

در مولتی پروسور حافظه به صورت **مشترک** می باشد

در مولتی کامپیوتر حافظه **مشترک** نمی باشد.

مهمترین وظیفه سیستم عامل توزیع شده: شبیه سازی حافظه **مشترک** برای کاربر است. به عبارتی این تصور را برای کاربر به وجود می آورد که حافظه به صورت مشترک می باشد.

یک سیستم عامل توزیع شده چه کاری را انجام می دهد.

- مدیریت منابع در یک سیستم توزیع شده (به طور شفاف از کاربر)
- مثل یک سیستم عامل متمرکز یکپارچه به نظر می رسد. (با این که روی پردازنده های مستقل زیادی عمل میکند)
- ایجاد شفافیت (از نظر محل، تغییر مکان پردازنده ها، همروندی و ...)
- ارائه یک پردازنده مجازی واحد به کاربر

Disdtributed shared memory system(DSM)

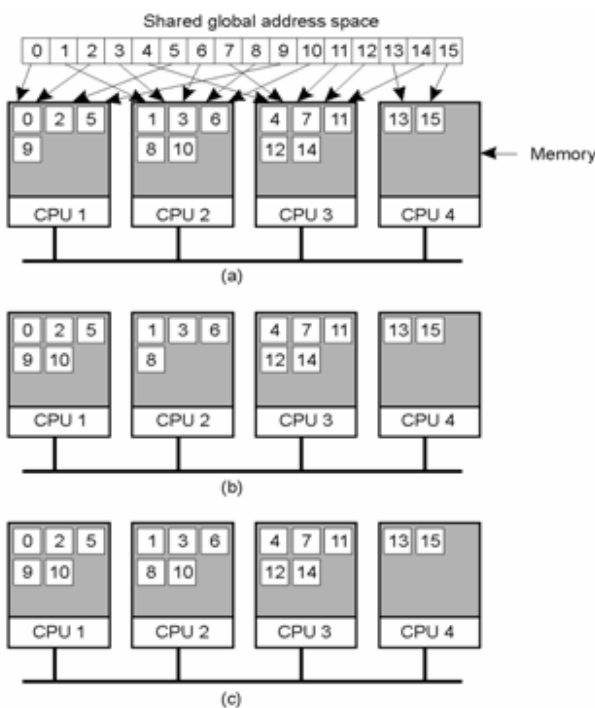
جهت پیاده سازی DSM از مفهوم حافظه مجازی هر گره استفاد می شود با این تفاوت که Remote RAM به جای دیسک استفاده می شود. در واقع به این شکل عمل می شود که فضای آدرس به صفحه هائی تقسیم می شود (معمولا به اندازه 4k یا 8k) که این صفحات در حافظه پردازنده های مختلف پراکنده هستند، زمانی که یک پردازنده به ادرسی مراجعه می کند که در صفحات موجود در حافظه محلی خود وجود ندارد، نقض صفحه رخ داده و سیستم عامل صفحه ای را که آدرس مورد نظر در آن وجود دارد را به حافظه محلی خود انتقال می دهد. این مفهوم را در شکل زیر ملاحظه می کنید (فضای آدرس با 16 صفحه و 4 پردازنده).

(a) فضای آدرس صفحاتی را نشان می دهد که مابین 4 ماشین (پردازنده) توزیع شده اند.

(b) حالتی را نشان می دهد که CPU 1 به صفحه شماره 10 مراجعه می کند (آن را به حافظه محلی انتقال می دهد).

(c) حالتی را نشان می دهد که صفحه شماره 10 فقط قابلیت خواندن داشته باشد (قابلیت نوشتن یا تغییر نداشته باشد)

در این مثال اگر پردازنده شماره 1 به دستور یا داده ای در صفحات 0، 2، 5 و 9 مراجعه کند، یک ارجاع محلی صورت می گیرد ولی مراجع به سایر صفحات باعث ایجاد نقض صفحه خواهد شد مانند مراجعه به آدرسی در صفحه شماره 10 که باعث انتقال صفحه شماره 10 از ماشین 2 به ماشین 1 می شود. (در شکل هم نشان داده شده است) یک روش برای افزایش کارائی سیستم ها این است که صفحاتی را که قابل تغییر نیستند (Read only هستند)؛ Replicate شوند (یعنی ماشینی که به آن صفحه نیاز دارد یک کپی از آن برای خود ایجاد کند نه اینکه آن صفحه را به حافظه خود از ماشین دیگری انتقال دهد شکل C)



نکته: اگر page ها قابل نوشتن هم باشند مشکل ساز گاری داریم به عنوان مثال ممکن است محتویات صفحات هم شماره یکسان نباشند.

