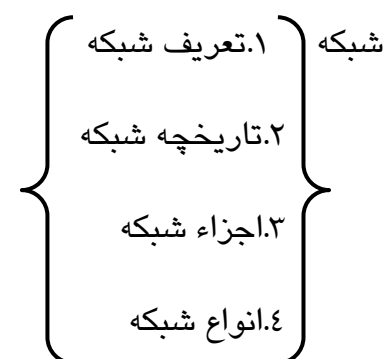


شبکه‌های کامپیوتری

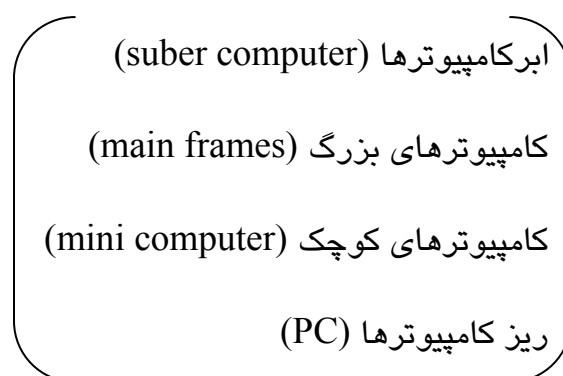
استاد وزیری



تعریف شبکه:

مجموعه‌ای از تعداد کامپیوترهای مستقل است که با یک تکنولوژی واحد به هم متصل‌اند، تعدادی کامپیوتر مستقل به هم متصل را شبکه گویند.

تاریخچه شبکه‌ها:



Main frames

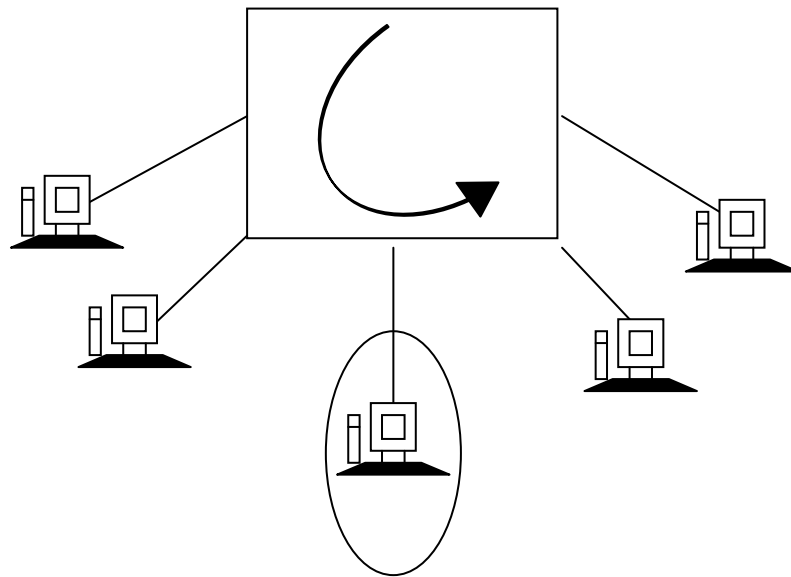
کامپیوترهای خیلی بزرگ اولیه بسیار گران و هزینه زیادی داشتند.

به این کامپیوترها دستگاههای ورودی و خروجی که به آن ترمینال گفته می‌شد، متصل بودند.

: Terminals

دستگاههای که ورودی و خروجی دارند ، ترمینال پردازشگر ندارد. همه به یک جا وصلند (به یک

processor) ارتباط هم در این ناحیه بین بقیه ترمینالها انجام می‌شود.



PC:

با استفاده از فن آوری روز کامپیوترهای کوچک و شخصی بوجود آمد. تمام کارها توسط pc ها انجام می شود. روز به روز آنها از نظر پردازش قویتر شده اند، مشکل آنها اینست که قدرت تبادل اطلاعات نداشتند. با متصل کردن و اشتراک گذاشتن کامپیوترها توانایی کامپیوترها را بیشتر می کند. تلفیقی از مشکل آنها اینست که قدرت تبادل اطلاعات را نداشتند. با متصل کردن و اشتراک گذاشتن کامپیوترها توانایی کامپیوترها را بیشتر می کند. تلفیقی از pc ها و ترمیالها را شبکه می گویند. شبکه شامل تعدادی کامپیوتر مستقل و متصل بهم اند.

مزیت های شبکه:

۱- اشتراک منابع (resource sharing)

۲- بالا بودن نسبت کارایی به قیمت

۳- از بین بردن بعد فاصله

۴- قدرت و قابلیت توسعه (Scability)

۵- تحمل خطا (futtoloerar)

۶- قابلیت اطمینان (reliability)

قابلیت اطمینان شبکه بهتر است چون Single point of نیست.

شبکه‌های توزیع شده (distributed network) در ساده‌ترین حالت همان شبکه‌های کامپیوتری هستند و برای راحت‌تر کردن ارتباط برای همه از این شبکه‌ها استفاده می‌شود (استفاده راحت‌تر برای همه کاربران آشنا و ناآشنا به کامپیوتر).

مشکل شبکه‌ها:

امنیت آنهاست چون اطلاعات خاص به یک کامپیوتر در تمام کامپیوترهای شبکه شده تبادل می‌شود. پس امن نیست. شبکه‌ها از نظر ساختار و اجزاء شامل سه جزء اصلی:

- ۱. کامپیوترها (client)
- ۲. کابل ارتباط
- ۳. قراردادها (Protocol)

می‌باشند.

تقسیم‌بندی شبکه‌ها:

- ۱- اندازه
- ۲- فن‌آوری انتقال
- ۳- نحوه ارتباط
- ۴- توزیع‌پذیری

انواع شبکه از نظر اندازه:

Local Area Network (LAN)
Metopolintion Area Network (MAN)
Wide Area Network (WAN)

:LAN

کوچکترین نوع شبکه و محدود به یک ساختمان را گوییم. شبکه‌ها اکثراً محلی هستند.

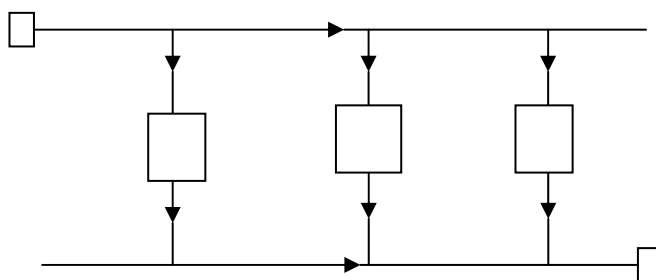
مزایا: هزینه پایین و خطا در آنها بسیار کم است و تکنولوژی خیلی بالا و سرعتشان (ظرفیت) خیلی بالایی دارند، 1000mb را می‌توانند تبادل کنند. و همچنین الگوریتم کارایی آنها ساده است.

:MAN

حدود آنها به اندازه یک شهر است.

: Headend

مرکز فوق توزیع که سیگنالها به آن وارد می‌شوند تا از آنجا توزیع شوند و این دو با هم در ارتباطند.



:WAN

محدودیت ندارند و بزرگ هستند، به خاطر فاصله زیاد خطا دارند، هزینه‌شان بالاست، تکنولوژی پیچیده دارند و سرعتشان پایین است.

:Topology

طریقه بهم بستن کامپیوترها را گویند که در lan به سه صورت می‌باشد:

1.BUS

2.Star

3.Ring

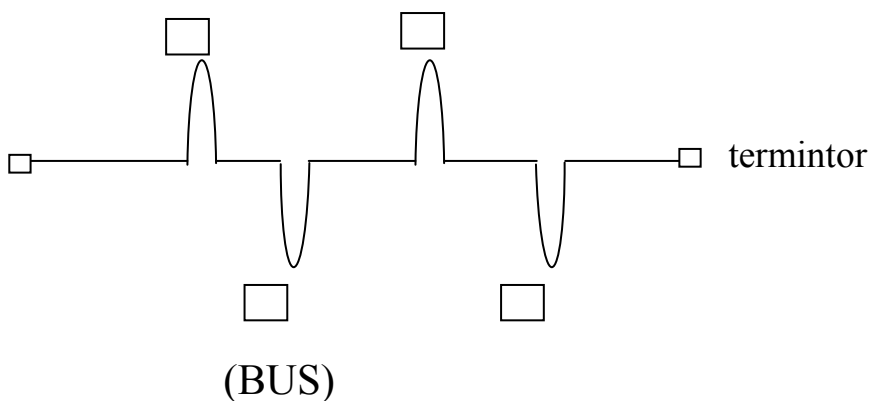
:BUS

ساده‌ترین همبندی است. در این نوع توپولوژی از کابل‌های به نم کواکسیال استفاده می‌شود، که برای متصل کردن کابلها از کانکتورهایی که شبیه T می‌باشند و به همین نام مرسومند استفاده می‌شود، زیرا طول

کابل حداکثر تا 180 متر باشد. حداکثر سرعت در آن 10MB/S است. در ابتدا و انتهای bus وسیله‌ای به نام terminator قرار می‌دهند.

Terminator: یک مقاومت 50 اهمی است که برای جلوگیری از تصادم اطلاعات در ابتدا و انتهای bus قرار می‌دهند. هر اطلاعاتی که آمد با وجود آن از بین می‌رود.

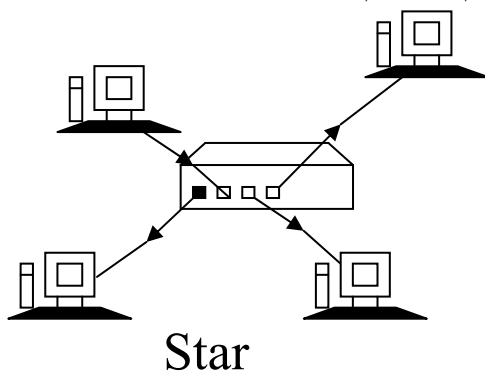
Terminator



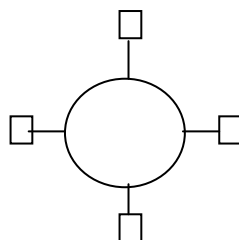
STAR

یک جزء مرکزی در آن بنام (hub) است، که روی یکسری port نصب شده . به ازای هر port یک چراغ داریم. کامپیوترهای شبکه‌شده هر کدام وصل به یک port هستند. در این توپولوژی از کابلهایی به نام utp استفاده می‌شود و کانکتور آنها کانکتورهای RG نام دارند.

مزیت: وجود چراغ مشخص می‌کند که کدام سیستم خراب است.

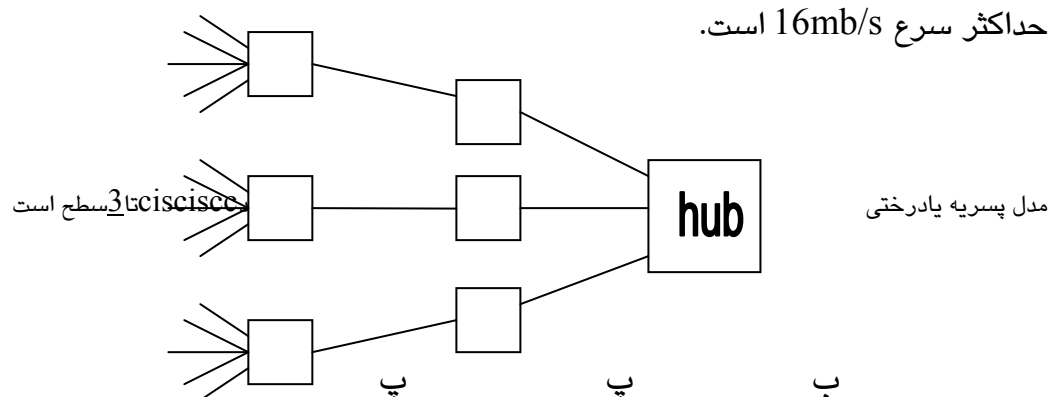


:Ring

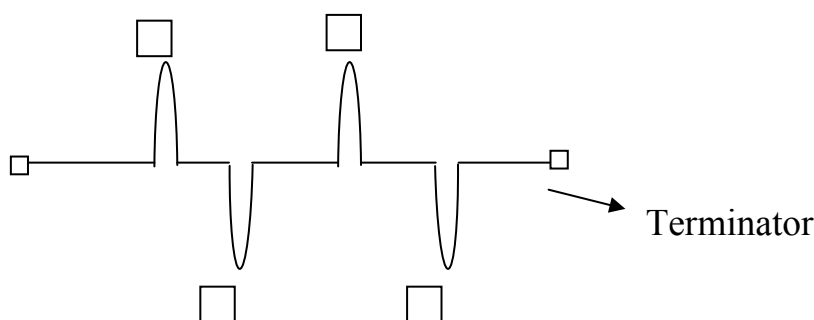


در این شبکه هر bit اطلاعات به صورت مستقل و بدون اینکه منتظر سایر bit ها ، بسته‌ای که به آن تعلق دارد، شود در شبکه پخش می‌شود. Token یک سیگنالی است که در شبکه وجود دارد و در حال چرخیدن است. برای فرستادن شرط اینست که آن سیگنال را کامپیوتر دریافت کرده و داده جدید را بوسیله token دوباره وارد شبکه بکن. Token یک چیز فیزیکی نیست بلکه منطقی است. احتمال تصادم داده‌ها را کم می‌کند.

حداکثر سرعت 16mb/s است.



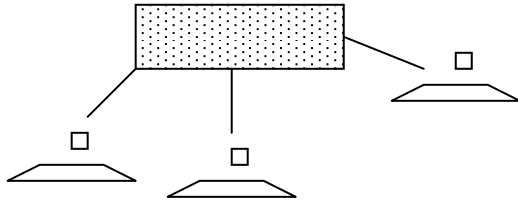
bus: ساده‌ترین همبندی است.



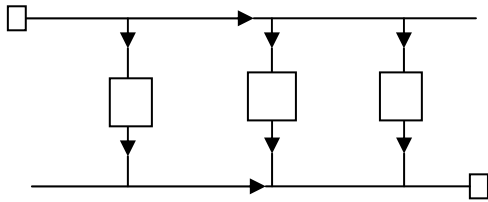
Terminator: یک نوع مقاومتهای #50 هستند. برای جلوگیری از تصادم اطلاعات آنها را در ابتدا و انتهای bus قرار می‌دهند. هر اطلاعاتی که آمد با وجود Terminator از بین می‌رود. حداکثر فاصله 180m است. مشکل عمده این نوع در پیدا کردن محل خطا است. چون مشخص نمی‌شود که کدام یک از کامپیوترها خراب است.

Star: یک جز مرکزی بنام (hub) در آن است. ک روی یکسری port نصب شده. به ازای هر port یک چراغ داریم. کامپیوترهای شبکه شده هرکدا وصل به یک port هستند

مزیت: وجود چرخ مشخص می‌کند که کدام سیستم خراب است.

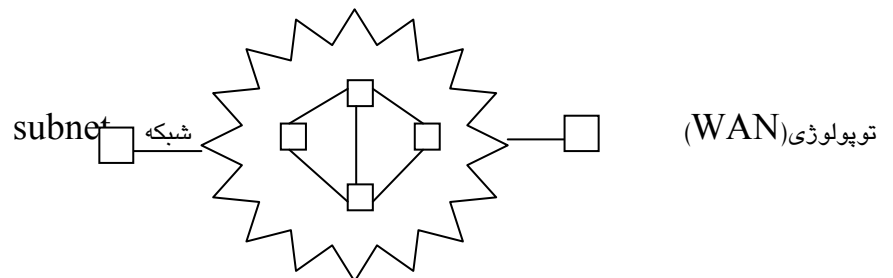


حدود آنها به اندازه یک شهر است.



headent: مرکز فوق توزیع که سیگنالها به آن وارد می‌شوند تا از آنجا توزیع شوند این دو با هم یکجوزی در ارتباطند.

(WAN) ۱- محدودیت ندارند و بزرگ هستند. ۲- به خاطر فاصله خطا زیاد ندارند. ۳- هزینه بالا ۴- تکنولوژی پیچیده ۵- سرعت پایین.



فن‌آوری انتقال شبکه:

broad Cast پخش کردن

point-to -point فقط مبدأ و مقصد از داده‌هایی که داده می‌شوند، مطلع هستند.

نحوه ارتباط در این مدل ارتباط افراد دیگر در یک گروه غیرثابت و ناپایدار صورت می‌گیرد. در واقع هر شخص می‌تواند مستقیماً با هر فرد یا افرادی تماس بگیرد. Peer-to-peer : تمام کامپیوترهای شبکه دارای اولویت یکسانی هستند.

Server based : یکسری از کامپیوترها وظیفه‌شان سرویس‌دهنده است.

Client-Server: یکسری از کامپیوترها وظیفه‌شان سرویس‌گیرنده است.

از نظر شبکه‌کردن تا 10 تا Server مشکل نیست اگر بیشتر خواستیم از peer-to-peer استفاده می‌کنیم.

توزیع پردازش: مرکزی ← Centraliz

توزیع‌شده ← Distributed در این نوع همه چیز به اشتراک گذاشته می‌شود جز Process

SQL همکاری شده ← Goop Crative در این نوع همه چیز به اشتراک گذاشته می‌شود حتی Process یعنی

CPU خودمان را در اختیار کامپیوتر قراردهیم.

Bandwith (پهنای ماده)

یعنی مقدار ظرفیتی که یک رسانه در واحد زمان اطلاعات را منتقل می‌کند. واحد انتقال منظور bit است. پهنای

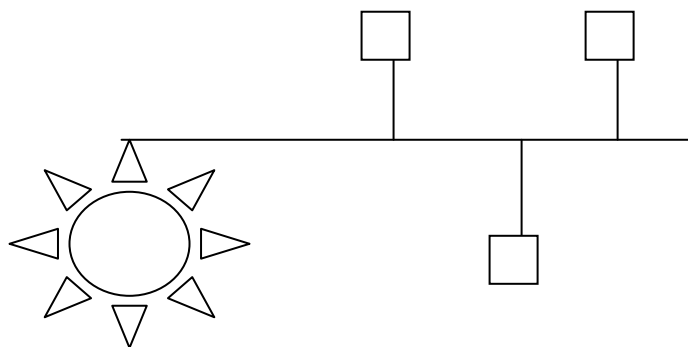
!مقدار خروجی اطلاعات از رسانه در واحد ثانیه . منظور از سرعت شبکه یعنی سرعت eها بیشتر است.

Wireless (بی‌سیم)

به هر ارتباطی که نیاز به وسایل فیزیکی نباشد، می‌گویند.

: Internet (work)

شبکه‌ای از شبکه‌هاست.



Server based: شامل Print server, fax server, application server, file server می‌باشد.

۱- برای تبادل اشتراک‌گذاری در فایلها مانند کپی

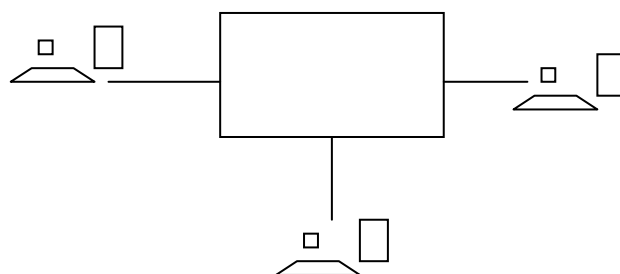
۲- Serverهای کاربردی

۳- مدیریت فلکسها

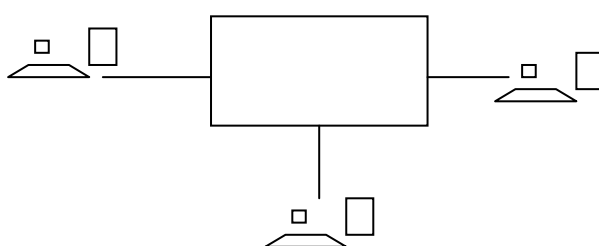
Dedicate (اختصاصی بودن): Server اختصاصی یا dedicated است که فقط کار سرویس دادن باشد و هیچکس نتواند با خودش کار کند. کسی نمی‌تواند برنامه روی خودش اجرا کند.

Server ، Non dedicated غیر اختصاصی یعنی با خودش می‌توان کار کرد ولی اینکار توصیه نمی‌شود چون ممکن است خطا پیش بیاید و سرویس بخوابد و قدرت پردازش پایین بیاید.

I یک ایده است در شبکه‌ها که شما وقتی خواستید کاری انجام دهید به ازای هر کامپیوتر یک hard قوی بگیرد . فایلها را روی آن بریزد. همه با هم در ارتباط باشند و از آن استفاده کند.



II ایده دیگر اینست که می‌گویند برای یکی یک Server قوی بگیرد و همه را روی آن بریزد تا همه با هم در ارتباط باشند. در اینجا باید از کامپیوترهای خاص استفاده شود. از نظر امنیتی ایده اولی بهتر است.



RAID: یعنی امکانات خاصی که در II قرار می‌دهند که اگر اطلاعات از بین رفت باز هم بتوان به آن دسترسی پیدا کرد.

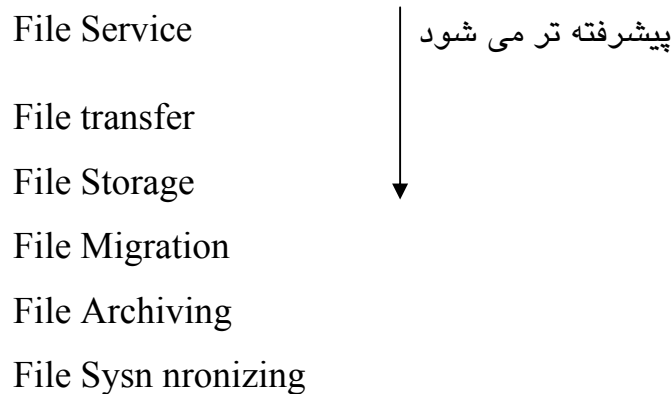
RAID یکسری از دیسکهای ارزان و متنوع . اولین استفاده از RAID ها به خاطر سرعت آنهاست. دیگری آئینه‌گذاری یا mirroring است. یعنی یک hard داریم . هرچه در آن حذف یا پاک می‌شود روی دیگری هم عیناً

حذف و یا اضافه می‌شود. کاربردش اینست که یک online, back up از اطلاعات داریم و اگر یکی از بین رفت دیگری موجود است.

برای RAID ها ممکن است hard4 داشته باشیم که اگر یکی از آنها سوخت یکی که اضافه است یعنی Spare یا یک جای آن hard سوخته را می‌گیرد و آن اطلاعات hard سوخته بازیابی می‌شود.

دسترسی محلی به II راحت تر است چون کامپیوتری که در II است hard قوی‌تر است و از همه نظر بهتر است. بنابراین کارهایی که محلی است، سبکتر است و با یک کامپیوتر قوی راحت‌تر انجام می‌شود. ایده II یک SPF دارد که اگر بشکند ارتباط قطع می‌شود. اما در I اینطور نیست اگر یکی بشکند بقیه جاها کار می‌کند و فقط آن یک نقطه کار نمی‌کند. اگر در I بخواهیم ارتقاء بدهیم همه را باید ارتقاء بدهیم ولی در II اگر بشود Server ارتقاء یابد همه ارتقاء یافته‌اند.

Bottel neck: اگر همه کارها روی یک Server بیاید، این عمل رخ می‌دهد. زیرا که کار زیاد از مرکز می‌خواهد در نتیجه نمی‌شود. ارتقاء دادن آن بهتره چون یک کامپیوتر بیشتر نیست، به صرفه‌تر است. از نظر مدیریت هم باز این بهتر است.



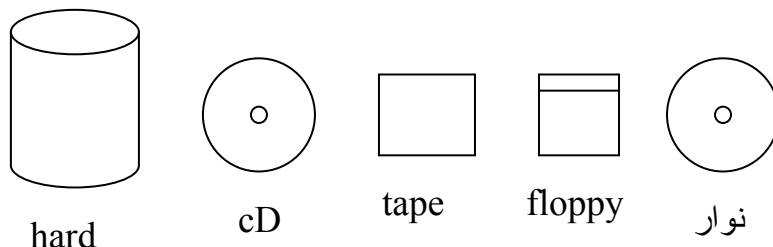
:File tranfer

کارهایی مربوط به دریافت و ارسال و اشتراک‌گذاری فایلها، مجوزهای مربوط به دسترسی باید رعایت شود. که اشخاص صحیح در زمان صحیح به اطلاعات صحیح دست یابند.

File stran: فایلها را در بسته‌های داده ای بتوانیم ذخیره کنیم. با توجه به اینکه منابع ذخیره‌سازی

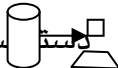
قیمتهای متفاوتی دارد، از نظر قیمت بسته به کاربرد فایلها روی منابع ذخیره‌سازی بهینه کنیم. منابع ذخیره

سازی شامل:



که بررسی می‌کند با توجه به کدام به صرفه‌تر است.

انواع دسترسی به file strage : near line, off line, on line

on line: همان hard است.  و سرعت بسیار بالا است.

Off line: روی hard چیزی نمی‌ریزیم هرچه را که قدیمی شد روی floppy -CD و غیره می‌ریزیم و به یک

مسئول بایگانی می‌دهیم و اطلاعات را از آن می‌گیریم و عیب آن این اس که اپراتور ما یک انسان مانیتک ربات

ساده است.

Near line: ترکیب این دو است. یک دستگاهی است که یک اهرم دارد و CD ها و ... به آن معرفی شده و

اطلاعاتی را که می‌خواهیم از روی آن load کنیم را پیدا کرده و به ما می‌دهد.

Off site Backup: هایی که در ساختمان مربوطه نشیند یعنی در جای دیگر گذاشته می‌شوند. سیستم عامل

بر اساس پارامترهایی که ما تعریف می‌کنیم، فایلها را رتبه‌بندی می‌کند و روی انواع و ساز ذخیره‌سازی ذخیره

کند.

منتقل کردن اطلاعات و داده‌ها براساس پارامترهای تعریف شده بصورت خودکار که مثلاً خودش براساس

اولویت و پارامترهای دیگر اطلاعات را روی device ذخیره‌سازی مناسبی ذخیره می‌کند پارامترهای سیستم

را خودش بصورت اتوماتیک ارزشیابی می‌کند. براساس آن کار می‌کند.

File Migration Supervisor حذف می‌شود در تشخیص online... چون Supervisor پارامترها را برای Server قبلاً تعریف کرده و براساس پارامترهای تعریف‌شده فایلها را طبقه‌بندی می‌کند. تا تصمیم بگیرد کدام online است و کدام off line

File Archiving , قانون moor در مورد احتمال خطا در شبکه توضیح می‌دهد. یک bit از بین برود هیچ‌کس مسئول نیست در شبکه‌ها اهمیت Back up را نشان بدهد. اطلاعاتی که برای ما حیاتی است اگر از بین برود قابلیت بازیابی دارد.

