریف می کند.
بازیگر می تواند یک انسان یا یک سیستم بیرونی دیگر باشد.



❖ «نمودار موارد کاربرد» Use Case Diagram را می‌توان مانند یک منو تصور کرد که شباهت زیادی به منوی رستوران‌ها دارد. با نگاه کردن به منو میدانیم که چه چیزهایی در اختیار ما قرار دارند. خوراکی‌های منفرد و نیز قیمت‌های آن‌ها در دید ما قرار می‌گیرند. همچنین متوجه می‌شوید که رستوران چه نوع غذاهایی مثلاً ایتالیایی، مکزیکی، چینی و غیره عرضه می‌کند.

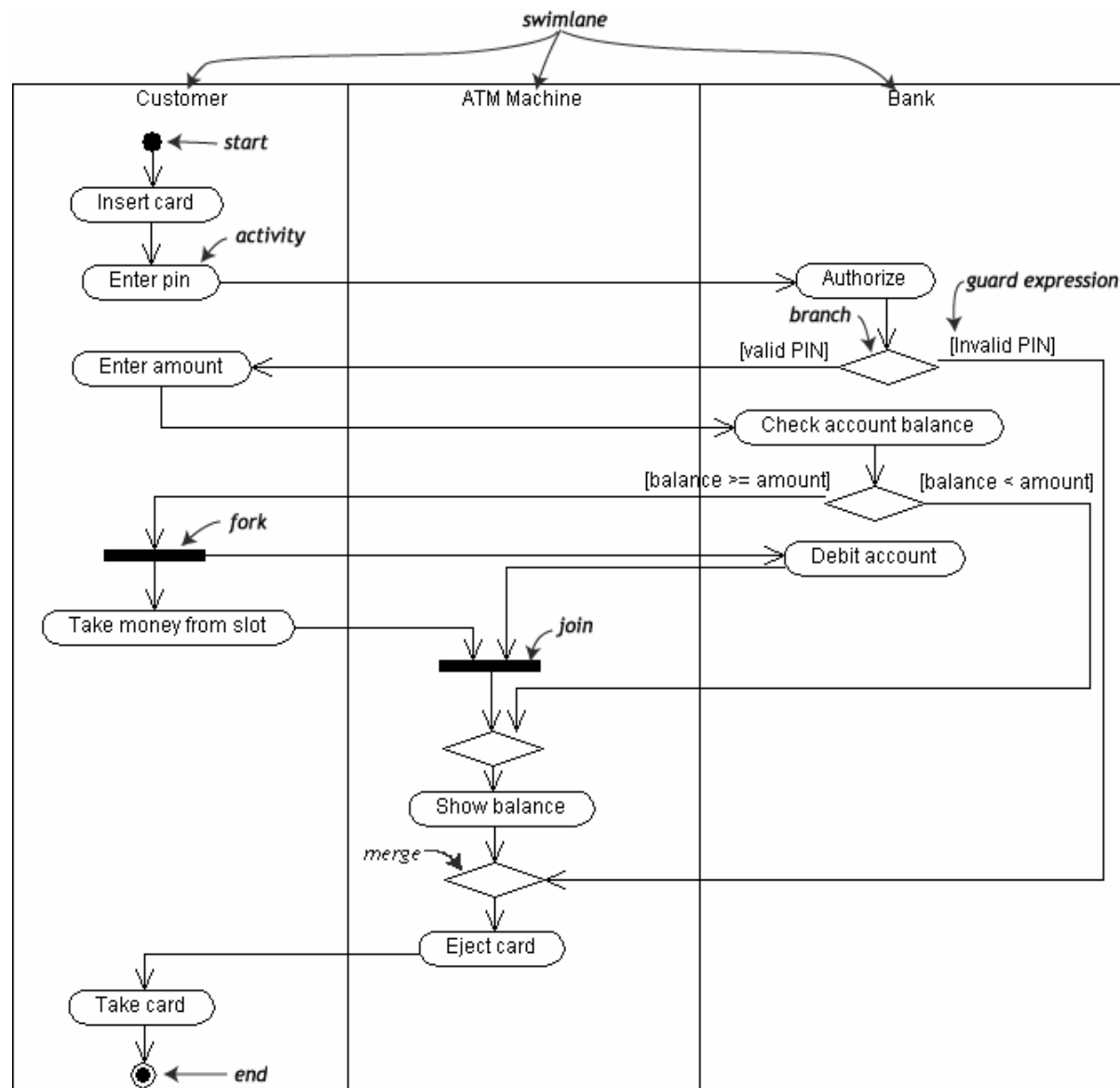
❖ با نگاه کردن به نمودار می‌توانید درکی کلی از تجربه تغذیه‌ای که انتظار دارید، داشته باشید. این منو در عمل رفتار رستوران را مدل‌سازی می‌کند. از آنجا که این نوع نمودار یک ابزار برنامه‌ریزی بسیار قدرتمند است، به طور کلی در همه فازهای چرخه توسعه از سوی همه اعضای تیم مورد استفاده قرار می‌گیرد.

