

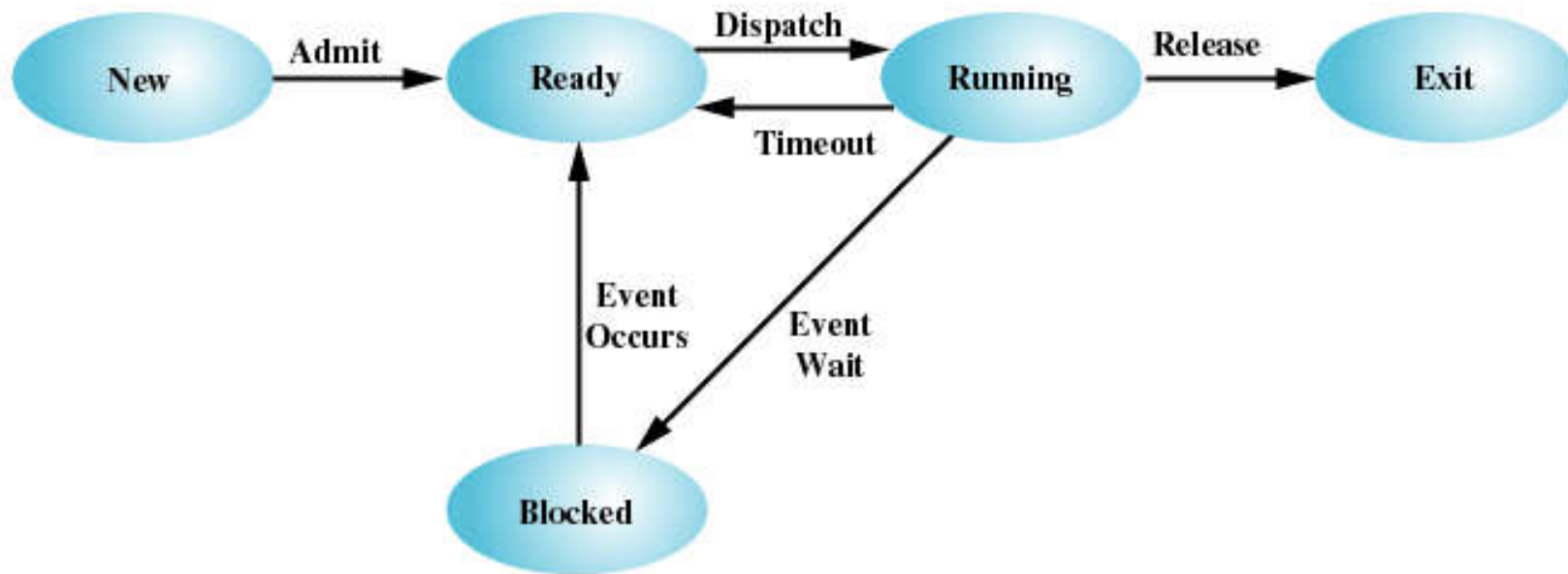
به نام خدا

فصل سوم شرح و کنترل فرآیند (بخش دوم)

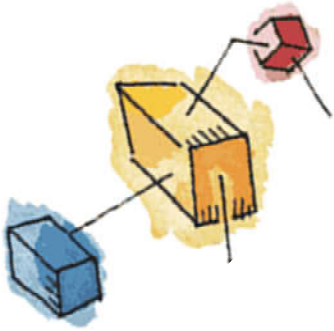
Process Description and Control



مدل ۵-حالته فرآیند



مدل ۵ حالته فرآیند



• جدید - New

– تازه ایجاد شده و به علت عدم بار گذاری (کامل) در حافظه، قابل اجرا نیست.

• آماده - Ready

– آماده اجرا بوده و تنها نیازمند پردازنده است.

• اجرا - Running

– پردازنده را در اختیار داشته و در حال اجرا است.

• مسدود - Blocked

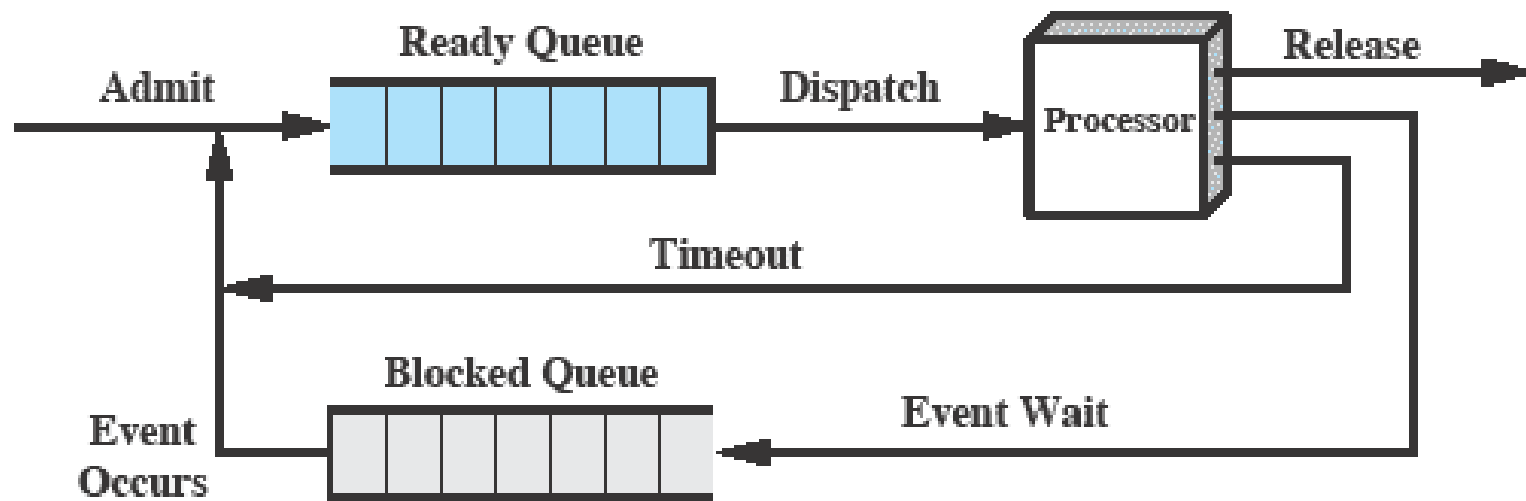
– تا زمان وقوع رویداد مشخصی (همچون اتمام عمل ورودی/خروجی) نمی تواند اجرا شود.

• خروج - Exit

– فرآیند به هر دلیل خاتمه یافته است.

