

فناوري اطلاعات پیشرفته

تعریف فناوری اطلاعات : گردآوری، سازماندهی و فرآوری داده‌ها و دانش پراکنده در سطح دنیا است به گونه‌ای که بتوان از این دانش گردآوری شده، معرفت و دانش جدید تولید کرد.

شبکه های کامپیوتری : مجموعه‌ای از کامپیوترهای "مستقل" است که به نحوی با یکدیگر اطلاعات و داده مبادله می نمایند.

مفهوم استقلال در شبکه های کامپیوتری : هر ماشین میتواند حتی بدون حضور در شبکه کار کرده و از شبکه فقط برای تبادل داده ها استفاده کند.

تبادل داده از دیدگاه شبکه های کامپیوتری : کامپیوترها در شبکه بتوانند با یکدیگر داده رد و بدل کنند بدون آنکه نوع کانال اهمیت داشته باشد.

Link از دیدگاه شبکه : وقتی 2 یا چند دستگاه بر اساس قواعد از قبل مشخص و استاندارد داده ها را به روش قانونمندی نشانه گذاری و سازماندهی و سپس بین هم مبادله کنند link پدید آمده است که در حقیقت يك کانال مخابراتی است که جزییات تکنیکی آن مشخص و به صورت فراگیر استانداردسازی شده و هیچ نقطه مبهمی ندارد.

Web : روشی برای سازماندهی اطلاعات است به گونه ای که غیر از کنار هم قرار دادن متن، صدا، تصویر، فیلم و گرافیک، بزرگترین حسن آن سادگی دسترسی به اطلاعات پراکنده در دنیا، از طریق مفهومی به نام "ابریوند" است.

تفاوت اینترنت و web : اینترنت یک زیر ساخت برای ممل و جابجایی اطلاعات است که از 3 دهه قبل وجود داشته در حالی که وب معماری سازماندهی و دسترسی به اطلاعات توزیع شده در سطح جهان است که خود جزیی از اینترنت است و عمر آن مدود 15 سال است.

کاربردهای شبکه های کامپیوتری : 1- اشتراک منابع 2 - حذف محدودیتهای جغرافیایی در تبادل داده ها 3 - کاهش هزینه ها 4 - بالا رفتن قابلیت اعتماد سیستمها 5 - افزایش کارایی سیستم

تعریف منبع از لحاظ شبکه : هر موجودیت شامل سخت افزار و نرم افزار یا يك آیتم داده دارای هویت مثل يك فایل یا رکورد اطلاعاتی منبع یا Resource گویند.

تقسیم بندی شبکه از لحاظ سخت افزار شبکه : 1- تکنولوژی انتقال به این معنی که در شبکه از چه نوع کانالی به عنوان واسطه انتقال استفاده می شود. 2 - در تفکیک شبکه ها مقیاس بزرگی شبکه و ناحیه تحت پوشش آن است.

دسته بندی شبکه ها از دیدگاه تکنولوژی انتقال : شبکه های پخش فراگیر Broad cast : در این شبکه ها انتقال اطلاعات از طریق يك کانال فیزیکی که بین تمام ایستگاههای شبکه مشترك است انجام می شود. همه ایستگاهها

موظفند به طور دائم به خط گوش بدهند و برای ارسال نیز مجبورند اطلاعات را روی همین کانال منتقل نمایند . بنابراین در چنین شبکه هایی هر ایستگاه باید يك آدرس يكتا داشته باشد تا گیرنده پیام بتواند از بین پیام هایی که بین شبکه مبادله می شود پیام مربوط به خودش را تشخیص داده و برای پردازش های بعدی از روی کانال به حافظه اصلی از آن استفاده کند . شبکه های نقطه به نقطه Point to Point : بین دو ماشین در شبکه ، یک کانال فیزیکی و مستقیم وجود دارد و هیچ ماشین دیگری به آن کانال متصل نخواهد بود . عبارت ساده تر به یک کانال فیزیکی فقط و فقط دو ماشین متصل است .

معایب Broadcast : مدیریت پیچیده کانال - امنیت کم - کارایی پایین

پروتکل نظارت بر واسطه انتقال پروتکلی است که وظیفه آن این است که پیام ها را نوبت بندی کرده و سپس اقدام به ارسال نماید . ارسال همزمان دو ایستگاه در این نوع شبکه منجر به تصادم شده و در نتیجه داده های ارسالی خراب و فاقد اعتبار می شود .