نکته مهم دیگر در طراحی سیستم های DSM کارآمد تعیین **انداز صفحات** (page ها) می باشد.

- اگر اندازه صفحات کوچک باشد عمل نقل انتقال زیادی خواهیم داشت
- اگر اندازه صفحات بزرگ باشد مشکل false sharing خواهیم داشت.

منظور از false sharing چیست.

وجود داده متعلق به دو پردازنده را در یک صفحه (page) **false sharing** گویند. که این موضوع باعث می شود آن صفحه مکررا بین دو پردازنده ای که به آن نیاز دارند رد و بدل شود شکل زیر را ملاحظه کنید.

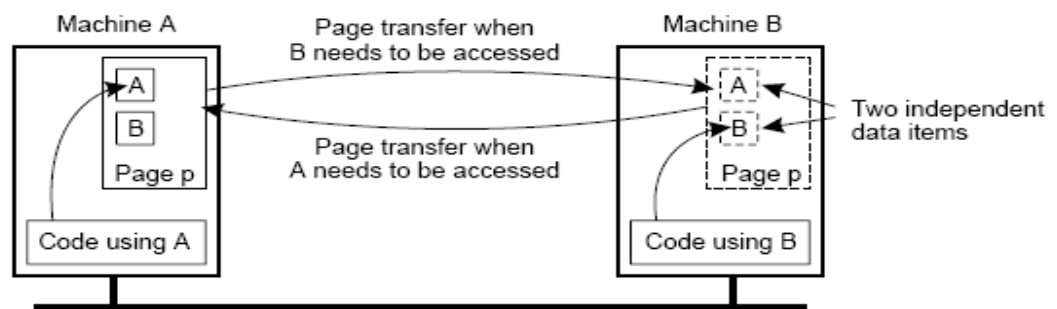


Figure 1-18. False sharing of a page between two independent processes.

سیستم عامل شبکه (NOS):

بر خلاف سیستم عامل های توزیع شده، سیستم عامل های شبکه می توانند بر روی سخت افزار های **ناهمگن** (از نوع های مختلف) اجرا شوند و عمل **مخفی سازی** در این سیستم ها وجود **ندارد** مثلا در کپی فایل در شبکه مشخص است که از کدام کامپیوتر کپی می شود. به عبارتی این سیستم عامل ها دید واحد و یکنواختی از سیستم را به کاربر ارائه نمی دهند. این سیستم ها از مجموعه ای از سیستم های تک پردازنده ای ساخته شده اند که هر کدام سیستم عامل خاص خودش را دارد. همانطور که در شکل زیر هم نشان داده شده است ماشین ها و سیستم عامل شان ممکن است متفاوت از هم باشند. اما تمامی این ماشین ها از طریق شبکه با هم در ارتباط هستند و سیستم عامل شبکه امکاناتی را فراهم می کند تا به کاربران اجازه دهد که از سرویس موجود در یک ماشین خاص استفاده کنند.

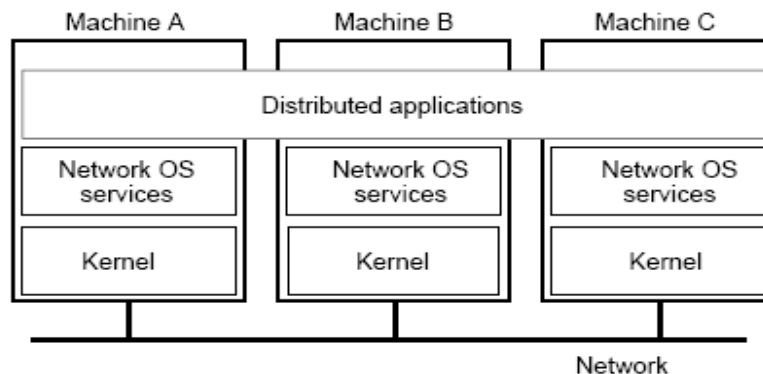


Figure 1-19. General structure of a network operating system.