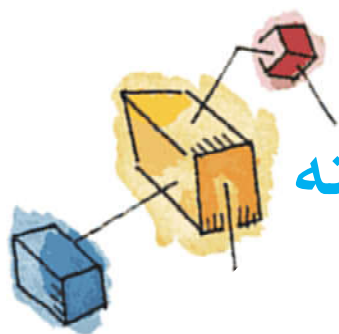


استفاده از دو صف

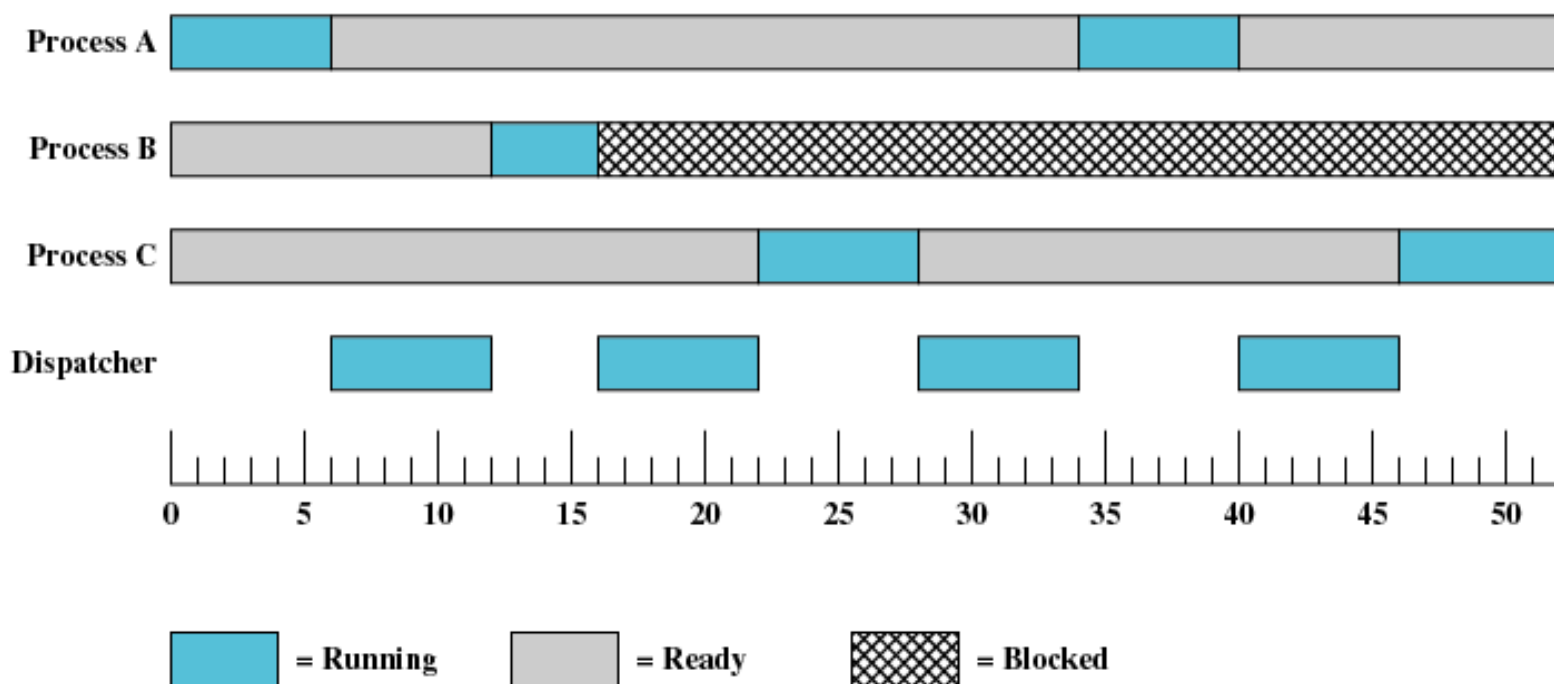


(a) Single blocked queue





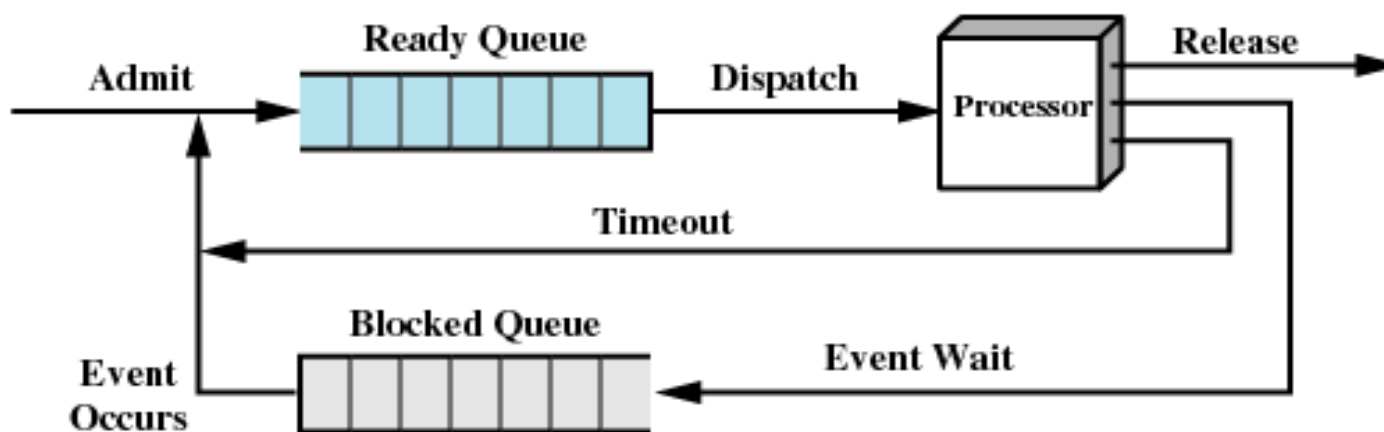
چگونگی اجرای فرآیندهای نمونه تحت مدل ۵-حالت



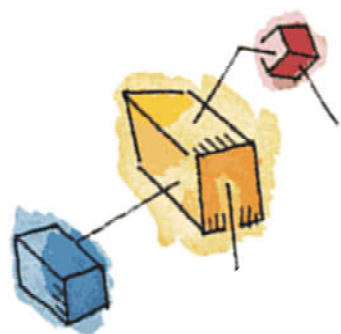
صف بندی در مدل ۵-حالته

• صف مسدود واحد

- تمام فرآیندهای مسدود در یک صف واحد قرار می گیرند.
- با بروز یک رویداد، صف مسدود جستجو می شود و فرآیند منتظر آن رویداد، یافته شده و از صف خارج می شود و به صف آماده منتقل می شود (اگر نوبتش باشد).



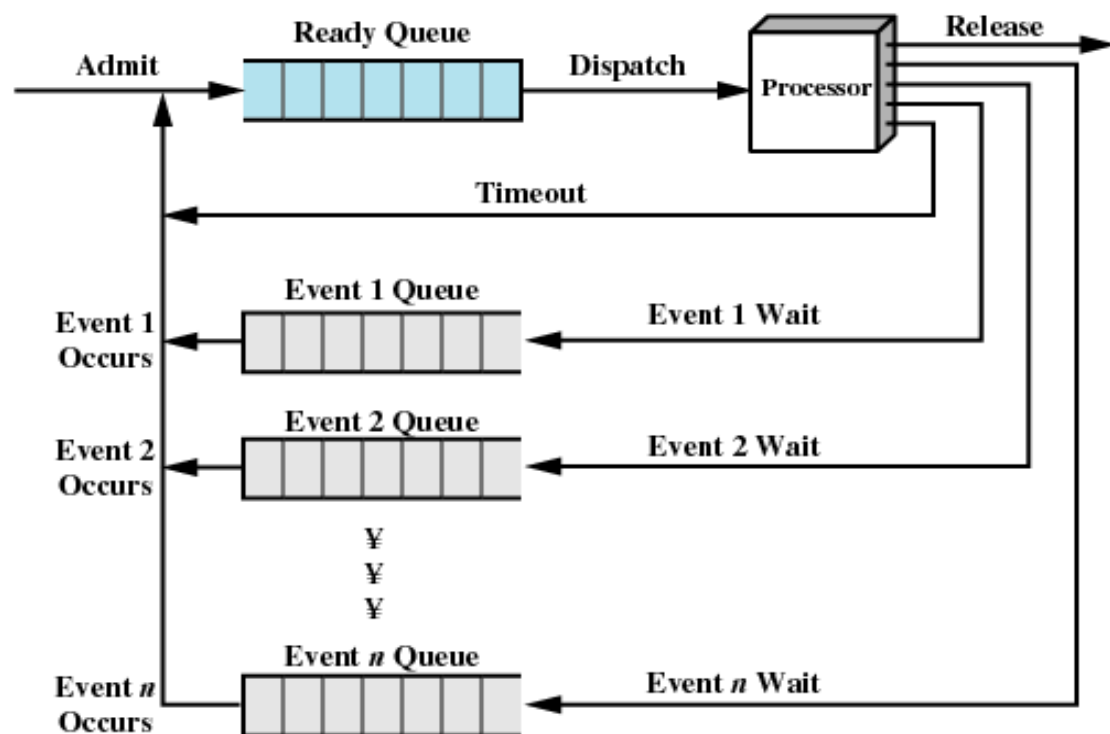
(a) Single blocked queue

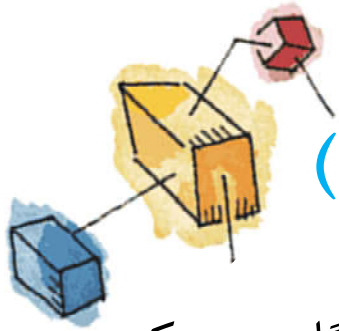


صف بندی در مدل ۵-حالته

• صف مسدود چندگانه

- به ازای هر رویداد یک صف در نظر گرفته می شود.
- با بروز رویداد، فرآیندی (که نوبتش است) از صف مربوط به آن رویداد، به صف فرآیندهای قابل اجرا منتقل می شود.



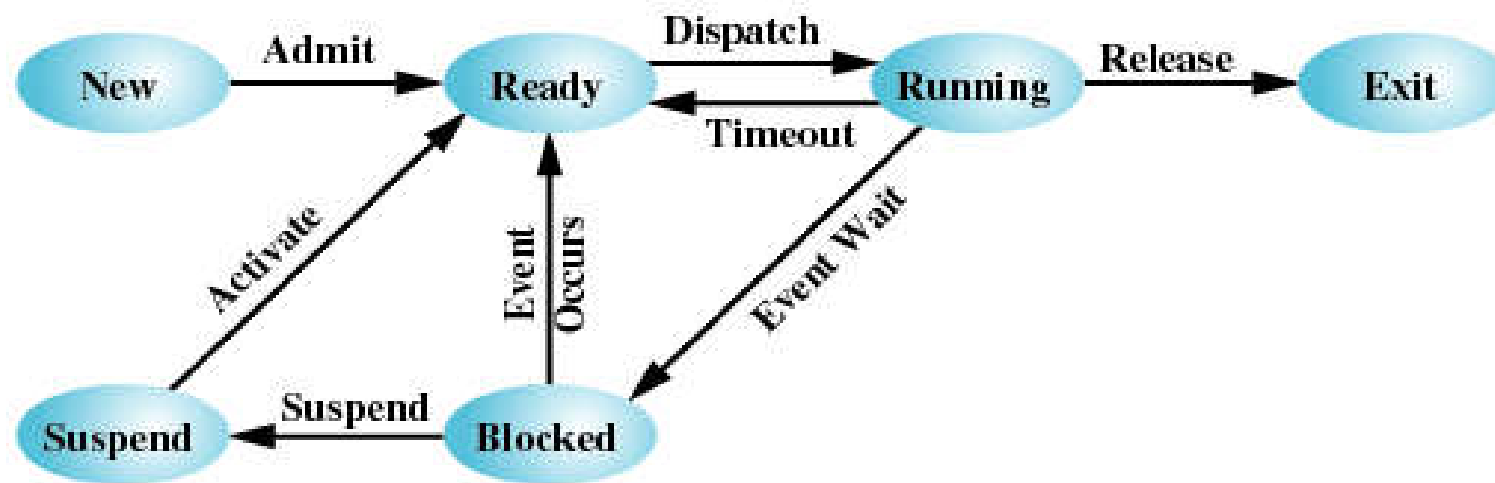


فرآیندهای معلق (Suspended Processes)

- پردازنده ها بسیار سریعتر از دستگاه های ورودی/خروجی اند، بنابراین گاهی ممکن است تمام فرآیندها برای عملیات ورودی/خروجی منتظر بمانند.
- می توان فرآیندهای منتظر را به حافظه ثانویه (دیسک) انتقال داد تا فضای آزاد بیشتری در حافظه اصلی داشته باشیم.
- فرآیندهای مسدود انتقال داده شده به دیسک را فرآیندهای **معلق** می گوئیم.
- دو حالت جدید مطرح می شود:
 - **معلق و مسدود:** فرآیند بر روی دیسک در انتظار یک رخداد است.
 - **معلق و آماده:** فرآیند بر روی دیسک است، اما به محض بار شدن در حافظه قابل اجرا خواهد بود.



