

فصل ششم

مدل سازی نیازمندی ها

جریان، رفتار، الگوها و برنامه های تحت وب

راهبردهای مدل سازی نیازمندی ها

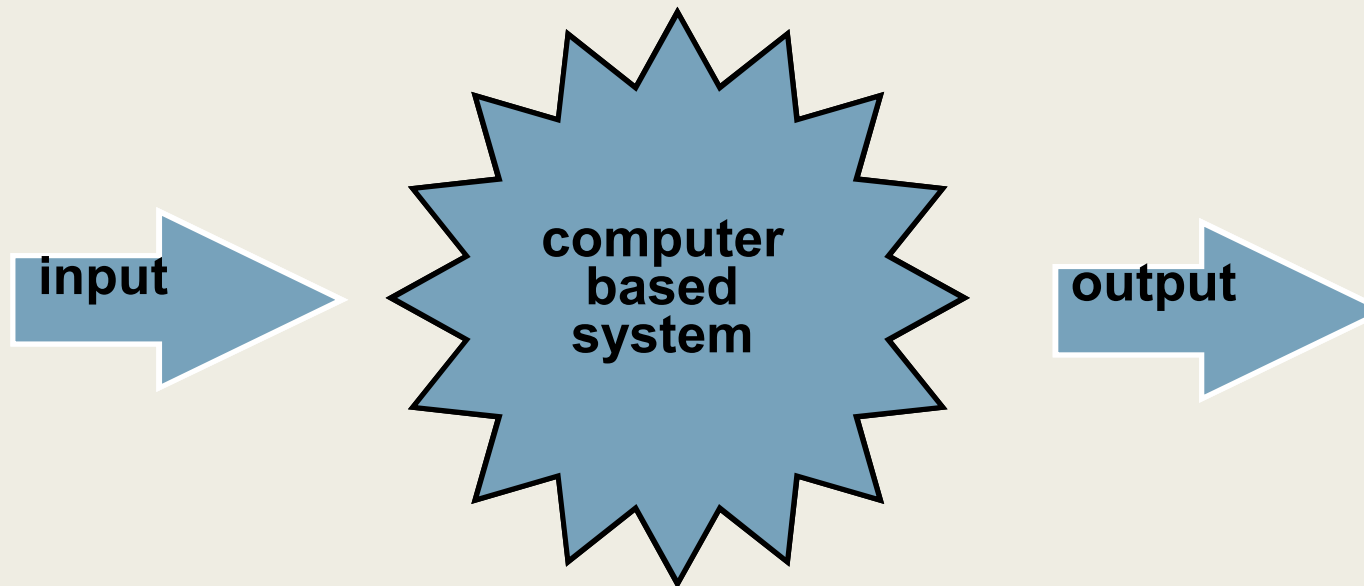
- یک دیدگاه مدل سازی نیازمندی ها، به نام **تحلیل ساخت یافته**، داده ها و فرایندهایی که داده ها را تبدیل می کنند، به صورت موجودیت های جداگانه در نظر می گیرند.
 - اشیاء داده ای به گونه ای مدل شده اند که ویژگی ها و روابط آنها را تعریف می کند.
 - فرایندهای که اشیاء داده ای را دستکاری می کنند به شیوه ای مدل می شوند که نشان می دهد چگونه داده ها را به عنوان جریان اشیاء داده ای از طریق سیستم منتقل می کنند.
- رویکرد دوم مدل سازی تحلیل، به نام **تحلیل شی گرا**، بر موارد زیر تمرکز دارد
 - تعریف کلاس ها و
 - شیوه ای که در آن کلاس ها با یکدیگر همکاری می کنند تا بر نیازهای مشتری تأثیر بگذارند.

مدل سازی جریان گرا

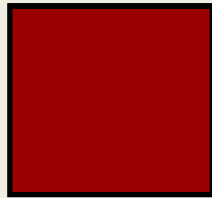
- نمایش می دهد چگونه اشیاء داده ای با حرکت در سیستم تغییر می کنند
- **نمودار جریان داده (DFD)** شکل نموداری است که مورد استفاده قرار می گیرد
- از دید بسیاری از افراد به عنوان یک رویکرد " قدیمی " است اما همچنان دیدگاهی از سیستم ارائه می دهد که منحصر به فرد است – باید برای تکمیل سایر عناصر مدل تحلیل استفاده شود.