شبکه ها کامپیوتری از نظر موقعیت جغرافیایی :

- 1 - شبکه PAN : این سیستم مانند سیستم های منزل است و برای کمتر از 10 متر کارایی دارد .
- 2 - شبکه LAN : برای يك یا دو کیلومتر کارایی دارد .
- 3 - شبکه MAN : موقعیت جغرافیایی این شبکه در محدوده شهر است و برای 100 تا 200 کیلومتر کارایی دارد .
- 4 - شبکه RAN : شبکه ایالتی یا استانی است .
- 5 - شبکه WAN : شبکه جهانی است .

نکته : اینترنت مالکی خاصی ندارد در صورتی که اینترنت مالک خصوصی دارد . پروتکل اینترنت و اینترنت یکی است .

نکته : هرگاه internet با حروف کوچک باشد بحث روی شبکه های داخلی است و هرگاه INTERNET با حروف بزرگ باشد بحث روی شبکه جهانی است .

محاسن شبکه های LAN :

- 1 - هزینه نصب و راه اندازی این نوع شبکه چندان بالا نیست.
 - 2 - در این نوع شبکه با توجه به محدود بودن تعداد ایستگاهها ، مدیریت شبکه آسانتر از بقیه شبکه ها است .
 - 3 - با توجه به کوتاه بودن طول کانال ، اولاً افت سیگنال کم و طبعاً نرخ خطا بسیار پایین است ثانیاً نرخ ارسال میتواند بسیار بالا باشد.
- نکته : تاخیر انتشار در کابل های فیبر نوری حدود 33 میکروثانی و در کابل های مسی یا رادیویی حدود 5 میکرو ثانیه به ازای هر کیلومتر خواهد بود .

انواع توپولوژی های LAN :

- 1 - توپولوژی خطی یا BUS : از رده خارج شده است .
- 2 - توپولوژی حلقه : ظاهرش Point to Point اما کارش به صورت Broad cast است . ارسال اطلاعات به صورت ساعتگرد یا پادساعتگرد است .
- 3 - توپولوژی ستاره یا Star
- 4 - توپولوژی درختی : مجموعه ای از چند توپولوژی star است . در این نوع توپولوژی ماشین ها در يك الكوي درختی به هم متصل می شوند . برگ های این درخت همان ماشین ها و گره های میانی عناصری مانند سوئیچ یا هاب هستند . این نوع توپولوژی در حقیقت از به هم پیوستن توپولوژی های ستاره پدید می آید که بسیار استفاده از آن رایج شده است .
- 5 - توپولوژی با اتصال کامل : در این نوع توپولوژی بین هر دو ماشین در شبکه يك کانال اختصاصی و مستقیم وجود دارد .

وب : یک روش معماری یا عبارتی یک نظام برای ذخیره سازی و دسترسی به مستندات به هم پیوندخوردهای است که روی هزاران هزار ماشین در کل جهان پراکنده و توزیع شده اند.

وظایف URL : (1) نام فایل یا منبع چیست (2) محل دقیق ذخیره شده فایل کجاست؟ (3) به چه روشی باید به فایل دسترسی داشت و طبق چه قاعده ای میتوان آن فایل را انتقال داد؟

IP : نام حوزه ماشین سرویس دهنده در آدرس URL معادل با آن جایگزین شود . استفاده از آدرس های IP به جای نام نمادین مرسوم نیست زیرا به راحتی به خاطر سپرده نمی شود ولی در مجموع امکان پذیر است و چون نیازی به ترجمه نیست طبیعتاً اندکی سریعتر عمل می کند .

معماری سیستم وب : کلاً از دیدگاه مسائل فنی ، سیستم وب در دو بخش سازماندهی میشود: الف) برنامه سمت سرویسدهنده وب و برنامه سمت مشتری وب ب) پایگاه اطلاعاتی توزیع شده از صفحات ابرمتن و فایل های داده مثل صدا ، تصویر و...

نکته : برای دسترسی به IP يك سایت بایست در نرم افزار Command Prompt دستور Ping به اضافه سایت مورد نظر را وارد نموده و جهت دسترسی به IP خودتان دستور IP Config در سطر فرمان وارد می نمایم .