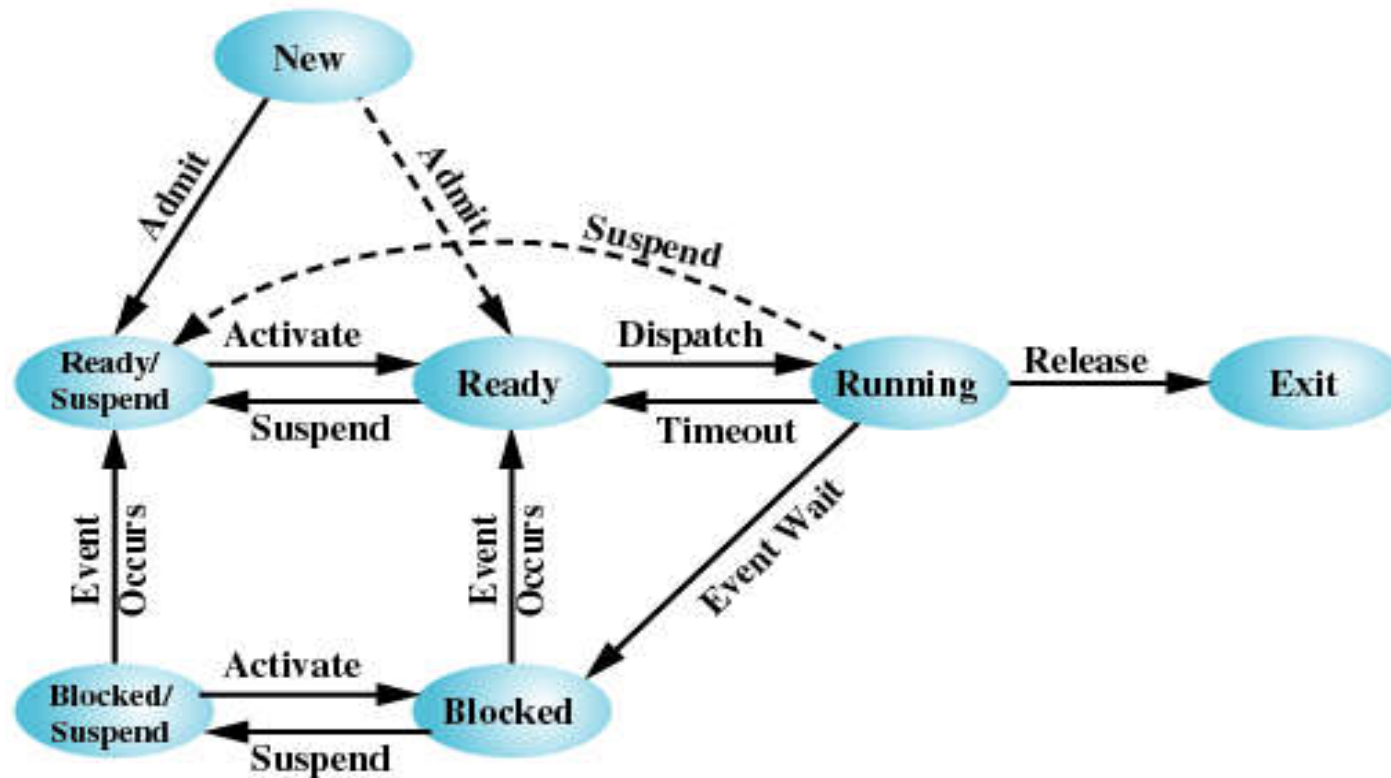
یک صف معلق



(a) With One Suspend State



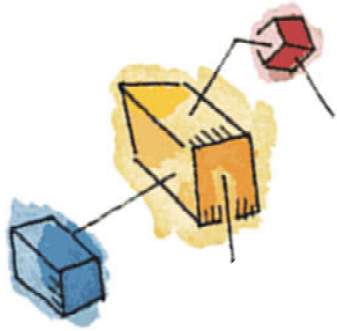
دو صف معلق



(b) With Two Suspend States



دلایل تعلیق فرآیند



- مبادله (Swapping)

- برای آوردن فرآیند آماده به اجرا نیاز به فضای حافظه آزاد است.

- دلیل دیگر سیستم عامل

- ممکن است سیستم عامل فرآیندی که مزنون به ایجاد مشکل است را معلق کند.

- درخواست کاربر محاوره ای

- ممکن است کاربر بخواهد به منظور اشکال زدایی یا استفاده از منابع، اجرای برنامه ای را معلق کند.

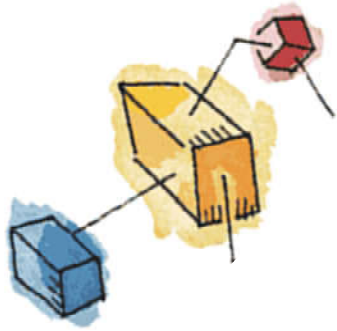
- زمان بندی

- ممکن است فرآیندی به طور دوره ای اجرا شود و هنگام انتظار برای دوره بعدی به صورت معلق باشد.

- درخواست فرآیند پدر

- ممکن است فرآیند پدر به منظور بررسی، ایجاد تغییر، و یا ایجاد هماهنگی، فرآیندهای فرزند خود را معلق کند.

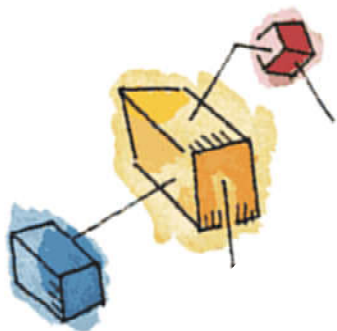




سرفصل مطالب

- فرآیند چیست؟
- حالات فرآیند
- شرح فرآیند
- کنترل فرآیند
- اجرای سیستم عامل
- مدیریت فرآیند در UNIX SVR4





فرآیندها و منابع

- فرآیند P_1 در حال اجرا است.
- فرآیند P_2 به دستگاه I/O ای نیاز دارد که به P_1 داده شده است و مسدود است.
- فرآیند P_n بیرون از حافظه اصلی است و معلق است.

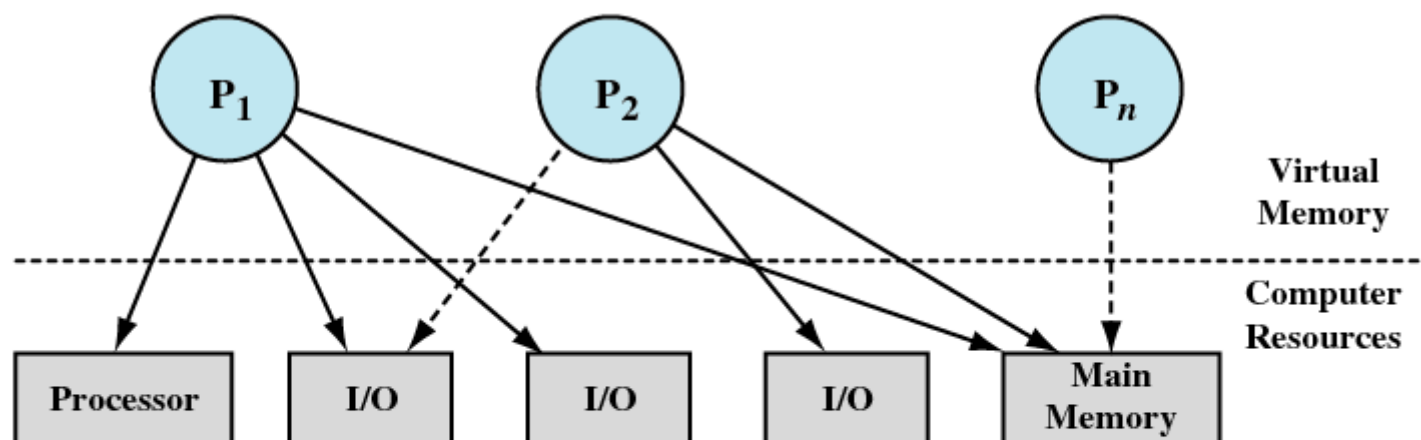
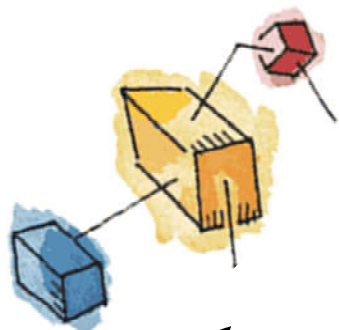


Figure 3.10 Processes and Resources (resource allocation at one snapshot in time)

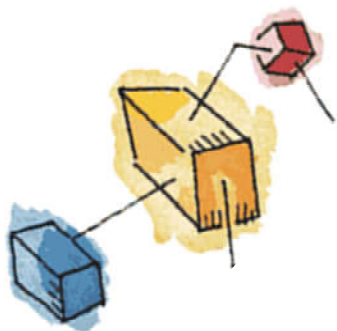


ساختار کنترلی سیستم عامل

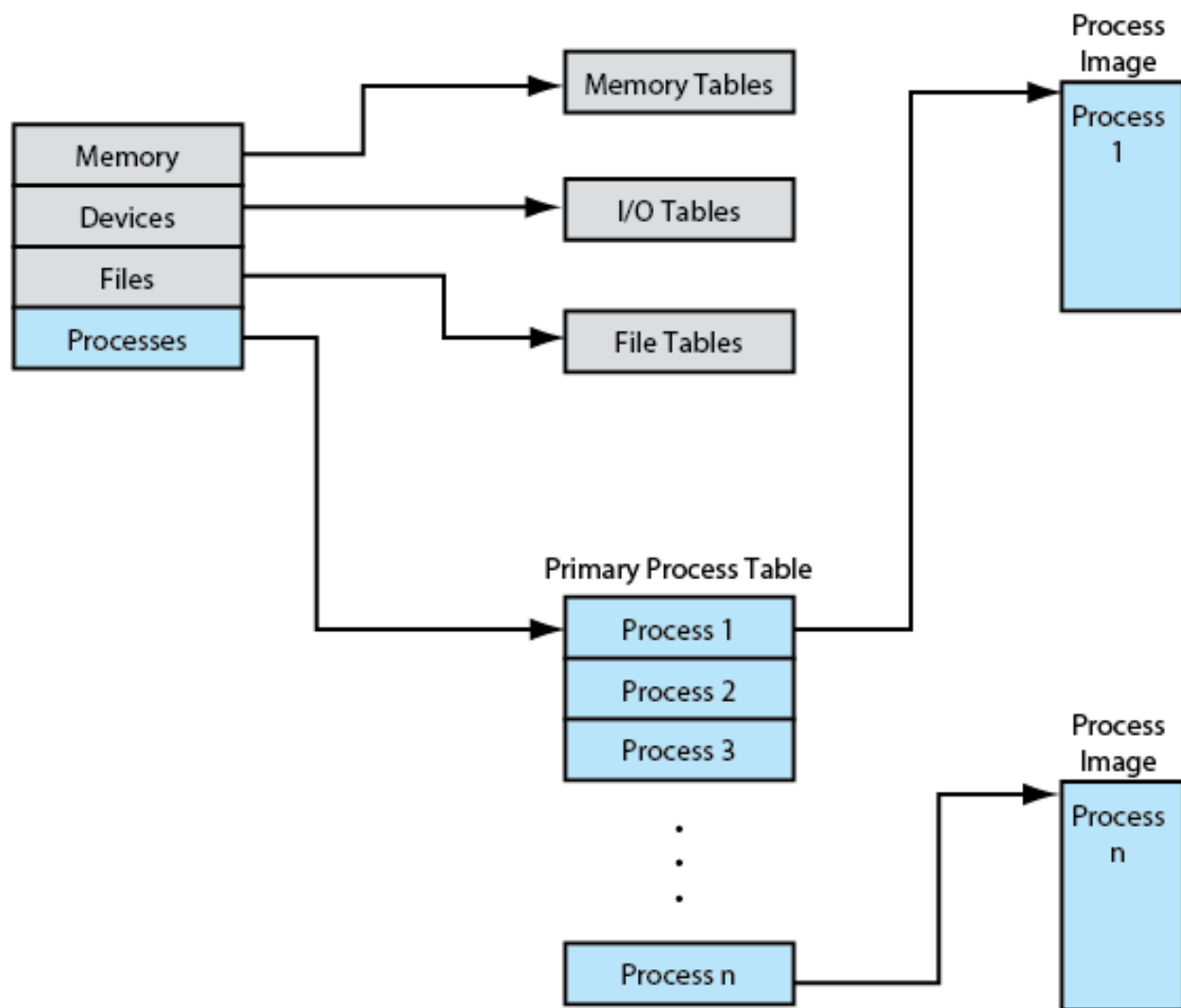


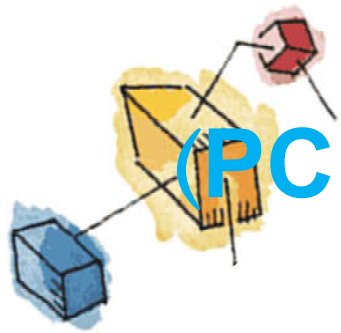
- سیستم عامل برای مدیریت فرآیندها و منابع، باید اطلاعاتی در مورد وضعیت کنونی هر منبع و فرآیند داشته باشد.
- برای تهیه این اطلاعات سیستم عامل برای هر موجودیتی که مدیریت می کند، جداول اطلاعاتی لازم را ساخته و نگهداری می کند.