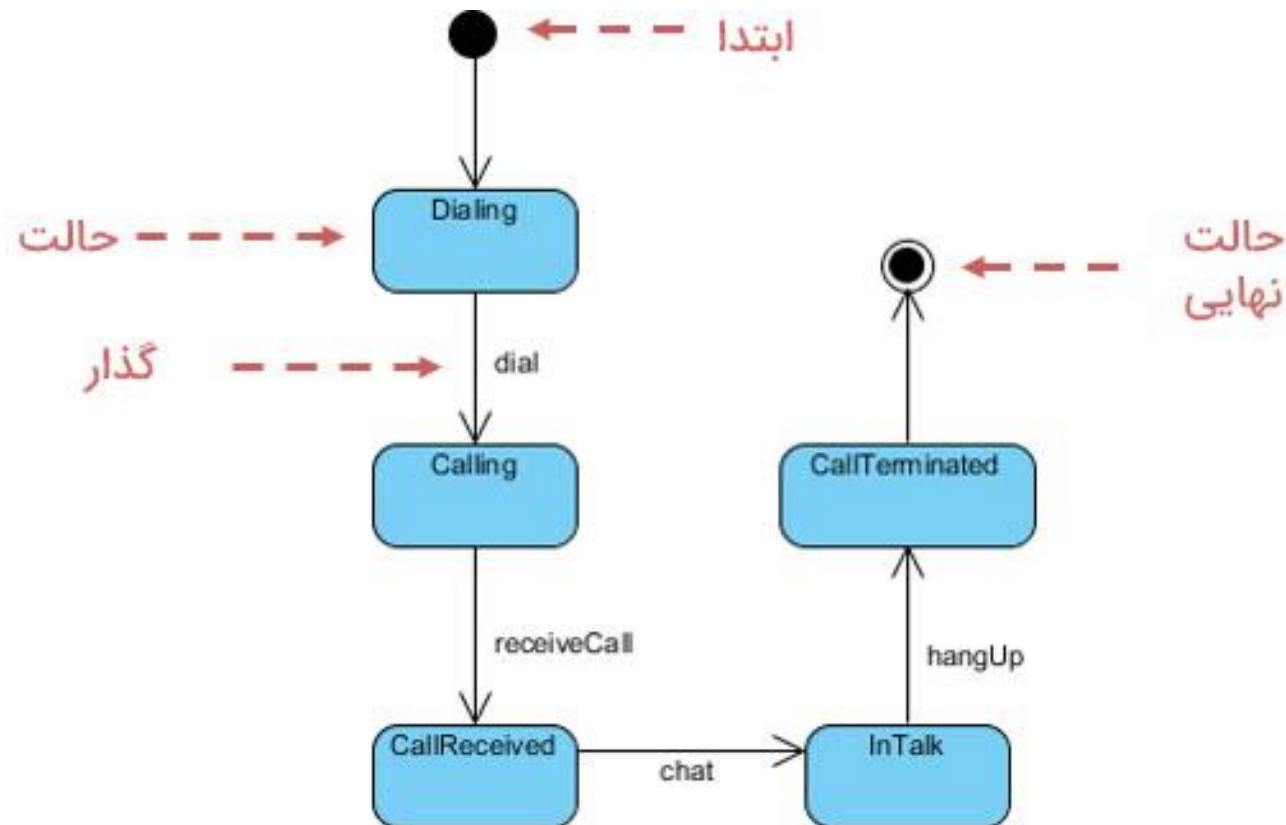
Behavior Diagrams

Activity Diagram

❖ «نمودار فعالیت» Activity Diagram به بازنمایی گرافیکی از گردش کارهای گام به گام فعالیتها و اقدامات گفته می شود که از انتخاب، تکرار و همزمانی نیز پشتیبانی می کند.