اطلاعات نامنظم را Back up Normal می‌گوییم. در نوع دیگر هم Back up داریم به نام Diffrential , intermantal در intermental : در آن اینطوری است که Back up گرفته می‌شود که هر بار که back up گرفتیم از تغییرات دفعات قبل تا الان back up می‌گیرد.

Diffrention : از آخرین Normal هرچه از وقت تغییر کرده تا الان را back up می‌گیرد. هر نقطه که باشد تمام فایل‌های تغییرکرده را از آخرین Normal back up یا intermental معمولاً Normal می‌گیرد و ذخیره می‌کند. ولی کاری به بیت آرشیو ندارد. اگر اطلاعات از بین برود ابتدا اولین آرشیو را که Normal است و بعد آخرین آرشیو را Diffrentional را می‌گذاریم.

File synronizing: نیاز به آخرین نگارشها و تغییرات را داریم. این وظیفه سیستم عامل است که همیشه آخرین نگارشها را در اختیار می‌گذارد.

نرم افزار شبکه:

۱- سخت افزار

۲- نرم افزار

۳- FW (feir ware): یک چیزی بین سخت افزار و نرم افزار است مثل RAM.

شبکه‌ها وقتی می‌خواهند پروتکل را پیاده‌سازی کنند، نیاز به نرم‌افزاری دارند که بتواند آنها را پیاده‌سازی کند. وقتی کامپیوتر می‌خواهد از شبکه‌ها استفاده کند، باید نرم‌افزاری در آن باشد که ارتباط را برقرار کند.

Redirector: جزیی است که باعث می شود ایستگاهها از امکانات شبکه استفاده کند. از جمله وظایف آن عبارتند از:

- ۱-بدیهی است امکان log out, login از شبکه است یعنی ورود و خروج به شبکه
 - ۲-امکان استفاده شبکه و منابع موجود یعنی نرم افزار باید امکان این را بوجود آورد که از منابع و امکاناتی که در شبکه وجود دارد، استفاده کنیم.
 - این نرم افزار سالهای گذشته بصورت اضافه (خارج از سازمان) به سیستم عامل می چسبد ولی از windows به بعد این نرم افزارها بصورت جزیی از سیستم عامل درمی آمد.
 - مزیت: سیستم عامل، شبکه را جزء بطن خود می شناسد. کارکردن با آن راحت می شود و سرعت بالا می رود.
 - این نرم افزار بسیار پیچیده است. و با خیلی مسائل باید کار کند.
- مشکلات:

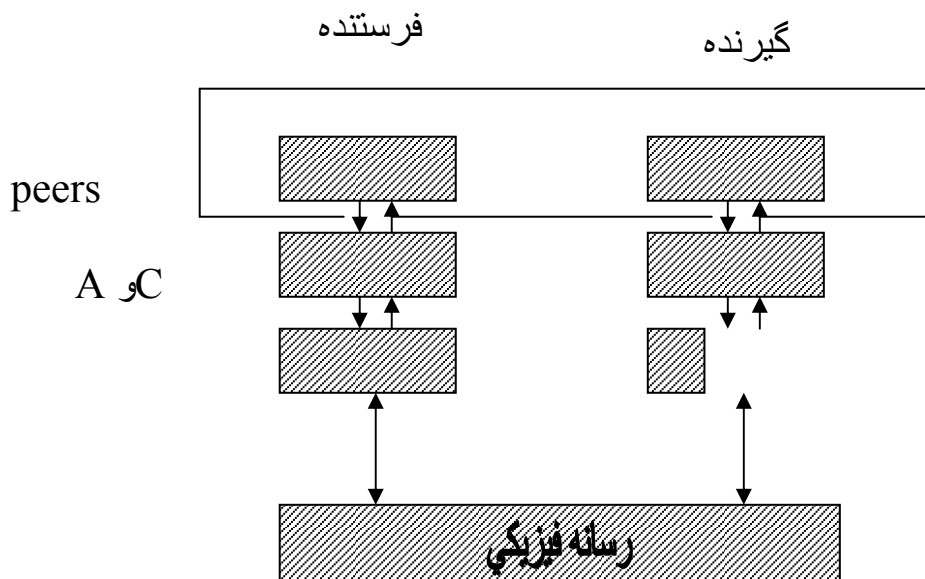
- ۱-ما وسائل ارتباطی که داریم می تواند خیلی متفاوت باشد مثلاً از تلفن گرفته تا به بالا مثل ماهواره و... این نرم افزار به دلیل سرو کار داشتن با طیف وسیعی از اینگونه مسائل پیچیده است.
- ۲-بحث ارسال و دریافت بحث بزرگی است. یعنی انتقال بیت هایی که اطلاعا رابا خود می برند یعنی کامپیوترهای واسط که اطلاعات ما از Switch می کنند تا به مقصد برسد packet اطلاعات که می روند در شبکه ممکن است به ترتیب ارسال به مقصد نرسند. نرم افزار شبکه باید تضمین کند اگر از نقطه ای به نقطه ای دیگر اطلاعات می فرستیم تضمین انتقال شوند. در خط با توجه به اثرات منابع خارجی و... ممکن است اطلاعات تغییر کند. سپس باید ملاکی برای تشخیص خطا داشته باشیم.
- به دلیل حساسیت کاربرد نرم افزارها مسئولیت نرم افزارها سنگین است. نرم افزارها الان به آن اندازه ای که باید قوی باشند، قوی نیستند. الان نرم افزارها بسیار پیچیده اند ولی مدیریت انسان به روی آنها کم است و پیشنهاد شده است که به این مسأله فکر کنیم.

۱-روش Top-down :

یعنی اگر مسأله سختی داریم آنرا بشکنیم به زیر مسأله‌های کوچکتر که اگر آنها حل شوند، مسأله بزرگتر حل شد، عمل شکستن را تا آنجایی ادامه بدهیم تا به مسأله‌ی برسیم که زیر مسائل قابل حل و احاطه باشند. مزیت دیگر این روش آنست که این روش مطابق با خصیصه‌های مغز انسان است.

۲- روش لایه‌بندی:

این است که هر مسأله‌ای که پیش آمد آن را لایه‌لایه کرده و هر لایه را خاص یک کاربرد بگیرد. و در واقع لایه‌های دیگر را درگیر آن لایه‌های دیگر نکنیم. به این می‌گویند object oriented یعنی از سطوح مختلف به آن مسأله نگاه کنیم. نرم‌افزار شبکه را بصورت زیر می‌بینید: می‌گویند چند لایه است. می‌گویند نرم‌افزار هم در طرف گیرنده و هم فرستنده مثلاً 3 لایه‌شده و هرکدام وظیفه خاصی دارند. در شبکه‌ها به هرکدام از این لایه‌های نظیر peers می‌گویند. هرکدام از آنها برای ارتباط با یکدیگر نیاز دارند روی یک قراردادی باهم توافق برسند. پس در نتیجه به تعداد لایه‌های گیرنده و فرستنده پروتکل داریم. به مجموع پروتکل‌های قوی یک نرم‌افزار شبکه پروتکل Stack می‌گویند یا protocol Suite به عنوان مثال TCP/IP یا IPX/SPX وظیفه هرکدام از این شاخه‌ها اینست که به لایه‌هایی بالایی خود یکسری خدمت یا Service بدهند. منتها این استفاده به این شکل است که لایه‌بالایی از یک فاصله‌هایی که به آن Interface می‌گویند به آنها وصل می‌شود و از خدمات استفاده می‌کند. درشی‌گرایی یا object oriented معنایی داریم بنام Encapsulation که در آن هرچیزی که می‌خواهد به داده‌ای دسترسی پیدا کند. باید از مرحله‌هایی که آن تعریف کرده به آن دسترسی پیدا کند. اینجا هم همینطور چون از Interface های تعریف نشده به آن دسترسی پیدا می‌کنند.



تعریف فاوول Interface :

مرحله‌هایی که امکان ارائه سرویس به لایه بالایی می‌دهد و لایه بالایی باید از آنها استفاده کند تا به لایه پائینی دسترسی پیدا کند.

تعریف معماری شبکه:

مجموع پروتکل‌های شبکه لایه‌های یک شبکه را معماری شبکه گویند.

اولین مزیت لایه‌ها این است که هر لایه‌ای کار خودش را انجام می‌دهد و درگیر جزئیات کار بقیه نشود. مزیت دیگر در ازاء تغییر فن‌آوری است.

چون درگیر جزئیات کار هم نمی‌شود از طریق فاصله از سرویس‌های دیگری استفاده می‌کند. هر لایه را به لایه دیگر با تکنولوژی بالاتر جایگزین کنیم. به این شرط که Interface عوض نشود. یعنی مزیت این است که لایه‌های مابین راحتی با لایه‌های بهتر جایگزین می‌شوند.

فرض کنید فرستنده ما رئیس یک شرکت باشد. در یک کشوری و بخواهد اطلاعات را به کس دیگری در جای دیگر برساند. پس باید متن اطلاعات او ابتدا ترجمه شود. فرض کنید در هر دو طرف منشی داریم. منشی اول به زبان A, C و طرف دوم به زبان B, C مسلط است. لایه مدیر ابتدا سرویس ترجمه را از لایه پائینی یعنی مترجم می‌گیرد. لایه پائینی هم ارسال و دریافت پیام می‌کند. از طریق منشی. منشی‌ها با هم از طریق رسانه فیزیکی در ارتباطند. لایه‌های نظیر وقتی دارند با هم صحبت می‌کنند فکر می‌کنند فقط دارند با هم صحبت می‌کنند. در صورتیکه گردش اطلاعات از بالا به پایین فرستنده و سپس رسانه و سپس از پایین به بالای گیرنده. مسأله اینست که هر کدام از لایه‌ها می‌توانند یکسری اقلام داده‌ای اضافه کنند تا لایه نظیر آن بتوانند از اطلاعات استفاده کند و آن لایه نظیر می‌تواند آنرا استفاده کند و سپس آنرا پاک کند. و فقط به درد لایه نظیر آن می‌خورد و اطلاعات فرستاده شده، باید مورد نیاز آن باشد به این اطلاعات beader می‌گویند. لایه‌ها به راحتی می‌توانند روش‌های خود را عوض کنند و به لایه‌های دیگر لطمه نخورده و هر کدام از لایه‌ها، اطلاعاتی را به اطلاعات لایه قبلی اضافه می‌کند. و لایه گیرنده، اطلاعات را حذف می‌کند.

اصطلاحات شبکه ای:

addressing: کامپیوترها وقتی می‌خواهند ارسالی انجام دهند باید آدرسش را داشته باشند. اینکار راهی ندارد جز اینکه همه کامپیوترهای شبکه یک آدرس منحصر به فرد داشته باشند که کامپیوتر دیگری نداشته باشد. و معمولاً دو-سه نوع addressing داریم.

نوع اول physical addressing و نوع دیگر logical address است. اولی آدرسهایی است که یک آدرس ثابت به کامپیوتر نسبت داده نمی‌شود. و بر روی آن حک میشود. مثلاً شبکه‌ها آدرس unic دارند که 48 bit و معمولاً 8 bit اول برای کد کارخانه و بقیه شماره کارتهاست. در دومی که قابل تعویض است مثل IP Address. نوع دیگر، آدرسهایی، آدرس سرویسهاست و منطقی هستند و کامپیوتری که از آنها استفاده می‌کند، معمولاً چندین سرویس می‌دهد. به همین خاطر logical به تنهایی کافی نیست چون client باید هنگام استفاده از ای آدرس باید نوع سرویس را مشخص کند. اینجا پس آدرس سرویس یا port مطرح می‌شود. آدرس فیزیکی در LAN ها و آدرس logical در شبکه‌های بزرگتر است.

۲-Communication director:

نام دیگر آن سمت ارتباط است. ارتباط ممکن است به سه شکل برقرار شود. یعنی جهتش به سه طرف باشد. ۱-Simplex: یعنی ارتباطی که در آن یک طرف همیشه فرستنده است و یک طرف همیشه گیرنده است. مثل رادیو.

۲-half duplex: در هر لحظه اطلاعات در یک سمت می‌تواند برود یعنی دو طرفه است ولی غیر همزمان مانند بی‌سیم پلیس.

۳-full duplex: یعنی همزمان می‌تواند هم بفرستد و هم بگیرد. مانند تلفن

۳-Error direction & correction:

در شبکه‌ها خطاها به وفور پیدا می‌شود. اگر به این مفهوم توجه نشود، مفهوم شبکه زیر سؤال می‌رود و بحث حیاتی است. Detection یعنی خطا را تشخیص دهیم که رخ داده. Corection یعنی خطای تشخیص داده شده را بتوانیم صحیح کنیم.

۴-Packet sequence control:

کنترل ترکیب پارکتها، یعنی لایه‌ها مثل لوله باشند یعنی به هر بیتی که در داخل لایه‌ها ریخته می‌شود . در طرف دیگر به همان ترتیب دریافت شود.

۵-flow control:

یعنی خیلی وقتها فرستنده و گیرنده قدرت یکسان ندارند. اما اگر گیرنده قدرتش کم باشد، ممکن است داده از بین برود. بحث اینست که یک گیرنده قوی اطلاعات را بتواند کنترل کند و چه جوری یک فرستنده قوی و یک گیرنده ضعیف با هم کار کنند.

۶-Multi plening :

استفاده همزمان و اشتراکی از یک منبع گران قیمت است. یعنی همه اشتراکی و همزمان از یک خط استفاده کنند.

۷-routing :

یعنی مسیریابی در WAN معنی دارد و در LAN چون یک مسیر مشترک وجود دارد، معنا ندارد . از بین مسیرهای موجود بهترین مسیر را براساس پارامترهای مختلف انتخاب شده است.

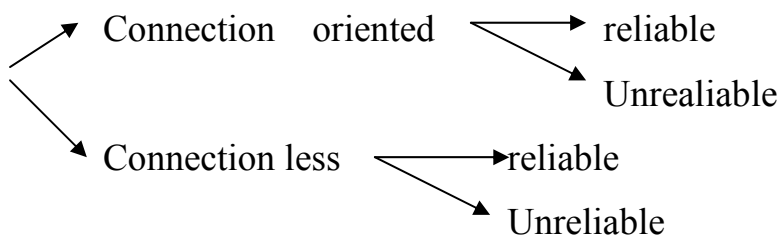
۸-Service:

سرویسی که لایه به لایه بالایی خودش می دهد. می تواند انواع مختلف داشته باشد، قابل اطمینان بودن و نبودن یا قابل اتصال بودن یا نبودن. قابل اطمینان بودن یعنی لایه به لایه بالایی هر سرویسی داد. حتماً به همان شکل در آن طرف تحویل می دهیم. قابل اطمینان نبودن یعنی سعی می کنیم اگر شد که شد . ولی با اینکه مزیت آن نسبت به قابل اطمینان بهتر است ولی هزینه هم دارد، قابل اطمینان نبودن سرویسی است که به این درد می خورد که ما انتخاب داریم. اگر مهم نباشد چه جوری باید از سرویس غیرقابل اطمینان استفاده می کنیم. ولی اگر مهم باشد از سرویس قابل اطمینان استفاده می کنیم.

سرویسها می توانند connection oriented باشند یا بدون اتصال connection less .

Connection oriented یعنی این سرویس باید با طرف گیرنده یک توافق اولیه را انجام دهد و اگر آن توافقات برقرار شود در آن صورت ارتباط را برقرار کند به عبارتی به اتصال وابسته است . در connection

less بدون هیچگونه توافق اطلاعات فرستاده می شود: مثل تلفن. برای اولی مثل تلفن و برای دومی پست با email



موسسه‌ای استاندارد به نام ISO یک پروتکلی ارائه کرد که همه شبکه‌ها حتی اگر بهم وصل نبودند با رعایت کردن آن چهارچوب می‌توانستند با هم کار کنند. آن چهارچوب نرم‌افزارهای شبکه بود.

Open system interconnection OSI

Open: استانداردها یا open هستند یا Close. Close: آنست که شرکت تجاری آنرا ارائه می‌دهد و برای محصول خودش ارائه نمی‌دهد و در نتیجه مشخصات محصول خود را زیاد ارائه نمی‌دهد. به این دلیل به آن close می‌گویند. چون کسانی دیگر نمی‌توانند اطلاعات خودشان را بنویسند یا نه آن اطلاعات اضافه کنند. Adhed: استانداردهایی که زیاد استفاده شده‌اند و مصطلح شده‌اند.

مدل OSI گفت نرم‌افزار شبکه باید ۷ لایه باشد و هر لایه باید چه وظیفه‌ای داشته باشد. لایه اول physical، لایه دوم data link، لایه سوم Network، لایه چهارم transport، لایه پنجم Session، لایه ششم presentation و لایه هفتم Application نام دارد. مدل OSI مفهوم Service دادن را به خوبی دید و مفاهیم Protocol Inter face ... را بخوبی رعایت کرد.

Physical-۱

هرکار مرتبط با لایه فیزیکی یا Media یا خط انتقالی را به عهده لایه فیزیکی می‌گذاریم. هر چیزی مرتبط با خط یا حامل باشد. مدیر تیش را به لایه فیزیکی می‌دهیم. مثلاً pinها، به لایه Datalink سرویس می‌دهد. از این سمت بیتها را می‌گیرد. و در سمت دیگر تحویل می‌دهد.

۲-Datalink: به لایه بالاتر این امکان را می‌دهد که داده‌هایی را که network برای آن می‌فرستد، انتقال بدهد. این انتقال منطقی است و فیزیکی نیست. تنظیم سرعت بین فرستنده و گیرنده multiplexing و اولین

شرط آن این است که (مهم) وقتی دارد با طرف دیگر لایه Datalink متناظرش صحبت می‌کند هر دو لایه باید روی یک خط باشد و خط مال آنها باشد. (مبدأ و مقصد باید روی یک خط باشد) و کار دیگرش frame کردن است یعنی اول و آخر داده را می‌بندد و محکم می‌کند و می‌فرستد.

۳- Network :

می‌گویند من اگر وجود داشته باشم در شبکه، آن شبکه برای فرستادن اطلاعات از مبدأ به مقصد راه‌های مختلفی وجود دارد که انتخاب بهترین مسیر با لایه Network است و اصطلاحاً به آن routing یا مسیریابی می‌گویند و در شبکه‌های WAN به درد می‌خورد. و با توجه به وضعیت شبکه و اتفاقاتی که الان دارد انجام می‌شود مسیر را انتخاب می‌کند. بهترین مسیر خودش ندارد. بهترین مسیر بسته به نظر شبکه یا Supervisor دارد. کار دیگر آن اینست که وظیفه کنترل ترافیک را دارد. اگر تداخلی هم پیش بیاید وظیفه Network است که آنرا حل کند. کار دیگر آن تبدیل پروتکل‌هاست. یعنی مثلاً دو شبکه بهم وصل کردیم که با هم متفاوت هستند. اگر بخواهند با هم کار کنند باید پروتکل‌های آنها بهم تبدیل شود. یکی از وسایل خیلی مهم در Network مسیریابها یا routerها هستند. Routerها، مثلاً برای لایه Datalink مودم هستند.

۴- Transport:

اولین وظیفه‌اش این است که بین جزئیات شبکه و کاربر خط بکشد. و کاربر خیلی درگیر مشخصات فنی شبکه شود. کارش شبیه لوله است. ممکن است چیزی که به آن می‌دهیم به چند قسمت تقسیم می‌کنند. حال با توجه به اینکه لایه پائینش مسیریاب است پس مسیرهای متفاوت را برای لایه تقسیم‌شده انتخاب می‌کند. پس ممکن است ترتیب آنها بهم بخورد. وظیفه لایه transport است که لایه‌ها را مرتب کند. یعنی به هر ترتیبی که رفتند به همان ترتیب هم آن سمت به مقصد برسند. اولین کارش شکستن اطلاعات است. Error control , flow control را انجام می‌دهد. واحد انتقال segment, transport و واحد انتقال Data, frame است.

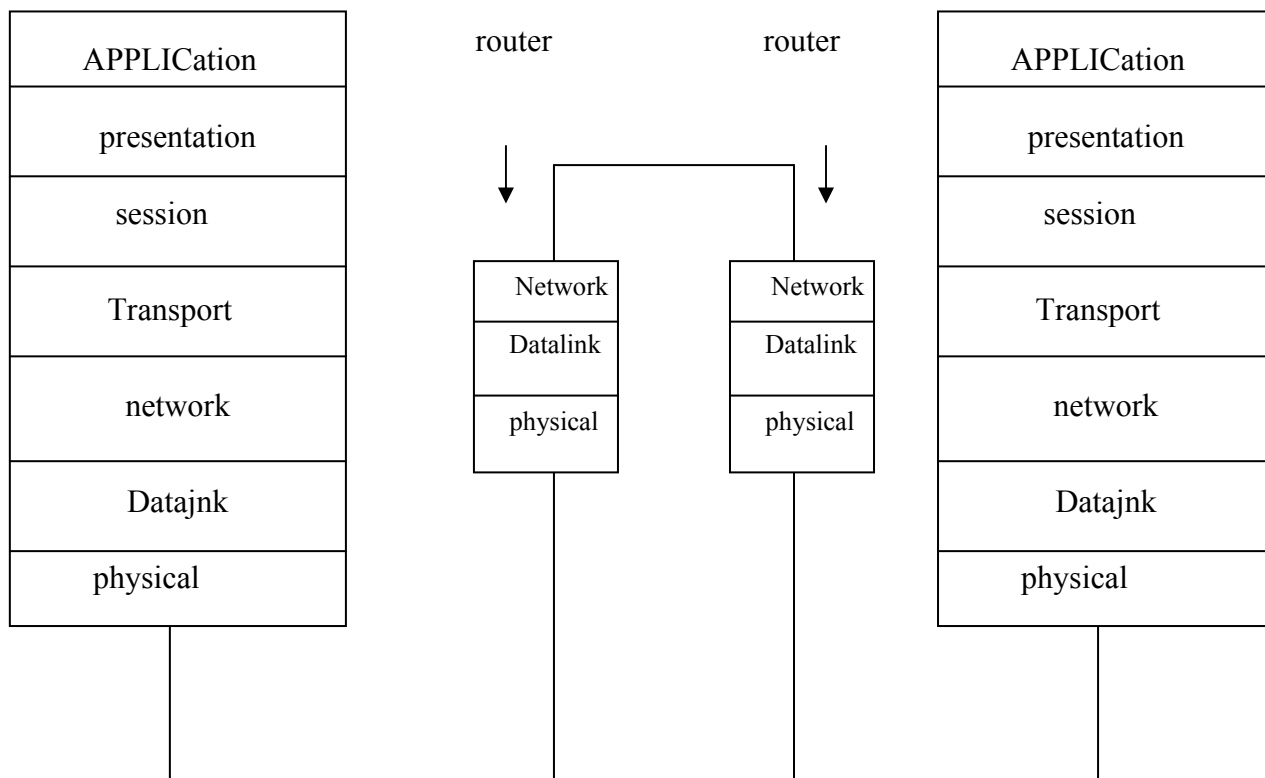
خیلی از کارهاییکه transport انجام می‌دهد Data link انجام می‌دهد. این پیشنهاد است که کدام بهتر است استفاده شود. یعنی بسته به مورد استفاده از هر دو باید استفاده شود. اما در پیاده‌سازی اشکال در خطاست که ممکن است برای داده اشکال بوجود بیاید. اگر در شبکه خطا وجود داشته باشد ممکن است با

خطا دوباره بفرستد یا مشکل آنرا درست کند یا اینکه آنرا در مقصد درست کند ولی در Datalink باید بین بیشتر را انتقال دهد. امروزه چون دیگر خطاها کم شده، احتیاج به این همه محکم‌کاری نیست. تفاوت Transport با Datalink: با توجه به فرض که در Datalink داشتیم یعنی روی یک خط‌بودن دو طرف، ای شرط در transport وجود ندارد.

رجوع شود به جدول صفحه ۱

اما در پیاده‌سازی اشکال در خطاست که ممکن است برای داده، اشکال به وجود بیاید اگر در شبکه خطر وجود داشته باشد ممکن است یک خطا دوباره بفرستد (یعنی اطلاعاتی که دچار خطا شده را دوباره بفرستد) یا اینکه آنرا در مقصد درست کند. ولی در Datalink بای بیت بیشتری را انتقال دهد. امروزه چون دیگر خطا کم شده، احتیاج به این همه محکم‌کاری نیست.

تفاوت Datalink، transport: با توجه به شرطی که در Datalink داشتیم یعنی روی یک‌خط‌بودن دو طرف، این شرط در transport وجود ندارد.



وظیفه‌اش ارتباط بین کامپیوتر مبدأ و مقصد را برقرار می‌کند. و آن را کنترل می‌کند. مثلاً بی‌سیم پلیس که نوبت‌دهی حرف‌زدن بین پلیس‌ها به نحوی وظیفه‌ی لایه جلسه است.

۶-presentation:

زیاد با 0 و سروکار ندارد و با محتوای داده‌ها و شکلشان کار دارد. خیلی وقتها ممکن است نیاز باشد شکل داده‌ها عوض شود. مثل ZIP کردن یا رمزگذاری. کاردیگرش تبدیل فرصتها به همدیگر است و استاندارد on wire دارد یعنی ما اطلاعات را می‌فرستیم . نمی‌دانیم فرمت کامپیوتر دیگر چیست؟ onwire یعنی هرکس خواست اطلاعات را بفرستد، باید یکسری استانداردها را رعایت کند.

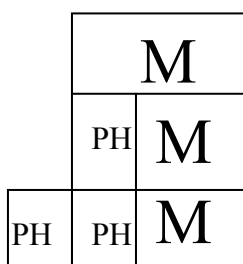
۷-application:

معمولاً یکسری استانداردهاست که دیگر مستقل از ماشین و شبکه‌اند که سرویس‌هایی را به برنامه‌ی ما یا خود ما می‌تواند ارائه کند مانند HTTP, Web و....

مدل OSI در تبادل اطلاعات اینگونه عمل می‌کند:

مدل Application ممکن است یک تکه که به آن header می‌گویند برای داده بالایی اضافه کند.