ساختار عمومی جداول کنترلی سیستم عامل

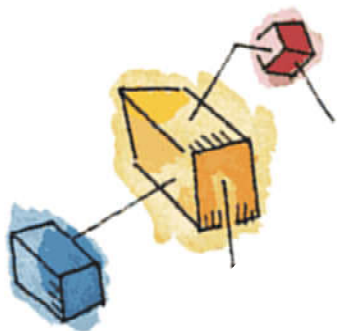




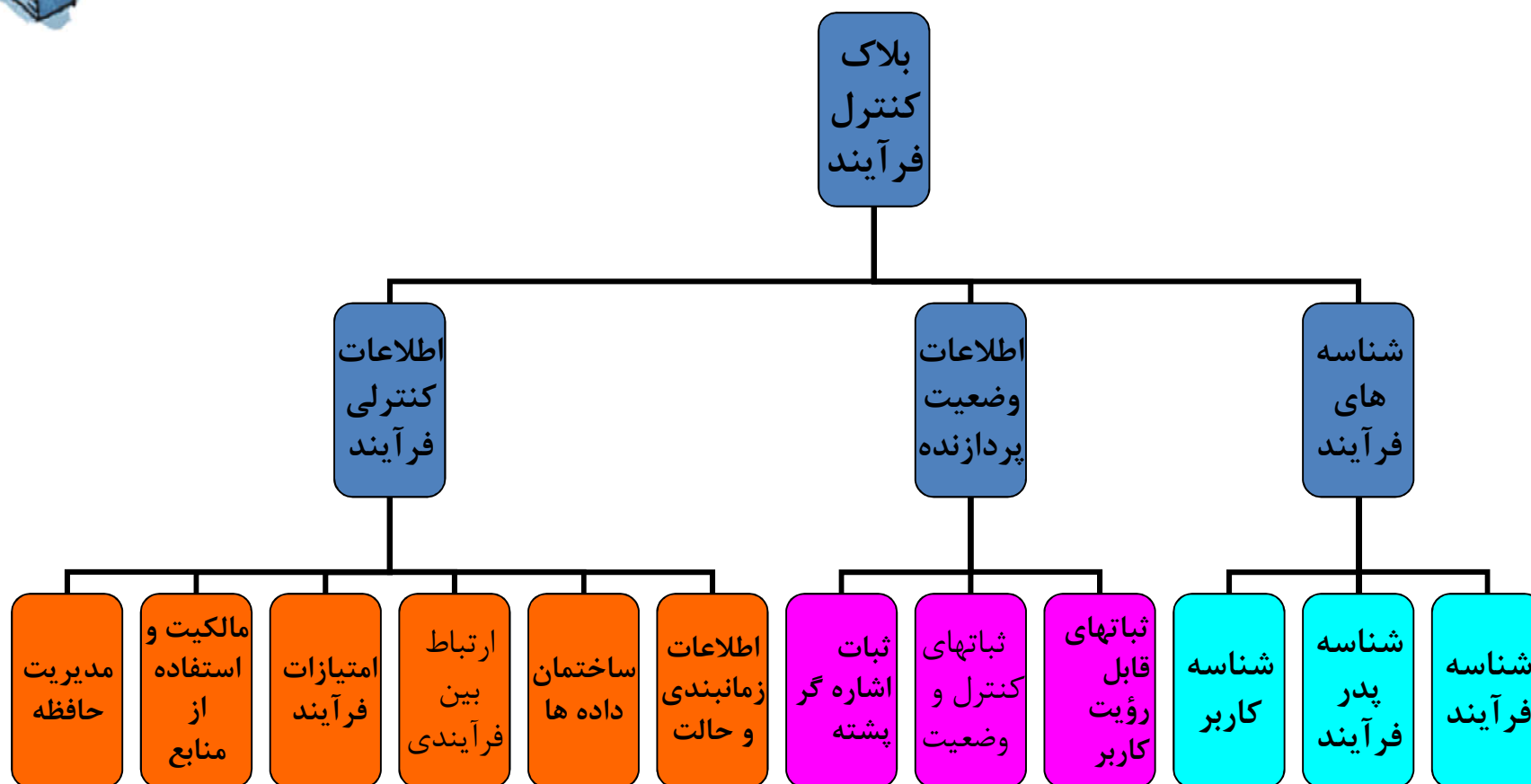
اجزای متداول بلاک کنترل فرآیند (PCB)

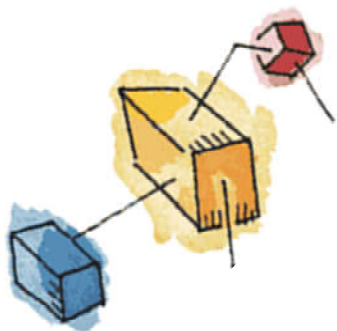
- شناسه های فرآیند
- اطلاعات وضعیت پردازنده
- اطلاعات کنترلی فرآیند



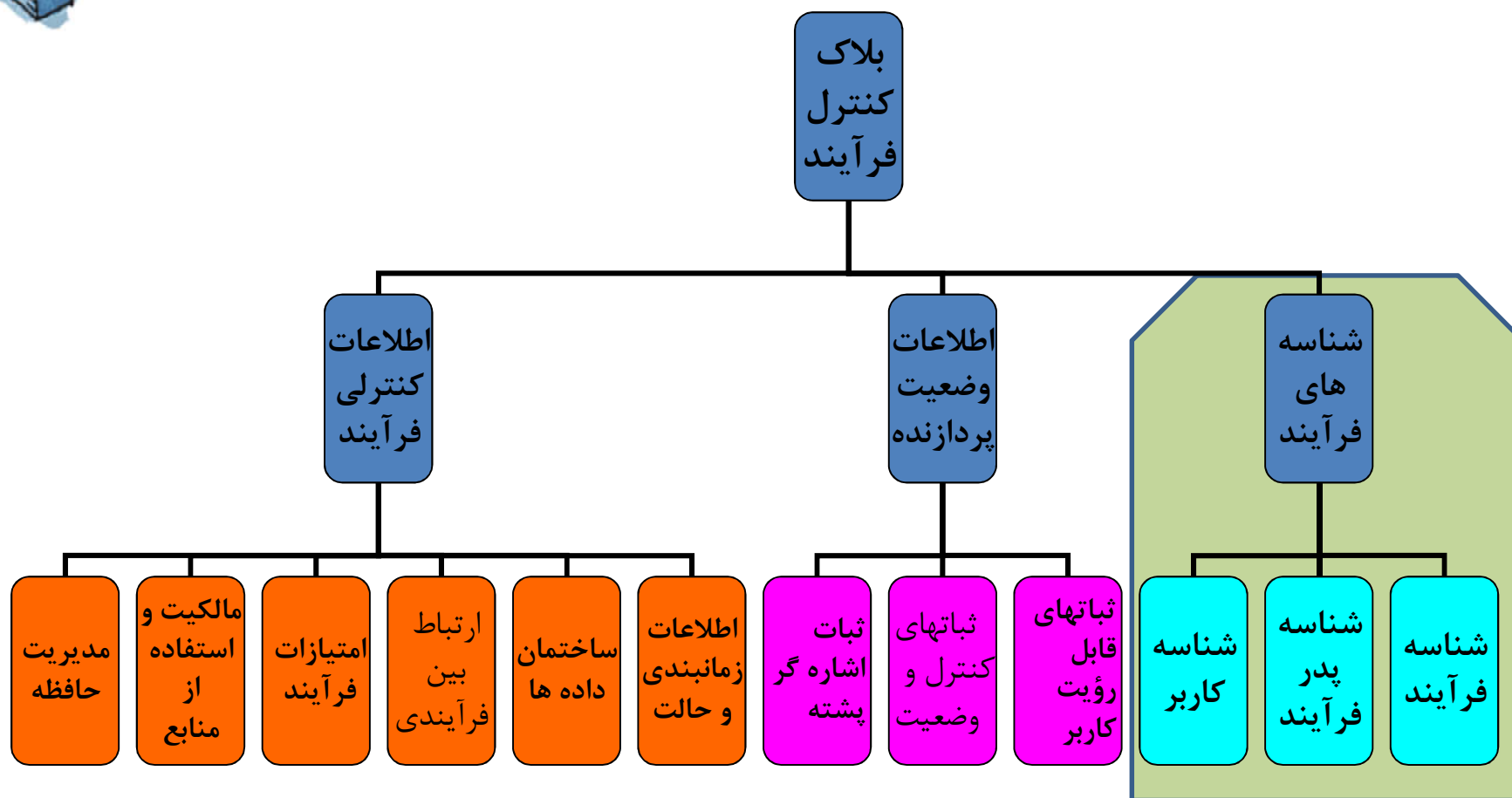


اجزای متداول بلاک کنترل فرآیند (PCB)

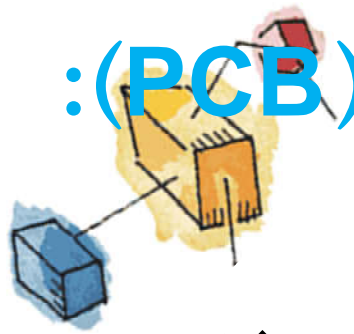




اجزای متداول بلاک کنترل فرآیند (PCB)