نکته دیگر این است که Client ها می توانند از طریق ادرس های متفاوت (مسیر های متفاوت) به Server ها دسترسی داشته باشند. شکل زیر این موضوع را نشان می دهد همانطور که در شکل ملاحظه می شود یکی از Server ها دایرکتوری به نام game را در خود دارد و دیگری دایرکتوری به نام work؛ و هر کدام از دایرکتوریا شامل چندین فایل می باشد این دو Client به هر دو Server نصب هستند اما با مسیر های متفاوت، در واقع Client 1 به دایرکتوری اصلی آنها دسترسی دارد و می تواند به شکل /games و /work به Server ها دسترسی داشته باشد. Client 2 نیز به دایرکتوری اصلی Server 2 دسترسی دارد (همانند Client 1). اما با این دید که بازی کردن شاید به عنوان یک کار Private (شخصی) باشد Client 2 یک دایرکتوری به نام Private ایجاد می کند و از طریق آن به Server ی که دایرکتوری game را دارد دسترسی پیدا می کند در نتیجه اگر Client 2 بخواید به pacwoman دسترسی داشته باشد مسیر به شکل /Private/game/pacwoman خواهد شد ولی Client 1 با مسیر /game/pacwoman به همین دسترسی دارد.

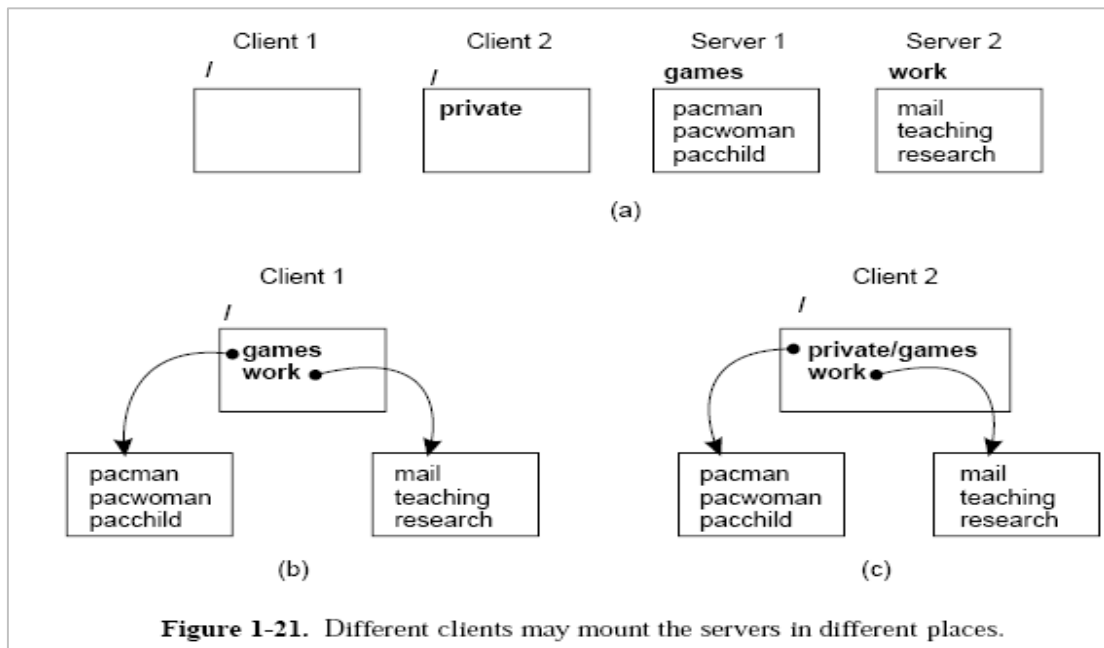


Figure 1-21. Different clients may mount the servers in different places.

میان افزاز (middleware)

هیچ کدام از سیستم عامل های شبکه یا سیستم عامل های توزیع شده، سیستم عامل توزیع شده واقعی (آنچه که قبلا تعریف شده) نیستند، سیستم عامل های توزیع شده نمی توانند مجموعه ای از کامپیوتر های مستقل از هم را پشتیبانی کنند و سیستم عامل شبکه یک دید منسجم و واحدی را از سیستم به کاربر ارائه نمی دهد، حال سؤال این است که آیا می توان سیستم عامل توزیع شده ای داشت که دو مورد قبلی را نیز نداشته باشد (قادر به اداره کامپیوتر های مستقل از هم باشد و دید واحدی از سیستم ارائه دهد) به عبارت دیگر سیستم عامل توزیع شده ای که قابلیت توسعه (مقیاس پذیری) و باز بودن (openness) سیستم عامل شبکه را داشته باشد و قابلیت مخفی سازی و ارتباط آسان سیستم عامل های توزیع شده را، راه حل به این شکل می تواند باشد که از یک لایه نرم افزاری اضافی در سیستم عامل های شبکه استفاده کنیم طوری که این لایه بایستی ناهمگونی های سخت افزاری را پنهان کند و برای ایجاد شفافیت یا پنهان سازی کسترش داده شود. که به این لایه میان افزاز (middleware) گفته می شود. شکل زیر جایگاه این میان افزاز را نشان می دهد.