APPLICation
presentation
session
Transport
network
Datajnk
physical



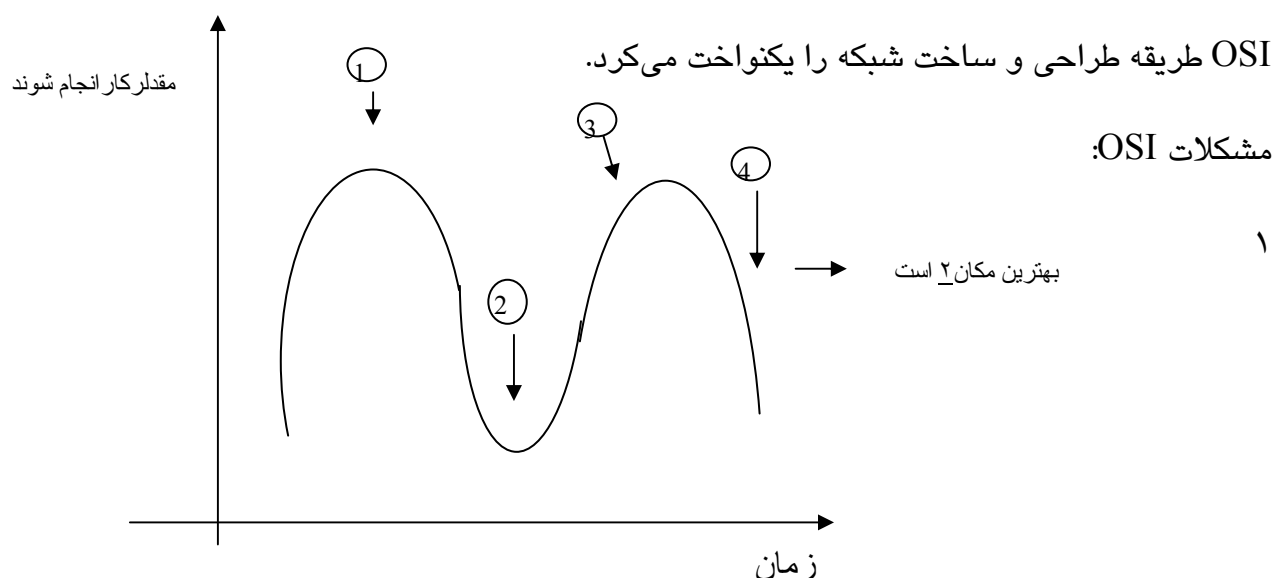
چون حجم اطلاعات زیاد بود و برای خط بزرگ بود ، شکسته شد.

Applicati
on
transport
internet
Host
network



header
TCP/IP 4 لایه دارد و لایه‌های session, presentation, datalink از OSI را ندارد.

روند تولید و توسعه‌اش (TCP/IP) بر عکس IP است. در TCP/IP یکسری مهندس اول پیاده‌اش کردند. سپس برایش طرح درآوردند. کسانی که TCP/IP را طراحی کردند به ازاء مسئله به آن رسیدند ولی دید کلان نداشتند پس open نیست ولی OSI، open است. TCP/IP مفاهیم سرویس پروتکل و... را پشتیبانی نمی‌کند. TCP/IP با همه صفت‌هایش حریف OSI شد. پیاده‌سازی‌ها همه براساس TCP/IP است. به این دلیل که زمانیکه OSI آمد TCP/IP وجود داشت و ثانیاً وقتی موضوع جدیدی پیش می‌آید دانشمندان و متخصصان روی آن بررسی می‌کنند و سپس بعد از اینکه همه تحقیقات روی آنها انجام شد، حال شرکت‌های سرمایه‌گذاری شروع می‌کنند به سرمایه‌گذاری روی اینها.



۱- وجود TCP/IP: وقتی OSI آمد، زبان بدی بود چون TCP جای خود را باز کرده بود و از استانداردها و سرمایه‌گذاری گذشته بود. این مشکل را تحت عنوان bad timing می‌شناسیم.

۲- درست است که روی طراحی آن کار زیادی انجام شده بود اما یکسری اغراض سیاسی و شخصی در آن خیل شده بود و بدون غرض طراحی نشده بود در خیلی لایه‌ها توازن وظیفه روی آن رعایت نشده بود یعنی روی خیلی لایه‌ها وظیفه کم بود. مثل presentation و روی بعضی لایه‌ها زیاد بود. مثل Datalink و این کار را به دلیل اینکه خواستند یکسری لایه وجود داشته باشد این لایه‌ها را وجود آورده بودند به این می‌گویند، bad technology

۳- پیاده‌سازی بد: طراحان OSI محقق و دانشمند بود ولی آن چیزی که ارائه کرده بودند باید می‌رفت در عمل درست می‌شد. یعنی مهندسی میشد. برای همین چون OSI، حجم سنگینی داشت می‌خواست یکسری الگوریتم را پیاده‌سازی کند در حقیقت آن پروتکل عملی نبود. این مشکل را تحت عنوان bad implement می‌شناسیم. در مقابل OST، TCP/IP بود. توسط مهندسین طراحی شده بود. و کاری به چیزهای جانبی‌اش نداشتند. بنابراین TCP/IP زود به جواب رسی. ولی در ادامه‌اش مشکل داشت. TCP/IP مفهوم سرویس، پروتکل، Interface نداشت و کلان کسری نداشت. بنابراین سیستم open واقعی نبود. تنها شانس‌اش زود جواب‌دادنش بود و کارش را خوب انجام داد. امروزه هیچ پیاده‌سازی‌ای از OSI نیست ولی در دنیا آن را به عنوان مرجع قبول ندارد.

TCP/IP در لایه‌Transport و پروتکل را دارد که متفاوتند که هم connectionless و هم connection oriented را پشتیبانی می‌کند. لایه‌host آن را به شبکه وصل می‌شود. لایه‌Application, transport, Network آن متناظر با همان لایه در OSI است و بقیه لایه‌ها را ندارد. Net از پروتکل IP استفاده می‌کند و routingهای مابراساس استانداردهایی است که در این لایه تعریف می‌شوند.

Transport دو پروتکل را پشتیبانی می‌کند: TCP, UDP. اگر دقت شود سرویسهای بالا Application می‌تواند از سرویسهای پائین استفاده کند اما می‌تواند به صورت انتخابی استفاده کند. مثلاً E-mail از UDP استفاده می‌کند.

مزیت TCP/IP نسبت به OSI: در لایه transport به لایه Application دو نوع سرویس حق انتخاب داده است در OSI، یک سرویس transport داریم که unreliable است.

شبکه‌های ناول (Novel Network):

شبکه ساده و محکمی بود. به درد کارهای فایلی می‌خورد. Share، تبادل فایلهای را انجام می‌داد. بیشتر file server بود. پروتکل آن به شکل زیر بود. در لایه physical، Datelink، استانداردهای token Ring، ethernet، Arcnet را پشتیبانی می‌کرد. در لایه Network از استاندارد TPX استفاده می‌کرد که

unreliable بود. در لایه transport دو نوع پروتکل داشت. SPX, NCP CO . در لایه application پروتکل print server, file server را داراست.

Application
Transport
Network
Datalink
physical

اینترنت:

سال ۶۰ وقتی جنگ سرد بین آمریکا و شوروی مطرح بود آمریکا کم کم متکی می شد به شبکه هایش برای انجام کارهای نظامی ولی ترسشان از این بود که اگر جنگ به وجود می آمد، ممکن بود تکه ای از شبکه هایشان از بین برود به فکر افتادند که کاری کنند که اگر قسمتی از کامپیوتر از بین رفت، بقیه کارشان را بکنند و خیلی به هم وابسته نباشند. پروژه ای مطرح شد توسط دولت DOD آمریکا. دانشگاه های آمدند خطوطی طراحی کردند. خطوطی بود که اوایلش S6K بود. راه حلی که مطرح کردند این بود که هر کدام از کامپیوترها حداقل به ۲ یا ۳ تا دیگر ضرورتاً وصل باشد تا اگر یکی خراب بشود، آنهای دیگر کار می کنند. ارتباط در اینجا connection oriented نبود. چون چیزهای ثابت را قبول نمی کرد. و الگوریتم های پویا می خوسات تا بسته به شرایط الان ارتباط را برقرار کند. برای همین در لایه Network پروتکلی که استفاده کردند، IP بود و reliable هم نبود. اسم این شبکه Arpanet بود. حال دانشگاه ها وارد کار نشدند و ...

NSFNET:

دانشگاه ها، خودشان شبکه راه انداختن، NSFNET, Arpanet با هم موازی کردند تا اینکه در یکسالی به هم وصل شدند و انفجاری در شبکه رخ داد تا اینکه internet اولیه ره وجود آمد. کار آنها به router بستگی داشت و اینجوری نبود که خودشان مسیریابی کنند.

لایه فیزیکی:

به قسمت فیزیکی شبکه کار دارد. یعنی واسط یا مربای ارتباطی اش در لایه فیزیکی قرار دارد. مثلاً connector ای که به شبکه می خورد باید یکسری قواعد داشته باشد. یکی از بیهوش ترین لایه هایی است که در شبکه وجود دارد. و یکسری بیت می دهند تا آنها را انتقال دهد و هر کاری هم خواست آنجا دهد در هین سطح است. مثالی از بیهوشی اش hube است هر داده ای بهش بدهیم 4 تا port خروجی داشتیم آن داده را روی تمام خطها می فرستد و شلوغ می کند. Repearator hube از نمونه های بیهوشی شبکه هستند. و کار این لایه این است که می آید اطلاعا را روی یکسری موج الکترومغناطیسی سوار می کند و آن را روی آها می فرستد، و شکل آن موج بسته به داده ما تغییر می کند. این امواج carniel نام دارد و گیرنده با توجه به شکل موج اطلاعات را حدس می زند. موج یا آنالوگ هستند یا دیجیتال که آنالوگ می تواند هر مقداری را بگیرد ولی دیجیتال فقط ۱ و ۰.

رسانه ما که امواج روی آنها ارسال می شود مانند کابلها به دو شکل می تواند باشد guided یا هدایت کننده . unguided.

هدایت شونده:

امواج را مجبور می کند از روی خودشان رو شوند مثل ریل قطار، که قطار باید در آن حرکت کند. غیر هدایت شونده:

امواج را پخش می کنند تا به مقصد برسد مثل ماهواره.

امواج حاملهایی هستند که ما داده امان را روی آها می فرستیم.

انواع امواج الکترومغناطیسی (EM):

تعاریف مربوط به الکترومغناطیس:

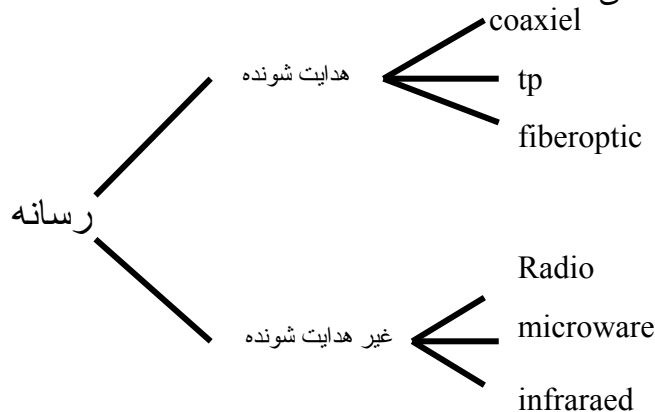
۱-فرکانس یا بسامد:

تعداد امواج در واحد زمان که باعث تغییر در رفتار امواج هستند. گوش انسان در دامنه 20-30KHZ می شوند. Infrared مانند دستگاه کنترل تلویزیون. در جدول پایین هرچه بالاتر می رویم تولید فرکانس

مشکلتر می‌شود. و از طرفی خطرناک هم هست. امواج ماهواره در سطح micro wave است. اولین نکته در ماهواره دید مستقیم یا onsite است. که باید آن دودیش به هم point to point باشند.

Infrared ها برد خیلی کمی دارند و در وسایل خانگی از آنها استفاده می‌شود خیلی راحت تولید می‌شوند

ولی خیلی ضعیف هستند نور خورشید آنها را خراب می‌کند.



اولین معیاری که در شبکه‌ها مهم است cost و هزینه است. بعد از آن installation requirement : نیازهای نصب، بعد از آن band width و سپس Attenuation مهم هستند.

تعریف Attenuation: مقدار ضعیف شدن امواج که روی رسانه قرار گرفتند در واحد فاصله واحدی که نشان می‌دهد به ازاء امواج فاصله امواج ما چقدر می‌میرند هرچه مقدار Attenuation کمتر باشد، بهتر است.

معیاری بعدی که در شبکه‌ها مهم است اثر امواج خارجی EM است امواجی که به دو دسته broad band, based band تقسیم می‌شوند.

Based band یعنی در هر لحظه روی media فقط یک موج باشد ولی broad band اینطور نیست.

کابل‌های Coaxial در واقع اسمشان coaxis است و به شکل زیر هستند و معمولاً در شبکه‌های boss استفاده می‌شود دو نوع هستند SON و 75π .

SO ای‌ها در (هم ۰,۲) اینچ قطر نشان است ولی 75π ای‌ها (04). SO^π ها

base band هستند و دیجیتال. ولی 75π هم آنالوگ هستند هم دیجیتال. 75π هم base band هستند و هم

broad band ولی بیشتر broad band هستند. این کابلها ارزان هستند. هزینه از 75π از SON بیشتر

است. نصبشان خیلی آسان است. پهنای باندشان $10 \frac{MB}{SEC}$ است. 50π میرایی استان 180mI خوب است

ولی 75π تا soom اثر امواج خارجی روی آنها کم است.

کابل‌های tp یعنی جفتهای به هم پیچیده . جدیداً در تمام شبکه‌ها استفاده می‌شوند. و علت اینکه پیچیده شده‌اند این است که امواج‌شان همدیگر را خنثی کنند. معمولاً یک جهت استفاده نمی‌شوند بلکه چند تا از آنها را جمع

می‌شوند.

می‌کنند و به دو دسته تقسیم

Unsheldedtp

⇒

یعنی TP هایی که پوسته ندارند

UTP

sheldedtp

⇒

TP هایی که هم کابل پوسته دارد و هم مجموعشان

STP

استانداردهای مختلف دارند به نام category از ۱ شروع می‌شوند به ۶ می‌رسند و براساس خصایص این کابلها است. Castes نقطهٔ اوج کابل‌های UTP بود که تا 100MB را پشتیبانی می‌کرد که شامل UTP,4 بود.

هرچه مقدار پیچش در سیم به هم بیشتر باشد کابلی بهتر کار می‌کند. پیچیده‌شدن این سیمها به هم یک

استانداردی دارد. در این کابلها از ۸ کابل، ۴ تا از آنها استفاده می‌شود. یکی از کابلها Send است و یکی Receive

. اگر در hub به هم وصل شوند. و send یکی به send دیگری وصل شود که نوبت اطلاعات به هم می‌خورند

و از بین می‌روند ولی باید send یکی به Receive دیگری بخورد به این سیمها Cross می‌گویند.

نیازهای نصب این سیستمها TP کم است. هزینه‌اش کم است. پهنای باند نسبت به Categor فرق می‌کند ولی

بیشتر از ۲ است. امواج خارجی روی آنها اثر دارد و baseband هستند و هزینه استان هم کم است.

فیبرهای نوری:

ارسال اطلاعات یک طرفه است. یک طرف یک منبع نوری داریم که نور ایجاد می‌کند طرف دیگر یک گیرنده

است. و به ازاء ۱ و ۰ که به آنها داده می‌شود به ازاء ۱ها پالس می‌فرستد و تبدیل به نور می‌شود و به طرف

دیگری فرستد. یعنی فرستنده در ازاء تعریف پالس ۱ دارد و در غیراینصورت ۰. و بنابراین ۰ و ۱ها را تنظیم می‌کند. قسمت اصلی زمانی که در اینکار هدر می‌رود موقع انتقال است که با تغییر تکنولوژی می‌شود سرعتش تا ۵۰۰۰۰ برابر بشود. نوری که می‌تاباند با زاویه خاصی به داخل همان لوله‌های شیشه‌ای می‌تاباند که به طور منظم بشکنند. طرف گیرنده هم نوری را دریافت و تبدیل می‌کنند. به بعضی از فیبرهای نوری می‌توان با زوایای مختلف تاباند که به این سیمها multimode می‌گویند و چند فوتون می‌تواند در طول سیم برود. Singlemode ها فقط یک موج در آنها حرکت می‌کنند و در آنها میرایی کم است.

تولیدکننده موج در نوع می‌تواند باشد LED , KED. LASER CONDUCTOR از دیود دیگری از امواج لیزری استفاده می‌کند. خرج ASERCON از LED بیشتر است و عمر سن هم کمتر است. برای محیطهای انسانی خطرناک هستند. نرخ انتقال LASERCO خیلی بیشتر از LED است هم multimode هست و هم Single mode ولی LED multimode است. مسافت انتقال اطلاعات LASER CON از LED بیشتر است. هزینه فیبرهای نوری زیاد است. نیازهای نصبشان خیلی زیاد است. تا ۱۰۰۰MB را راحت پشتیبانی می‌کند. مرئی‌شان تا حد کیلومتر می‌شود. امواج خارجی که به آنها اثر ندارد و معمولاً baseband هستند. محل استفاده انسان در baseband شبکه است. ارتباطات و hubهای اصلی شبکه را back bone شبکه گویند. معمولاً wireless هستند بدیهی ترینشان از امواج رادیویی استفاده می‌کنند امواج رادیویی نوع خاصی از امواج است که تولید آنها ساده و ارزان است مسافت زیادی را پوشش می‌دهند از دیوارهای ساختمانها به راحتی رد می‌شوند و چند عیب دارند:

۱- امواج رادیویی را اگر کسی بخواهد استفاده کند، باید حتماً مجوز بگیرد چون هرکسی از هر فرکانسی خواست برای ارتباطش نمی‌تواند استفاده کند چون امواج روی هم می‌افتد خراب می‌شوند (امواج یکدیگر را خراب می‌کنند) به عبارتی دیگر مجوزگرفتن سخت است.

۲- خطای زیادی دارند از نظر noise مشکل دارند.

امواج ماکروویو: تبادل اطلاعات حجیم بین نقطه point to point مثلاً تلفن، امواج ماهواره، خطوط مخابراتی.

اگر بخواهیم شبکه‌ای در سازمانهای اجرا کنیم باید به این نکته توجه کنیم که سازمانها دارای این خاصیت هستند که مراکزی دارند که از هم جدا هستند یکی از معزلات این است که چه جوری سازمانهایی که از نظر فیزیکی از هم دور هستند به هم وصل کنیم. به عبارتی روشهای اتصال این نوع شبکه‌ها بحث ما است.

یک راه این است که خودمان کابل بکشیم یا اینکه دیش بگذاریم با ماهواره با هم ارتباط برقرار کنیم. همه امکانات در اختیار خودمان است. عیبی که این روش دارد این است که خیلی گران تمام می‌شود. اگر قرار باشد هرکسی اینکار را بکند هم دوباره کاری نمی‌شود و هم سازمانهای خیلی بزرگ می‌توانند هزینه‌ان را بدهند. اینجا بود که گفتند می‌توانیم در هر کشور آن را بر عهده مخابرات بگذاریم تا متولی برقراری ارتباطی شود که هرکس بخواهد به کسی وصل شود از امکانات شرکت مخابرات استفاده کند. این ایده از تلفن آمد چون مردم دیدند کابل کشیدن برای برقراری تماس مشکل است. همی ایده را خواستند توسعه دهند برای Data. یعنی یک شبکه اطلاعات در کشور راه بیندازیم. هرکس به این شبکه وصل شده به بقیه جاهایش هم وصل شود. همین امر باعث شد اگر کشورها به سمت شبکه Data بروند پروتکلی که استاندارد کاری بود X2s بود که خیلی کند بود و سرعتش 19200 بود. این پروتکل در سه لایه پایین را پوشش می‌داد.

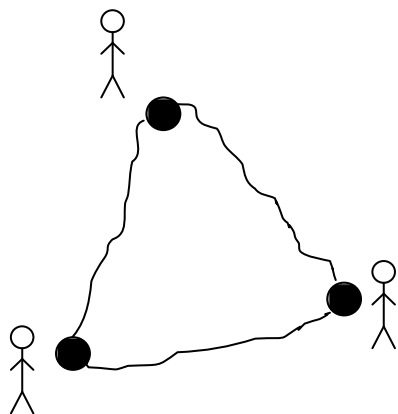
خطوطی که ما می‌توانیم بگیریم دو نوع است در کار ارتباط و اتصال.

۱-تلفن (psta)

۲-خطوط اختصاصی یا Leased Line

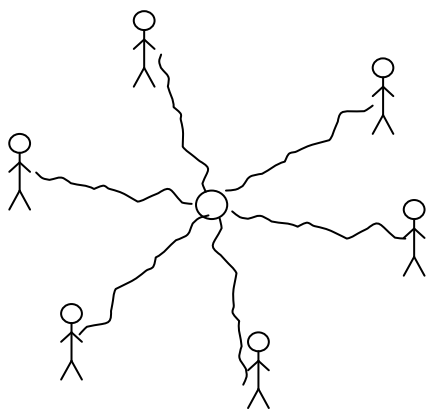
نکته‌ای که وجود دارد این است که اینها تا قبل از اینترنت بود اما الانه روشهای دیگری هم هست مثل EPN. توسط بل، تلفن کشف شد. و اینگونه بود که هرکس می‌خواست تلفن داشته باشد و با کسی صحبت کند باید یک خط به آن شخص می‌داشت و اگر می‌خواست با کس دیگری یا چندین نفر ارتباط داشته باشد، باید به همه آنها کابل می‌کشید.

راه‌حلی که به نظر می‌رسید استفاده از مرکزی بود که هرکس برای برقراری ارتباط باید به آن مرکز وصل می‌شد. مشکل اینجا این بود که باید سر این سیمها به هم وصل میشد که با هم ارتباط برقرار کنند.



یک راه حل این بود که یک OPERATOR آنها را به هم وصل می‌کرد و OPERATOR یک شخص بود. مدتی بعد سیستم operator از بین رفت و Switching جایگزین شد. مشکل روش Switching این بود که اگر شهر بزرگ بود کابل‌کشی به آنها مشکل بود. مشکل بعدی این بود که اگر زیاد بود (جمعیت شهر) باز هم

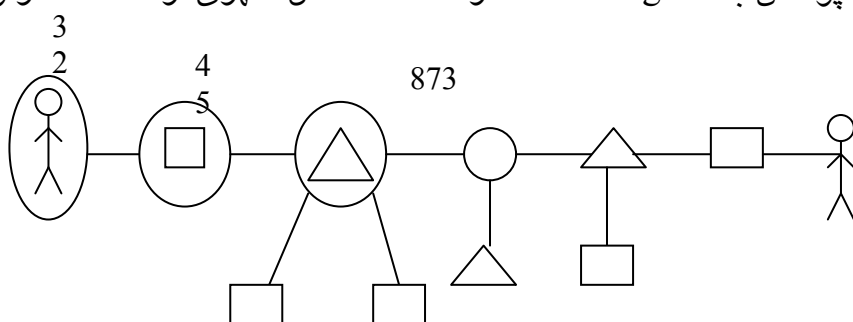
کابل‌کشی مشکل بود.



راه حل اینجا این بود که مراکز را زیاد کردند و فقط شرطش این بود که مراکز باید با هم در ارتباط باشند و خطوط بین مراکز حجیم‌تر و قوی‌تر بود. مشکل این راه این بود که مراکز شروع به پیشرفت از نظر تعداد کرد پس مشکل اولیه دوباره رو شد یعنی افزایش کابلها. پس آمدند، همان ایده قبلی را دوباره مطرح کردند یعنی مرکز سطح بالاتر در نظر گرفتند که همه به آن وصل شوند. گفتند یک مرکز در نظر می‌گیریم که همان مرکز محلی ما است و یک مرکز سطح بالاتر در نظر می‌گیریم. که همه ب آن وصل شوند و گفتند مراکز با هم مراکز با مشترک متوسط توسط Switch با هم در ارتباط باشند و همینطور این مراکز سطح بالاتر را پیش می‌گرفتند. تا آنجایی که مشترک را تحت پوشش بدهند. Switching در مکالمات داخل شهری از همه کمتر و

در خارج کشور از همه بیشتر است.

مثلاً تلفن ۸۷۳۴۵۳۲



خطوط تلفن باید امکان ارتباط همزمان چندین نفر را بدهد. در سه نقطه بحث دیجیتالی کردن را داریم. کابل بین ما و مرکز خطوط مراکز خود مراکز .

مزایای دیجیتال:

۱-علاوه بر صوتی که آنالوگ می فرستد قادر به ارسال تصویر، Data و صوت است . آنالوگ به درد صدا می خورد.

۲-علیرغم اینکه موجهای دیجیتال زودتر از آنالوگ خراب می شود یعنی Attenuation بیشتر است ولی به راحتی می شود محاسبه کرد که دیجیتال تا چه مسافتی اگر برود مشکلی برایش پیش نمی آید.

۳-نگهداری کم هزینه تر و بهتر است.

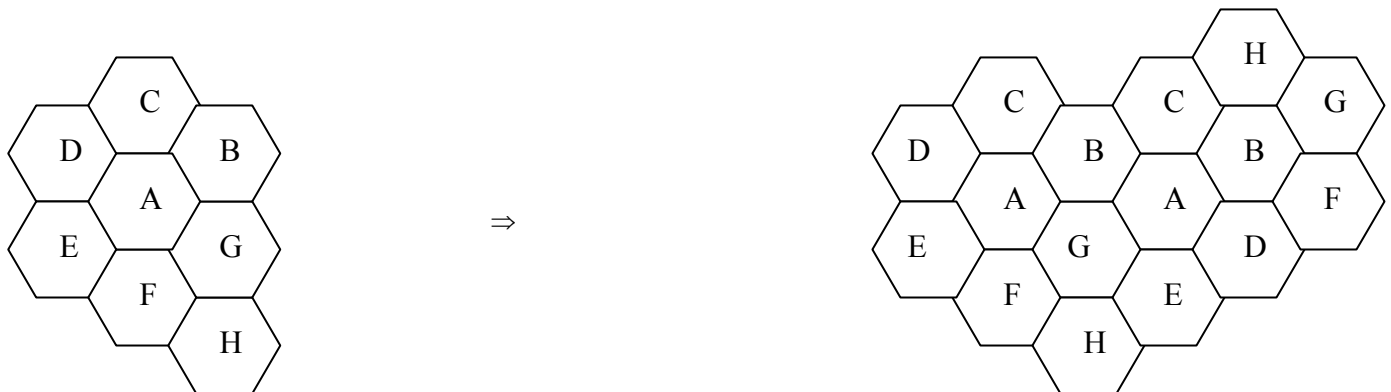
۴-ارزان تر هستند.

به همین دلایل خطوط به سمت دیجیتال شدن می روند. وقتی ما به شخصی زنگ می زنیم 3 Switching انجام می شود همین وضعیت برای Data هم هست.