اجزای متداول بلاک کنترل فرآیند (PCB): شناسه های فرآیند



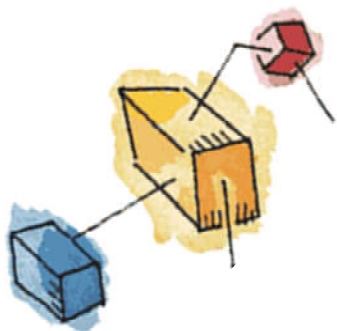
شناسه های عددی زیر ممکن است با جدول کنترل فرآیند ذخیره شود:

– شناسه فرآیند

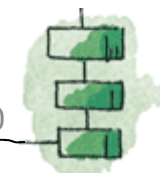
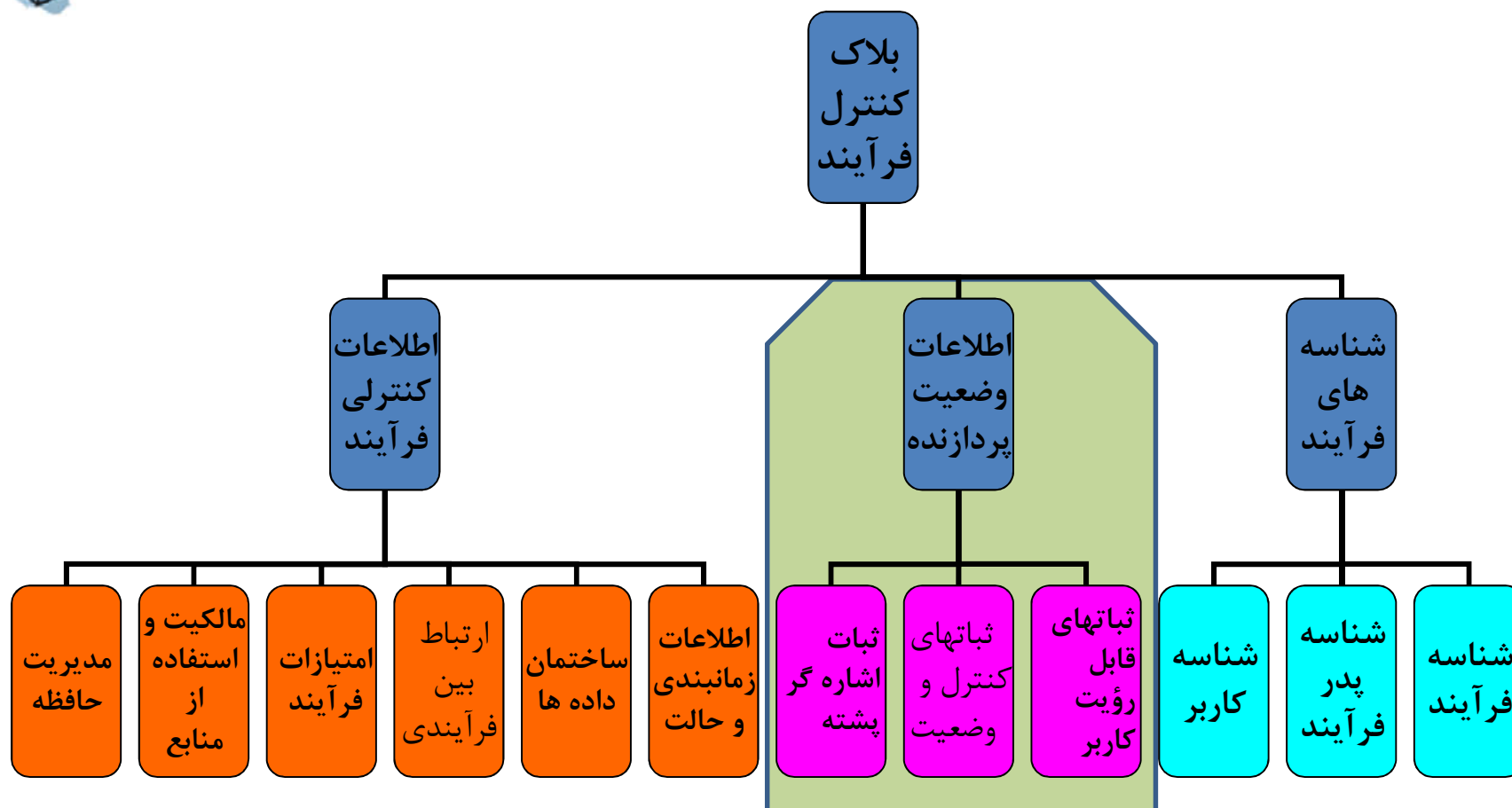
– شناسه فرآیندی که این فرآیند را ایجاد کرده است (فرآیند پدر)

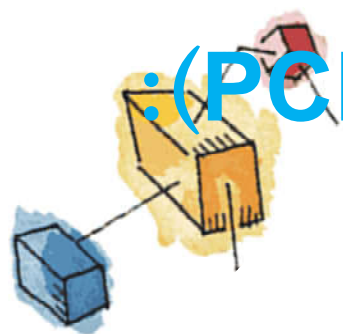
– شناسه کاربر





اجزای متداول بلاک کنترل فرآیند (PCB)





اجزای متداول بلاک کنترل فرآیند (PCB):

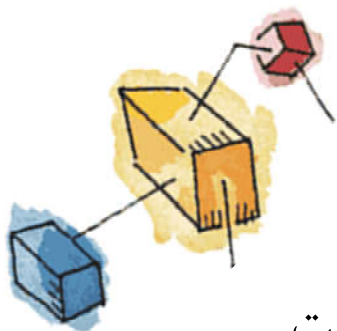
اطلاعات وضعیت پردازنده

- ثبات های قابل رویت برای کاربر
- ثبات های قابل رویت برای کاربر یعنی ثباتی که به وسیله دستورالعمل های زبان ماشین، که پردازنده در حالت کاربر اجرا می کند، بتواند مورد مراجعه قرار گیرد.
— معمولاً ۸ تا ۳۲ تا از این ثبات ها در پردازنده وجود دارد. و در بعضی پیاده سازی ها بیش از ۱۰۰ تا.

• ثبات های کنترل وضعیت

- بعضی از ثبات های پردازنده برای کنترل عمل پردازنده به کار می روند. مانند:
 - شمارنده برنامه: حاوی آدرس دستورالعمل بعدی که باید واکنشی شود.
 - کدهای شرایط: نتیجه آخرین عمل محاسباتی یا منطقی (علامت، صفر، رقم نقلی، مساوی، سرریز)
 - اطلاعات وضعیت: پرچم های از کار انداختن و به کار انداختن وقفه

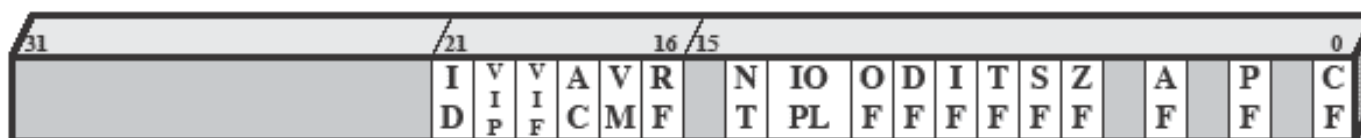




ثبات EFLAGS در Pentium II

تمام پردازنده ها شامل ثبات یا مجموعه ای از ثبات ها برای اطلاعات وضعیت هستند.

نمونه ای از یک PSW (کلمه وضعیت برنامه) از این ساختار در سیستم عاملی که روی پنتیوم اجرا می شود، استفاده می شود.



ID = Identification flag
VIP = Virtual interrupt pending
VIF = Virtual interrupt flag
AC = Alignment check
VM = Virtual 8086 mode
RF = Resume flag
NT = Nested task flag
IOPL = I/O privilege level
OF = Overflow flag

DF = Direction flag
IF = Interrupt enable flag
TF = Trap flag
SF = Sign flag
ZF = Zero flag
AF = Auxiliary carry flag
PF = Parity flag
CF = Carry flag