مدل جریان

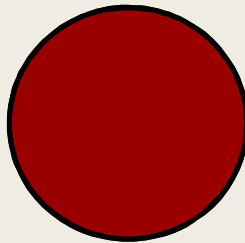
■ هر سیستم مبتنی بر کامپیوتر یک مدل داده است



نمادهای مدل سازی جریان



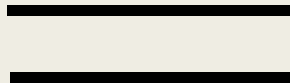
موجودیت خارجی



فرآیند



جریان داده



ذخیره داده

موجودیت خارجی

تولید کننده یا مصرف کننده داده

مثال: یک فرد، یک دستگاه، یک سنسور

مثالی دیگر: سیستم مبتنی بر کامپیوتر

داده باید همیشه از جایی شروع شود و باید همیشه به چیزی ارسال شود

فرآیند

یک مبدل داده (ورودی را به خروجی تغییر می دهد)

مثال: محاسبه مالیات، تعیین مساحت، گزارش قالب،

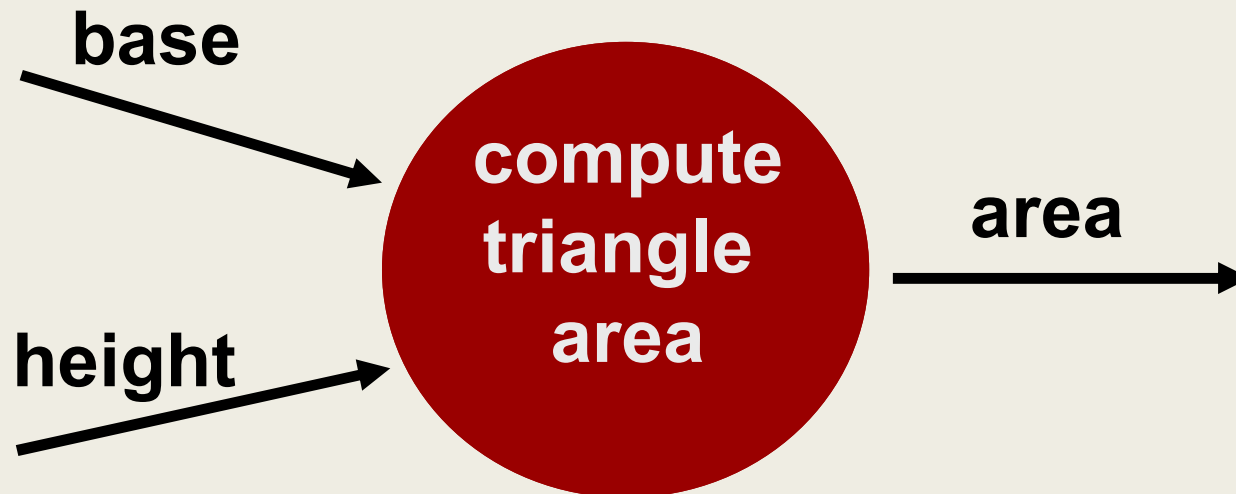
نمایش گراف

داده باید همیشه به شکلی پردازش شود تا عملکرد