تفاوت http و html : http یا پروتکل انتقال صفحات hyper text زبان یا قراردادی برای صحبت کردن مشتری با سرویسدهنده وب ولی html یا زبان نشانه گذاری صفحات hypertext زبانی برای قالببندی و صفحه آرائی اطلاعات متنی است .

به عنوان مثال : www.salman.ac.ir/ICT/student.html

عملیات انجام شده توسط پروتکل http :

1 - Get : http/1.1 /ICT/student.html

2 - Host : www.salman.ac.ir

opera/ 9.0 : User-agent - 3

: If modified-sence - 4

CRLF - 5

در شماره 1 : با فرمان Get و مشخص کردن موقعیت دقیق فایل و همچنین تعیین نسخه پروتکل HTTP تقاضای دریافت فایل داده می شود .

در شماره 2 : نام نمادین میزبان وب تعیین شده است .

در شماره 3 : ماهیت و نام برنامه مرورگری که این درخواست را تولید و ارسال کرده است .

در شماره 4 : مرورگر تمایل دارد صفحه مورد نظر را در صورتی دریافت کند که بعد از آخرین تاریخ مشاهده مورد نظر تغییر کرده باشد در غیر این صورت خودش نسخه قدیمی آن را نمایش می دهد .

در شماره 5 : طبق قرارداد باید خالی بماند (این سطر به معنای خاتمه درخواست تلقی می شود)

پاسخ ارسالی از سرویس دهنده وب :

http / 1.1 - 1

date - 2

server - 3

last modified - 4

content Length - 5

content type - 6

CRLF - 7

در شماره 1 : سرویس دهنده وب ضمن معرفی نسخه پروتکل يك كد سه رقمی که وضعیت اجرای فرمان را مشخص می کند به همراه يك رشته متنی قابل فهم مثل OK بر می گرداند .

در شماره 2 : تاریخ و زمان ارسال پیام آورده می شود .

در شماره 3 : ماهیت و نام سرویس دهنده اعلام می شود .

در شماره 4 : تاریخ آخریت تغییر در سند درخواستی .

در شماره 5 : طول سند .

در شماره 6 : نوع سند .

در شماره 7 : پس از ارسال يك خط خالي داده هاي سند درخواستي به صورت متني خط به خط به مشتري تسليم مي شود .

انواع متود ها در HTTP :

- (1) **Get :** مرورگر با ارسال اين متود از سرويسدهنده تقاضا ميكند كه يك صفحه وب يا يك فايل دودويي مثل فايل تصوير يا صدا برایش ارسال شود.
- (2) **متود Head :** اين متود از سرويسدهنده تقاضا ميكند كه فقط سرآيند صفحه وبی را كه نام آن در جلوی متود درج شده ، ارسال نمايد . اين متود چند کاربرد دارد : اول آنكه مشخصات صفحه وب ، شامل تاريخ آخرين تغيير ، عنوان صفحه ، نام تدوين كننده و صاحب اصلي آن و برخی از مشخصات اختياری كه در سرآيند صفحه وب درج شده ، ارسال ميشود و اين اطلاعات ميتواند برای مقاصدی همانند تهيه بانك اطلاعاتی از صفحات وب و طراحي جستجوگرهای وب مفيد واقع شود . دوم آنكه ميتوان با اين متود صحيح بودن يك URL و وجود يك صفحه وب را ارزيابی كرد
- (3) **متود PUT :** اين متود عكس عمل Get است يعنی مرورگر تقاضا ميكند كه يك صفحه وب را بر روی ماشين سرويسدهنده ذخيره نمايد . اين متود را سرويسدهندهائی حمايت ميكند كه بخواهند صفحات برخی از کاربران را دريافت كرده ضمن ذخيره ، آنها را در اختيار ديگران قرار بدهند .
- (4) **متود Post :** اين متود از سرويسدهنده تقاضا ميكند كه دادههائی را به يك منبع موجود (مثل يك صفحه وب يا يك فايل) اضافه كند . اين متود برای ايجاد "صفحات آزاد خبری" ، "تابلو اعلانات" ، محيطهای نظر خواهی يا ارسال اطلاعات برای يك پروسه تحت وب همانند برنامه های CGI مورد استفاده قرار ميگيرد
- (5) **متود Trace :** اين متود براي اشكال زدائي و بررسي سلامت سرويس دهنده کاربرد دارد . اين متود از سرويس دهنده مي خواهد كه درخواست را به همان شكلي كه دريافت كرده بازگرداند .