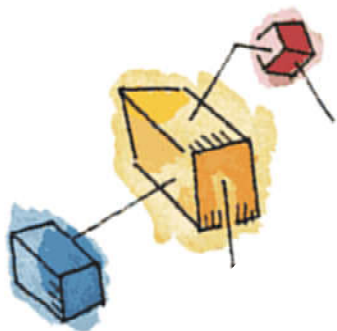


اجزای متداول بلاک کنترل فرآیند (PCB): اطلاعات وضعیت پردازنده

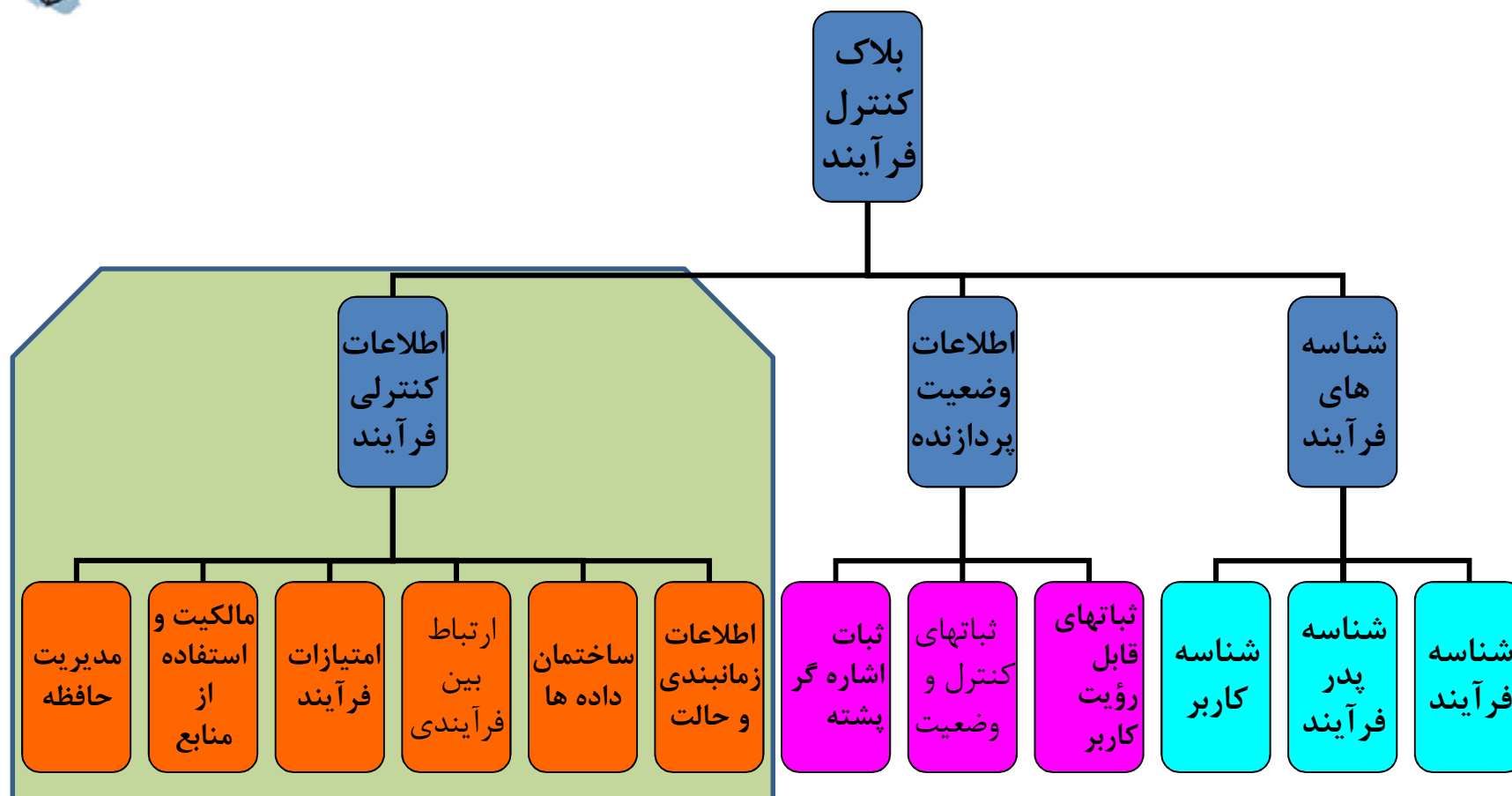
• اشاره گرهای پشته

- به هر فرآیند یک یا چند پشته تخصیص داده می شود.
- پشته برای ذخیره سازی پارامترها و آدرس های فراخوانی رویه و سیستم به کار می رود.
- اشاره گر پشته به بالای پشته اشاره می کند.





اجزای متداول بلاک کنترل فرآیند (PCB)





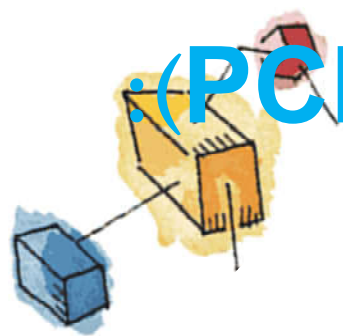
اجزای متداول بلاک کنترل فرآیند (PCB): اطلاعات کنترلی فرآیند

• اطلاعات زمانبندی و حالت

– یعنی اطلاعاتی که سیستم برای انجام زمانبندی نیاز دارد. مانند:

- وضعیت فرآیند: مانند: در حال اجرا، آماده، منتظر، متوقف
- اولویت: برای بیان اولویت زمانبندی فرآیند استفاده می شود (مقدار جاری، مقدار پیش فرض، بالاترین مقدار مجاز)
- اطلاعات مربوط به زمانبندی: این اطلاعات به الگوریتم زمانبندی مورد استفاده بستگی دارد. مثلاً مدت زمانی که این فرآیند منتظر بوده است و یا مدت زمانی که این فرآیند آخرین بار اجرا شده است.
- رویداد (حادثه): حادثه ای که تا وقتی رخ دهد این فرآیند باید منتظر بماند.





اجزای متداول بلاک کنترل فرآیند (PCB): اطلاعات کنترلی فرآیند

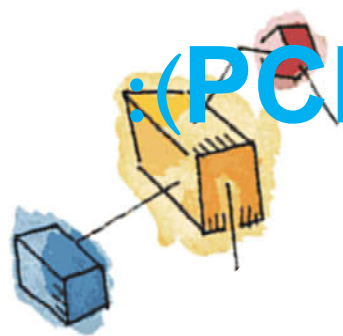
• ساختمان داده ها

– ممکن است فرآیند در یک صف، حلقه، یا ساختمان داده های دیگر، به فرآیندهای دیگر متصل باشد. برای حمایت از این ساختمان داده ها، ممکن است بلوک کنترل فرآیند اشاره گر هایی به فرآیندهای دیگر داشته باشد.

– ارتباط بین فرآیندها (Interprocess Communication)

– برای ارتباط بین دو فرآیند، ممکن است پرچم ها و علائم و پیام های مختلفی مطرح باشد. ممکن است تمام یا بخشی از این اطلاعات در PCB ذخیره شود.





اجزای متداول بلاک کنترل فرآیند (PCB): اطلاعات کنترلی فرآیند

• امتیازات فرآیند

– ممکن است برای حافظه ای که فرآیند می تواند دسترسی داشته باشد، و انواع دستورالعمل هایی که می تواند اجرا کند، امتیازاتی داشته باشد.

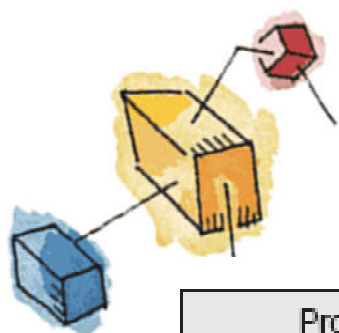
• مدیریت حافظه

– شامل اشاره گرهایی به جدول قطعه و یا جدول های صفحه مجازی این فرآیند

• مالکیت و استفاده از منابع

– منابعی که به وسیله این فرآیند کنترل می شوند (مثل فایل ها) می توانند در اینجا نشان داده شوند. تاریخچه استفاده از پردازنده نیز ممکن است نگهداری شود.





تصویر ساختار فرآیندها در حافظه مجازی

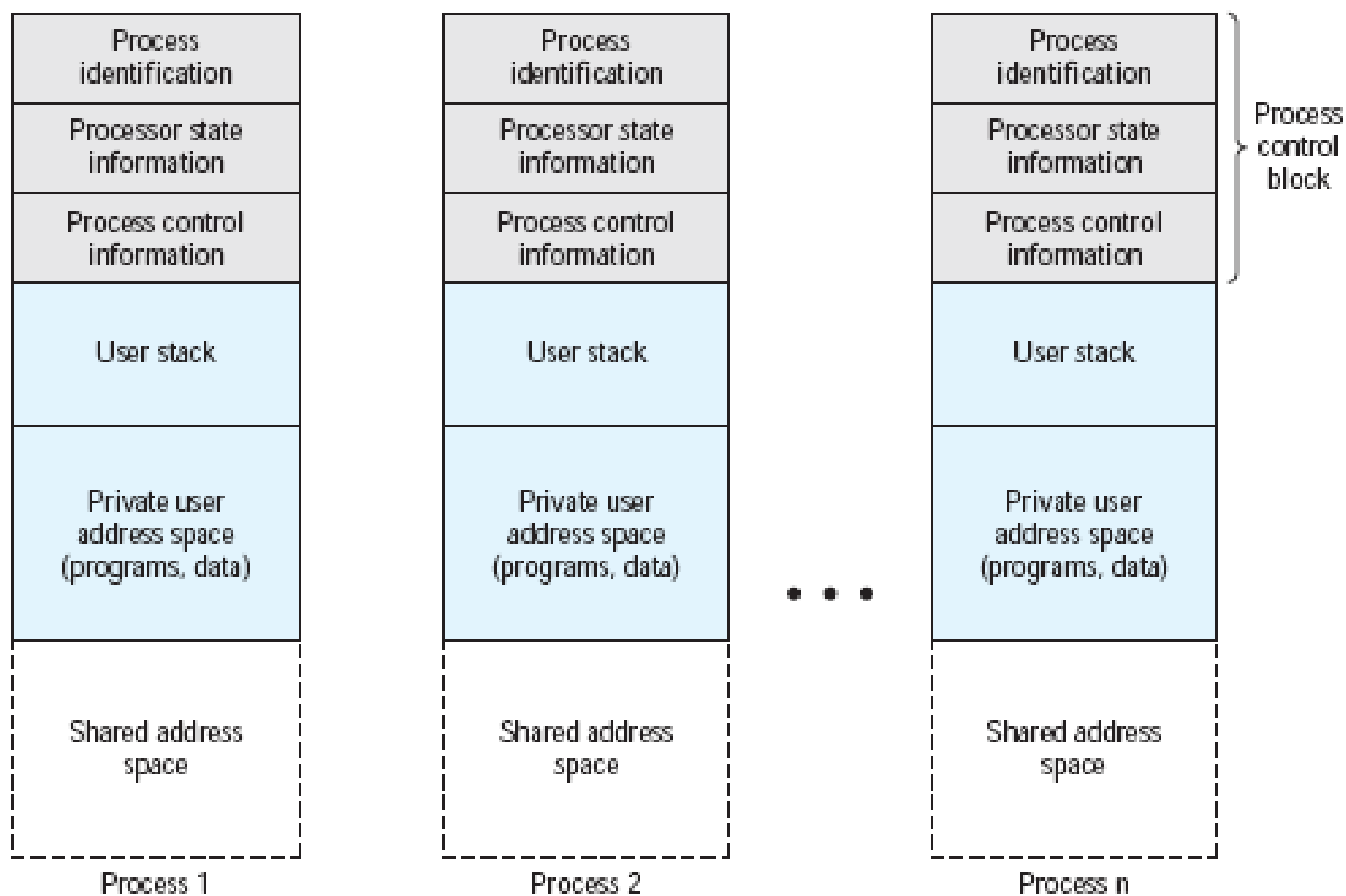
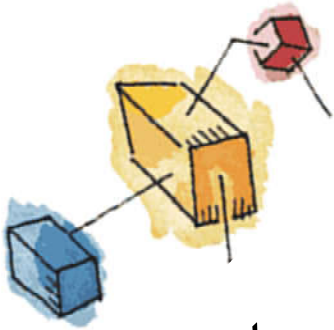


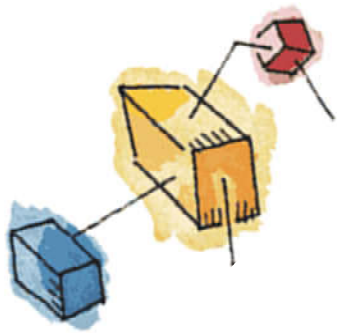
Figure 3.13 User Processes in Virtual Memory

نقش بلاک کنترل فرآیند



- بلاک کنترل فرآیند، مهم ترین و محوری ترین ساختمان داده در سیستم عامل است.
 - تمام اطلاعات مورد نیاز سیستم عامل در مورد یک فرآیند را در بر دارد.
 - این اطلاعات توسط مولفه های مختلف سیستم عامل مورد استفاده و تغییر قرار می گیرند.
- هر تغییر در طراحی و معنی بلاک کنترل فرآیند، ممکن است تعداد زیادی از مولفه های سیستم عامل را تحت تاثیر قرار دهد.