اختصاصی هستند. دو شعبه داریم که می خواهیم دائماً به هم وصل باشند و تبادل اطلاعات کنند. در تلفن تماس که می گیریم باید از اولاً خط آزاد باشد. در اینجا ما اشغالی نداریم. در تلفن اگر تماس قطع شود تمام شده . خطوط اختصاصی همیشه در اختیار ما هستند این خطوط برای ما اینگونه است که ما به دستگاههای Switching وصل می شدیم. مادامی که آبونمان بدهیم، خطوط به هم وصل هستند. هزینه اش بیشتر است و خطوط اشغال نیست و بوق ندارد. اینجا ارتباط توسط مودم های خاصی برقرار می شود. این خطوط بسیار متنوع هستند.

ایده ای بود که قدیمی نیست. ۲۰-۳۰ سال عمر دارد. همگونی که Wile less، police را به هم وصل می کنم چرا کاری نکنیم که افراد عادی هم بتوانند از این ارتباطات استفاده کنند. یک گیرنده و فرستنده و دیش می خواست یک باند فرکانسی برای دیدن و یک باند فرکانسی برای شنیدن . همینطور برای طرف مقابلشان داشتیم یعنی نیاز به سیم نباشد. با فرکانسی ۸۳۰ کانال 4k هر تری بود که به هر شخص اختصاص می یافت. پس می توانست ۴۰۰ ارتباط برقرار شود . در هر زمان (چون هر شخص ۴۰۰ تا می خواست پس اگر دو نفر با

هم صحبت می کردند $800 = 2 \times 400$ می شد.) این مقدار برای یک شهر بزرگ مقدار کمی بود. به سال ۱۹۸۲ کمپانی ای که این ایده را مطرح کرده بود ایده ای بنام Amops مطرح کرد که گفت: رابه سلولهای تقسیم کنیم. آن سلولها در اقع گرد بودند. آن باند فرکانسی را بین نواحی تقسیم می کرد که به هر کدام ۱۰۰ تا می رسید. بین مناطق برای تماس خودشان هیچ تداخلی نداشتند. قدرت فرستنده ، گیرنده با فرکانسهای باند را جوری تنظیم کرد که دو تا خانه آنطرفتر امواجی از این سلول را دریافت نکند. یعنی دو تا خانه آنطرفتر ما بتوانیم از این امواج استفاده کنیم بدون تداخل یعنی مثلا A نباید با A تداخلی داشته باشد. ولی A با C. اگر در منطقه ای تماس زیاد بود مشکل ایجاد می شو چون ۱۰۰ کانال کافی نبود. راه حل این بود که سلول را به سلولهای کوچکی تقسیم کردند. به ازاء هر کدام تقسیم سلولی انجام بدهیم. منتها قدرت فرستنده را کم می کنیم یعنی امواج هر خانه ای (که دوباره تقسیم شده بودند) دوباره از خودش بیرون نرود.



:Base Station

یک گیرنده و فرستنده که محدوده خودش را مدیریت می کند. ۴ دسته کانال در موبالی است:

۱-کانال کنترل.

۲-کانال paging

۳-کانال Data

۴-کانال access

فقط ۲۱ تا از ۸۳۲ تا به کنترلی اختصاص دارد. کانالهای کنترلی بسته به اینکه تماس از کجاست و کجا کشور است، تشخیص می دهد بند فرکانس کنترلی کدام است و تشخیص می دهد از آن ۲۱ تا کدام قویتر است. کانال

paging کانالی است که وقتی تماس گرفته می شود تشخیص می دهد که تماس با ما از کجاست و اینکار از طریق ایستگاه انجام می دهد. کانال access وقتی تماس می گیریم با Setup اولیه سروکار دارد. کانال Data، کانالی است که می توانیم از طریق آن Data رد و بدل کنیم.

وقتی دستگاه روشن می شود ، می گردد دنبال کانال کنترلی وقتی پیدا کرد اطلاعات مربوط به شماره تلفن و یک کد خاص مربوط به آن موبایل خاص را که حدود ۳۲ بیت است را می فرستد به مرکز که این کار یعنی می خواهیم وارد شو. Data base اطلاعات مربوط به مشترک جدید را به مرکز بزرگتری به نام MTSLE وصل می شود که Data Base بزرگ است که رکود شخص وارد شده را وادار می حکند اگر شخص خواست کسی را بگیرد، اولاً شماره شخص شوم را می گیرد سپس شماره روی از روی access به Base station می فرستند و می گوید این شخص کجاست. و اگر در گیرنده باند فرکانسی خالی باشند به شخص مخاطب از طریق paging اطلاع می دهد و بعد از اینکه شخص گوشی را برداشت ارتباط اتصال می یابد.

حرکا موبایلها:

فرض کنید با شخصی تماس گرفتید و در حال حرکت صحبت می کنید. اگر مرکز عوض شود این است که مرکز اولیه حس می کند داریم از مرکز دور می شویم از امواج ب سه سلول که در آن مجاورت است پیغام می دهد کدام امواج مشترک من به آن نزدیکتر است. اولی کانال را از می گیرد و دومی مشترک را Switch می کند به کانال جدید.

موبایل وسیله شخصی است و مشکلش این است که چون Wireless است ارتباط ما پخش می شو که شخص می تواند اگر بخواهد آنتنی وصل کند و تماس مارا بگیرد. و همچنین می تواند pincode موبایل ما را بگیرد و از طریق شماره ما تماس بگیرد و استفاده کند. راه حلش این است که کد کنیم. در اینصورت با جنایتکاران دچار مشکل می شویم چون طرف می تواند جایش را عوض کند.

برای تبادل اطلاعات ۵۰-۶۰ سال پیش از بالن استفاده کردند و این به دلیل انحنای زمین بود که در روی آن بالن منعکس می کرد. ایده دیگر این بود که از ماه استفاده کنند. ایده دیگر استفاده از ماده مصنوعی بود. یعنی ماهواره: یعنی اگر چون ماهواره ها در مدار 36000k زمین بچرخد. حرکتش با حرکت زمین منطبق می شود.

در هر هم یک ماهواره می‌توانیم داشته باشیم پس کلاً می‌شود ۱۸۰ ماهواره پس برای این اساس سهم هر کشور را سهمیه‌بندی کردند. فرستادن از کانال به ماهواره گران است چون باید کانال داشت باشیم. همه با ریز وصل می‌شوند و به مرکز و کار اصلی را مرکز انجام می‌دهد.

عیبی که ماهواره دارد تأخیر زیاد است که 270ms می‌شود و 270ms هم برمی‌گردد و در نتیجه روی هم 0s ثانیه می‌شود. امواج ماهواره میکروویو است. از نظر حجم تبادل اطلاعات فیبر بیشتر از ماهواره است نصب ماهواره آسان است. ماهواره وابسته به مسافت نیست از نظر هزینه ماهواره برای مسافت دور به صرفه است. ماهواره امکان اضافی می‌دهد به کسانی که می‌خواهند از محدودیت مخابرات استفاده نکنند. و می‌توانند سیستمهای مخابراتی کشورها را مختل کند. فیبر تحت تأثیر هیچ چیزی قرار نمی‌گیرد ولی بعضی چیزها روی ماهواره تأثیر می‌گذارند.

تلفن ماهواره ای

شرکت motorola پروژه‌ای مطرح کرد که ۷۷ ماهواره را در کره می‌چید و وقتی زمین می‌چرخید هر کدام از ماهواره‌ها که از دید ایستگاه زمین خارج میشد ماهواره دیگری را جای آن جایگزین می‌کرد. و محدودیت ۳۶۰۰۰ از بین می‌رفت. با بررسی‌ای دیگر این ۷۷ تا به ۶۶ ماهواره رسید یعنی ۷۷ ماهواره روی مدارات مختلف زمین بود.

از وسایل لایه فیزیکی می‌توان repeater یا hub را نام برد. کارهای لایه فیزیکی غیرهوشمند است. به این دلیل می‌گوئیم لایه فیزیکی غیرهوشمند است که چون داده‌ای را که بهش می‌دهیم تمام خروجی هایش می‌فرستد ولی hub Switch تشخیص می‌دهد به کجا بفرستد که مقصد لایه ۲ است. وسیله دیگر modem است. لایه ارتباط داده‌ها:

مهمترین لایه در لایه‌بندی شبکه است. Datalink ما بین physical, Network است. وظیفه اصلی این لایه این است که داده‌ای که از طرف Network می‌گیرد را ارسال کند به گیرنده. نکته در اینجا این است که point to point است یعنی لایه گیرنده و فرستنده حتماً باید هر دو و فقط و فقط آن دو روی یک خط باشند. این لایه ۴- ۳ وظیفه عمده دارد:

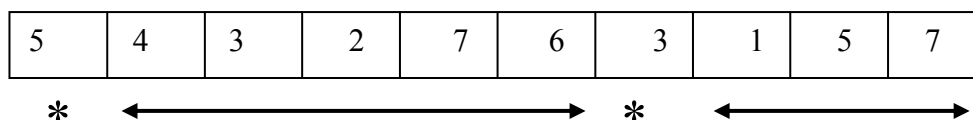
۱- فرم بندی یا Framing ۲- Error control یا کاهش خطا ۳- Flow control

۱- فرم بندی یا framing:

یعنی داده‌ای که Network می‌دهد و می‌گوید ارسال کند را با توجه به شناختی که از نقطه پایین دارد آن را کوچک کند و بفرستد. کار دیگر این است که آن را بسته‌بندی کند و ارسال کند. طرف گیرنده هم وقتی داده‌ها را گرفت، بسته‌ها را می‌چسباند به هم و برای صخقن می‌فرستد. بسته‌بندی ابتدا و انتهایش را مشخص می‌کند. فرم‌بندی یعنی بستن اول تا آخر پاکت تکه‌تکه کردن آن. بستن اول تا آخر کار ساده‌ای نیست. چون اطلاعات دنباله‌ای از ۱۰ و ۱ است نکته این است که برای 08,0- برای ۱، می‌گذارد تا قاطی نشود مشکل دیگر بستن این داده ۱۰ و ۱ است، چیزی که درست است نباشد.

۱- روش شماره کاراکترها:

هنگام فرستادن همان بیت اول را تعداد بایتهای اطلاعات بگذاریم. مشکل این است که اگر یک noise بیفتد همه اطلاعات خراب می‌شود.



۲- روش کاراکترهای ابتدایی و انتهایی:

کاراکترهای خاصی را اول داده‌ها می‌گذارد مثل STX و آخرشان هم همینطور. کاراکتر ۳۲-۰ اسکی کنترلی هستند که در داده‌ها همچنین کاراکترهایی نداریم اما مشکل این است که ممکن است در وسط اطلاعات STX خودش ظاهر شود.

STX	0 0 0 STX 0 0 0	STX
-----	-----------------	-----

برای حل این مشکل از روش زیر استفاده می‌شود.

۳- روش کاراکتر stuffing:

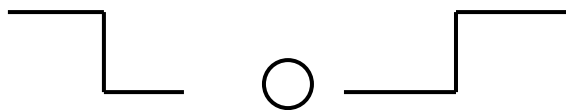
اگر در کاراکترها جایی STX بود پشت سرش یک STX دیگر اضافه می‌کند. فرقی این است که STX در آخر یکی است در وسط ۲ تا . در طرف دیگر یعنی گیرنده یک STX را حذف می‌کند.

۴-الگوی بیتی ابتدائی و انتهایی:

اول تا آخر داده الگوی خاصی قرار می‌دهیم. مثلاً 0111111 هم در آخر و هم در اول که مشکل بالا هم هست برای حل این مشکل روش Stuffing هم هست. بیت اضافه می‌کند یعنی هر جا فرستنده S تا ۱ ظاهر شد پشت سرش خودش یک ۰ اضافه کند تا با ابتدا و انتها اشتباه نشود. مشکل اینطوری حل می‌شود که اگر در طرف گیرنده S تا ۱ را دید بعدی‌اش را (بیت بعد) حذف کند.

۵-استفاده از کدهای غیرمجاز:

دو بیت فیزیکی روی خط یک بیت منطقی را نشان می‌دهد. چون اگر یکی بود به noise حساس می‌شود در نتیجه نسبتشان خیلی فرق نمی‌کند.



با توجه با وضعیتی که کدها دارند، ۴ وضعیت داریم که دو تاش برای کدینگ استفاده شده و از دو تای بقیه بستن استفاده شود از مدهای غیرمجاز برای بستن اول و آخر فرم استفاده می‌شود.

امروزه روش اول جلد و یکی و سه روش دوم استفاده می‌شود اگر درست بود که هیچ واگرانه می‌فهمند خطایی اتفاق افتاده.

برای ابتدا و انتهای فرم می‌توان از حالت‌های ترکیبی دیگر ولتاژ بالا و پائین استفاده کرد. و بقیه پیچها را از ابتدا و انتها جدا کرد. از کد × هیچ موقع در مورد بیت‌های وسط کاربرد ندارد بلکه شکل‌های کاربرد دارد.



error control (کنترل خطا):

وظیفه‌اش کنترل و پایبیشن و نظارت بر خطاست. یعنی اینکه در واقع این وظیفه خاص راجع به این است که تضمین اینکه داده‌ای که در شبکه ما منتقل شده و از یک نقطه به نقطه دیگری می‌رود خطا در آن پیش نیای اطلاعات یکسری ولتاژ است که تفاوت آنها زیاد است تحت تأثیر خیلی عوامل قرار می‌گیرد: رادیو- امواج- کابل و وظیفه لایه DataLink است که به ازاء داده‌هایی که می‌فرستد تضمین می‌کند که خطایی رخ نداده . error control در دو سطح مطرح می‌شود. یعنی در دو سطح خطا پیش می‌آید.

۱- سطح frame

مثل این است که فرمی بفرستیم کل فرم کم شود یا دچار مشکل شود. یا فرمی را بفرستیم برای اطمینان از رسیدن آن رسید بخواهیم رسید اصلاً کم شود.

۲- سطح بیتی

فرم به مقصد برسد و داده‌های فرم خراب بشود که به این سطح ، سطح بیتی گویند.

راه حل سطح اول: فرض کنید یک frame فرستادیم رمز رسده. طرف فرستنده ساعت داریم به نام ساعت رسید که وقتی مری را فرستادیم آن ساعت شروع به کار می‌کند و آن ساعت را ساعت‌بندی می‌کنیم. ساعت شروع به کار می‌ند. اگر فرم برشد طرف دیگر یک acknowledge آماد می‌کند که به عنوان رسید آن frame است. اگر frame ترسید ساعت enspire یا منقضی می‌شود و می‌فهمیم نیامد و دوباره می‌فرستیم وای نکته ریز اینجا این است که این برای وقتی که رسیدیم بشود درست کار کند یا نه برای زمانی که frame کم بشود فقط درست کار می‌کند. اگر دوباره بفرستد فرم قبلی دوباره فرستاده می‌شود و این مشکل است چون طرف گیرنده دو فرم یکسان گرفته و فکر کرده پشت سرهم هستند . در نتیجه داده ها خراب می‌شود. راه حل این است که به فرم‌ها شماره بدهیم و یا باید کاری کنیم که فرم‌هایی که دریافت می‌کند چون گیرنده شماره الان را داشته می‌فهمد که Acknoweldge گم‌شده و در نتیجه فرمی که دوباره فرستاده شده را حذف می‌کند.

برای سطح دوم مشکل سخت‌تر است و به دو شکل وجود دارد.

۱- خطای تکی

۲- خطای دسته‌ای (Burst)

(Burst) یعنی دنباله‌ای از بیتها پشت سرهم عوض شوند. معمولاً خطاهای شبکه دسته‌ای رخ می‌دهد. اینجا دیگر شخص گیرنده خیلی دچار مشکل می‌شود چون در هر دو طرف مجموعه‌ای از 0,1 ها را دارد.

Error control در سطح بیتی به دو بخش تقسیم می‌شود.

۱- تشخیص خطا error detector

۲- رفع خطا error recovery

بخش ۱ راحت‌تر از ۲ است چون در ۱ وقتی شخص تشخیص داد خطا دارد یا از آن استفاده می‌کند یا نه فاز بعد پیاده‌سازی است یعنی می‌شود تصحیحش کرد یا نه بگوئیم فرستنده دوباره بفرستد.

بیت ۲۵۶ حالت را می‌تواند کد کند و به هر کدام از ترکیبات این بیتها یک کارکتر خاص نسبت داده مثل کدهای اسکی مثلاً 256 به F حال اگر فرستنده abcd را بفرستد و در طرف گیرنده axcd برسد از طرف گیرنده با مشکل شناخته خواهد شد.

هر کد به یک کد مجاز تبدیل شده . یک جور فضای کدینگ را زیاد کنیم که هم فضا توسط کارکترها پر نشود، بنابراین یکسری از حالتها مجاز هستند و یکسری غیرمجاز. پس زمانی که خطا اتفاق افتاد طرف دیگر ممکن است که غیرمجاز را دریافت کند پس متوجه خطا می‌شود یکی از این روشها parity است. Parity تنها یک بیت خطا را تشخیص می‌دهد که اگر خطا در دو بیت اتفاق بیفتد دیگر متوجه نمی‌شود.

کدینگ زیر را در نظر بگیرید:

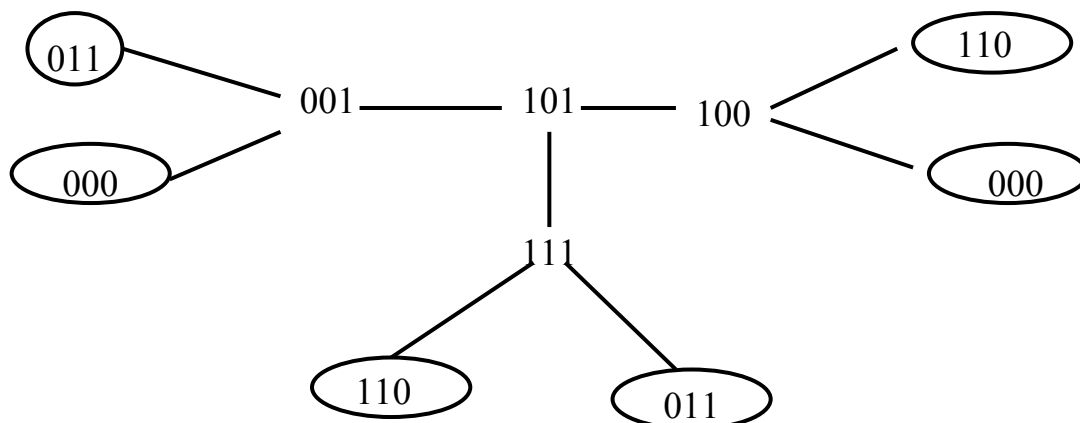
A 1111100000
B 0000011111
C 1111111111
D 0000000000



اگر خطا به جای اینکه در یک بیت اتفاق بیفتد در ۲ بیت اتفاق بیفتد دیگر ممکن است خطا قابل تشخیص نباشد یعنی A اگر یکبار انتقال یابد به  می‌رود و اگر دوبار انتقال دهد به  ی‌رود. حال با توجه به اینکه انتقال به فضای غیرقابل مجاز رفته قابل تشخیص بود ولی با دوبار انتقال به فضای مجاز رفته و قابل

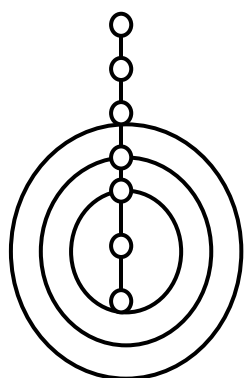
تشخیص نمی‌باشد. در شکل پائین اگر کدینگ‌های داخل دایره را انتخاب کنیم همه اینها درست با کد اصلی اختلاف دارند. اگر کدینگ جوری باشد که کدهای داخل دایره مجاز باشند خط غیرقابل تشخیص است اما قابل

تشخیص است.



فاصله همینگ: حداقل تعداد بیتی که در یک مجموعه کدگذاری باید عوض شود تا یک کد مجاز دیگر تبدیل شود را فاصله همینگ گویند. یعنی در یک کدگذاری خاص حداقل بیتی که باید عوض شود تا یک بیت مجاز به یک بیت مجاز دیگر عوض شود آن تعداد بیتیها را فاصله همینگ گویند. کد اسکی فاصله‌اش ۱ است و parity، ۲ است. فاصله همینگ کدهای ۱۰ بیتی باید S است یعنی هر بیت فاصله‌اش تا بیت‌های قبلی S بیت است یعنی اگر تا ۴ خطا رخ دهد می‌شود آن را فهمید. توجه به نحوه کدگذاری D,C,B,A می‌شود فهمید. اگر برای ۱۰ کد بالا فرستاده D را فرستاده باشد و 1100000000 ره طرف گیرنده رسیده باشد، او می‌تواند آن را راحت به کد

مجاز شبیه‌اش تبدیل کرده باشد.



$$\Rightarrow \left\lceil \frac{d-1}{2} \right\rceil =$$

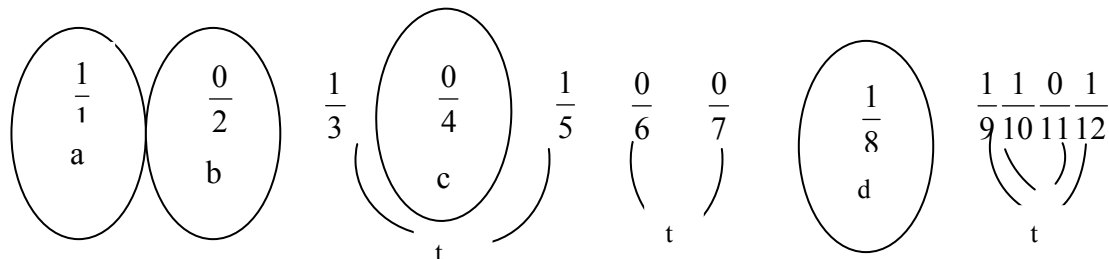
تعداد بیت تصحیح شونده

$\Rightarrow$

شعاع = تعداد بیت‌های خطا

داده‌های رسایی ما را به نحوی کدگذاری می‌کند که وقتی طرف گیرنده دریافت شد اگر یک بیت خطا اتفاق افتاد هم تشخیص می‌دهد و هم تصحیح می‌خواهیم A:11001101 را ارسال کنیم. فرض کنیم -----ها مقادیر ارسالی طرف گیرنده باشد. -----ها را شماره‌گذاری می‌کنیم موقعیت‌هایی که توان صحیحی از ۲

هستند را برای چک کردن بیتها استفاده کنیم. سپس داده‌ها را روی جاهای خالی می‌چینیم. به ازاء هر کدام از محل‌هایی که بیت‌های داده روی آن است را به صورت توانی از ۲ بفرستیم.



مجموع توان‌های در هم همان چک بیتها بودند. هر کدام از بیت‌های داده‌ای به موفقیت به وسله آن بیت‌هایی از چک بیتها که در آن ظاهر شده‌اند یک می‌شود. به اینصورت که هر کدام از این چک بیتها را بررسی می‌کنیم کدام‌ها را پوشش می‌دهد. مثلاً $a(3,5,7,9,11)$, $c(5,6,7,12)$, $b(3,6,7,10,11)$, $d(9,10,11,12)$. چکینگی که مار در اینجا لحاظ می‌کنیم. Parity است یعنی a,b,c,d باید از بیت‌هایی که پوشش می‌دهد parity زوج را عمل کند. مثلاً در مور a ، ۳ بیت ۱ دارد و b ، پرییتی‌اش ۰ است و c هم ۰ است.

اشکالاتی که در این روش وجود دارد عبارتند از:

۱- داده A ، ۸ بیت بود ولی هنگام انتقال به ۱۲ بیت تبدیل شد.

۲- به داده‌ای که اضافه شود به داده ما و خودش حضور داده نیست و برای کنترل و تشخیص خطا را انجام دهد Check sam می‌گویند.

اگر یک بیت از داده‌ها عوض شود مثلاً بیت ۷ آن به a,b,c به هم می‌خورد ولی d عوض نمی‌شود.

$$3 = 2^1 + 2^0 = a + b$$

$$5 = 2^2 + 2^0 = c + a \Rightarrow a(3,5,7,9,11)$$

$$6 = 2^2 + 2^1 = c + b \quad b(3,6,7,10,11)$$

$$7 = 2^2 + 2^1 + 2^0 = c + b + a$$

$$9 = 2^3 + 2^0 = d + a \quad c(5,6,7,12)$$

$$10 = 2^3 + 2 = d + b \quad d(9,10,11,12)$$

$$11 = 2^3 + 2^1 + 2^0 = d + b + a$$

$$12 = 2^3 + 2^2 = d + c$$

$$\begin{array}{cccccccccccc} 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ \hline 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 \end{array}$$

$$\begin{array}{ccccccc} a & b & c & \uparrow & d & & \end{array}$$

چون تعداد parity ها با بیت‌هایی که تحت پوشش دارند مطابقت دارد پس a,c غلط و d,b درست است چون مشترک C,a, 5,7 است اما مشخص شد که ۷ مشکلی ندارد پس ۵ است.

CYCLIC REDUNDANCY CHECK CRC

یکی از معروفترین کدهای تشخیص خط CRC است. امروزه دنبال خطا به وسیله بیت‌های خط نیستند امروزه ما معمولاً اول خط را تشخیص می‌دهیم سپس به فرستنده می‌گوییم بفرست. یکی از بهترین روشها برای تشخیص خط CRC است وقتی بخواهیم دنباله‌ای از داده‌ها بفرستیم مثل B فرض می‌کند بیت‌ها هر کدام مضرب توانی از X هستند که توان X به وسیله‌های آنها دو دسته مشخص می‌شود.

$$B: \frac{1}{X^{10}} \frac{1}{X^9} \frac{1}{X^8} \frac{1}{X^7} \frac{0}{X^6} \frac{1}{X^5} \frac{1}{X^4} \frac{1}{X^3} \frac{0}{X^2} \frac{1}{X} \frac{0}{X^0}$$

$$\Rightarrow P(X) = X^{10} + X^8 + X^7 + X^5 + X^4 + X^3 + X^1$$

طرف گیرنده و فرستنده روی چند جمله‌ای واحدی به نام GUO توافق کنند.

$$\Rightarrow C(X) = X^4 + X + 1 \quad 10011$$

طرف فرستنده وقتی خواست آنها را بفرستد، ابتدا داده را به چند جمله‌ای p(x) تبدیل و به Generator تقسیم می‌کند پس یک خارج قسمت و یک باقیمانده داریم. سپس داده ارسالی مثل p(x) را از باقیمانده کم می‌کند.

$$\begin{array}{r|l} A & B \\ \hline D & C \end{array} \quad \Rightarrow P(X) - R(N) = Q(X)$$

p(x) بر G(x) بخشپذیر است و باقیمانده صفر است.

$$\begin{array}{r|l} & 3 \\ \hline 24 & 8 \\ \hline & 1 \end{array}$$

حال $Q(x)$ برای فرستنده طرف دیگر $Q(x)$ را دریافت و دوباره بر $G(x)$ تقسیم می‌کند. اگر باقیمانده صفر بود می‌گوید به احتمال زیاد صفر است. و در غیراینصورت یعنی اصماند خطا رخ داده است.