ساختار كلي درخواست در http :

پروتكل يا شناسه مورد استفاده نام فايل يا شي درخواستي نام متود : سطر درخواست

CRLF مقدار نام فيلد مورد نظر

CRLF مقدار نام فيلد مورد نظر

CRLF مقدار نام فيلد مورد نظر

CRLF

داده براساس درخواست مورد نظر

مثال :

Mozilla www.IRIB.ir/TV1/film.html

Get /TV1/film.html

Host : www.IRIB.ir

User – agent : moaila / 4.0

If modified-sence : آخرین تاریخ رجوع به این صفحه :

CRLF

پاسخ کلی درخواست در http :

متن رشته ای معادل کد سه رقمی (نشاندهنده نتیجه درخواست) نام شناسه پروتکل : سطر پاسخ

CRLF مقدار نام فیلد مورد نظر

CRLF مقدار نام فیلد مورد نظر

CRLF مقدار نام فیلد مورد نظر

داده های ارسالی

انواع کدهای وضعیت :

1 - xx 1: اطلاعاتی (پاسخی جهت آگاهی بیشتر مشتری

2 - xx 2 : عمل درخواستی موفقیت آمیز اجرا شده است

3 - xx 3: URL مورد تقاضا تغییر آدرس داشته است

4 - xx 4 : در تقاضای ارسال شده از طرف مشتری خطایی وجود دارد

5 - xx 5 : در سرویسدهنده خطایی داخلی رخ داده است

نشست (Session) : مجموعه عملیاتی که پس از برقراری يك ارتباط بین دو پروسه و با يك توافق اولیه آغاز و سپس با انجام يك سري تراکنش ادامه می یابد و بعد در روالی هماهنگ و مورد توافق ختم می شود .

Session ID : مشخصه اي جهت شناسايي يك کاربر مجاز صادر شده تا عمليات مورد درخواست او با موفقيت به سرانجام برسد .

Cookie : در بيان ساده اطلاعاتي است كه توسط فيلد set cookie از طرف سرويس دهنده براي مرورگر ارسال مي شود و مرورگر آنها را در جايي ذخيره مي كند تا در آينده آن را براي سرويس دهنده پس بفرستد تا گذشته اش را به خاطر بياورد . cookie معمولا يك فايل كوچك حداكثر تا 4kb يا يك فايل رشته متني است .

انواع فيلدهاي اطلاعاتي Cookie :

- 1 - **Domine :** مشخص كننده اين كه cookie از چه وب سايتي آمده است .
- 2 - **Path :** اين فيلد مسير دايركتوري هايي را كه اطلاعات وضعيت در آنجا ذخيره شده را مشخص مي كند .
- 3 - **Content :** تمام اطلاعات مورد نياز سرويس دهنده را براي حفظ حالت نشست و يادآوري گذشته آن شامل مي شود .
- 4 - **Expire :** اين فيلد مشخص مي كند كه اطلاعات cookie تا چه زمان معتبر است و بعد از آن اطلاعات وضعيت پاك مي شود و نشستي در كار نخواهد بود .
- 5 - **Secure :** مشخص مي كند كه مرورگر بايستي اين cookie را براي يك سرويس دهنده امن بفرستد چرا كه مثلا اطلاعاتي كه در خلال نشست مبادله مي شود بسيار حساس است .

Html : صفحات HTML متون غني شدهاي هستند كه اطلاعات موجود در يك سند را بصورت صفحهآراني شده و سازمانيفته ، در اختيار کاربر قرار ميدهند . HTML مجموعه اي از برچسب هاي خاص صفحه آراني و عوامل و اشيا ورود و خروج اطلاعات است .