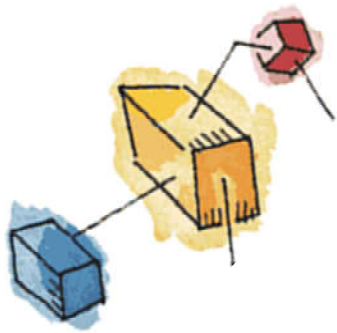




سرفصل مطالب

- فرآیند چیست؟
- حالات فرآیند
- شرح فرآیند
- کنترل فرآیند
- اجرای سیستم عامل
- مدیریت فرآیند در UNIX SVR4





حالت های اجرا

- اکثر پردازنده ها از دو حالت اجرا پشتیبانی می کنند:

– حالت کاربر

- حالت کم امتیاز
- برنامه کاربران در این حالت اجرا می شود

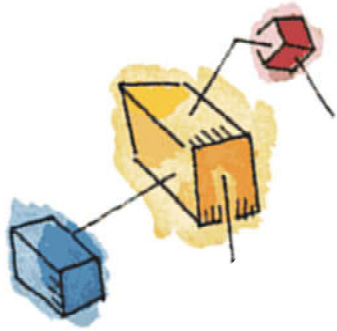
– حالت هسته (ممتاز/سیستم/کنترل)

- حالت پر امتیاز
- عملیات هسته سیستم عامل در این حالت اجرا می شود

- تغییر حالت معمولا در واکنش به وقفه ها، فراخوانی سیستم و همینطور بعضی دستورالعمل های خاص انجام می شود.



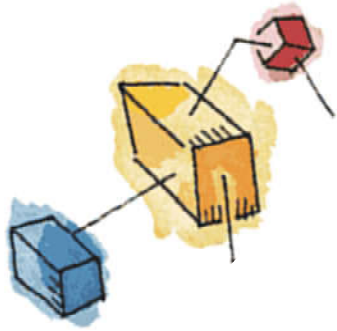
مراحل ایجاد فرآیند



- تخصیص یک شناسه منحصر به فرد به فرآیند
- تخصیص فضا به فرآیند
- مقداردهی اولیه بلاک کنترل فرآیند
 - شناسه فرآیند، شناسه پدر
 - جز شمارنده برنامه و اشاره گر پشته، بقیه با صفر مقدار دهی می شوند.
 - اطلاعات کنترلی فرآیند بر اساس مقدار پیش فرض مقدار دهی می شوند.
- برقراری پیوندهای لازم
 - به عنوان مثال، اضافه کردن فرآیند جدید به لیست پیوندی مورد استفاده برای صف زمانبندی
- ایجاد و گسترش ساختمان داده های دیگر
 - مثلاً تخصیص پرونده های حسابداری به فرآیند.



شرایط و عوامل تعویض فرآیند



- وقفه خارجی

- وقفه ورودی/خروجی

- وقفه ساعت

- فرآیند به میزان برش زمانی مجاز در حال اجرا بوده است.

- تله داخلی

- دستورالعمل غیرمجاز

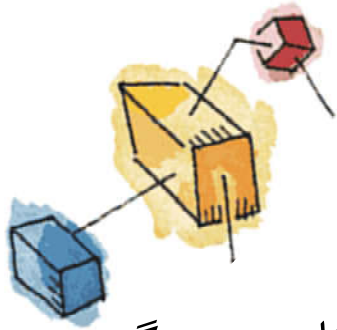
- خطای حافظه (صفحه)

- آدرس صفحه مورد ارجاع در حافظه مجازی قرار داشته و لذا لازم است به حافظه اصلی آورده شود.

فراخوانی ناظر (فراخوان سیستم)

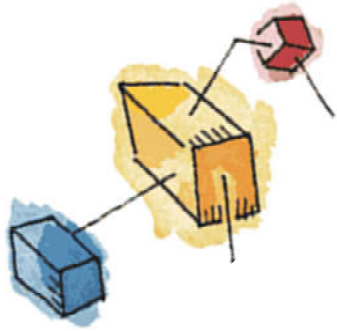


تغییر حالت فرآیند



- زمینه (Context) فرآیند یا وضعیت جاری پردازنده شامل شمارنده برنامه و دیگر ثبات ها، ذخیره می شود.
- حالت فرآیندی که در وضعیت اجرا قرار دارد به روز می شود (اجرا به مسدود، آماده، خروج)
- فرآیند به صف مناسب (صف مسدود، معلق، آماده) واگذار می شود.
- فرآیند دیگری برای اجرا انتخاب می شود.
- بلاک کنترل فرآیند انتخاب شده به روز می شود (تغییر حالت به اجرا)
- به روز کردن ساختمان داده های مدیریت حافظه
- زمینه فرآیند انتخاب شده بر روی پردازنده بارگذاری می شود.

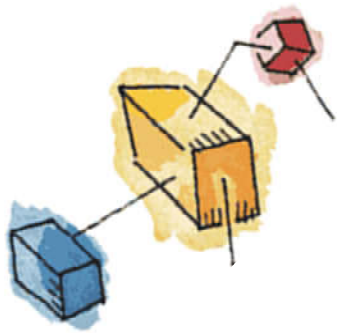




سرفصل مطالب

- فرآیند چیست؟
- حالات فرآیند
- شرح فرآیند
- کنترل فرآیند
- اجرای سیستم عامل
- مدیریت فرآیند در UNIX SVR4





اجرای سیستم عامل

- گفته شد که:

- سیستم عامل مثل نرم افزارهای معمولی کار می کند. یعنی سیستم عامل یک برنامه است که توسط پردازنده اجرا می شود.

- این سوال پیش می آید که:

- اگر سیستم عامل، مجموعه ای از برنامه ها است که توسط پردازنده اجرا می شود، آیا سیستم عامل هم یک فرآیند است؟

- طراحان سیستم عامل، در طراحی های خود سه روش مختلف را در نظر گرفته اند:

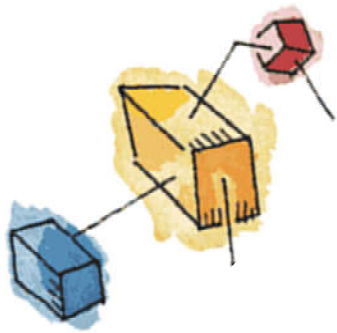
- هسته غیرفرآیند

- اجرا در خلال فرآیند های کاربر

- سیستم عامل مبتنی بر فرآیند

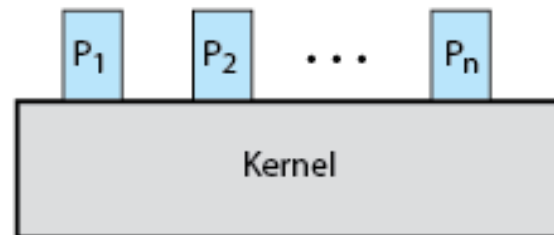


اجرای سیستم عامل



• هسته غیر فرآیند (Non-process Kernel)

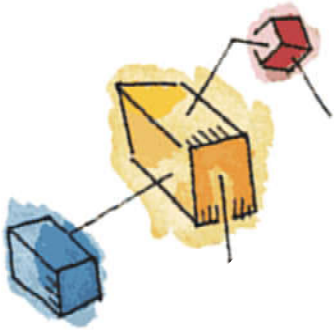
- هسته در خارج از متن فرآیند در حال اجرا، اجرا می شود.
- هسته، حافظه و پشته مخصوص به خود دارد.
- کد سیستم عامل به عنوان یک موجودیت جداگانه که در حالت ممتاز عمل می کند تلقی می شود.
- مفهوم فرآیند فقط به برنامه های کاربر اعمال می شود.



(a) Separate kernel



اجرای سیستم عامل



• هسته غیر فرآیند (ادامه)

- وقتی فرآیند در حال اجرا با وقفه مواجه می شود، یا یک دستور فراخوانی سیستم را اجرا می کند، متن مربوط به این فرآیند ذخیره می شود و کنترل به هسته منتقل می شود.

- سیستم عامل اعمال موردنظر خود را انجام می دهد و سپس بسته به علت و نوع وقفه و شرایط:

– اجرای فرآیند متوقف شده ی کاربر از سر گرفته می شود.

– یا

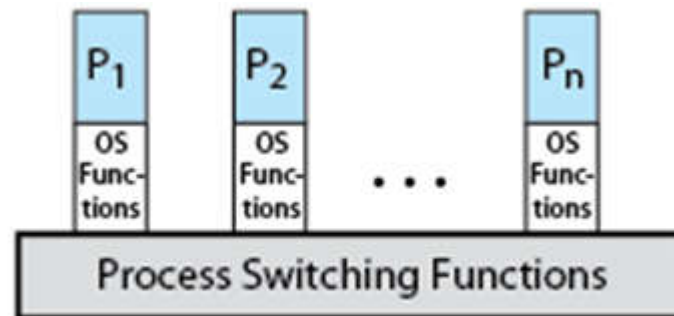
- عمل ذخیره سازی فرآیند جاری را تکمیل می کند و توزیع کننده، فرآیند دیگری را برای اجرا انتخاب می کند.



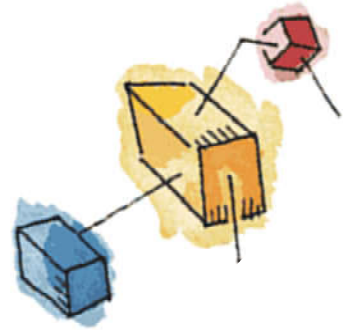
اجرای سیستم عامل

• اجرا در خلال فرآیندهای کاربر

- سیستم عامل مجموعه ای از روال ها است که کاربر آنها را برای انجام کارهای مختلف فراخوانی می کند تا در داخل محیط فرآیند کاربر اجرا شوند.
- فراخوانی های سیستمی در زمینه (متن) فرآیند در حال اجرا رخ می دهند.
- تصویر هر فرآیند، ساختمان داده های کاربر و سیستم را با هم دارا ست.
- هنگام اجرای کد سیستم عامل، فرآیند در حالت ممتاز اجرا می شود.
- Unix معمولا در این حالت اجرا می شود.



(b) OS functions execute within user processes



اجرای سیستم عامل

- اجرا در خلال فرآیند های کاربر (ادامه)
- نیاز نیست که برای اجرای روال های سیستم عامل، تعویض متن رخ دهد.
- کد و داده های سیستم عامل در فضای آدرس مشترک قرار دارد و برای تمام فرآیندهای کاربران مورد اشتراک قرار می گیرد.
- وقتی تله، وقفه یا فراخوانی سیستم رخ می دهد، پردازنده به حالت ممتاز می رود و یک تعویض به یک روال سیستم عامل انجام می شود ولی اجرا از داخل فرآیند جاری کاربر ادامه می یابد.
- بنابراین تعویض فرآیند صورت نمی گیرد. فقط در همان فرآیند، تعویض حالت صورت می گیرد.