❖ این نمودار گردش کنترل سیستم هدف مانند اکسپورت کردن قواعد و عملیات بیزینسهای پیچیده، توصیف مورد کاربردی و همچنین فرایند کسب و کار را نیز توصیف می کند. در زبان مدل سازی یکپارچه نمودارهای فعالیت به منظور مدل سازی فرایندهای محاسباتی و همچنین سازمانی (مانند گردش کارها) مورد استفاده قرار می گیرند.





❖ «نمودار حالت» State Diagram نوعی از نمودار مورد استفاده در UML است که رفتار سیستم را بر مبنای مفهوم نمودارهای حالت را توصیف می کند. نمودارهای حالت، حالت ها، گذارها و همچنین رویدادهایی که بر این گذار تأثیر می گذارند را توصیف می کنند.

❖ این نوع نمودار به بصری سازی چرخه عمر کلی اشیا کمک می کند و از این رو درک بهتری از سیستم های مبتنی بر حالت ارائه می کند.

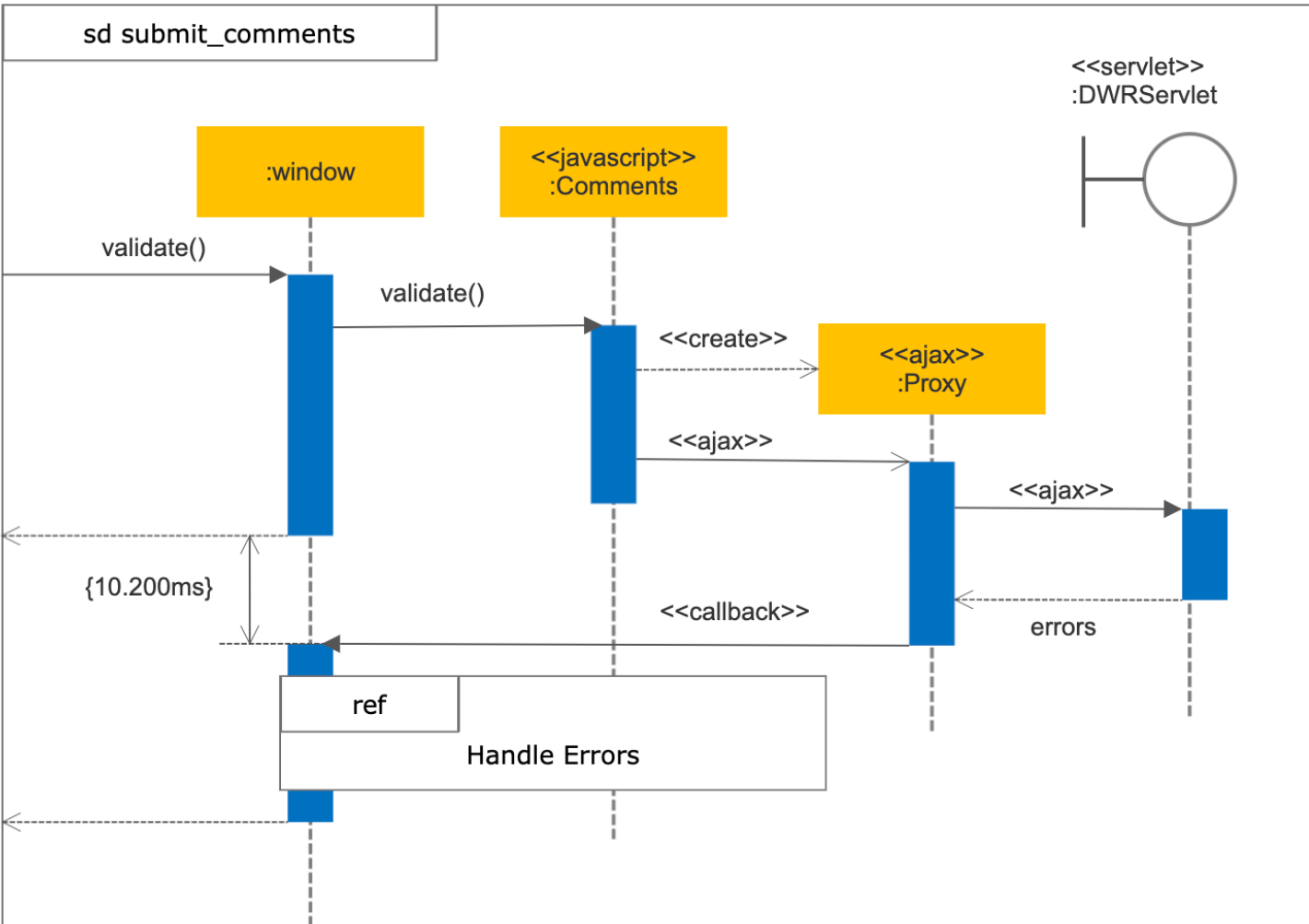
Behavior Diagrams: Interaction Diagrams

Sequence Diagram



❖ «نمودار توالی» (Sequence Diagram)

بر مبنای یک توالی زمانی می‌پردازد. این نمودار شیوه تعامل اشیا با همدیگر را برحسب یک سناریوی خاص در یک مورد کاربردی به تصویر می‌کشد.



[Fowler 2004]