ساختمان كلي يك صفحه HTML :

```
<HTML>
<HEAD>
<TITLE>
عنوان سند در اين ناحيه درج ميشود.
</TITLE>
</HEAD>
<BODY>
تمام اطلاعات صفحه وب در اين ناحيه درج ميشود.
</BODY>
</HTML>
```

مثال :

```

<html>
<head>
<title> salman </title>
</head>
<body>
<p align=" center ">
Salman
</p>
</body>
</html>

```

تگ های مختلف در HTML :

- (1) **P (پاراگراف)** : مرورگر پاراگراف را بر اساس عرض پنجره نمایش ، تنظیم میکند و در صورت لزوم جملات را شکسته و به خطوط بعد میبرد؛
- (2) **H1 تا h6** : جهت درشت یا نازک کردن نوشتار
- (3) **Bg color** : جهت رنگ پس زمینه و رنگ نوشتار به صورت زیر :

```
<body bg color =" " text=" ">
```
- (4) **نوع و اندازه قلم** :

```
<font face " " size=" " color=" " ">
```

 (تا نسخه 3.0 کاربرد داشت) در نسخه های بعدی به شیوه زیر عمل می شود :

```
<style="font-family: ;font-size: ;color:" ">
```
- (5) **Br** : معادل کلید Enter
- (6) **Hr** : جداسازی عناصر صفحه با یک خط افقی (

```
<hr width=" ">
```

)
- (7) **رسم جدول** : با تگ table می باشد و برای جداسازی سطر از tr و ستونها از td و خطوط بین جدول از border استفاده می شود . جهت تغییر رنگ هر خانه می توان در td از bgcolor استفاده کرد .
- (8) **Th** : اگر بخواهیم در جدول سطر یا ستونی به صورت سر تیتر باشد به جای td از th استفاده می شود .
- (9) **Caption** : عنوان بالای جدول
- (10) **لیست شماره دار** :

```
<ol><li></li></ol>
```
- (11) **لیست بدون شماره** :

```
<ul><li></li></ul>
```

 (جهت مشخص کردن پشت هر آیتم در تگ ul می توان type را : دایره توپر circle – مربع square – دایره تو خالی disc استفاده نمود)
- (12) **لیست تعریفی** : تگ این لیست dl است و dt جهت عنوان تعریف و تگ dd جهت متن تعریف است
- (13) **درج تصویر** : تگ مورد استفاده img است که داخل آن از

```
src="
```

 جهت آدرس تصویر و

```
alt="
```

 جهت متن تصویر و

```
align="
```

 جهت موقعیت مکانی تصویر استفاده می شود .
- (14) **Hyper link** : اتصال صفحات وب (

```
<a href=" "></a>
```

)
- (15) **Form** : جهت ورود اطلاعات به صفحه وب (

```
<form><input></form>
```

)

انواع صفحات در html :

- 1 - **صفحات ایستا** : آن دسته از صفحات وب که فقط ارائه کننده اطلاعات به کاربر هستند و هیچ تراکنشی با کاربر ندارند ایستا نامیده می شوند .
 - 2 - **صفحات پویا** : آن دسته صفحاتی که کاربر می تواند با يك صفحه وب در تراکنش باشد . به این معنا که قادر است در صفحه وب اطلاعاتی را وارد کند و آنرا به سمت سرویس دهنده ارسال نماید .
- تگ form : تگی که برای ایجاد صفحات وب استفاده می شود .
- ساختار کلی ورود اطلاعات در html :

```
<form>
```

```
<input>
```

```
<input>
```

```
.
```

```
.
```

```
.
```

```
</form>
```

انواع اطلاعات ورودی در فرم :

- 1 **متنی** : `<input type="text">`
- 2 **پسورد** : `<input type="password">`
- 3 **Check box** : انتخاب چند گزینه مثل علاقمندی ها در favorite پست الکترونیکی (`<input type="checkbox">`)
- 4 **radio** : انتخاب يك گزینه از بین گزینه ها (`<input type="radio">`)
- 5 **Drop-down** : با کلیک کردن لیستی باز می شود که از داخل آن کاربر يك گزینه را انتخاب می کند (`<form><select name=" "><option value=" "></option></select></form>`)
- 6 **Submit** : ارسال داده های وارده در قسمت فرم توسط کاربر به سرور (`<input type="submit">`)
- 7 **Reset** : پاک کردن اطلاعات وارد شده در فرم توسط کاربر و بازگشت فرم به حالت اولیه (`<input type="reset">`)