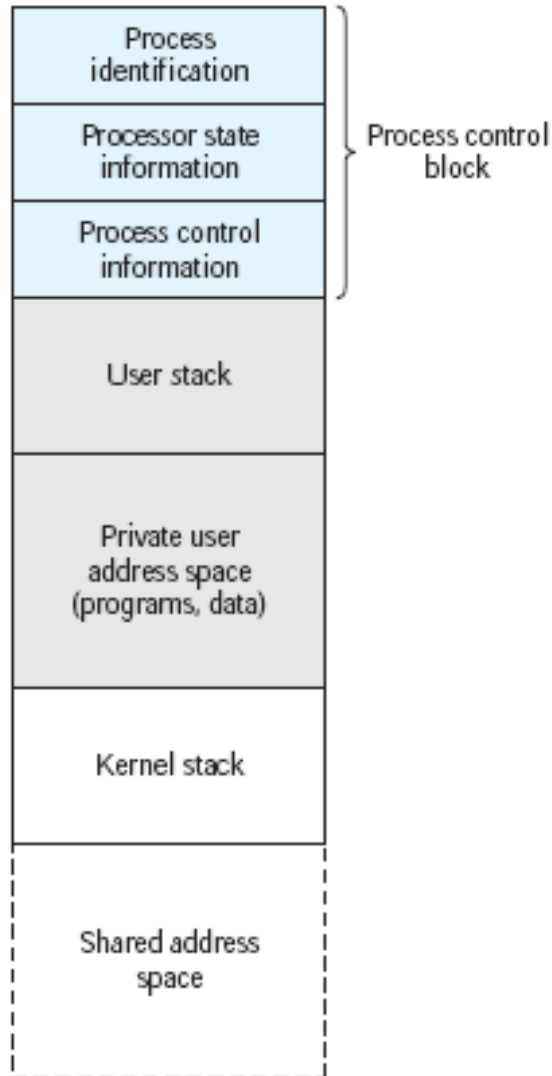
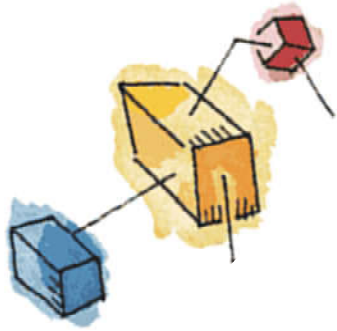


Figure 3.16 Process Image: Operating System Executes within User Space

اجرای سیستم عامل



• اجرا در خلال فرآیند های کاربر (ادامه)

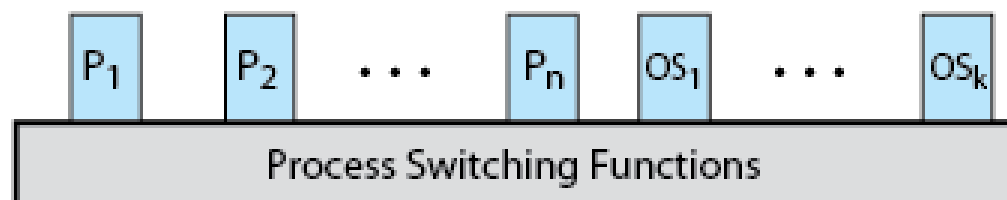
- پس در داخل یک فرآیند، ممکن است هم یک برنامه کاربر و هم برنامه های سیستمی اجرا شوند و برنامه های سیستم عامل که در فرآیندهای مختلف اجرا می شوند، یکسان هستند.
- یک مزیت: هنگامی که کاربر نیاز به به کارگیری یک روال سیستم عامل دارد، با دو تا تعویض متن مواجه نمی شود و فقط تغییر حالت (به حالت ممتاز یا هسته) صورت می گیرد.
- در مواردی که نیاز به تعویض فرآیند است، اینکه روال تعویض کننده فرآیند، در فرآیند جاری اجرا شود یا نه بستگی به نوع طراحی و پیاده سازی سیستم عامل دارد.
- منطقی است که اجرای این کد در خارج از همه فرآیندها صورت بگیرد.

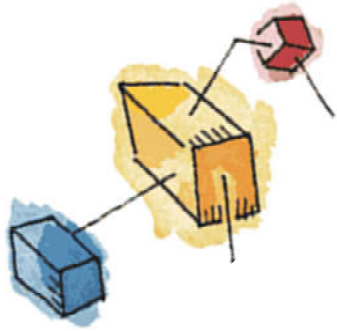


اجرای سیستم عامل

• سیستم عامل مبتنی بر فرآیند

- سیستم عامل به عنوان مجموعه ای از فرآیندهای سیستمی طراحی می شود.
- اعمال اصلی سیستم به صورت فرآیندهای جداگانه سازماندهی می شوند. باز هم ممکن است مقدار کوچکی از کد تعویض فرآیند، خارج از هر فرآیندی اجرا شود.
- یک مزیت: امکان طراحی سیستم عامل به صورت مولفه ای با ارتباط های مشخص بین مولفه ها
- در سیستم های چندپردازنده ای نیز کارآمد است.





سرفصل مطالب

- فرآیند چیست؟
- حالات فرآیند
- شرح فرآیند
- کنترل فرآیند
- اجرای سیستم عامل
- مدیریت فرآیند در UNIX SVR4



UNIX Process Image

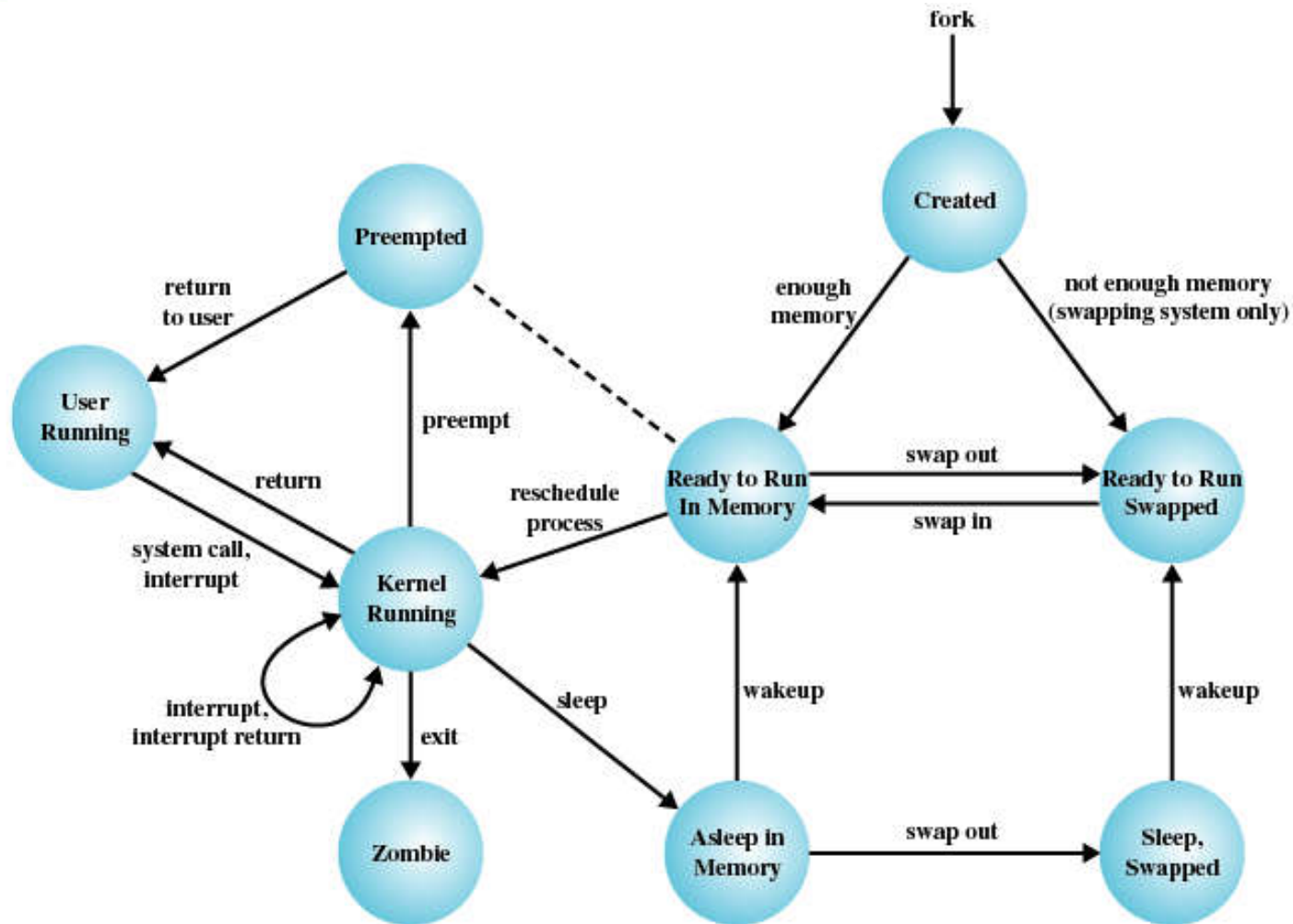
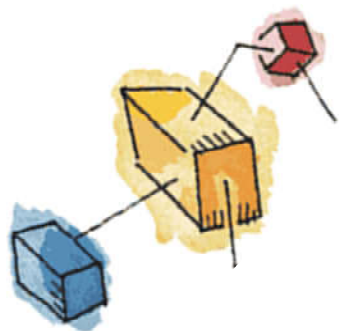


Figure 3.17 UNIX Process State Transition Diagram

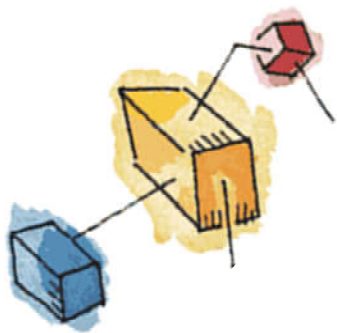


UNIX Process States

Table 3.9 UNIX Process States

User Running	Executing in user mode.
Kernel Running	Executing in kernel mode.
Ready to Run, in Memory	Ready to run as soon as the kernel schedules it.
Asleep in Memory	Unable to execute until an event occurs; process is in main memory (a blocked state).
Ready to Run, Swapped	Process is ready to run, but the swapper must swap the process into main memory before the kernel can schedule it to execute.
Sleeping, Swapped	The process is awaiting an event and has been swapped to secondary storage (a blocked state).
Preempted	Process is returning from kernel to user mode, but the kernel preempts it and does a process switch to schedule another process.
Created	Process is newly created and not yet ready to run.
Zombie	Process no longer exists, but it leaves a record for its parent process to collect.





پایان فصل سوم