سیستم محقق گردد

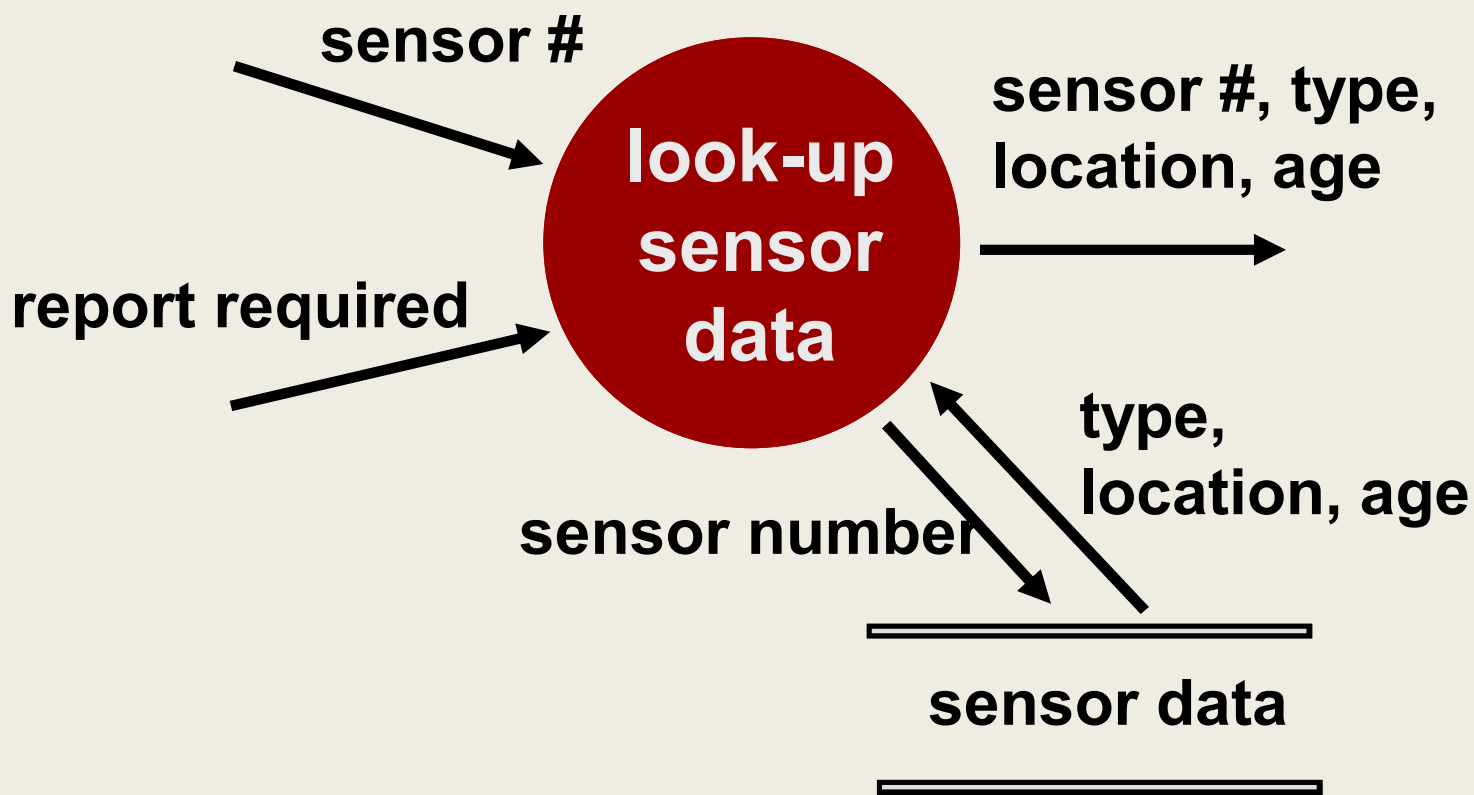
جریان داده

داده درون سیستم جریان می یابد، از ورودی آغاز می شود و به خروجی تبدیل می شود



ذخیره داده

داده اغلب برای استفاده آتی ذخیره می شود



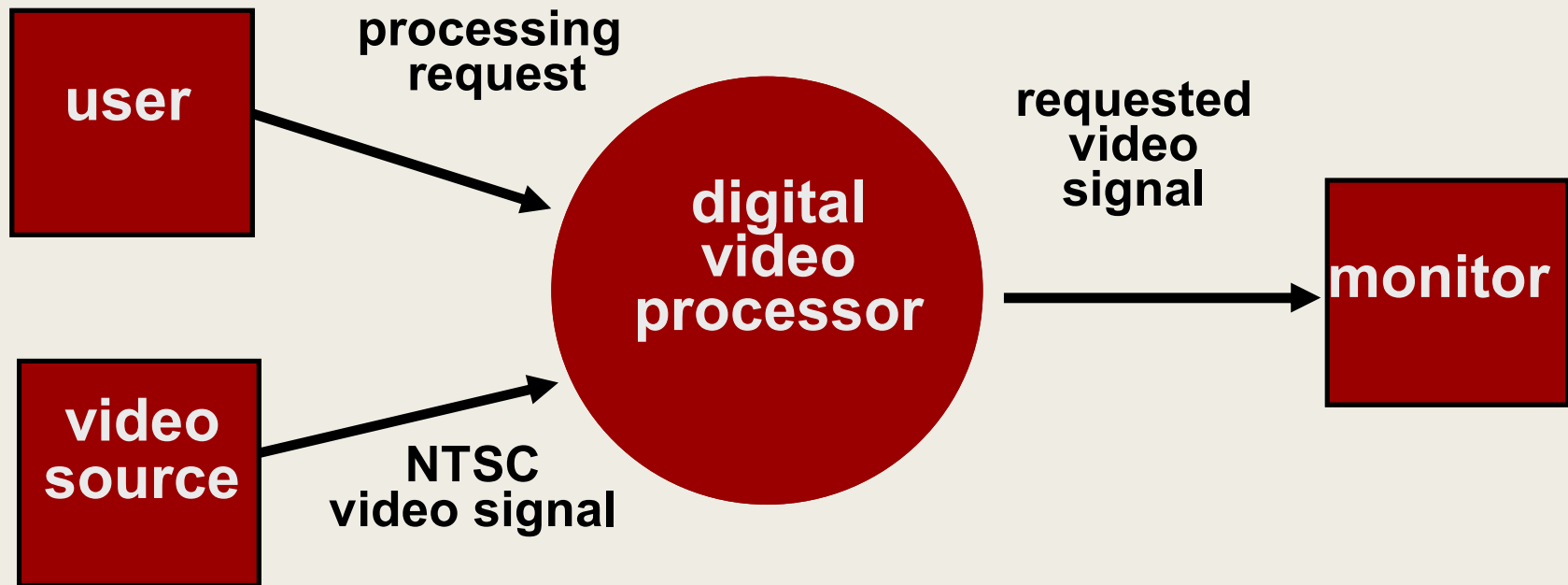
ترسیم نمودار جریان داده : راهنمایی

- تمام نمادها باید با نام های معنی دار مشخص شوند
- DFD از طریق تعدادی از سطوح جزئیات تکامل می یابد
- همیشه با یک نمودار سطح محتوا (با نام دیگر سطح صفر) آغاز کنید
- همیشه موجودیت های خارجی را در سطح صفر نشان دهید
- همیشه فلش های جریان داده را برچسب گذاری کنید
- منطق رویه ای را نشان نمی دهد