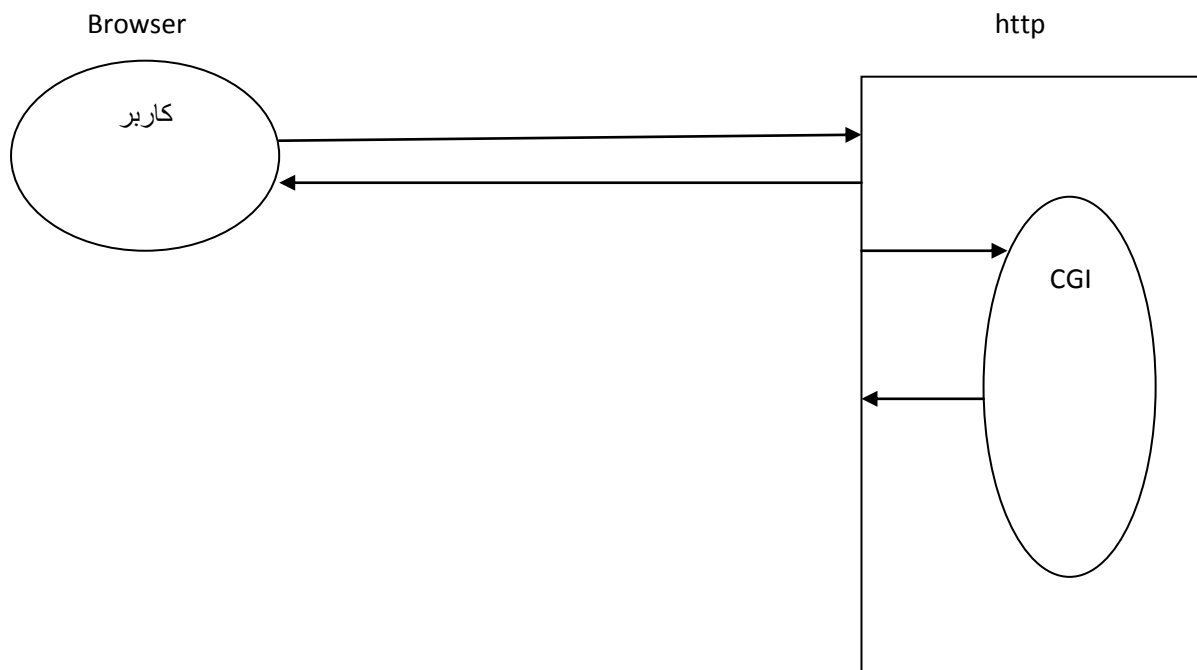
پارامترهای اساسی فرم :

- 1 **Action** : محل آدرسی که اطلاعات در آن ذخیره می شود (آدرس برنامه کاربردی که روی سرور است)
- 2 **Method** : متود های http است مه براساس داده های ارسالی یا دریافتی مشخص می شود .
- 3 **Enctype**

ویژگی های برجسته فرم :

- (1) **Action** : آدرس URL مربوط به محل قرار گرفتن برنامه CGI که اطلاعات ارسالی را دریافت و پردازش می کند .
- (2) **Method** : یکی از متود های http که توسط آن اطلاعات به سمت سرویس دهنده ارسال می شود را مشخص می کند .
- (3) **Enctype** : نوع کدگذاری داده های ارسالی را مشخص می کند . اگر این خصوصیت معرفی نشود اطلاعات ارسالی از نوع متن فرض خواهد شد .

CGI : استاندارد است که چگونگی ارتباط برنامه های جانبی با سرویس دهنده http را تبیین میکند. پروتکل http به تنهایی فقط قادر به ارسال و دریافت صفحات وب است و برنامه های CGI در کنار این پروتکل میتوانند یک ارتباط دوطرفه با کاربر ایجاد نمایند . (1- ارسال داده توسط کاربر 2- فراخوانی داده ها توسط CGI از http 3- ارسال پاسخ از CGI به پروتکل http 4- ارسال پاسخ از پروتکل http به کاربر)



- 1 - ارسال داده توسط کاربر
- 2 - فراخوانی داده ها توسط CGI از http
- 3 - ارسال پاسخ از CGI به پروتکل http
- 4 - ارسال پاسخ از پروتکل http به کاربر

□ برنامه های CGI را میتوان به زبانهای مختلفی نوشت و نیاز به ابزار خاصی ندارد. زبانهایی که امکان

نوشتن برنامه های CGI را فراهم آورده اند ، عبارتند از : C – C++ – Visual Basic – Shell Scripts
 Delphi – Aplle Script – UNIX) و

تفاوت های زبان برنامه نویسی CGI با زبان برنامه نویسی معمولی :

- 1 - داده های ورودی : در يك برنامه معمولی داده ها از طریق صفحه کلید یا موس دریافت می شود در حالی که برنامه های CGI ورودی های خود را از سرویس دهنده http دریافت می کنند .
- 2 - خروجی داده ها : که به صورت يك زبان نشانه گذاری شده (html) می باشد .