Behavior Diagrams: Interaction Diagrams

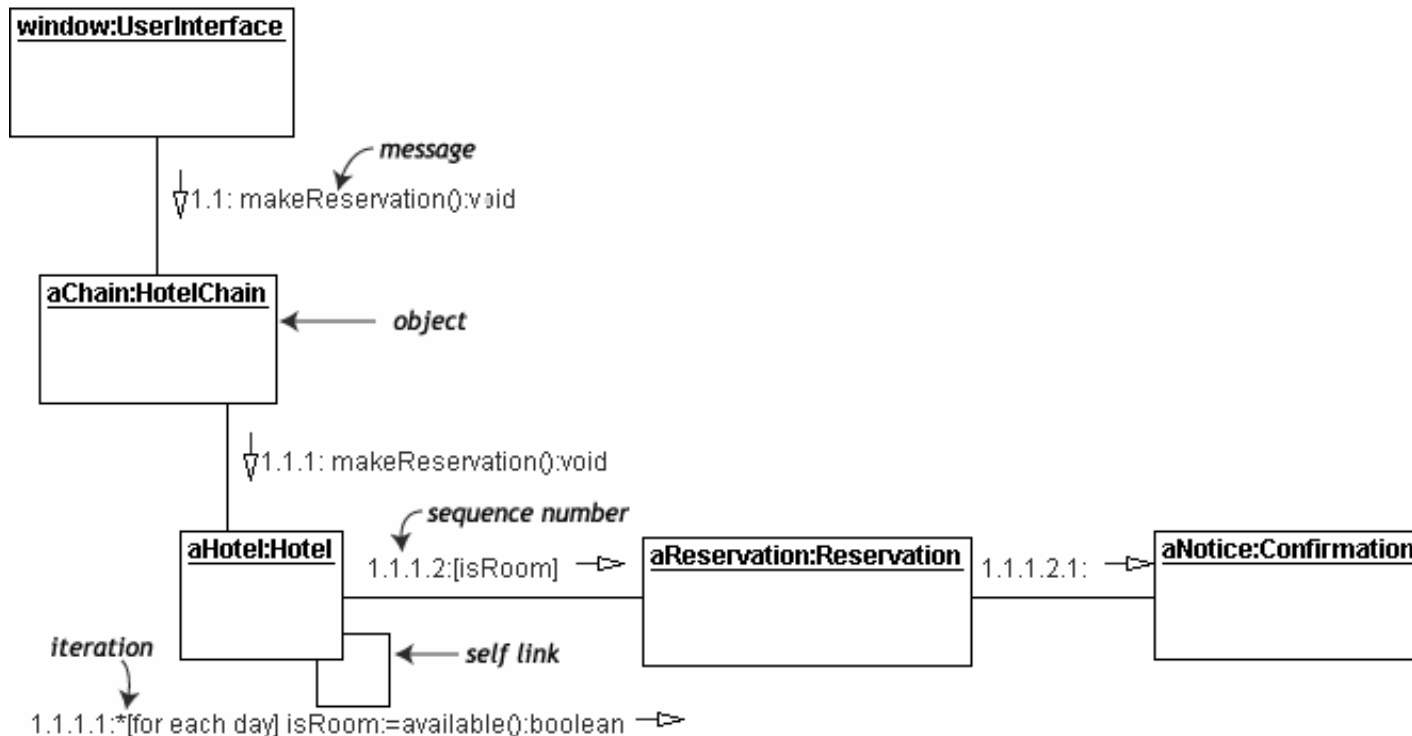
Communication/Collaboration Diagram



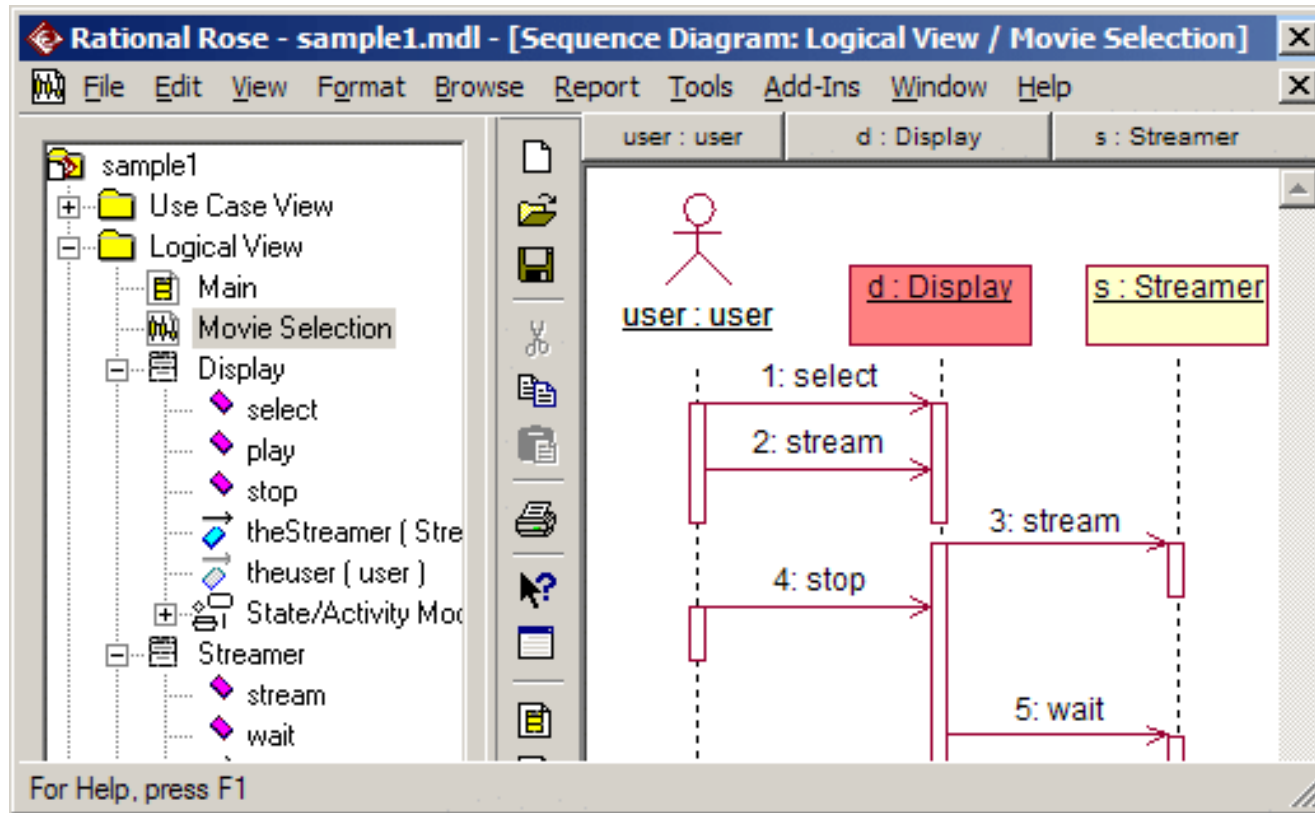
❖ «نمودار ارتباط»

Communication Diagram

مشابه نمودار توالی برای مدل سازی رفتار
دینامیک مورد کاربردی مورد استفاده
قرار می گیرد. زمانی که آن را با نمودار
توالی مقایسه می کنیم، می بینیم که
نمودار ارتباط تمرکز بیشتری روی
همکاری بین اشیا به جای توالی زمانی
این ارتباط دارد. این نمودارها در عمل از
نظر معناشناختی متناظر هم هستند، از
این رو برخی از ابزارهای مدل سازی آنها
مشترک هستند.