ساخت نمودار جریان داده سطح ۱ (DFD-۱)

- سناریوهای کاربر و/یا مدل داده ای را برای جداسازی اشیاء داده و استفاده از تجزیه دستوری برای تعیین "عملیات" بررسی کنید
- موجودیت های خارجی (تولید کنندگان و مصرف کنندگان داده) را تعیین کنید.
- یک نمودار جریان داده سطح صفر (DFD-0) ایجاد کنید

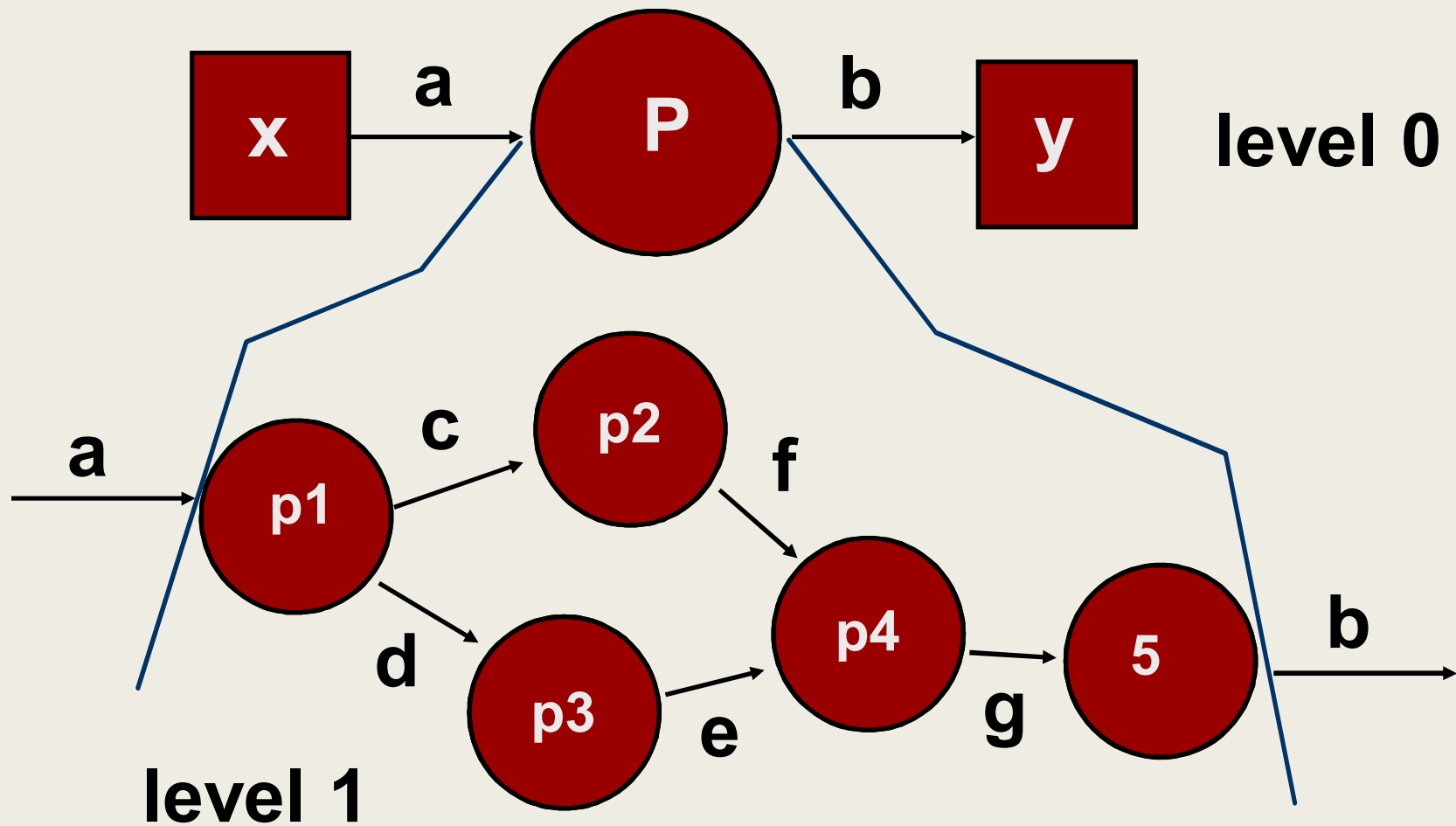
مثال نمودار جریان داده سطح صفر (DFD-0)



ساخت نمودار جریان داده سطح ۲ (DFD-II)

- تشریح غیر کتابی تبدیل را بنویسید
- برای تعیین تبدیل های سطح بعد آن را تجزیه کنید
- جریان را برای حفظ پیوستگی جریان داده "متوازن" کنید.
- یک نمودار جریان داده سطح یک (DFD-I) ایجاد کنید
- نرخ گسترش ۵:۱ (تقریبا) استفاده کنید