در صورتی خطا قابل کشف نیست که خطا یکجوری بیتها را عوض کند که چند جمله‌ای ساخته بشود که آن بر $G(x)$ قابل تقسیم باشد.

راه حل این است که $p(x)$ را در بالاترین توان x در GUO ضرب می‌کند.

$$\Rightarrow P(X) = X^{14} + X^{12} + X^{11} + X^9 + X^7 + X^5$$

می‌خواهیم دنباله‌ای از بیتها را بفرستیم اگر کدینگ ارسال کنیم همراه این بیتها بسته به این که خطا به این دسته‌ها فقط در یک بیت باشد می‌تواند تشخیص بدهد.

101011011011

011000100110

001010101001

111100011101

010101101110

010101101110

011011101101

001111101101

خطای Borst یا دسته‌ای:

داده‌ها را به شکل ماتریس کنار هم کی چینند و به جای اینکه روی سطرها اجرا کنند روی ستون اجرا می‌کنند. سپس به همان روش قبلی سطرها را می‌فرستند. اگر هم بیتها را عین ماتریس می‌چینند خطا را تشخیص می‌دهد.

خطای دسته‌ای اگر اتفاق بیفتد معمولاً سطر است یعنی هرکدام مربوط به یک ستون خاص هستند یعنی هرکدام از خطاها روی یک ستون می‌افتد. و در نتیجه همانطور که برای سطر قابل تشخیص بود یک خطا اینجا هم قابل تشخیص است. اگر مقدار خطا به اندازه طول یک سطر باشد قابل تشخیص است و اگر نه قابل تشخیص نیست.

ممکن است خطایی رخ دهد و مشخص نشود مثلاً $Q(n)$ را فرستادیم و در آن سمت که رفت چند تا ۰ و ۱ عوض شد. $0 \leftarrow 1$ یعنی یک ضریب از x عوض ستود می‌آید و ۱ اگر $0 \leftarrow 1$ تبدیل شده بود. را داریم و یک جمله‌ای به وجود می‌آید که می‌ستود. $Q(x)$ جمع ستود. ضریب ایجاد شده $E(x)$ است. یعنی آن چیزی دریافتی $Q(x)+E(x)$ است. اگر خطا رخ ندهد باقیمانده بر $G(x)$ صفر خواهد بود. داده‌ای که آنور رفته خودش نیست چون از $Q(x)$ آنهای $R(x)$ شده. وقتی می‌خواهند عمل روی $P(x)$ را انجام دهند آن را در بالاترین توان $G(x)$ ضرب می‌کنند.

$$\Rightarrow P(X) = 11010011010000, G(X) = X^4 + X + 1 = 10011$$

$ \begin{array}{r} 11010011010000 \\ \underline{10011} \quad \swarrow \\ 010010 \\ \underline{10011} \\ 000011101 \\ \underline{-10011} \\ 0 \\ 0 \\ \underline{0} \\ 00001 \end{array} $ باقیمانده	$ \begin{array}{r} 10011 \\ \hline 11000110111 \\ \Rightarrow 11010011010000 \\ \hline 1- \\ 11010011010001 \\ \searrow \end{array} $ اگر باقیمانده صفر بود خطا رخ نداده
---	---

CRC هرچه خطای تک بیتی باشد تشخیص می‌دهد. همینطور برای ۲ بیتی. اگر خطا رخ دهد $E(X)$ به وجود

می‌آید باقیمانده $Q(x)$ خواهد بود ولی باقیمانده $\frac{E(X)}{G(N)}$ معلوم نیست صفر بشود. یعنی خطا رخ نداده.

$$\frac{Q(X)+E(N)}{G(X)} = \frac{E(X)}{G(X)}$$

$$x^n + x^{n-1} + 000 + x^1 + x^0$$

فرض کنید دنباله‌ای از بیتها دارد ارسال می‌شود به طول n :

حال $G(x)$ را چطور انتخاب کنیم که اگر خطا رخ داد و در عین حال $\frac{E(X)}{G(N)}$ باقیمانده‌اش صفر شد بتوانیم خطا را تشخیص دهیم. چکار کنیم هیچوقت X^1 بر $G(X)$ قابل تقسیم نشود. کافی است $G(x)$ در جمله باشد، هیچوقت خطای تک بیتی ای نیست که کشف نشود پس باید حداقل دو جمله باشد.

خطای دوبیتی:

Check sum یعنی برای انجام عملیات خودمان داده‌ای را اضافه کنیم) برای این منظور داده را در بالاترین دتبه $G(x)$ ضرب می‌کنیم حال می‌خواهیم نمونه‌ای از آن را ببینیم.

$$11010011010000 \quad P(X) = X^9 + X^8 + X^6 + X^3 + X^2 + 1 \quad (\text{مثال})$$

$$10011 \quad 6(X) = X^4 + X + 1$$

$$11010011010000 \quad \begin{array}{|l} 10011 \end{array} \quad \div: \text{exclusiveor}$$

$$\begin{array}{r} 10011 \\ \hline 010010 \end{array} \quad \begin{array}{r} 1100011110 \\ 11000110111 \end{array}$$

$$\begin{array}{r} 100110000 \\ \hline 0000011101 \end{array}$$

$$\begin{array}{r} 100110 \\ \hline 011100 \end{array}$$

$$\begin{array}{r} 10011 \\ \hline 011110 \end{array}$$

$$\begin{array}{r} 10011 \\ \hline 011010 \end{array}$$

$$\begin{array}{r} 11001 \\ \hline 000110 \end{array}$$

$$Q(X) = P(X) - R(X) = 11010011010001$$

$$\frac{Q(X) + X^9}{6(X)} = \frac{Q(X)}{6(X)} + \frac{X^1}{6(X)}$$

× اگر $G(x)$ را در دو جمله‌ای انتخاب کنیم، حداقل باید خطاها در دو بیت باشد تا قابل تشخیص باشد و هیچ خطای تک‌بیتی نیست که کشف نشود.

حال اگر بخواهیم که دو جمله را تشخیص دهد، مطابق فرمول زیر باید $G(x)$ را طوری انتخاب کنیم که بر هیچ

$$\frac{Q(X) + X^I + X^J}{6(X)} = \frac{X^J (X^{I+J} + 1)}{G(X)} \text{ frame قابل تقسیم نباشد.}$$

ثابت شده است که اگر $G(x)$ را مضرب $(x+1)$ انتخاب کنیم، در آن صورت تمام خطاهای با طول فرد از بیت‌ها قابل تشخیص است. (یعنی ۱بیت، ۳بیت، ۵بیت و ...)

اگر توان $n, G(x)$ باشد در آن صورت تمام خطاهای دسته‌ای با طول $(n-1)$ قابل تشخیص است.

$$\frac{X^i + X^{i-1} + X^{i-2} + \dots + X^{j+1} + X^j}{X^j (X^{i-j-1} + X^{i-j-2} + \dots + X + 1)}$$

چرا که وقتی طول آن کمتر باشد در واقع مثل آن است که بگوییم X^2 بر X^3 بخشپذیر است یا نه جواب منفی است.

CRC(12) معمولاً به در ماشین‌های ۷بیتی می‌خورد.

$$\text{CRE12:} \quad 6(X)X^{12} + X^{11} + X^3 + X^2 + X + 1 \quad \longrightarrow \quad 7\text{بیتی}$$

$$\text{CRC16:} \quad 6(X) = X^{16} + X^{15}X^2 + 1 \quad \} \quad 8\text{بیتی}$$

$$\text{CCITT:} \quad 6(X) = X^{16} + X^{12} + X^5 + 1$$

$$\text{CRC32:} \quad 6(X) = X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + \quad 32\text{بیتی}$$

$$X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$$

در واقع CRC(12) دوازده بیت به داده اضافه می‌کند و crc32 که امروزه بیشتر از آن استفاده می‌شود ۳۲بیت به داده اضافه می‌کند.

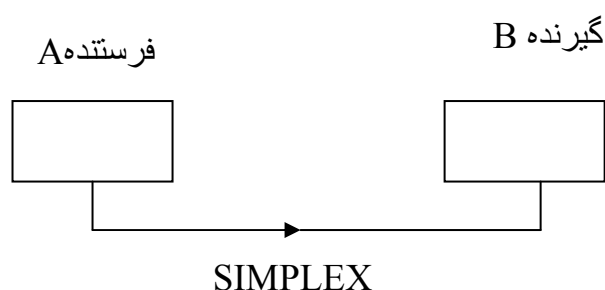
CRC12، خطای ۱۱بیتی و CRC32 خطای ۳۱بیتی را تشخیص می‌دهد.

ممکن است گفت شود CRC16 خطایی ۱۷ بیتی را تشخیص می‌دهد ولی معمولاً پروتکل شبکه به گونه‌ای است که این اتفاق نمی‌افتد.

پروتکل‌های مهم لایه Data link:

HDLC, SLP, PPP

Flow control: محافظت یک گیرنده ضعیف در مقابل یک فرستنده قوی. برخلاف ظاهر آن، اصلاً ساده نیست. باید دید که چکارکنیم دو فرستنده و گیرنده که از هم دور هستند سرعت همدیگر را رعایت کند. تا پروتکل وجود دارد.



فرض: ارتباط ساده، یکطرفه، خطا وجود ندارد و زمان ارسال کم و بدون معطلی است و با فرگیرنده نامحدود ← هرچه فرستنده، بفرستد گیرنده می‌گیرد و محافظت معنا ندارد.

اگر بخواهیم این دنباله از داده را بفرستیم کارهایی که می‌خواهد فقط framing است و نیاز به error control و داده‌های کنترلی ندارد.



۲-فرض: این بار بافر گیرنده محدودیت دارد. از آن جایی که فرستنده از گیرنده خبر ندارد. اولین کار توافق است که روی یک سرعت توافق کنند هر دو با هم بتوانند کار کنند ← Connection oriented در هر حال مناسب نیست چرا که ممکن است توافق کنند ولی گیرنده زمانی سرش شلوغ باشد یا سرش خلوت باشد و وقتش تلف شود. ممکن است توافق روی زمان به راحتی امکانپذیر نباشد.

۳-از یک Acknowledge استفاده کنیم، فرستنده داده را به گیرنده می‌فرستد و صبر می‌کند تا ACK از گیرنده بیاید که این جا به معنای error نیست و یعنی آماده است، دوباره اطلاعات را می‌فرستد به این روش پروتکل Stop&Wait گویند.

باید ACK را از داده جدا کنیم ← از یک فیلد Kiod استفاده می‌کنیم که نوع آن را معلوم می‌کند.

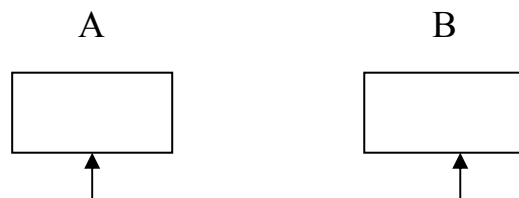


4 - فرض: خطا هم دارد. در سطح frame، علاوه بر flow control به error control هم نیاز است. وقتی داده را ارسال کرد یک ساعت را فعال می‌کند که کم می‌شود در آن محدوده باید گیرنده داده را ارسال کند که یعنی frame بعدی را بفرستد یا دوباره frame قبلی را ارسال کند.

۱-frame گم شده

۲-سیگنال ACK گم شده ← داده تکراری ← شماره می‌خواهد.

CS: Check sum برای خطایابی در داخل خود frame است.



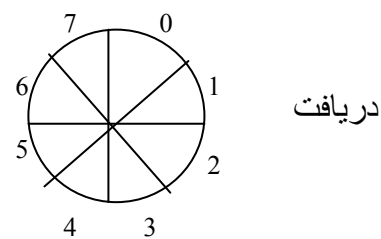
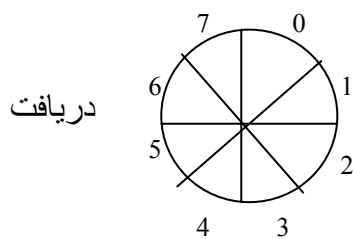
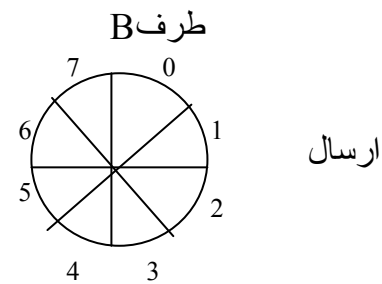
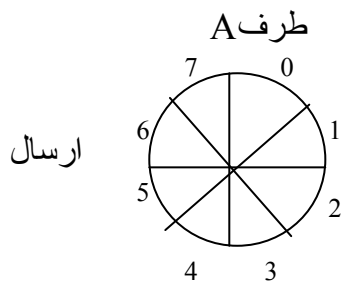
می‌خواهیم دوطرفه کار کنیم:

iggy backing: B داده را که دریافت کرد صبر کند ACK آنرا با داده برای A بفرستد. مشکل دارد چرا که ممکن است داده‌ای برای ارسال به A نداشته باشد.

برای رفع این مشکل B هریک ساعت دارد که اگر در زمانبندی آن داده داشت آن را با ACK به A می‌فرستد ولی اگر منقضی شده و داده نداشت خود ACK را به A می‌فرستد.

آنچه معلوم است این است که زمان ساعت B کوچکتر از A است تا زمان آن منقضی شود و داده دوباره ارسال نشود.

پنجره‌های غلطان یا Sliding Windows: یکی از پروتکل‌های معروفی است که امروزه در Windows هم مورد استفاده است. تبادل اطلاعات دوطرفه، با خطا و بافر محدود است. در واقع شرایط واقعی است چون دو طرفه است. هرطرف یک تعداد بافر دارند که به آن‌ها پنجره غلطان گویند که دو دسته است برای ارسال و دریافت.



ارسال: نشان دهندو حاوی frame هایی که ارسال شده اند و منتظر ACK آنها هستیم ، هستند.

دریافت: نشان دهنده frame هایی است که منتظر دریافت آنها هستیم.

می خواهیم ببینیم که چگونه کار می کند.

لحظه اول: A کاری نکرده و B منتظر دریافت B است. در واقع قبل از ارسال اطلاعات است.

لحظه دوم: A، 0 را ارسال کرده ولی هنوز به B نرسیده است.

لحظه سوم: B دریافت کرده پس می غلند روی ۱ ولی هنوز ACK به A نرسیده .

لحظه چهارم: ACK به A می رسد پس 0 خالی می شود و B هم ۱ پراست که frame ۱ است و بقیه خالی

است.

اگر داده به B نرسد با CK از B به A نرسد (یعنی لحظات دوم و سوم) زمان آن منقضی می شود در نتیجه A روی frame صفر است آن را دوباره می فرستد.

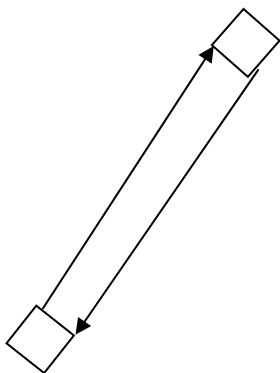
حالا می خواهیم زمان را هم از روی کار برداریم یعنی تاثیر نباشد.

در روش اخیر مشکل این است که به دلیل انتظار برای رسیدن ACK و ارسال دومی زمان زیادی تلف می شود. اگر فرض کنیم که زمان ارسال 20ms باش طول مسیر 540ms باشد 520ms وقت تلف می شود.

← 26 تا frame را با هم ارسال می کنند تا خط گران ما خیلی تلف نشود. این ارسال به گونه ای است که

آخرین frame که ارسال شد اولین ACK دریافت شود. در غیر این صورت ما فقط $\frac{20}{540} = 4\%$ کل خط را

استفاده می کنیم.



به دو طریق این کار انجام می شود:

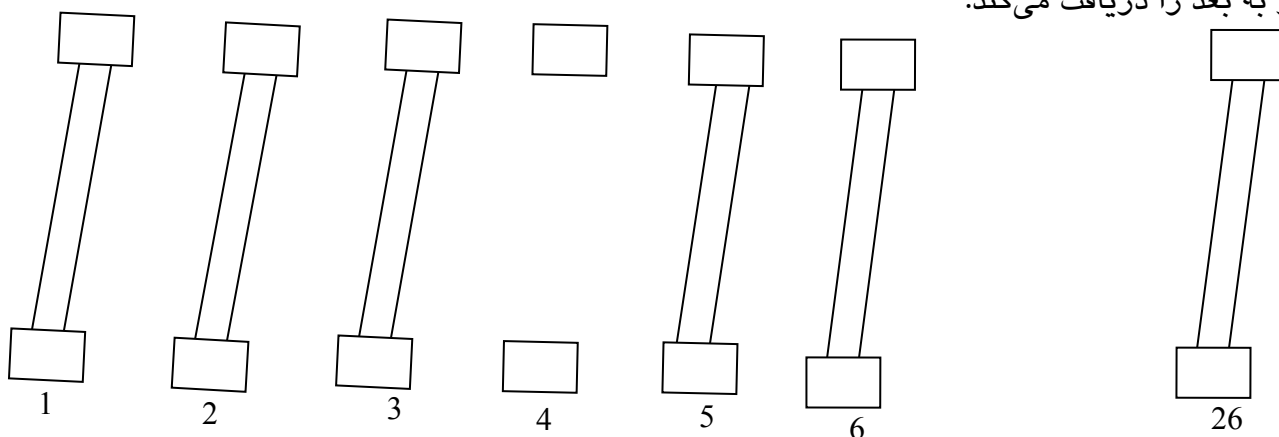
۱- Selective repeat: اگر مثلاً گیرنده، چهار را دریافت نکند می فهمد که نیامده، بقیه را می گیرد ولی در بافر

نگه می دارد و کاری انجام نمی دهد. بعد که فرستنده چون ACK چهار را دریافت نکرده، دوباره آن را

می فرستد روی بقیه کار انجام می دهد.

۲- go back repeat: اگر چهار را دریافت نکند، بقیه را نمی گیرد و فرستنده که کارش تمام شد، دوباره از

چهار به بعد را دریافت می کند.



اولی بافر را هدر می‌دهد ولی دومی پهنای باند را هدر می‌دهد. مزیت اینها بستگی به موقعیت ما دارد که آیا بافر داریم و پهنای باند گران است یا پهنای باند ارزان است ولی بافر گران است.

HDLC: high level Data Link Control

حدود سال ۷۰ IBM، پروتکل HDLC را که تجاری بود ارائه داد. و مؤسسات ANSI و ISO ←

ADCCP: ANSI

HDLC: ISO نام آن را تغییر دادند.

ویژگی‌های HDLC:

(۱) Bit Oriented: از الگوهای بیتی ابتدایی و انتهایی و برای حل مشکل از bit stuffing استفاده می‌کند.

01111110	Address	control	Data	ckecksnm	01111110
----------	---------	---------	------	----------	----------

ارتباطات Data Link در اینترنت:

پروتکل SLIP: serial link IP

ISP: Internet service provider client- ۱- اولین ارتباط PTP می‌خواهیم ارتباطات PTP که ماهیت

Datalink در اینترنت دارند را پیدا کنیم:

پروتکل ارتباطی مطرح می‌شود → ارتباط بین روترها

(۲) دومین ارتباط PTP

(۱) اولین پروتکل SLIP بود. مشکلی داشت:

۱- OPEN نبود ← فقط مربوط به IP بود و جای دیگر به درد نمی‌خورد و به سیستم‌های دیگر وصل نمیشد.

۲- Authenticalation بررسی مجوزها (برای دسترسی) نداشت.

۳- error detection correction به هیچ وجه نداشت.

۴- اگر دو طرف می‌خواستند با هم ارتباط برقرار کنند باید آدرس هم می‌داشتند و گرنه اتصال ممکن نبود چرا

که خودش نمی‌توانست حین اجرا آدرس را پیدا کند.

۵- استاندارد نبود و هرکس برای خود نسخه‌ای از آن را نوشته بود.

بنابراین روتکل PPP: point to point Protocol مطرح شد که اکنون 80% کامپیوترها از آن استفاده می‌کنند. مشکلات قبلی را ندارد. به جای Bit oriented، کاراکتر اُزینتید است و این تفاوت عمده آن با SLIP است و برای رفع مشکل از Character stuffing استفاده می‌کند و سروته داده را با کاراکتر می‌بندد.

لایه Physical:

از جمله وسایلی است که در این لایه هستند: عبارتند از:

(۱) repeater

(۲) بعضی از انواع hub

لایه اول و دوم را پوشش می‌دهند. → (۳) بعضی از کارت شبکه‌ها

بیشتر لایه اول و بعد دوم (بالا تر نمی‌آید) → (۴) مودم

انواع hub

Pasive: غیرفعال: n تا پورت دارند، داده ورودی را روی n پورت دیگری می‌فرستند.

Active: فعال: تقویت می‌کند.

Intelligent

اگر IP, VLAN را آدرس برای شناسایی قرارداد، لایه سوم است و مانند router است. اگر کارت شبکه باشد لایه دوم و data link است.

Passive: کابل UTP که حداکثر ۱۰۰ متر است در آن به کار می‌رود. حداکثر فاصله ۱۰۰ متر است.

: هیچ کاری روی سیگنال انجام نمی‌دهد.

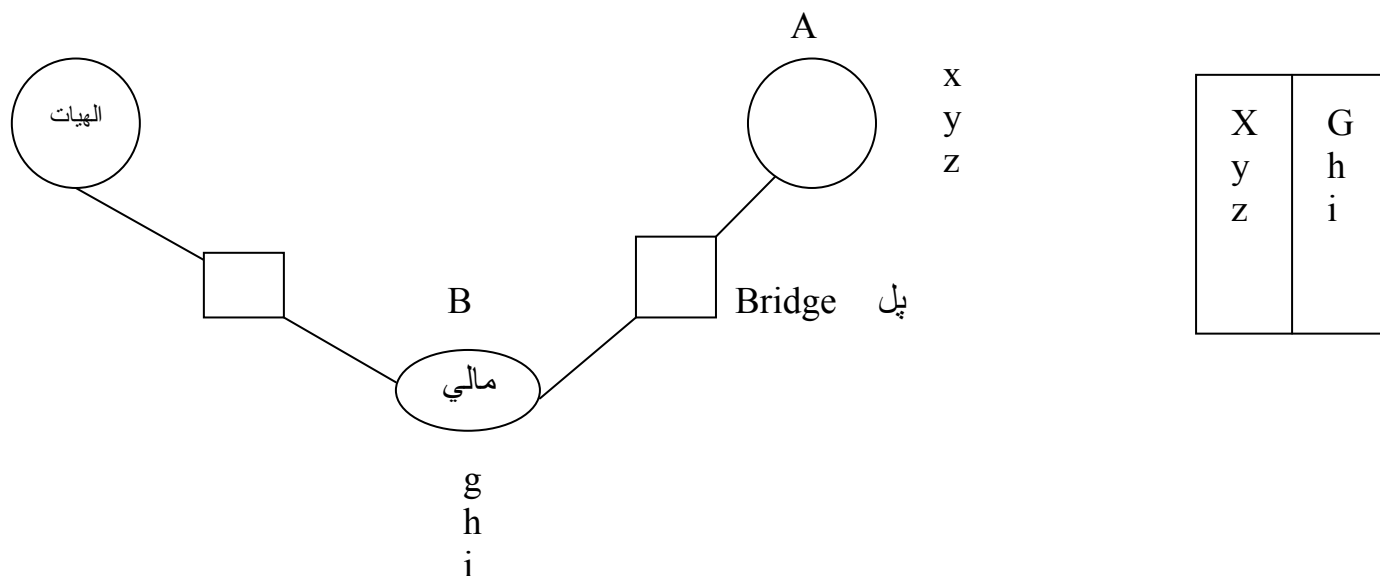
Active: چون تقویت می‌کند فاصله ۲۰۰ متر است.

Intelligent: به آنها Switch هم می‌گویند. هوشمند هستند و پروسور دارند. آدرس پورت‌ها را می‌دانند و داده

ورودی را روی پورت خاصی که آدرس آن در Packet است می‌فرستد و شبکه را شلوغ نمی‌کنند. خیلی

انعطاف دارند. بعضی سیستم عامل دارند و ویروس می‌گیرند. از پشت کامپیوتر قابل تنظیم هستند.

VLAN ، یک LAN مجازی است که بعضی سوئیچ ها امکان آن را می دهند.



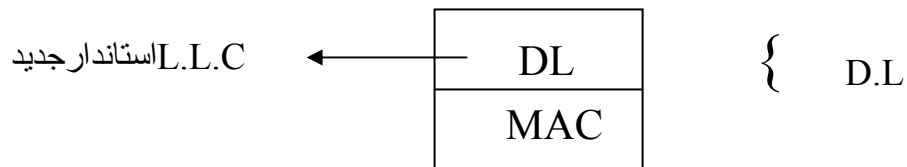
پل: حداقل دو تا کارت شبکه دارد و جلوی ترافیک کل شبکه را می گیرد و مانند یک فیلتر عمل می کند و ترافیک محلی می شود. (اگر مقصد در حوزه محلی باشد به آن اجازه رفتن به جاهای دیگر را نمی دهد)

این پل مشکلاتی دارد کامپیوترها عوض می شوند یا آدرس ها تغییر کند در آن صورت کار supervisor باید مدام ایجاد جداول باشد.

پل های جدید آمده است که هوشمند است و Intelligent Bridge , learning که می گوید ما نیاز به جدول نداریم ما را نصب کنید خودما جدول را ایجاد می کنیم. در ابتدا مانند hub عمل می کند و تا چیزی ننویسیم خودش را نمی شناسد کم کم که اطلاعات فرستاده شد یاد می گیرد و دیگر broadcast انتشار انجام نمی دهد نکته دیگر در مورد این الها این است که هرچند وقت یکبار خودش را update میکند یعنی پاک می کند و دوباره متشدد می کند تا جابجایی ها را پوشش دهد. کامپیوتر جدید که وارد شبکه می شود خودش را broadcast می کند و پل اینگونه او را می شناسد.