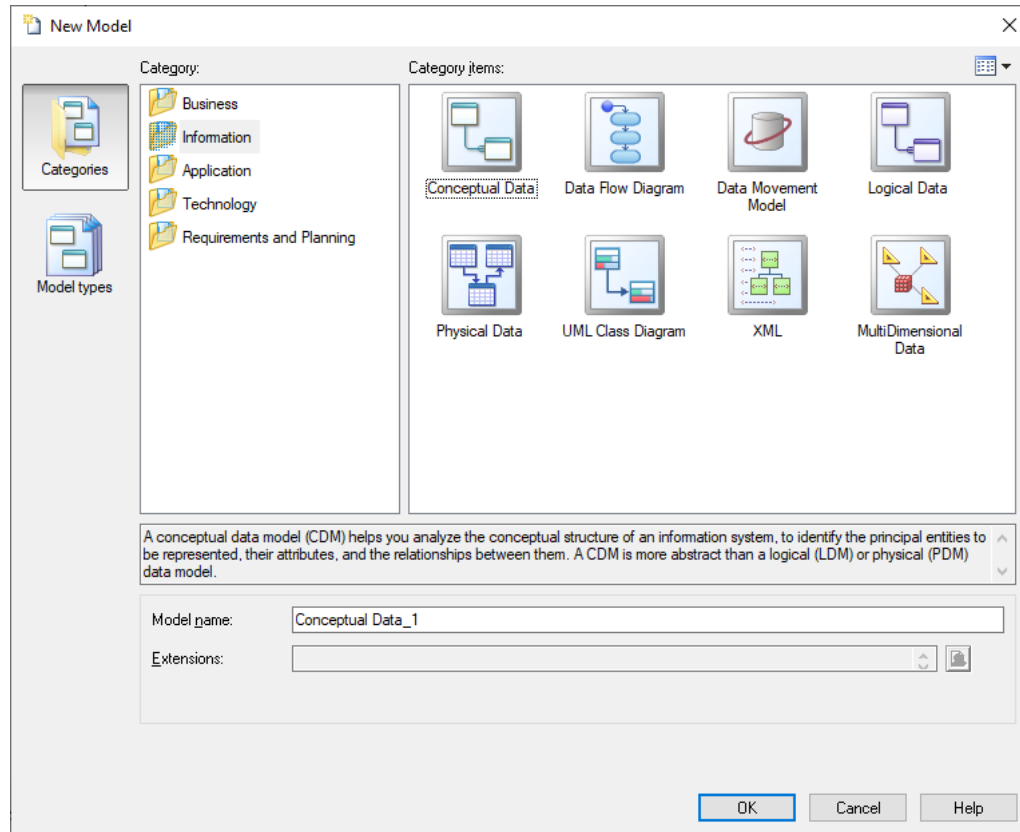
مدل سازی UML با نرم افزار Rational Rose



❖ Rational Rose ابزاری است که با هدف هدایت پروژه‌های بسیار بزرگ نرم‌افزاری و برای طراحی مدل‌های کد و ایجاد کدهای پیش‌تولید به کار می‌رود.

❖ با نرم‌افزار مذکور پیش از اینکه کدی برای یک پروژه بنویسیم، می‌توانیم سیستم مرتبط به آن پروژه را مدل کنیم و به این ترتیب اطمینان حاصل کنیم که سیستم موردنظر دارای معماری معتبری است.

مدل سازی UML با نرم افزار PowerDesigner



❖ **PowerDesigner** یکی از مهم ترین نیازهای دانشجویان و مهندسين نرم افزار، نرم افزارهایی جهت ساده تر کردن کار مدل سازی داده ها و نیز تحلیل و طراحی و نمایش گرافیکی سیستم های نرم افزاری است. شرکت Sybase یکی از بزرگترین شرکت های سازنده ابزارهای مهندسی نرم افزار است که برای مدل سازی داده ها، نرم افزار قدرتمند **Sybase PowerDesigner** را ارائه کرده است. این نرم افزار زبان مدلسازی UML را نیز پشتیبانی میکند و با ۶۰ دیتابیس مختلف نیز خروجی می دهد. این نرم افزار همچنین با اکثر محیط های برنامه نویسی مانند دات نت، جاوا، اکلیپس و ... میتواند تعامل داشته باشد.