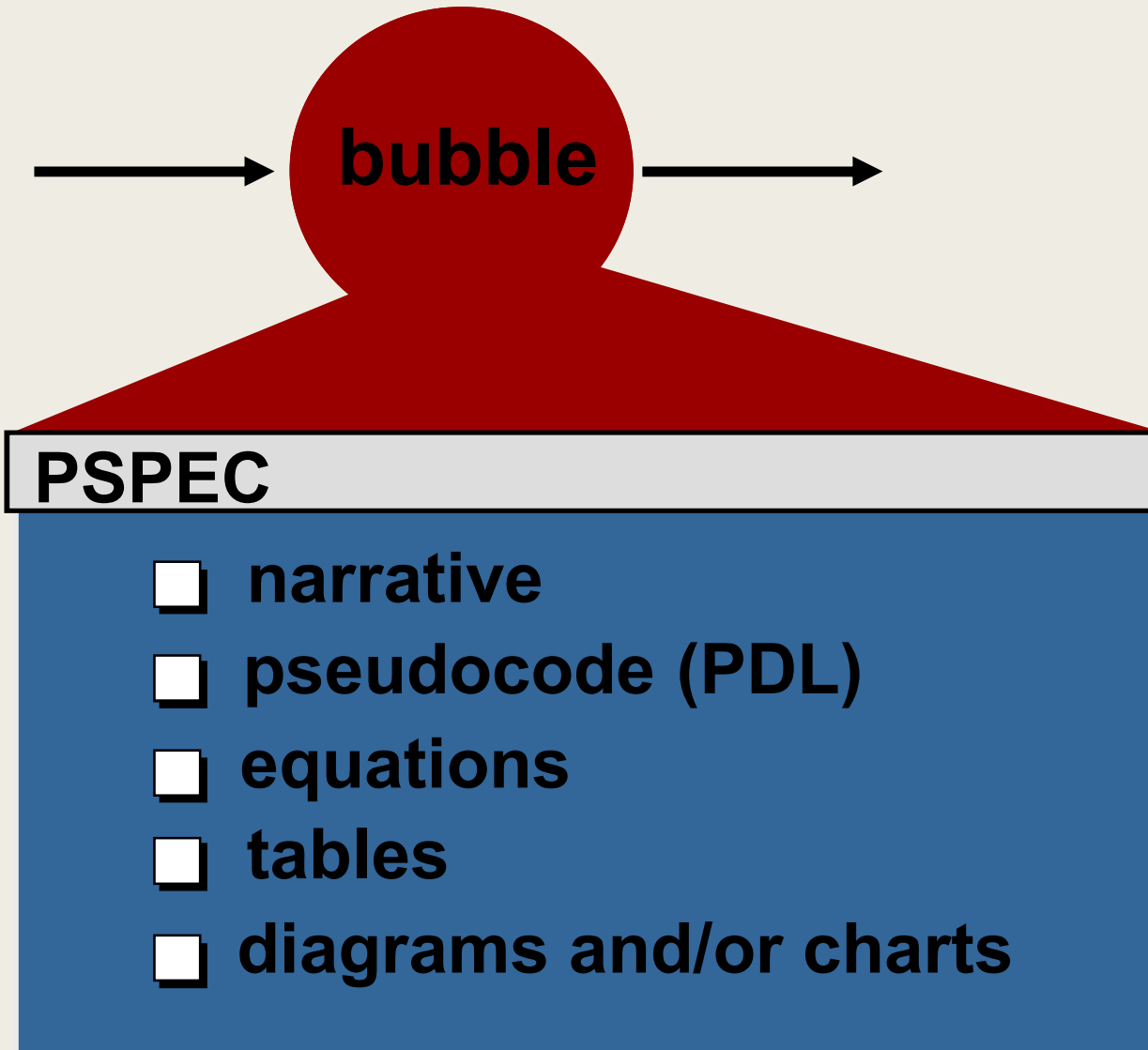
سلسله مراتب جریان داده



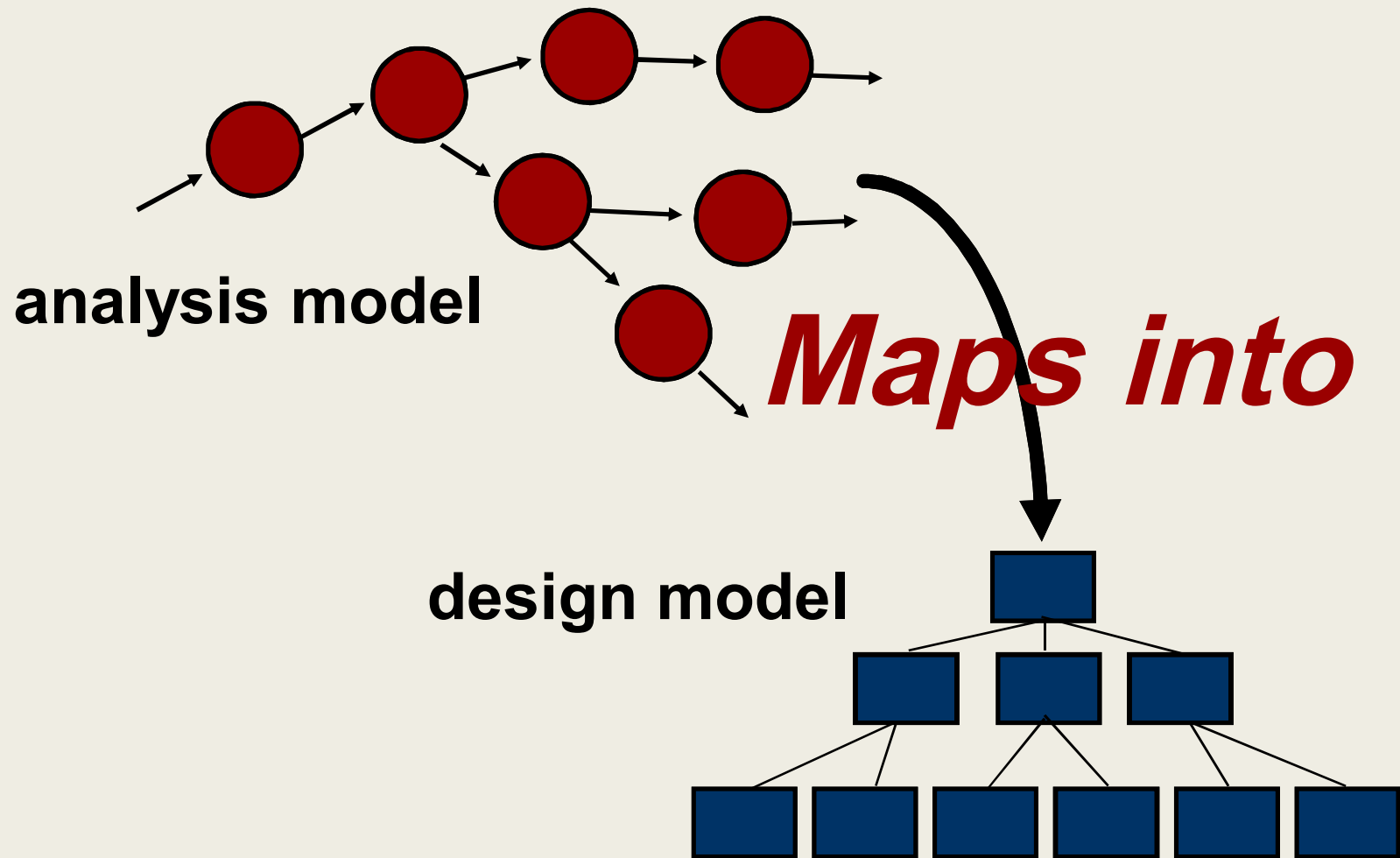
نکات مدل سازی جریان

- هر حساب تصفيه شده تا زمانی که فقط يك چیز را انجام دهد
- نسبت گسترش با افزایش شماره سطوح کاهش می یابد
- اکثر سیستم ها برای يك مدل جریان مناسب بين ۳ تا ۷ سطح نیاز دارند.
- يك آيتم جریان اطلاعات تک (فلش) ممکن است با افزایش سطح گسترش یابد (لغت نامه داده اطلاعات را فراهم می کند)

خصوصیات فرآیند (PSPEC)



نمودارهای جریان داده: نگاهی به جلو



مدل سازی جریان کنترل

- **"رویدادها"** و فرآیندهای مدیریت رویدادها را نشان می دهد
- یک "رویداد" یک شرط دودویی (Boolean) است که می تواند با موارد زیر محقق شود:
 - لیست کردن تمام حسگرهایی که توسط نرم افزار "خوانده" می شوند.
 - فهرست کردن همه شرایط وقفه
 - لیست کردن تمام "سوئیچ هایی" که توسط اپراتور فعال می شوند.