MAC: Media Access Control

گاهی کارت های شبکه را MAC گویند ولی MAC کلی تر است. وقتی D.L طراحی شد شبکه های LAN به وجود آمد و دیگر PTP نبود. فرض D.L هم این است که در PTP است و خط ارتباطی اختصاصی است ← به مشکل برخوردند. مشکل OSI نیز همین بود. بنابراین تکه جدید به نام MAC زیر آن اضافه کردند و به کل D.L گفتند. MAC کاری می کند که قسمت پایینی به صورت PTP برای بالا شبیه سازی شود.



وقتی شبکه‌ای مانند BUS داریم که چند کامپیوتر از یک خط مشترک به صورت سلامت می‌نوانند استفاده کنند، لایه MAC امکان این استفاده‌ها از یک Media مشترک ایجاد می‌کند.

پروتکل‌ها و الگوریتم‌های کارکرد MAC برای حل این مشکل:

کار ساده‌ای نیست که N تا ایستگاه که از هم خبر ندارند از خط مشترک به سلامت استفاده کنند.

الگوریتم‌ها ← ایستا Static : زمان را به n تکه مساوی تقسیم می‌کنیم و به هر ایستگاه یک بازه زمانی می‌دهیم.

۱- IDM: Time derision mudulation: تقسیم زمانی

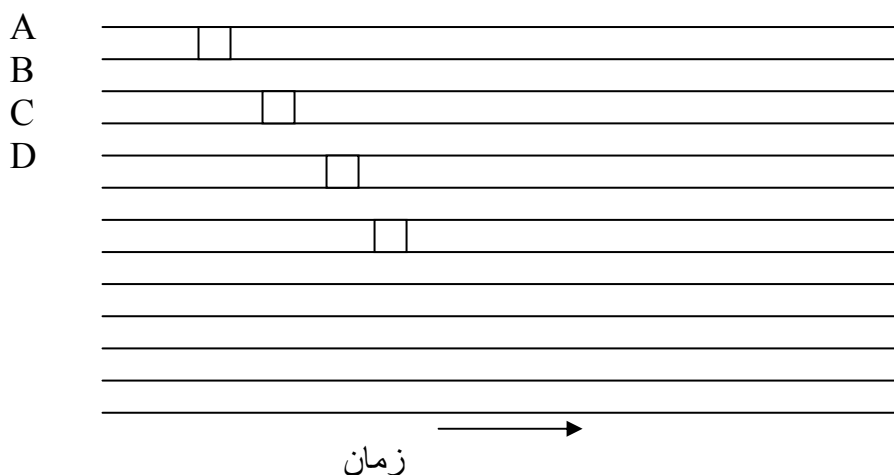
۲- FDM: Frequency DM: تقسیم باند فرکانسی

مشکل ۱- بهینه نیست، مثلاً ایستگاه داده ندارد ولی ظرفیت دارد. در کنار آن ایستگاهی داده زیاد دارد ولی زمان ندارد.

← پویا Dynamic : خودشان را بسته به شرایط تطبیق می‌دهند و ظرفیت‌ها را تغییر می‌دهند.

: ALDHA

Pure Aloha خالص



اولین الگوریتم ساده بود هیچ قانونی نداشت. ایستگاهها اطلاعات را می‌فرستند. اگر تصادم رخ داد، آن را ثبت می‌کنند و دوباره اطلاعات را می‌فرستند.

ایستگاه : Client های ورودی شبکه که از خط مشترک استفاده می‌کنند

تصادم: colision

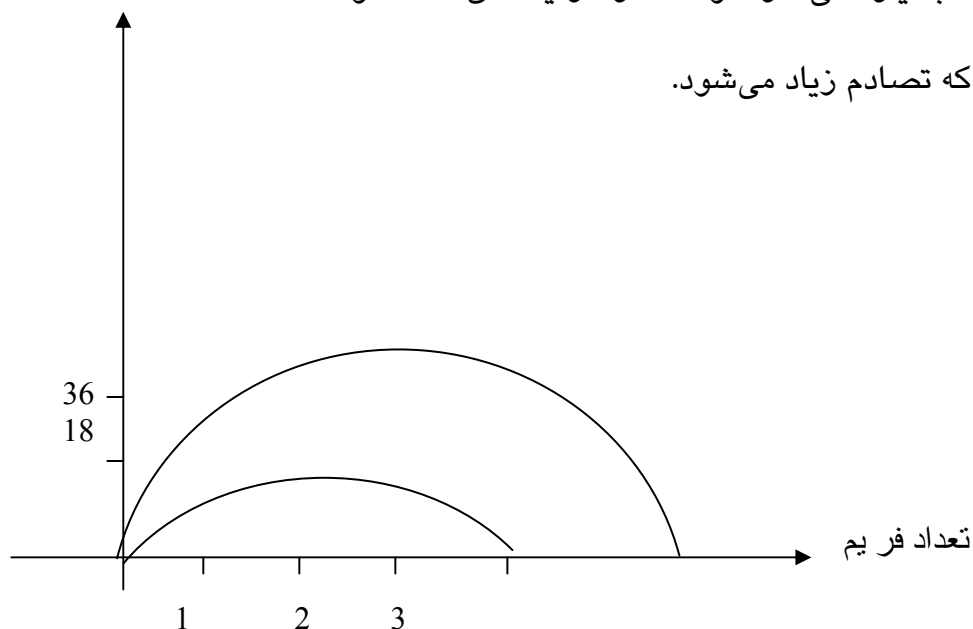
زمان پیوسته : Countionus time : هرکس هر وقت خواست داده می‌فرستد و محدودیت ندارد.

زمان تکه‌تکه: Stotted time : زمان تکه‌تکه‌شده و به slotهایی تقسیم شده و هرکس خواست اطلاعات بفرستد فقط در اول slot می‌تواند.

موج حامل Corrier: آن سیگنالی است که ما داد را روی آن سوار می‌کنیم و می‌فرستیم. یعنی ما موج را بر حسب داده تغییر شکل می‌دهیم و می‌فرستیم. نسبت به Corrier دو نوع ایستگاه ما داریم ← کاری به آن ندارد

← آن را حس می‌کنند و Carrier sence دارند.

Aloha بازده بسیار کمی دارند و حداکثر ظرفیت آن تا حدود ۱۸٪ است frame ما که زیاد شود به ۲ می‌رسد چرا که تصادم زیاد می‌شود.



تعداد فریم ارسال شو نده همزمان

عیب عمده این روش که باعث کم‌بودن بازده می‌شود این است که چون از هم خبر ندارند ممکن است ابتدای یک Frame به انتهای دیگر برخورد و هر دو خراب کند.

روش دوم برای حل مشکل Slotted Aloha ها هستند. در آنها زمان به اندازه حداکثر طول زمان بودن frame ها روی خط است و دیگر مشکل دوم پیش نمی آید. بازده به ۳۶٪ می رسد و با شلوغ شدن دوباره کم می شود.

CSMA Carrier Sense multipk Access : عیب عمده دو مورد قبل این بود که کاری به هم نداشتند. اما این مورد می گوید من carrier sense دارم و هر وقت خد اشغال بود صبر کنید و اگر خالی بود اطلاعات را می فرستد. ← به خط گوشی می دهد و هوشمندتر است. باز هم مشکلاتی دارد ←

۱- دو تا ایستگاه همزمان به خط گوش می کنند تا آزاد شود و اطلاعات را روی خط می گذارند.
۲- دو تا ایستگاه می بینند که خط خالی است و همزمان اطلاعات را روی خط می گذارند و تصادم رخ می دهد.
۳- فرض کنید تأخیر پخش امواج روی خط t ثانیه باشد. propagation delay این t خیلی کم است ولی صفر نیست. در این فاصله اگر کسی اطلاعات بگذارد تصادم رخ می دهد.
CSMA معمولاً یک خطایابی دارد و گوش می کند. اگر داده ها با آن چه گذاشته فرق داشته باشد می فهمد که تصادم رخ داده و دوباره آن را می فرستد.

انواع CSMA ۱- PERSISTENT

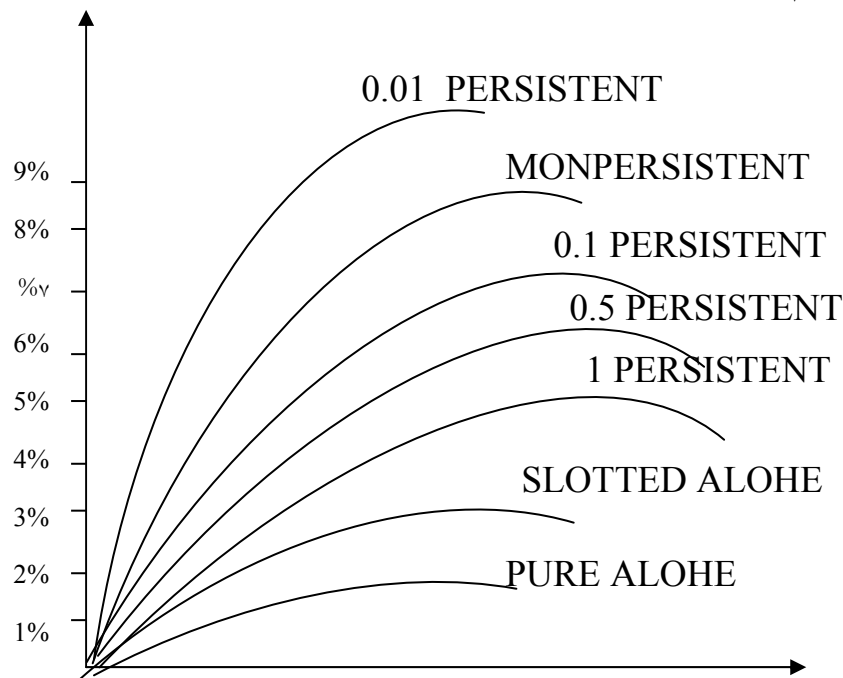
۲- non persistent

۳- p.persistenr

۱- Persistent : وقتی تشخیص داد خط خالی است با احتمال ۱ اطلاعات را می فرستد.
۲- Non Persistent : به خط گوش می کند اگر خالی نبود خط را ول می کند زمان تصادم صبر می کند و دوباره الگوریتم را اجرا می کند که مشکل اول را حل می کند. بازده تا ۷۰ درصد هم می رود.
← این زمانها به حد ms است.

p.Persistent: اگر حتی ببیند خط خالی است دلیل ندارد حتما بفرستد. با احتمال p می فرستد و با احتمال (1-

p) نمی فرستد. خواهیم دید که بهترین است. به عنوان نمونه احتمال ۱٪ از همه بهتر عمل می کند.



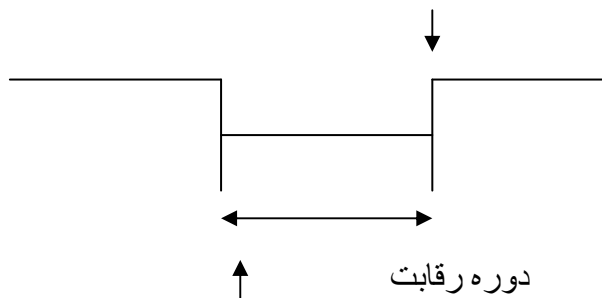
CSMA/CD ← CD: Collision Detection هم دارد و امروزه بسیار پرکاربرد است.

پویا ← With collision تصادم دارد. ← No collision یا collision free بدون تصادم ← limited

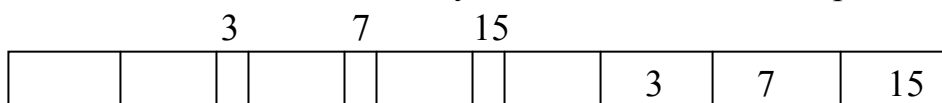
collisoin با تصادم محدود و رقابت محدود.

بدون تصادم ها، ادعایشان در آن است که اصلاً اجازه نمی دهد تصادم رخ دهد. برای توضیح آن باید روی یک مفهوم توافق کرد.

Contention Peroid: دوره رقابت. رقابت پیش می آید تا خط را در اختیار بگیرد.



روشهای بدون تصادم دو نوع است ← bit map ← binary Count down



: bit map

فرض کنید که n ایستگاه داریم و در دوره رقابت n تابت داریم.

نفر ایستگاه که اطلاعی برای ارسال دارد بیت متناظر خود را set می‌کنند. بعد از پایان دوره رقابت معلوم است که کدام ایستگاه‌ها اطلاع برای ارسال دارند به آنها سویت می‌دهد و آنها به ترتیب از کمترین شماره شروع به ارسال اطلاعات خود می‌کنند و در واقع به جای رقابت به وبت خط به آنها داده می‌شود و به این ترتیب تصادم رخ نمی‌دهد. بعد از تمام شدن دوباره رقابت است.

Binary couat down : شمارش دودویی به سمت پایین:

در این روش نیز باز فرض می‌کنیم که هر ایستگاه یک آدرس دارد از ۱ تا n. فرض کنید که چهار ایستگاه می‌خواهند اطلاعات همزمان ارسال کنند.

ایستگاه‌ها بیت اول خود را روی شبکه bread cast و پخش می‌کنند. اگر یک ایستگاه ببیند که دو بیت خودش است و بیت ۱ از بقیه روی شبکه‌ای از گردونه خارج می‌شود و بعدبیت دوم و ... در نتیجه رقابت آن ایستگاه که آدرس بزرگتر داده برنده می‌شود.

هیچگاه تصادم پیش نمی‌آید.

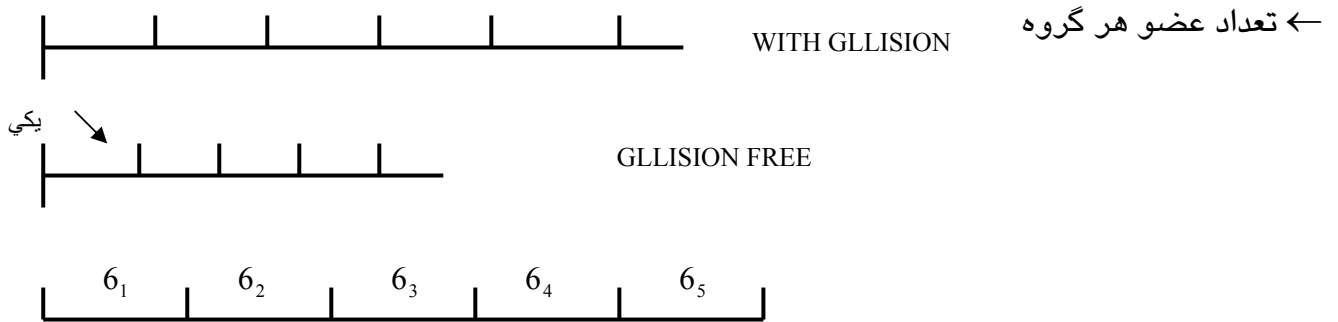
اگر بیت بزرگتر از خودش در شبکه باشد از رقابت خارج می‌شود. اگر خیر در رقابت می‌ماند و رقابت روی بیت‌های بعد صورت می‌گیرد.

نکته: درست است که wllision tree ها است و تصادم رخ نمی‌دهد. ولی این عیب را دارد که سربار ایجاد کند. البته اگر بار شبکه و load بالا باشد خیلی خوب و پرکاربرد است ولی اگر بار شبکه کم باشد باز هم الگوریتم سنگین و زمانبر را انجام می‌دهد. در نتیجه علاوه بر ایجاد سربار ایستگاه‌ها را معطل می‌کند. سربار بودن از ای جهت است که بیت‌ها راه عمل کرده است.

With wllision ها وقتی load پایین است خوب هستند ولی در load بالا کارایی را کاهش می‌دهند، collection tree ها برعکس هستند.

طراحان شبکه به این نتیجه رسیدند که باید الگوریتم‌ای ارائه دهند که در load بالا مانند بدون تصادم باشد و معطل نکند. و در load بالا کارایی را هدر ندهد. در واقع ایه این مسئله از انجا آمده که اگر شبکه slot بندی شده بود در روش collection free سربار مال یکی بود.

راه حل: n ایستگاه ، یا گروه که ایستگاهها را طبقه بندی می کنیم.

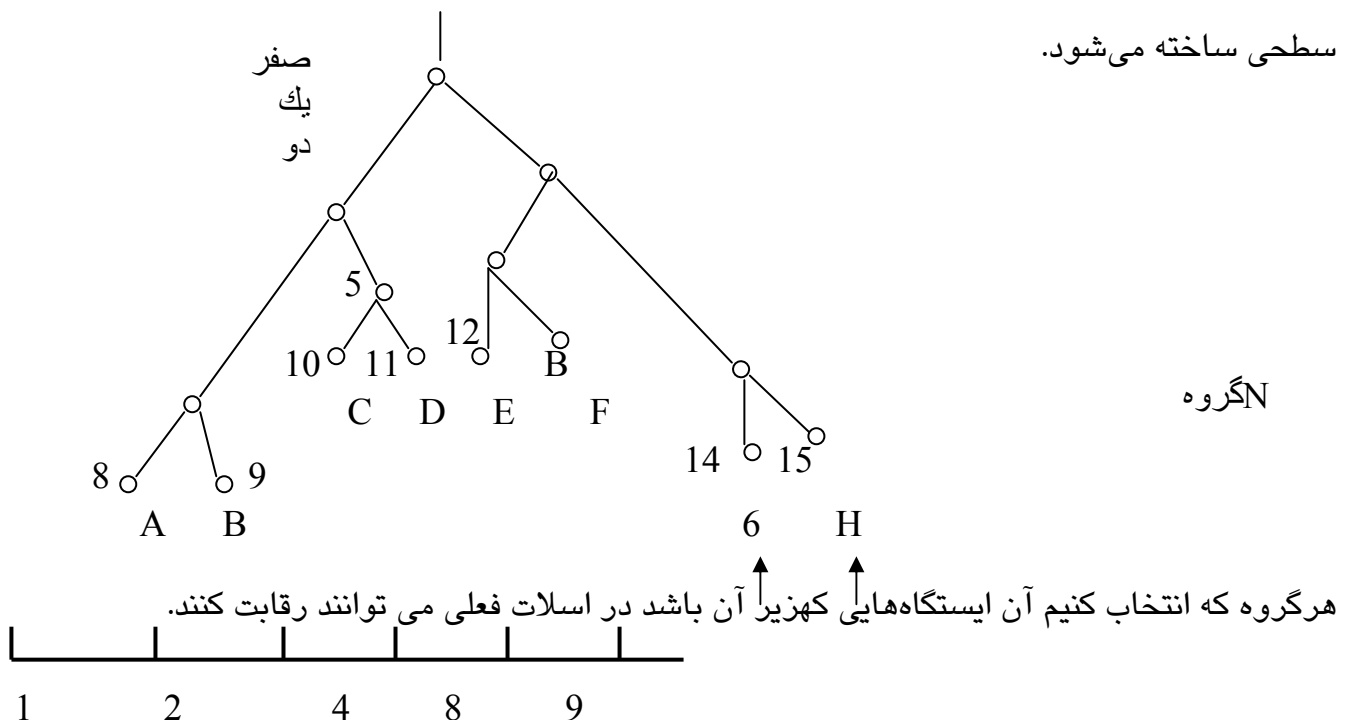


مراسلات را به یک گروه اختصاص می دهیم. اگر بتوانیم تعداد عضو گروهها را بسته به بار شبکه تنظیم کنیم الگوریتم درست شده است :

بار کم: تعداد عضو: ← مانند with collision

بار زیاد: تعداد عضو : یک ← مانند collision free

الگوریتم: فرض کنید ۸ تا ایستگاه داریم که هر کدام برگهای یک درخت دو دویی هستند ← یک درخت سه سطحی ساخته می شود.



وقتی روی سطح صفر هستیم تمام ایستگاهها می توانند مراسلات ی رقابت کنند. سه حالت رخ می دهد:

۱- تصادف رخ می دهد. شبکه شلوغ است و رقابت زیاد است.

۲- یکی از ایستگاهها اطلاعات ارسال می کنند ← باز هم نشان می دهد که شبکه خیلی شلوغ نیست.

۳- هیچکدام اطلاعات ارسال نمی کند ← خوب است و دوباره اسلات را به همان گره می دهد.

وقتی تصادف شد ← رقابت طبیعی نیست و باید عکس العمل نشان دهد ← یک سطح به سمت پایین حرکت می کند. پس اسلات ۲ را به گره های زیر گروه ۲ و اسلات ۳ را به گره های زیر گره ۳ می دهد. در حالی که اگر خلوت بود در همان سطح یک می ماندیم. ولی حالا که شلوغ بود به سطح پایین تر آمدیم. یعنی وقتی شلوغ ات آنقدر پایین می آییم که گروه ها تک عضو می شود و تا آخرین سطح پایین می آییم.

اگر n تا ایستگاه باشد منطقی نیست که از سطح صفر شروع کند و حتماً تصادم رخ می دهد وقتی n تا attempt برای گرفتن اسلات است از سطح @ شروع می کنیم.

این الگوریتم امکان بهینه سازی دارد:

فرض کنید که G,H را می خواهد بفرستد.



برای ارسال ۲ تا frame ۷ تا مصرف شد.

راه دوم: وقتی در ۱ تصادم است و ۲ خالی است پس تصادم در ۳ بوده. بنابراین اگر اسلات به آن بدهد می بایست آن را بشکنیم چون تصادم رخ می دهد. پس همان جا می شکند. در این جا باز هم به ۷ نمی دهد و



می شکند.

با ای هوشمندی ه تا frame مصرف شد و دو تا صرفه جویی کردیم.

استانداردهای IEEE :

یکسری اساندار برای شبکه های LAN تعریف کرده است. به استانداردهای 802 موسوم هستند و بیشتر در رابطه با لایه Datalink هستند. یکسری کتابچه ها این استانداردها را می سازند.

راجع به استاندارد و محتویات دسته ها → 802.1

→ 802.2

استانداردهای MAC

802.3 Ethernet

802.4 token bus

802.5 token ring

802.6 MAN

802.7

استانداردهای 802.3 تا 802.5 همگی LLC را قبول دارند. لایه MAC را پوشش می‌دهند و فقط پایه MAC
ففرق دارد.

LLC: یک روش 1.persistance است که با احتمال یک می‌فرستد. (به خط گوش می‌دهد و با احتمال یک
می‌فرستد)

در ethernet به چهار شکل شبکه‌های LAN می‌توان فرستد و هرکدام یک اسمی دارد.

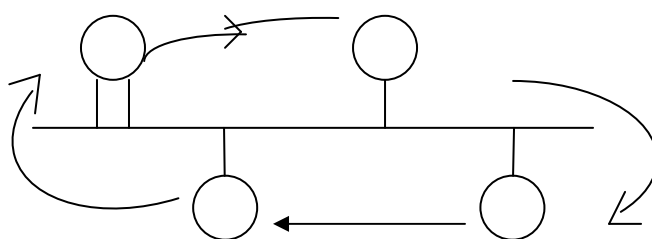
کابل‌هایی که هر زمان یک سیگنال روی آن است. Base band

همان شبکه‌های bus معمولی $\rightarrow$ 10 Base 2

کد شبکه‌های coaxial هستند. حداکثر اول segment شبکه 200m

segment = اندازه‌ای از media که بواسطه فیزیکی سیگنال ضعیف نمی‌شد و قابل قبول بود.

تا ۱۸۰ متر طول دارند، BNC و conector دارند. به این شبکه‌ها TioNet هم می‌گویند.



Couaxial. تا ۵۰ است.

معمولاً hub دارد و کابل UTP دارد.

نوع واسط مورد استفاده

فیبر دارد.

Token bus: دسترسی به لایه MAC از طریق token bus است. 802.4 بهترین شکل برای این شبکه‌ها
قراردادن سیستم bus بود. ولی عیب عمده این بود که نمی‌شد اولویت تعیین کرد. و ایستگاه‌ها اولویت را
بردارند. مثلاً در یک نیروگاه ممکن است حیاتی باشد. و دیگر آن که برای کاربردهای recal time جواب
نمی‌دهد یعنی سیستم‌هایی که کارهایشان باید در یک محدوده خاص انجام گیرد، چون در این شبکه‌ها سقف

زمانی معلوم نیست و اگر شلوغ باشد ممکن است هیچگاه نتواند اطلاعات را بفرستد در نتیجه گفتند ما از همین توپولوژی استفاده می‌کنیم ولی مفهوم token را وارد می‌کنیم. آنگاه هر ایستگاهی بخواهد اطلاعات را بفرستد باید token (یک فریم که فرمت خاصی دارد و متمایز از داده است) را بگیرد تا نتواند اطلاعات بفرستد. W token معمولاً در Ring به کار می‌رود ولی در این جا bus است ← این روش collision free است. Bus را بکار بردند چون ماهیت مسئله آنها خیلی هماهنگ بود. برای آنکه در خود منطقاً یک Ring را تشکیل دهند هر ایستگاه، بعدی خود را می‌شناسد.

شبکه broad cast است ولی هر ایستگاه آدرس کارت شبکه بعدی خود را می‌گذارند و برای همه می‌فرستد، همه می‌بینند ولی بعدی آن را برمی‌دارد. هرکدام که token را برداشت، خط مال اوست. اطلاعات را می‌فرستد و دیگر Ring را رعایت نمی‌کند چون خط مال خودش است، بعد که کارش تمام شد، token را ایجاد می‌کند. و روی خط می‌فرستد. (روی Ring)

مزایا ← ۱- در این روش به راحتی می‌تان تقدم و تأخر را معلوم کرد.

۲- حداکثر زمان توقف معلوم است. اگر n ایستگاه باشد، هر ایستگاه بیشتر از n ثانیه نمیتواند token را نگهدارد ← حداکثر زمان معطلی برای خط nt است.

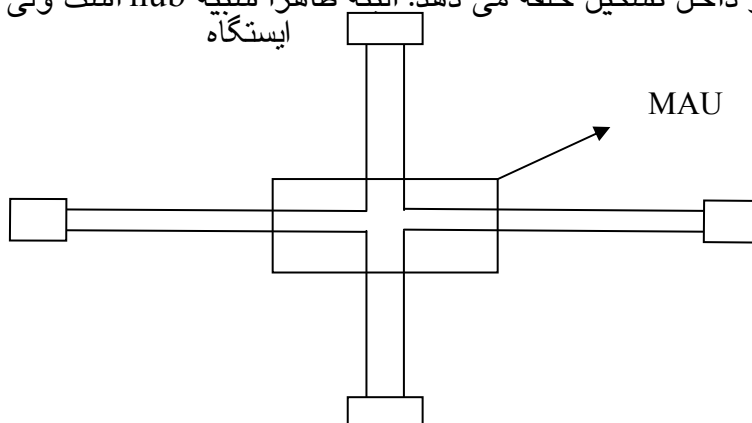
معایب ← پیچیدگی الگوریتم است. تعداد زیادی متغیر و زمانبند و ... دارد.