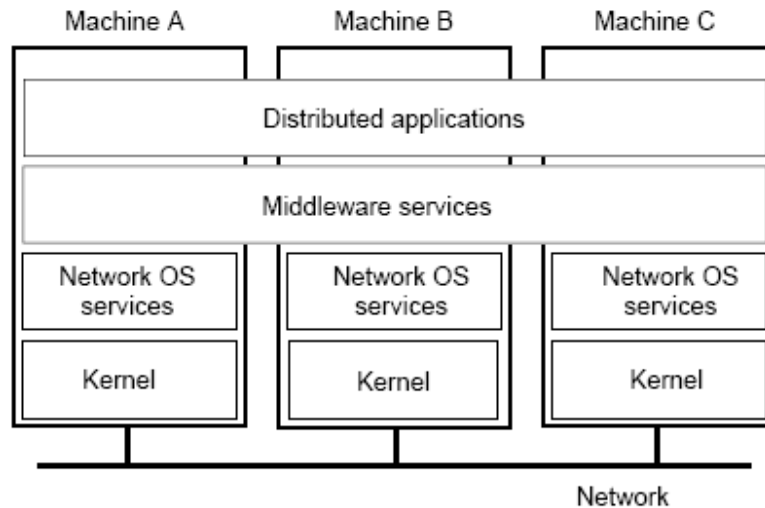


Figure 1-22. General structure of a distributed system as middleware.

پس می توان نتیجه گرفت که m.w نسبت به شبکه و سیستم های توزیع شده هم اشتراک منابع را دارد و هم مخفی سازی و هم امتیازات دیگر را
مزایای مولتی کامپیوتر نسبت به مولتی پروسسور:

- رشد افزایشی: در مولتی کامپیوتر به راحتی می توان یک کامپیوتر را اضافه کرد ولی در مولتی پروسسور به سختی امکان پذیر است(امکان پذیر نیست)
- افزایش نسبت سرعت به هزینه: در مولتی کامپیوتر بیشتر است
- تحمل خطا: تحمل خطای مولتی کامپیوتر بیشتر است اگر یکی خراب شود مشکلی ایجاد نمی شود ولی در مولتی پروسسور مشکل ایجاد می شود.
- خاصیت توزیعی: به عنوان مثال اگر یک کامپیوتر برای پردازش تصاویر مناسب باشد کار سرویس پردازش تصاویر را به آن محول می کنیم.
- مزایای مولتی کامپیوتر نسبت به تک کامپیوتر همان مزایای شبکه است نسبت به تک کامپیوتر
- مزیت سیستم توزیع شده مولتی کامپیوتر نسبت به شبکه مخفی سازی می باشد.

مدل های های پیاده سازی میان افراز:

{ service:file Distributed File
 { communication : file

RPC (فراخوانی راه دور پردازش Remote Procedure Calls): در واقع سرویس ها را در پردازش ها توزیع می کنیم .

{ service:procedure
 { communication:file(message)

RMI (Remote method invocation) : در این روش سرویس ها را در Object ها قرار می دهیم و معمولا در خود

Object ، distribution نداریم

{service:object
communication:file(message)

URL

{service:document
communication:link

مقایسه سیستم ها با یکدیگر:

Item	Distributed OS		Network OS	Middleware-based DS
	Multiproc.	Multicomp.		
Degree of transparency	Very high	High	Low	High
Same OS on all nodes?	Yes	Yes	No	No
Number of copies of OS	1	N	N	N
Basis for communication	Shared memory	Messages	Files	Model specific
Resource management	Global, central	Global, distributed	Per node	Per node
Scalability	No	Moderately	Yes	Varies
Openness	Closed	Closed	Open	Open

Figure 1-24. A comparison between multiprocessor operating systems, multi-computer operating systems, network operating systems, and middleware-based distributed systems.