 - فهرست کردن تمام شرایط داده ای
 - به یاد آوردن تجزیه اسم/فعل که در روایت پردازش اعمال شد، بررسی تمام "موارد کنترل" به عنوان ورودی/خروجی های CSPEC ممکن

مشخصات کنترلی (CSPEC)

■ CSPEC می تواند موارد زیر باشد:

■ نمودار حالت (جنبه های ترتیبی)

■ جدول تغییر حالت

■ جدول تصمیم

■ جدول فعال سازی

جنبه های ترکیبی

مدل سازی رفتار

■ مدل رفتار نشان می دهد چگونه نرم افزار به رویدادهای خارجی یا محرک ها پاسخ می دهد. برای ایجاد مدل، تحلیلگر باید مراحل زیر را انجام دهد:

- تمام UseCase ها را ارزیابی کنید تا کاملاً متوجه توالی تعاملات درون سیستم شوید.
- رویدادهایی را شناسایی کنید که توالی تعاملات را به راه می اندازند و درک کنید که این رخدادها چگونه با اشیائی خاص مرتبط اند.
- یک توالی برای هر UseCase ایجاد کنید.
- یک نمودار وضعیت برای سیستم بسازید.
- مدل رفتار را برای تحقیق صحت و ثبات بررسی کنید.