محاسن عمده ← ۱- در لایه فیزیکی می‌توان از کابل تلویزیون استفاده کرد.

۲- پس ارزان است و broud baod است و چند سیگنال را میتواند همزمان بفرستد.

To cleen ring: مانند قبلی است ولی منطقاً و فیزیکی یک Ring است. 802.s شکل ظاهری مانند star

است. در مرکز یک Mau دارد که در داخل تشکیل حلقه می‌دهد. البته ظاهراً شبیه hub است ولی فرق دارد.



* بالایی دوتا زائده دارد. اگر برق آن قطع شود زائده ها کنار هم قرار می گیرند و ایستگاه بالا را by poss می کنند، قدرت تصحیح و رفع اشکال خود را تنها این سیستم دارد. چرا که در صورت قطع برق بالایی از شبکه خارج می شود و شبکه خودش را اصلاح می کند.

در یک مقایسه 802.3 از همه ساده تر، ارزان تر، و پرکاربرد تر است. از نظر نصب هم بسیار راحت است و همان طور که شبکه کار می کند می توان ایستگاه کم یا اضافه کرد و نیاز نیست شبکه را خواباند. در حالی که در to ken Ring و حتی to ken bus این طور نیست. غیر قطعی یا nondeterministic است و زمان پاسخ قطعی آن معلوم نیست. تقدم و تاخر ندارد. عیب دیگر آن هم ای ناست که جزئی در آن وجود دارد که آنالوگ است. ولی ما در شبکه به سمتی باید پیش برویم که دیجیتال بشود. باید جزئی آنالوگ باشد که نویز آنالوگ را تشخیص دهد و بفهمد که تصادم رخ داده و باید جزئی مانند کابل شبکه باشد.

از کابل تلفن استفاده شده، ارزان است. تقدم دارد. در اکثر زمان پاسخ دارد. 802.4 پیچیده است. چون broad bad است جزء Anologe engineering دارد. فیبر نوری قابل نصب نیست. عیب دیگر آن است که کم نصب شده و جاهای خاصی کاربرد دارد. الگوریتم پیچیده و رمامبر دارد که دچار مامبر دارد.

عیب سرعت پایین، مزیت ارتباطات نمابر point_to point است. مهندسی آن ساده است: 802.5 خودش می تواند خودش را تصحیح کند. Aouto callect است، در اکثر زمان پاسخ دارد. عیب Ingle point of failinre دارد. اگر مرکز خراب شود شبکه می خوابد.

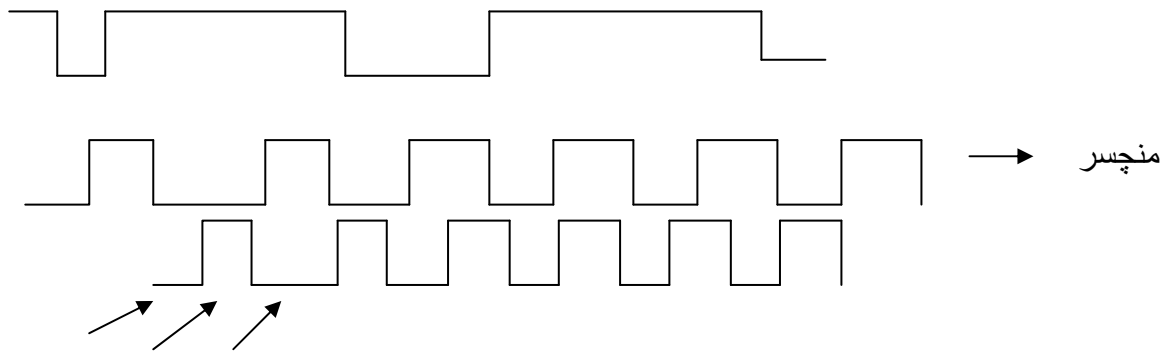
کدینگ منچستر: در مورد کدینگ بیت های منطقی روی کامپیوتر به تبدیل بیت های فیزیکی روی media صحبت می کند.

در کامپیوتر معمولاً یک منطقی به یک ۱ فیزیکی و ۰ فیزیکی پشت آن تبدیل می شود.

صفر منطقی به یک ۰ فیزیکی و ۱ فیزیکی پشت آن تبدیل می شود.

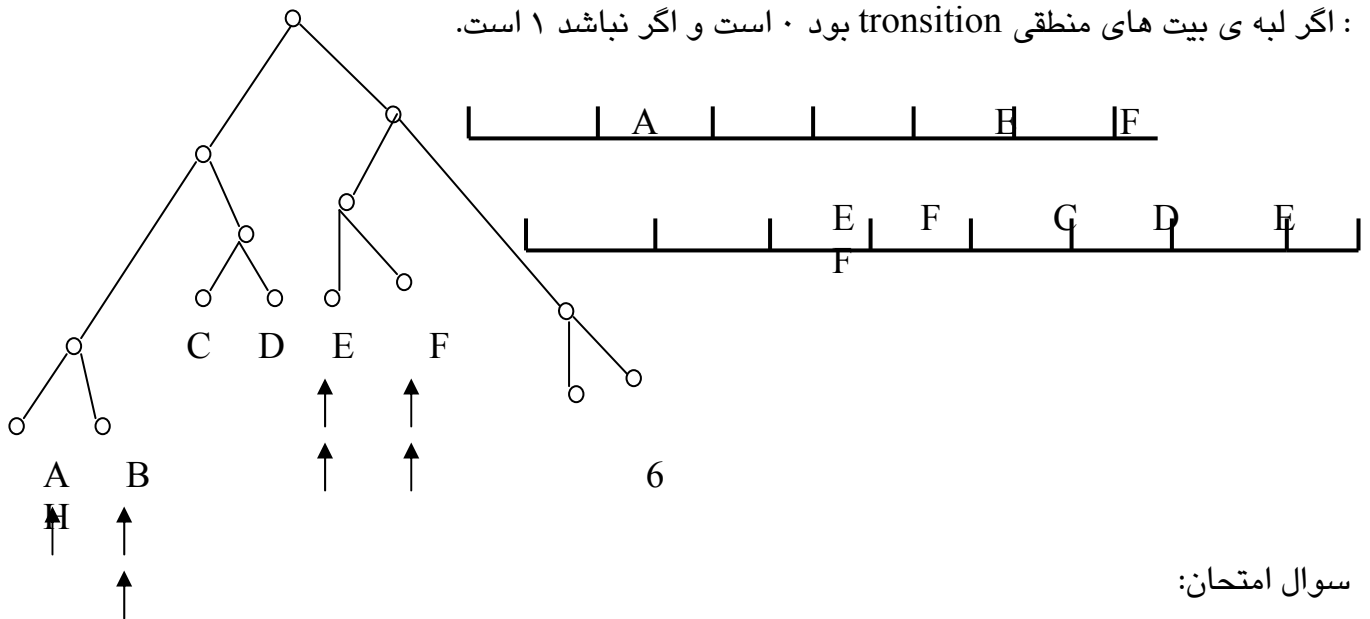
یعنی ظرفیت منطقی به دو بیت فیزیکی تبدیل می شود.

معنای آن این است که اگر می توانستیم همین گونه روی شبکه بفرستیم می شد:



در منچستر به transition حساس است و اگر موج ها ب انویز تغییر کند باز هم transition می ماند.

: اگر لایه ی بیت های منطقی transition بود ۰ است و اگر نباشد ۱ است.



سوال امتحان:

۱- بین دو شبکه قرار می گیرد تا اطلاعات که تبادل می شود را فیلتر کند و از ترافیک شبکه جلوگیری می کند.

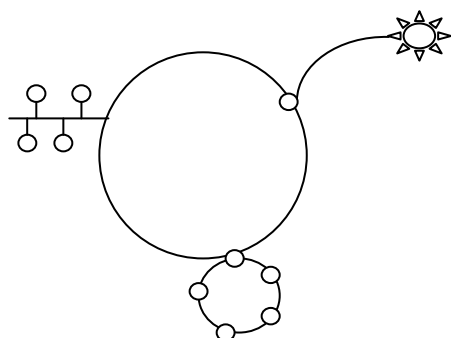
۲- کار دیگر آن تبدیل پروتکل های رویه MAC به یکدیگر است. مثلا 803- 802 و یا ترکیبی از آن ها. (تبدیل پروتکل دو لایه Data link).

IEGE: سرعت یا ظرفیت شبکه را نشان می دهد که تمایز آن چه دیدیم ۱۰ بود. ابتدا ICME خیلی خوب بود ولی بعد ها جو انگیز نبود و گفتند باید تکنولوژی را به گونه ای معرفی کنیم ه سرعت بالا برود. سراغ high speed lon رفتند. دو نوع lon پر سرعت (100 MB) وجود دارد. (امروزه 1 KB= 1000MB نیز آمده است).

1) FDDI. Fyner Distributed Data Intciface

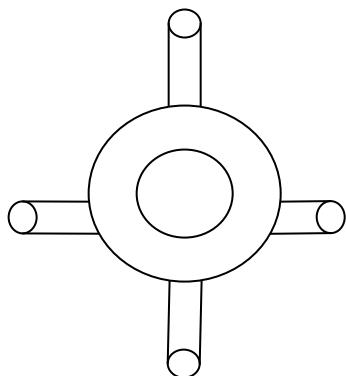
در آن نیز به کار رفته است و توپولوژی Ring را معمولاً دارد. در جایی کاربرد دارد که شبکه های مختلف را در فواصل فیزیکی دور بر هم وصل کنیم. هر کدام از شبکه ها را یکی از کامپیوتر هایش را با فیبر به شبکه دیگر وصل می کنیم. چون فیبر می تواند کیلومترها طول داشته باشد شاید تعریف Lon چندان برای آن درست نباشد.

برای منبع نور از LED و فیبر Multi media استفاده می کنند. هزینه ی آن کمتر است ولی ساخت کمتری را



پوشش می دهد

توپولوژی پیاده سازی آنها به شکل روبرو است که دو تا Ring تو در تو دارد که back up هم هستند. حتی اگر جایی هر دو قطع شوند به هم وصل کنیم یک Ring دیگر ایجاد می کند و باز می توان اطلاعات را ارسال کرد.



FDDI ها از نظر کاربردی خوب بودند ولی از نظر پیاده سازی توپولوژی آن ها سخت و گران بود. پس باید نگاهی جدیدی را ارائه می دادند. پس فکر کردند از روش تست موجود استفاده کند ولی سرعت آن را افزایش دهند. دو راه داشتند:

۱- استاندارد جدید با سرعت 100

۲- تغییر استاندارد که سرعت 100.10 شود و قبلی را توسعه دهند.

آنها به سه دلیل راه و نوع را انتخاب کردند:

۱- گستردگی اینترنت در وب که هزینه متغیر زیاد بود.

۲- مشکلات اینترنت تا آن زمان حل شده بود و امتحان خود را پس داده بود و عادی شده بود.

۳- اگر معطل می کردند تا استاندارد را ارائه دهد، bad timing رخ می داد و همان بلایی سر آن می آمد که برای OSI رخ داد. دیگر سرمایه گذاری زیادی انجام شده بود و بازگشت امکان نداشت.

802.3 u_upgrade را ارائه دادند. که در واقع سه تا cabling مختلف دارد.

کابل تلفن که 4سیم دارد.

100 base - T4

100 base - T4

100 base - F

دیگر در آن توپولوژی bus وجود ندارد چون نمی تواند 100 باشد. حتما باید Itub به عنوان قطعه مرکزی باشد. 95% شبکه ها نوع enbling آن upp است. خیلی شبکه ها وجود داشت که باید آن ها را هم پشتیبانی می کردند. در T4 از کابل تلفن استفاده کردند که می توانست تا سرعت 100 باشد (CAT3 تلفن است). ولی Half dublen است و سرعت مجموع ارسال و دریافت 100 است.

TX : Cats است و Full duplex است و سرعت ارسال دریافت هریک 100 است.

F: از فیبر نوری استفاده می کنند.

امروزه طوری است که سیم ها امکان انتقال ۱۰ و ۱۰۰ و ۱۰۰۰ را دارد و بسته به آن که ایستگاه چه ویژگی هایی داشته باشد از آن استفاده می کند.

لایه Network:

مابین لایه Datalink و Transport قرار گرفته است. هر چیز مرتبط با بستر شبکه به آن مربوط است.

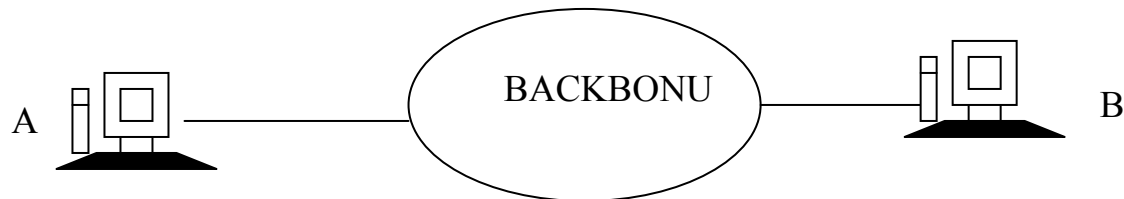
وظایف عمده:

۱- مسیریابی که از همه مهمتر است.

۲- کنترل ترافیک یا ازدحام.

۳- تبدیل پروتکل‌های مختلف لایه شبکه به همدیگر.

در واقع یکی از وظایف عمده لایه شبکه این است که ها راز لایه transport می‌گیرد و آن را در شبکه‌های WAN کنترل می‌کند تا به مقصد برسد. مهمترین لایه است. در LAN چندان دیده نمی‌شود چو مسیریابی معنا ندارد.



به قسمت‌های اصلی و حیاتی شبکه اینترنت گویند.

شریان‌های تبادل هستند در واقع هرچه به شبکه وصل است را شامل می‌شود. مانند: خیابانهای اصلی شهر که جاهای دیگر را تغذیه می‌کنند. در شبکه کامپیوتر هم جابجایی اسامی را گویند ک اصلی و حیاتی است و ظرفیت آن راز زیاد است و جاهای مهم را به هم وصل می‌کند. دو جزء اصلی دارد:

۱- اتصالات

۲- روترها

که بستره کاری و انتقال اطلاعات را به وجود می‌آورد.

اصلی که snbrut ها هیچگاه فراموش نمی‌شود این است که بین روترها چندین ارتباط وجود دارد که اگر یکی از کار افتاد کارش را ادامه دهد (حداقل سه ارتباط) به سه طریق اطلاعات از A به B منتقل می‌شود.

Switching : در روترها و شبکه‌ها به این معنا است که وقتی اطلاعات به روتر می‌رسد ب توجه به مقصد معلوم کند که بهترین خط خروجی کدام است و ورودی را به آن خط خروجی سوئیچ کند. در آن لحظه روتر براساس جدول routing این کار را انجام می‌دهد دنباله‌ای از تصمیمات محلی در روترها مسیر بهینه را ایجاد می‌کند چون تک‌تک آنها مسیر بهینه را انتخاب کرده‌اند.

روش اول circuit Switching :

شبکه‌های تلفن نیز که قدیمی نیز هستند از همین روش است با این تفاوت که سلسله مراتبی است و همه در یک سطح نیستند.

ویژگی اصلی این شبکه آن است که مانند تلفن وقتی ارتباط برقرار شد تا مادامی که ارتباط است ظرفیت را که بین ارتباط اختصاص داده است نگاه می‌دارد و دائم مال شما است.

روش دوم Message Switching

ظرفیت را اختصاصی نمی‌دهد. تمام message را در یک pocket قرار می‌دهند که با اندازه mus بزرگ می‌شود. تا مادامی که انتقال شود خط را اشغال می‌کند ولی اختصاص نمی‌دهد. به این روش storage & forward گویند. ذخیره می‌کند و بعد به روتر بعد forward می‌کند. وقتی تعدادی بسته می‌آید آنها را در صف قرار می‌دهد. تک‌تک بررسی می‌کند. مسیر بهینه را که انتخاب کرد ارسال می‌کند. اشکالی که وجود دارد چون message ها می‌تواند بزرگ باشند می‌تواند در RAM بگذارد چون بفرستد، زیاد است. پس آنها را در hard ذخیره می‌کند و در صف می‌گذارد و این باعث کند شدن آن می‌شود.

روش سوم: Pocket Switching

تفاوت عمده این است که اندازه pocketها استاندارد و کوچک شده است و هر pocket یک مسیر را انتخاب می‌کند ← ممکن است پس و پیش هم بشود ولی ترتیبی که دارد چون کوچک است دومی RAM می‌تواند ذخیره بشود و پردازش شود.

دو نوع است:

۱- Data Gram : هر روتر با توجه به شرایط مسیر بهینه را انتخاب می‌کند. Pocket دوم و سوم

و ... می‌توان از مسیرهای دیگر منتقل شوند و هیچ ربطی به هم نداشته باشند. در واقع هیچ قیدی

رویشان نیست. میتوانند هم مسیر باشند یا خیر.

۲- Virfual Circuit : از خط اختصاصی استفاده می‌کند ولی pocket سوئیچ می‌کند مانند

موبایل‌های دیجیتالی. می‌خواهد اطلاعاتی از A به B بفرستد. به آن شماره می‌دهد. مانند ۱۰۰. همینطور در طول روترها مسیر بهتر برای ۱۰۰ معلوم می‌شود و الی آخر. بعد که کار تمام شد روترها مسیر را در حافظه دارند اگر اطلاعاتی از نوع ۱۰۰ بیاید از هما مسیر قبلی منتقل می‌کند. دیگر مبدأ و مقصد لازم نیست. و شماره مسیر لازم است. یعنی یک Setup اولیه می‌خواهد و دیگر مانند تلفن است و خط اختصاصی را می‌دهد. یعنی روترها از مسیرها مطلع هستند.

مقایسه:

۱- خرابی روترها در روش دوم تأثیر زیادی دارد ولی در اولی اگر خراب شود در مسیر دیگر می‌رود و اثر ندارد.

۲- در روش دوم حافظه RAM معرفی بیشتر است.

۳- در رومی ترافیک شبکه کمتر است و ظرفیت کمتری از شبکه را مصرف می‌کند. چرا که قبلاً یک آدرس مقصد داشت یعنی ۸ بایت آدرس ولی در این روش فقط شماره ارتباط است که حداکثر ۲ یا ۳ بایت است. <=> پس پکت‌ها کوچکترند و کمتر شبکه مصرف می‌کند.

۴- سرعت دومی بیشتر است زیرا در اولی باید تکتک برای انجام شود و مسیریابی کند. ولی در دومی همان مسیر قبلی را می‌رود.

سرورها دو نوع هستند <=> State : اطلاعات را نگه می‌دارد.

Stnteless : اطلاعات را نگه می‌دارد در خوابیدن سرور بهتر است چون چیزی ندارد که از بین برود.

۵- کنترل ترافیک در دومی بهتر است چون هر روتر می‌داند چند Connection از آن رد می‌شود و پارامتر خوبی است و هرچه بیشتر باشد یعنی سر آن شلوغ تر است ولی در اولی معیاری ندارد و گاهی ممکن است خلوت و گاهی شلوغ باشد. در نتیجه اگر شلوغی خط پارامتر بهینه باشد در انتخاب مسیر بهینه کمک می‌کند.

در Virtual cirenit مجازاً یک خط را ایجاد کرده است ولی واقعاً این طور نیست .

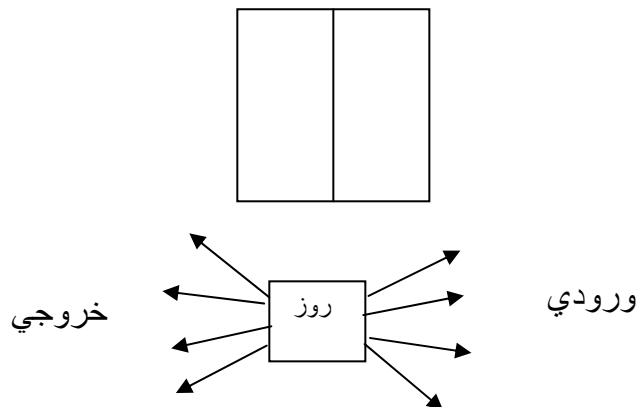
Multi plening چندین pocket از همان خط ارسال می‌شود.

وظایف روترها:

۱-تبدیل پروتکل‌ها ⇐ تبدیل پروتکل‌ها به یکدیگر است.

۲-کنترل ترافیک

۳-مسیریابی



مسیریابی:

Pocket که از ورودی می‌آید براساس جدول مسیریابی به یکی از خط‌های خروجی Switch می‌کنند. چون هرکدام محلی بهینه می‌کند مجموع آن یک مسیر بهینه است.

الگوریتم‌های مسیریابی:

۱-Static: ایستا: ساده‌تر ولی انعطاف ندارد.

۲-Dynamic: پویا: پیچیده‌تر ولی انعطاف دارد.

ایستا: مسیریابی که خود را با شرایط اصلاح نمی‌کنند و بازخورد ندارند بست به شرایط جدید خود را تغییر نمی‌دهند.

پویا: ویژگی آنها پویا بودن است. هر اتفاقی و حادثه‌ای اعم از خوب یا بد (آیا منبعی اضافه می‌شود یا کم) برای آنها مهم است و به آن پاسخ می‌دهند. و رفتار خود را بسته به آن متناسب می‌کنند.

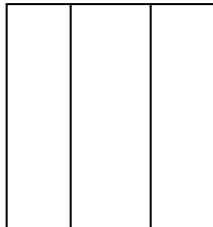
الگوریتم‌های ایستا:

۱) کوتاهترین مسیر Short test path :

گران شبکه را تعیین می‌کند. WPC را گره و ارتباط بین آنها را که اتصالات جهت‌دار است که هرچه دارد یک گران است جهت دارد ، وزن دار داریم. الگوریتم دیکسترا را روی آن قرار می‌دهیم و براساس گره‌ها و هزینه‌ها بهترین مسیر را انتخاب می‌کند.

بسته به هر خط خروجی هزینه را تا مقصد معلوم می کنند. خط خروجی توسط الگوریتم تعیین

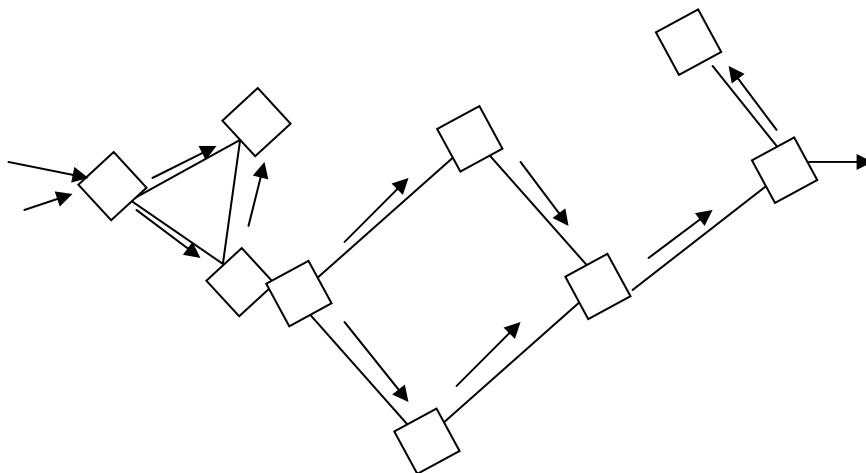
می شود.



(۲)مرز ایستا و پویا است flooding (مانند سیل جاری شدن)

یک الگوریتم خاصی یا کاربردهای خاصی است. هر روتر هرچه بهش برسد روی خطهای خروجی خو

پخش می کند . pocket مثل سیل جاری می شود و ممکن است دهها نسخه از آن موجود باشد.



مشکلات:

۱- ایجاد loop

۲- گم شدن پکت در شبکه

از مفهوم P.T.L (Time to life) که در TCP/IP است. برای حل این مشکل استفاده می کند.

یک فیلدی به نام T.T.L در pocket قرار می دهند. به ازاء هر پکت کد ارسال می شود طول بیشترین

هزینه ای که مصرف می شود تا به مقصد برسد را برای آن قرار می دهند. اگر به مقصد برسد هزینه کسر

می شود اگر + یا شد یعنی به مقصد رسید ولی اگر گم شده باشد و- می شود. و می فهمیم که سرگردان شده

است. وقتی بسته به مقصد رسید براساس هزینه میزان بهینگی رسیدن بسته می توانند بفهمند که از کجا آمده .

چون روترها مسیر را ثبت می‌کنند. این یکی از راه‌های کشف هک است. معمولاً پکت که اول می‌رسد بهینه‌تر است.

مزایا:

۱- بهترین مسیر که flooding مشخص می‌کند می‌تواند معیاری باشد برای سنجیدن سایر الگوریتم‌ها که اگر مثلاً یکی ۷۰٪ و دیگری ۲۰٪ مطابقت با آن داشت دومی بهتر است.

۲- برای کاربردهای حیاتی، مثل ایده اولیه اینترنت که جنگ بود و می‌خواستیم که اگر ردی بود حتماً به مقصد برسد.

۳- اگر خود روترها بخواهند با هم تبادل اطلاعات کنند باید همه با هم ارتباط داشته باشند، او نتیجه معمولاً از روش flooding استفاده می‌کنند.