نمایش های وضعیت

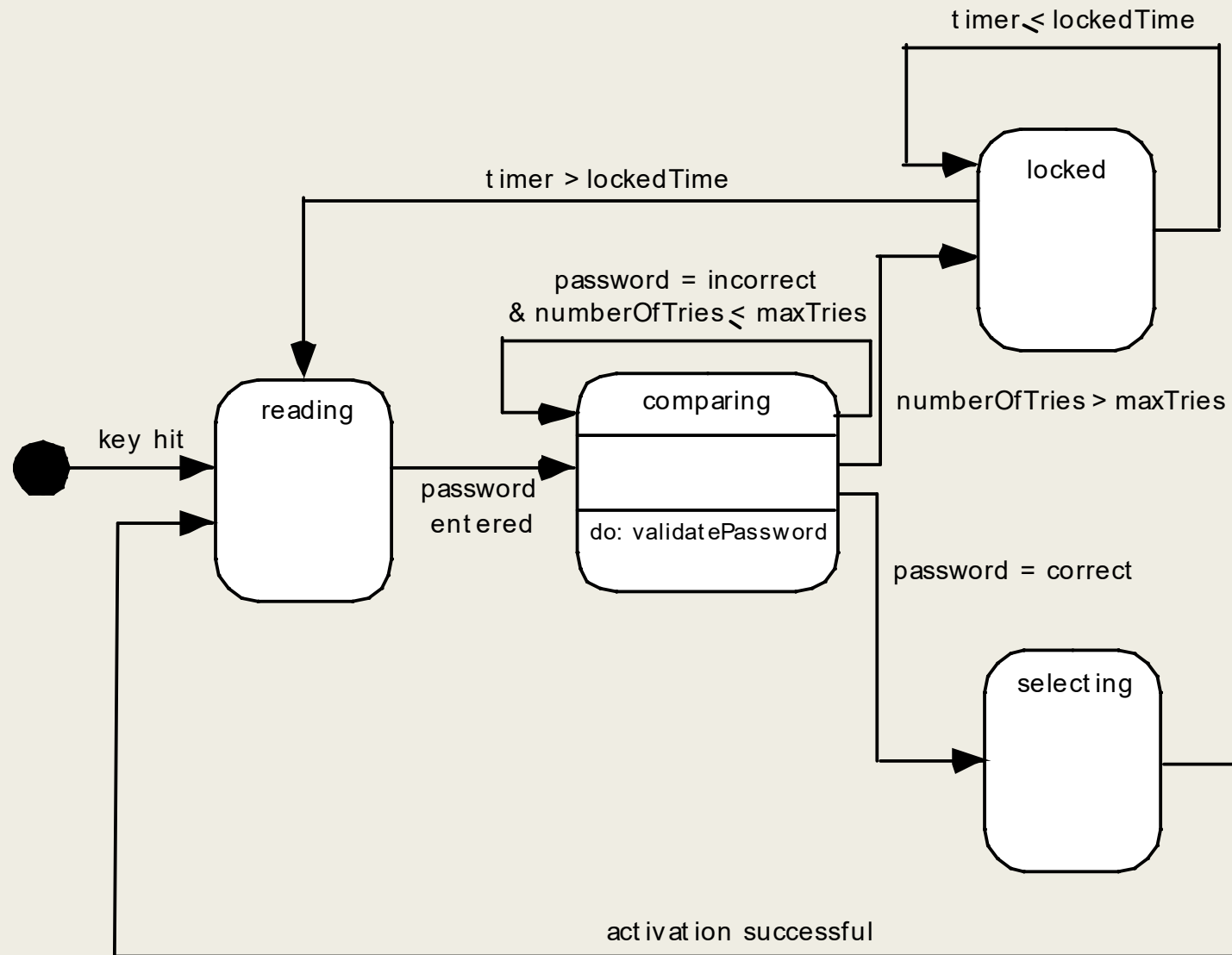
■ در زمینه مدل سازی رفتار، دو ویژگی متفاوت حالت ها باید مورد توجه قرار گیرد:

- وضعیت هر کلاس به گونه ای که سیستم عملکردش را انجام می دهد و
- وضعیت سیستم آن گونه که از بیرون سیستم آن گونه عملکردش را انجام می دهد

■ وضعیت یک کلاس با دو جنبه منفعل و فعال مواجه می شود

- **حالت منفعل** به طور ساده وضعیت فعلی تمام ویژگی های یک شی است.
- **حالت فعال** یک شی نشانگر وضعیت فعلی شی است آن گونه که یک تداوم تبدیل یا پردازش را تحمل می کند.