دینامیک:

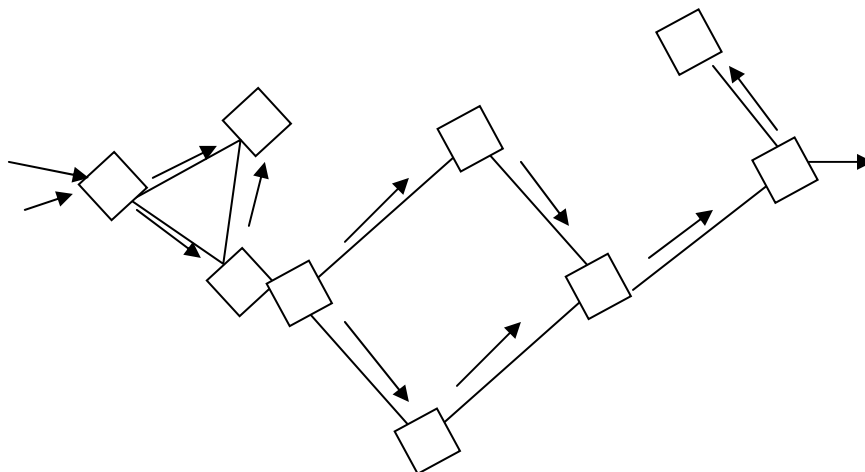
۱- Distance vector بردار فاصله 56k

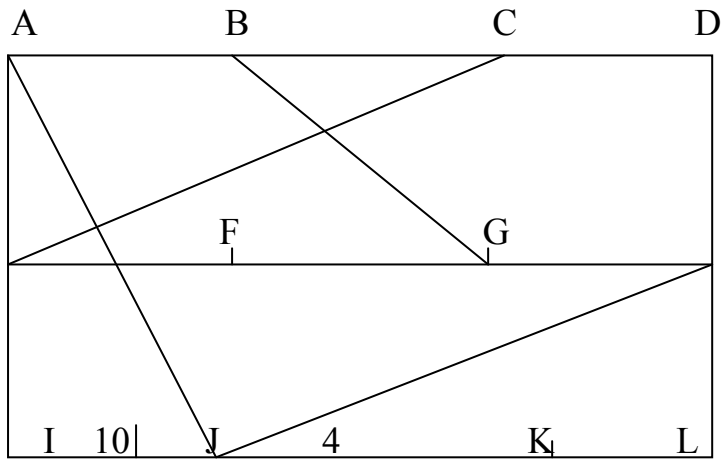
۲- huk State وضعیت اتصال

هر روتر اطلاعات مربوط به روترهای همسایه را بدست می‌آورد و هزینه رسیدن به آن را مشخص می‌کند. بعد از آن زمان را اسلات‌بندی می‌کنند. بعد از هر t ثانیه روترها اطلاعاتی را که از همسایه‌های خود بدست آورده‌اند با هم تبادل می‌کنند. در گام سوم، هر روتر که n همسایه دارد، تا n pocket دریافت می‌کنند با استفاده از n تاپکت که اطلاعات جداول همسایه‌ها است باید بتواند بهترین مسیر را پیدا کند.

هوب: حرکت یا جهش از یک روتر به روتر دیگر را گویند. پس هزینه حرکت از یک روتر به روتر همسایه را

یک هوب گویند.



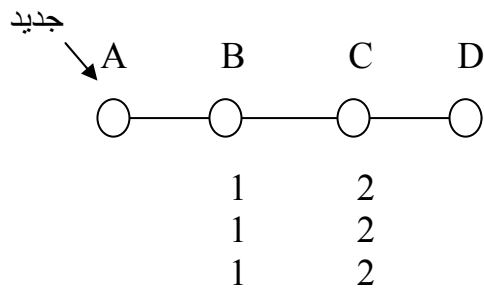


A	2	8
B	2	20
C	1	26
D	3	28
E	1	17
F	1	
G		
H	3	12
I	1	10
J		
K	4	6
L		

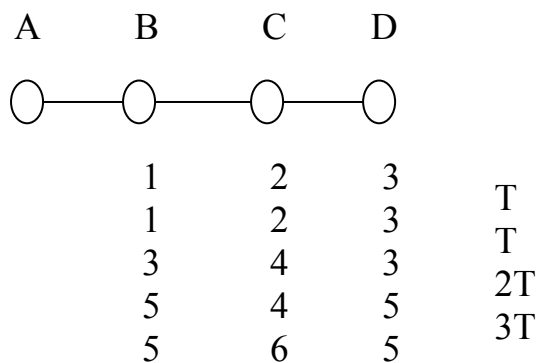
A	0	24	20	21
B	12	36	31	28
C	25	18	19	36
D	40	27	9	24
E	14	7	30	22
F	23	20	19	40
G	18	31	6	31
H	17	20	0	19
I	21	0	14	22
J	9	11	7	10
K	24	22	22	0
L	29	33	9	9

← می‌خواهیم بدانیم انتشار یک خبر خوب مانند اضافه‌شدن روتر در شبکه چگونه است.

کاری به جدول خودش ندارد و غیر از اطلاعات ثابت (همسایه‌ها) همه را دوباره می‌سازد.



← حال بینیم که خبر بد چگونه است . به سادگی انتشار خبر خوب نیست.



می‌بینید که خودش به A نمی‌رسد ولی با ۱ به C می‌رسد و هزینه A تا C هم ۲ است. یک هزینه خودش را ۳ می‌کند.

می‌فهمد که خط وجود ندارد ولی دیر متوجه می‌شود.

۲- در الگوریتم قبل هیچ جا به ماهیت خود خط و ظرفیت آن نمی‌پردازند. از پروتکل‌های معروف آن پروتکل

RIP در W NT که برای مسیریابی در NT پیاده‌سازی شده است. IPX, NOVEL و روترهای apple talk

SISCO, از این روش استفاده می‌کنند. مشکلات فوق باعث شد به سمت روش دوم پیش بروند. در این روش از

پنج گام استفاده می‌شود.

قدم اول تشخیص همسایگان و ادرس آنها

قدم دوم برآورد هزینه دسترسی به همسایه‌ها که براساس پارامتر بهینگی است.

قدم سوم ساختن یک pocket برای تبادل با روترهای دیگر

← تا این جا مانند روش قبل است و از این جا به بعد متفاوت است.

قدم چهارم: این pocket را برای تمام همسایه‌ها و همه روترهای شبکه می‌فرستد.

قدم پنجم: زمان ارسال آن و شاختش متنوع تر است.

به قدم دو که رسید که هزینه ارسال به همسایه‌ها را به دست آورده‌است و در گام سوم packet را آماده می‌کند و بعد آن را برای $(n-1)$ روتر دیگر موجود در شبکه ارسال می‌کند. برای ارسال به زمانها تنوع داده‌است.

چند رش وجود دارد:

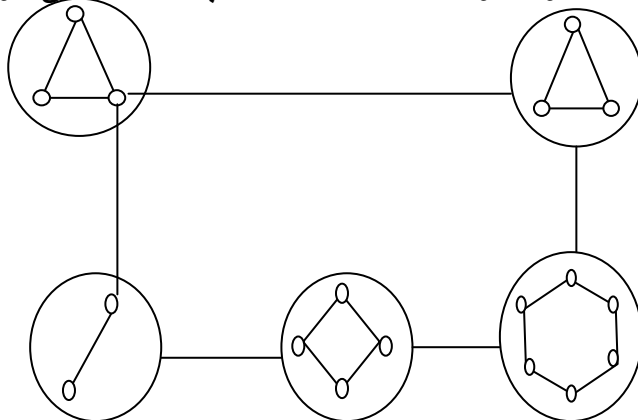
-یک روش آن است که هر t ثانیه یکبار می‌فرستد و کاری ندارد که در شبکه چه اتفاقی افتاده‌است و جداول را می‌فرستد. حال آن که اگر اتفاق خوب یا بد رخ نمی‌داد لازم بود و فقط بار شبکه را هدر می‌داد.

-روش دیگر آن است که هر وقت اتفاقی افتاد تبادل انجام یزد انعطاف بیشتری به زمانبندی‌های خود می‌دهد. حال که مشخص شد کی packet را ارسال می‌کند در گام چهارم packet را برای تمام روترهای شبکه ارسال می‌کند. در نتیجه هر روتر اطلاعات تمام روترهای دیگر شبکه را به‌ان می‌رسد در نتیجه می‌تواند گران جهت‌دار شبکه را بسازد و الگوریتم کوتاهترین مسیر را روی آن پیاده‌سازی کند. پس گام پنجم ساخت گران شبکه و پیاده‌سازی الگوریتم دیکسترا روی آن است.

← معروفترین این روترها OSPF است که الگوریتم معروفی است. دیگر IS-IS است که کمتر مورد استفاده است.

OSPF در اینترنت بسیار فراگیر دشه است.

مشکل روش‌های روتینگ: این است که اگر شبکه بزرگ شود، روترها از نظر حافظه دچار مشکل می‌شوند.



همان طور که می‌بینیم برای هر روتر باید یک جدول با ۱۸ سطر داشته باشیم.

A		
B		
C		
D		
E		

حال اگر تعداد روترها زیاد شود این جدول بزرگتر می‌شود.

برای رفع این مشکل از دید منطقه Zone استفاده می‌کنند. این منطقه از نظر فیزیکی است آنهایی که به هم وصل هستند یک منطقه را تشکیل می‌دهند. هر zone یک پیش شماره دارد ← شماره‌ها یکتا است.

هر منطقه با یک خط به مناطق دیگر وصل است. این مناطق از طریق روت‌های ورودی و خروجی به هم وصل هستند.

هر زودتر یکسری اطلاعات همسایه خود را دارد و دسترسی به ماطق دیگر یعنی به تکتک روزهای مناطق دیگر وصل نیست و فقط به دروازه ورودی و خروجی می‌تواند وصل شود.

A_2	1	2
A_3	1	1
B	2	2
C	3	2
D	3	2
E	2	2

← در این روش به جای 18 تا سطر 6 سطر داریم.

← تعداد سطرها از فرمول زیر بدست می‌آید:

(۱-) تعداد مناطق + تعداد همسایه‌ها

(۲-) به این روش مسیریابی سلسله‌مراتبی Hirarchinal Routing گویند.

۳- این روش خیلی بیشتر خطوط تلفن و شبکه‌های تلفن است ولی عیب در این است

که بعضی اوقات ممکن است مسیرها بهینه نباشد.

۴- مثلاً برای رسیدن به C6 جدول به ما می‌گوید که از خط ۲ برود و می‌شود ۵ ولی مسیر بهینه علت:

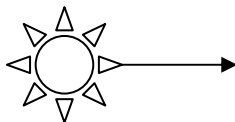
مسیر بهینه است ولی توپولوژی آن منطقه چون بهینه نیست، جواب بهینه شده است. در واقع اگر توپولوژی مناطق بهینه نباشد ممکن است گاهی جواب بهینه نباشد.

← در بدترین حالت تعداد سطرهای جدول ۱۰ تا است که مربوط به روتری است که در منطقه‌ای قرارداد که تعداد همسایه‌های آن از همه بیشتر است.

وظیفه دوم لایه شبکه (کنترل ترافیک)

خیلی از روش ما مانند کنترل ترافیک در خیابانهاست. اگر شبکه شلوغ شود، ازدحام یا ترافیک زخدلثسفهذ به وجود آمده است. بسیار مهم است چون اگر مشکل رادر یک جا حل نکنیم در تمام شبکه پخش می‌شود.

← می‌خواهیم بدهیم لایه شبکه با چه مفاهیمی برای کنترل ترافیک درگیر است. معمولاً سه فاکتور باعث به وجود آمده ترافیک می‌شود:



۱-پروسور ضعیف برای پروتکل

۲-بافر ضعیف برای پروتکل

۳-اگر در زمان خاصی تعدادی ورودی به یک روتر وارد شود و همه ب یک خروجی اعمال شوند ترافیک به وجود می‌آید.

← حتی بافر نامحدود باشد ممکن است بگوید که همه را می‌گیرم و در بافر قرار می‌دهم و به نوبت پردازش می‌کنم. این باعث می‌شود زمان پردازش بالا می‌رود. ← چون فرستاده ACK دریافت نکرده فکر می‌کند

گم شده و با آن را ارسال می‌کند. این خودش باعث شلوغ‌تر شدن شبکه می‌شود.

← دو علت عمده برای شلوغ شدن شبکه وجود دارد:

۱-کمبود منابع

۲-بالارفتن تقاضا

بنابراین راه کلی برای رفع ای مشکل وجود دارد عبارت است از:

۱-زیاد کردن منابع

۲-کم کردن تقاضا ← مثلاً روتر که فهمید که همسایه‌اش شلوغ است یک پارامتر منفی در هزینه‌های آن بیاورد و کمتر pocket برای آن ارسال کند.

معمولاً برای کنترل ترافیک از تئوری کنترل استفاده می‌کنند. این تئوری سیستم‌ها را دو نوع در نظر می‌گیرد:

۱-سیستم‌های open ۲-سیستم‌های close

open: طوری ساخته شده‌اند که ازدحام به وجود نیاید. در واقع حالت پیشگیری دارد: در نتیجه خیلی هم کاری به شبکه و کنترل آن ندارد چون طوری است که ازدحام رخ ندهد:

close: مدام به شبکه‌نگاه می‌کنند و feeol backy از شبکه‌می‌گیرند که آیا شلوغ است یا خیر. یعنی بیشتر دید درمانی دارد اگرچه سعی می‌کند طراحی هم خوب باشد.

در این سیستم‌ها سه مرحله وجود دارد:

۱-تشخیص ازدحام : روش اول: خودش اطلاع دهد روش دوم: همسایه‌اش بفهمد

۲-اطلاع‌دادن ازدحام به مراجعی برای رفع آن

۳-اصلاح عملیات سیستم برای رفع ازدحام

مثال) بعضی روترها اگر شلوغ باشند آنرا در شبکه broad cast می‌کنند و برای رفع آن همسایه‌ها برایش packet ارسال نمی‌کنند.

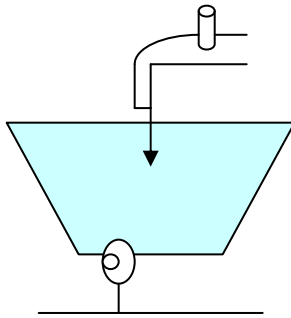
یکی از علل بوجود آمدن شبکه آن است که قابل پیش‌بینی نیست و شکل خاص و مدل ندارد .

← حال اگر این ترافیک شکل داشت می‌توانستیم طوری برنامه‌ریزی کنیم که آنرا حل کنیم. بنابراین یکی از الگوریتم‌های مناسبی برای آن الگوریتم Tercitic Shoping یا شکل‌داده به ترافیک است.

عملکرد می‌خواهد کاری کند که با ورودی به شبکه قابل محاسبه باشد. مثلاً می‌گوید در واحد ثانیه 5 تا

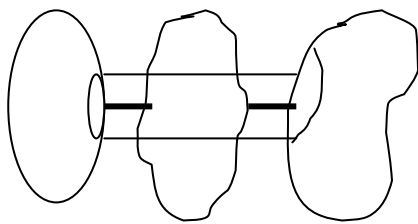
pocket بیشتر ارسال نمی‌شود. یکی از الگوریتم‌های این روش Leaky bucket است.

در این روش کاری کنیم که هر ایستگاه ظرفیت خروجی خاص خود را داشته باشد. اگر کمتر باشد هم باز در ثانیه 5 pocket می فرستد. زیاد هم باشد باز 5 تا می فرستد. و اگر زیاد باشد برای خود ایستگاه، مشکل پیش می آید و سرور می شود.



: routing

دو نقطه داریم که از نظر فیزیکی مجزا هستند. و می خواهیم آنها را به هم وصل کنیم.



راههای مختلفی وجود دارد.

۱- خطوط تلفن

۲- ماهواره، هزینه آن زیاد است و افراد خاصی می توانند .

۳- خط کشیدن که ممکن نیست.

ایده هایی که وجود دارد، روش tunuing یا تونل گذاری است که در یک بستره عمومی کمک پروتکل ها به طور اختصاصی استفاده می کنند.

Pocket روی بستره عمومی قرار می گیرد و در طرف مقابل تحویل گرفته شده و باز می شود اسم آن در پیاده سازی VPN= Virtual Private Network است که tuning نام تئوری آن است. یک روش با نفرینه قابل قبول برای اتصال سازمان ها است.

دیوار، آتش fire wal:

بیشتر در لایه Network مطرح می شود. به طور کلی هرگاه یک منبعی داشته باشیم که در آن اطلاعات است و بخواهیم آن را از دسترس افراد غیرمجاز بیرونی خارج نگه داریم روی آن forewol می گذاریم . به این دلیل می گوئیم منبع که دیوار آتش سطوح مختلف دارد. در واقع تا زمانی که در شبکه وصل نشده ایم مشکلی نیست

ولی وقتی وصل شدیم دیگر هیچ تضمینی نیست. شاید حتی ندانیم که از ما استفاده می شود. پس بحث مهم شبکه ها امنیت است.

کار دیواره آتش امنیت و محافظت از مجوزها است. که اطلاعات غیرمجاز از بیرون به داخل و در داخل به خارج منتقل شود. خود آنها سطوح مختلف دارند.

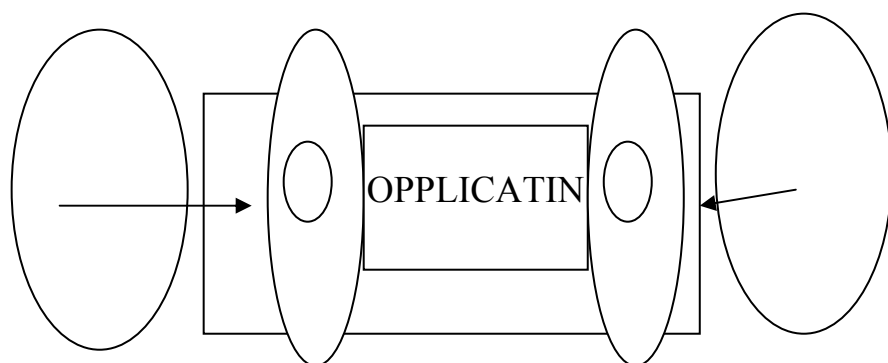
۱- تنها آدرس ها را چک می کنند که اگر غیرمجاز باشد اجازه نمی دهند.

۲- مفاهیم را بررسی می کنند.

۳- بعضی ها به سمت هوشی مصنوعی پیش می روند.

دو نوع کلی دارد: ۱- نرم افزاری ۲- سخت افزاری: مدارهایی که چک را انجام می دهند.

Fire wall از نظر ساختاری:



روزهای ورودی و خروجی که می تواند سخت افزاری یا نرم افزاری باشد که خیلی هوشمند نیستند و در حد آدرس یا متن است. و گاهی نمی توانند آن را حل کنند. برای رفع مشکلات یک نرم افزار احاطه کرده اند. هرچه بهرتر ورودی و خروجی بیاید از آن رد می شود و روی آن نرم افزار می گذارند. این نرم افزارها می توانند هوش مصنوعی هم باشد.

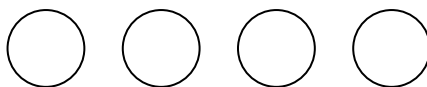
: IPV6

به عنوان یکی از پروتکل های مطرح دنیا وجود دارد. در اوایل برای پوشش شبکه arpanet به وجود آمد و با گسترش آن خودش فراگیر و استاندارد شد. و اکسون در LAN ها و VAN ها بکار می روند. به عنوان یک مثال می توان در مورد نسخه چهار صحبت کرد.

TCP/IP در شبکه:

هر ایستگاهی در شبکه برای اتصال به یک آدرس IP وجود دارد. حالا چه شبکه LAN باشد و چه VAN.

در LAN ها نیاز به ثبت آنها نیست و لازم نیست Valid,IP باشد.



--	--

از ۰ تا ۲۵۵ تغییر می‌کند.

که معمولاً ۱ و ۲۵۵ روروشده‌اند.

۵ کلاس IP داریم: C, B, A که دو تای دیگر مربوط به Multi cast است که حد فاصل PTP . Broad cast است.

← انواع ارتباطات: PTP ← یک فرستنده و یک گیرنده.

Broad cast ← یک فرستنده و n گیرنده

Multi cast ← که در شبکه تعدادی را گزینش می‌کند.

← ویژگی‌هایی که نسخه بعدی IP باید داشته باشد تا جوابگو باشد:

۱- ظرفیت بیشتر آدرس دهی ← در آینده هر انسان یک IP دارد و IP4 جوابگو نیست.

۲- در IP4 باید الگوریتم های سنگین error detection , error correction می‌داشت اما اکنون که باید پیشرفت فناوری خطا کم است و نیاز به الگوریتم ساده خطایابی است.

IP4 چون در محیط پرخطا به وجود آمدن در نتیجه Packet های آن بسیار بزرگ بود و Header آن بزرگ است. عیبی که وجود دارد:

۱- هدر دادن پهنای باند

۲- هدر دادن زمان پردازش روترها

اما IPV6 لازم نیست پروتکل الگوریتم سنگین داشته باشد.

۳- ساده تر کردن پروتکل ها ← پردازش روترها ساده تر است.

۴- امنیت باید بیشتر باشد.

ه-یک Host بتواند جایش عوض شود ولی IP آن عوض نشود. در واقع می‌خواهند پایه mobik را بگذارند.

← IPv6 مدتی است که آمده اما هنوز استفاده نشده است چون قبلی هنوز هست و احتمالاً زیاد از آن استفاده می‌کنند که TCP/IP بیاورد. چون هزینه تغییر زیاد است.

UOIPAND AMATEUY RADIO

فرستادن محوت از طریق پروتکل اینترنت

امروز بیماری از افراد از اینترنت به عنوان پلی برای فرستادن صوت به نقاط دور دست دنیا استفاده می‌کنند . البته این تبادل اطلاعات ترکیبی از اینترنت و فرستنده گیرنده VHF یا UHF FM است، نه آنچه که افراد از اینترنت تا به امروز انتظار داشتند .

امروزه آنتن های آماتور UOIP فراوانی در سراسر دنیا وجود دارند که از اینترنت جهت تبادل صوتی استفاده می‌کنند .بدین منظور از NEPEATER های متوالی در فاصله های طولانی استفاده می‌شود .وسیله دیگری که بدین منظور مورد نیاز است به نام SIMPLEX شناخته می‌شود که کره مرکزی بین چندین کاربر است .

بدین ترتیب افراد مختلف می‌توانند به سادگی با افرادی که کیلومتر ها از دستگاه فرستنده گیرنده FM آنها فاصله دارند تماس بگیرند .

در این مقاله به چندین نمونه از این نوع آنتن های آماتور اشاره شده است .

CCHOLINK :

CCHOLINK توسط JONATHAN در اوایل سال ۲۰۰۲ ساخته شده به سرعت در سراسر جهان مورد توجه عموم قرار گرفت .پس DOWNLOND کردن نرم افزار این دستگاه به صورت جهانی از آدرس WWW.ECHOLINKORG پس از اولین بار W، کردن دستگاه پس از اتصال به اینترنت به سایت ECHOLINK و به SORRER آن متصل می‌شود و پس از VALIDATE کردن شما در البته این کار تنها یک بار صورت می‌گیرد ما دقیقاً همانند یک SWITCHBORD در دنیای کامپیوتری امروز فعالیت خواهد کرد.

: LINK

سازنده این سیستم GRRRAEM BRANES می باشد روند کاری LINK بیمار به روند کاری EEHOLINK شبیه است با این تفاوت که کار بران LINK ملزم به استفاده از واسطه های رادئویی خاصی از جمله VA3TO یا ULIL هستند نرم افزار این سیستم را نیز به صورت رایگان از آدرس WWW.OACNET.NETRADIO دریافت کرد.

LINK از چندین SERRER جدا از هم استفاده می کند که هر یک مشابه S ERRER های ECHOLINK هستند. ولی استفاده از LINK در ماه های اخیر به شدت کاهش یافته است .

EQSO

هنگامی که سیستم EQSO توسط PANL DAVIE ساخته شد هدف اصلی آن تبدیل شدن به سیستمی بود که به صورت یک شبکه رادئویی جهانی عمل می کند .

این سیستم بر اساس DEDICATED SERRER و که گونه ای طراحی شده است که می توان به سادگی و از طریق یک کامپیوتر شخصی یا اتصالات رادئویی که به درگاه های R F معروف هستند از آن استفاده کرد. نرم افزار این سیستم را نیز می توان به صورت رایگان از آدرس WWW.EQSONET دریافت کرد.

IRLP

IRLP که مخفف THE INTERNET RODIO LINKING PROJECT می باشد سیستمی است که تنها راه دست یابی به آن استفاده از امواج رادئویی است.

مخترع این سیستم DARID COMERON است ولی اولین گیرنده های IRLP که ونکوور را به بر تیش کلمبا متصل می کرد توسط MICHEAL JLLING BY و SINPLEX طراحی شده است

بر خلاف دیگر سیستم ها این سیستم از نرم افزار های تحت INTERFACE,LIUT های سخت افزاری استفاده می کند .

نقشه ای IRLP وگره های خاص آن در آدرس STATUS.IRLP.NET موجود است. جهت اتصال به سیستم IRLP ابتدا کار بر باید خود را معرفی کرده و کد دست یابی DTMF را بفرستد. پس باید عدد ۴ رقمی را که آدرس گره مقصد وارد کنید.

WB8IMY ACCESSING NOD 5555 هنگامی که تماس برقرار می شود یک ID صوتی ارببر فرستاده خواهد شد. پس از دریافت تاثیر به ID صوتی کاربر می تواند تماس خود را برقرار کند.

WIRES.II

INTERNET REPEATER ENHONCEMENT SYSTEM شبکه VAIP است که توسط YAESU

طراحی و ساخته شده است و روش کار آن بیمار شبیه به RLP II است یا این تفاوت که WIRES تحت

WINDOW کار می کند. واسط سخت افزاری W IRES II دستگاه HRI-100 است که یک کامپیوتر

متصل به اینترنت با سرعت بالا وصل می شود و به عنوان واسط بین گره های و کامپیوتر کار می کند.

اگر چه این دستگاه نیز در شرکت YAESO ساخته شده است، ولی قابلیت اتصال به هر فرستنده گیرنده ای را دارد و دارای دو M ODE عملیاتی است .

SRE به اربران این مکان را می دهد تا به هر گره دیگری وصل شوند هم چنین همانند I RLP می توان از

D TME استفاده رد از صدا جهت دست یابی استفاده می کند سیستم F RE است با این سیستم میتوان به

تمام گره های موجود در جهان متصل شد و حتما می توان به طور همزمان بین ۱۰ گره کنفرانس ایجاد کرد .

UOIP INTEMET VERCE FCC COND UMER FACLS

UOIP سیستمی است ه به وسیله آن می توان به جای استفاده از تلفن های آنالوگ معمولی از کامپیوتر و اینترنت جهت ایجاد تماس های تلفنی استفاده رد .

UOIP سیگنال آنالوگ صوت را به سیگنال دیجیتال تبدیل می کند و اگر کار به یک تلفن آنالوگ متصل شود

این سیگنال مجدا به شکل سیگنال آنالوگ تبدیل خواهد شد. جهت استفاده از این سیستم برخی سرویس دهنده

ها این امکان را به کاربران خود می دهند تا از تلفن های معمولی متصل شوند. در این صورت نیازی به

دستگاه های اضافی نخواهد بود و تنها یک آداسپتور نیاز است . ولی برخی سرویس دهنده ها کاربران را

ملزم به استفاده از کامپیوتر می کنند که د راین صورت کاربر جهت اتصال به سیستم می باشد از یک کامپیوتر استفاده می کند که در این حالت نیازی به آداپتور نیست .این سیستم محدودیت خاصی ندارد و با استفاده از آن می توان به تلفن های معمولی یا موبایل ها یا تماس های راه دور یا بین المللی دست یابی داشت قیمت این تماس ها هر کدام توسط سرویس دهند ها شخص می شود.