نمودار حالت برای کلاس صفحه کنترل



حالت یک سیستم

- **حالت** – مجموعه ای از شرایط قابل مشاهده که رفتار یک سیستم را در یک زمان معین مشخص می کند
- **انتقال حالت** – جابجایی از یک حالت به حالت دیگر
- **رخداد** – وقوعی است که موجب می شود سیستم برخی رفتارهای قابل پیش بینی را نمایش دهد
- **عمل** – فرآیندی که به عنوان یک نتیجه یک انتقال رخ می دهد

مدل سازی رفتار

- لیستی از وضعیت های مختلف یک سیستم تهیه کنید
(سیستم چگونه رفتار می کند؟)
- نشان دهید چگونه سیستم از یک حالت به حالت دیگر
منتقل می شود (سیستم چگونه تغییر وضعیت می دهد؟)
 - رویداد را نشان دهید
 - عمل را نشان دهید
- یک نمودار حالت یا نمودار توالی ترسیم کنید

نمودار توالی

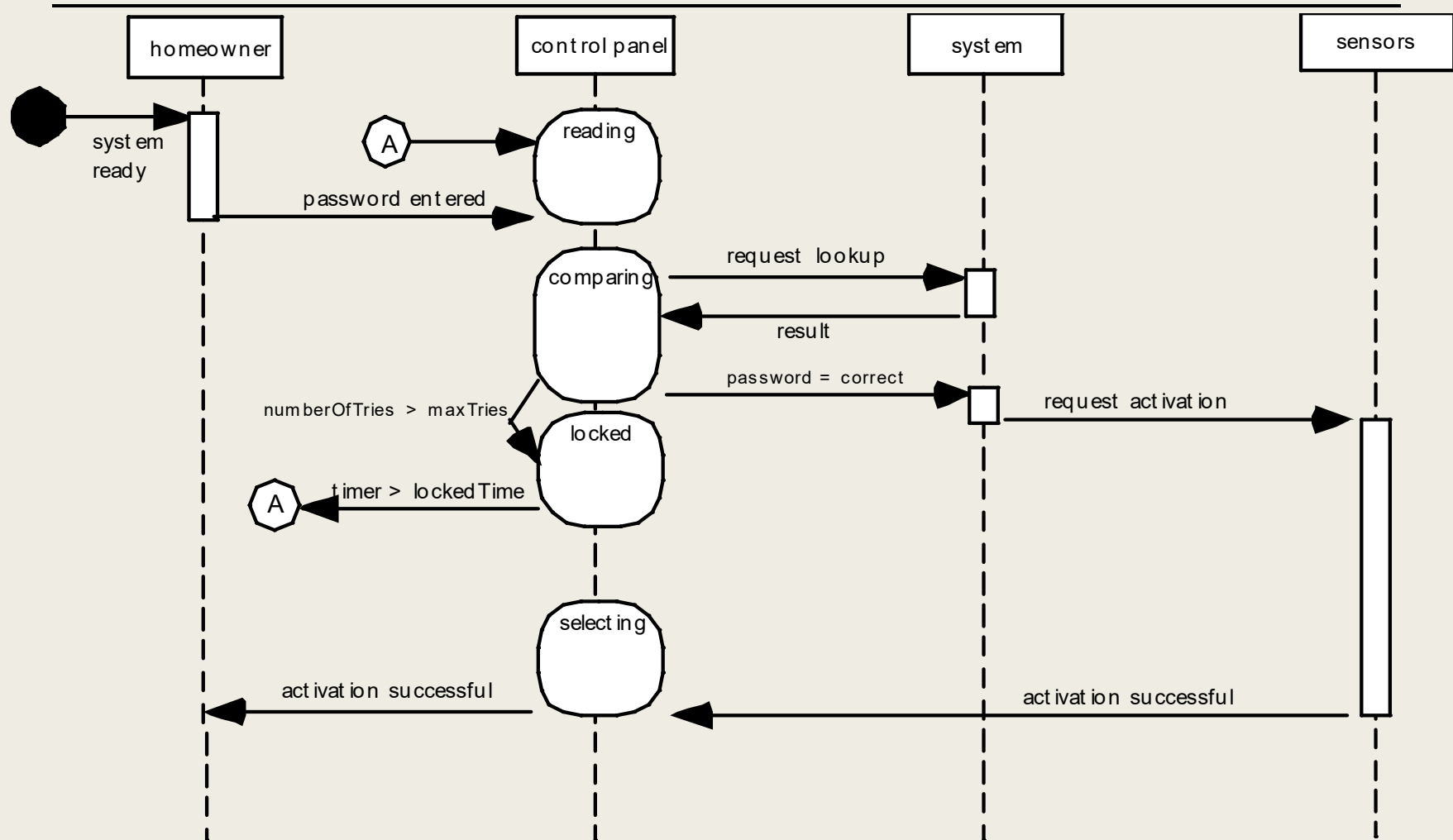


Figure 8.27 Sequence diagram (partial) for *SafeHome* security function 25