ساختار کلی Get :

http / 1.0 & مقدار = field name 1 ؟ آدرس برنامه مورد نظر

نکته : تمام فیلدها در متغیر Query String در برنامه CGI پروتکل http ذخیره می شوند .

پس از ارسال دستور Get سرویس دهنده http این رشته را دریافت می کند . برنامه مشخص شده را اجرا کرده و ادامه رشته را از کاراکتر ؟ به بعد در يك متغیر محیطی به نام Query String ذخیره می کند و سپس برنامه CGI متغیر های مورد نظر را فراخوان می نماید .

مفهوم مقیاس مجازی (VR) : شبیه سازی فضای سه بعدی توسط تکنیکهای نرم افزاری روی فضایی که ذاتاً

دو بعدی است؛ چگونه ای که کاربر میتواند به غیر از مشاهده این فضا ، در آن سیر کرده و در محیط تغییراتی را نیز ایجاد نماید.

تفاوت محیط های چند رسانه ای و VR : محیط های چندرسانه ای با داده هائی سروکار دارند که از قبل جمع

آوری و برنامه ریزی شده اند . اینها در حقیقت یک سری اطلاعات دسته بندی و ذخیره شده هستند که به صورت متوالی و با یک روال از قبل مشخص در یک فضای دو بعدی به نمایش درمی آیند . ولی محیط های حقیقت مجازی کاملاً محاوره ای است و قابلیت سازگاری و تغییر دائم دارد و کاربر میتواند اشیاء محیط را جابجا کند ، یا آنها را از زوایای مختلف نگاه کند .

تعریف زبان VRML : زبانی است که توسط آن ، هر شیئی در محیط VR با علائم مخصوص و گرامر خاص یک

زبان نشانه گذاری بصورت استاندارد ، مدل میشود . یعنی VRML قالب استاندارد فایلی است که درون آن اطلاعات مربوط به اشیاء محیط VR بصورت متنی تعریف میشود .

مراحل نمایش فایل متنی VRML :

- 1 - خواننده شدن فایل متنی VRML و استخراج تك تك اشیاء و پارامترهای آن و تبدیل آن به يك ساختمان داده مناسب برای تغییر و نمایش . (این قسمت از کار را برنامه مرورگر انجام می دهد)
- 2 - تحویل داده ساختمان داده تشکیل شده برای اشیاء به برنامه ای که آن را پردازش و تفسیر کرده و نهایتاً نمایش می دهد (به این موتور تفسیر یا Rendering Engine گفته می شود)

اصول VRML : پس از بار شدن یک فایل VRML مرورگر آنرا پویش کرده و مجموعه ای از ، ساختمان داده ها را

در حافظه تولید میکند . پس از تشکیل ساختمان داده های لازم ، مرورگر "موتور تفسیر" را برای نمایش اشیاء فرا می خواند .

وظایف موتور جستجو : محاسبات و پردازش لازم - رسم تصاویر

□ موتور تفسیر را میتوان پروسه ای در نظر گرفت که کارش رسم تصاویر محیط VR با استفاده از ساختمان داده حاصل از پویش فایل VRML می باشد .

خصوصیات زبان VRML :

- 1 - کاملاً متنی
- 2 - شی گرایی
- 3 - ارث بری پدر و فرزند
- 4 - برداری

محاسبات و پردازش های موتور جستجو جهت رسم يك شي :

- 1 - بررسی موقعیت يك شي نسبت به اشیا دیگر
- 2 - محاسبه درخشندگی شي
- 3 - محاسبه سطوح قابل رویت
- 4 - محاسبه پرسپکتیو يك شي : با توجه به آنکه صفحه نمایش تخت و دو بعدی است ولی تصاویر توصیف شده در فایل VRML ذاتاً سه بعدی هستند ، موتور تفسیر باید به نحوی مختصه Z از هر شیئی را روی صفحه دو بعدی X-Y بنگارد ، تا جسم دوبعدی توسط چشم ، سه بعدی تجسم شود.

اجزای يك فایل VRML :

- 1 - خط سرآیند (#VRML V1.0 ascii)
- 2 - خط توضیح (با علامت # شروع می شود)
- 3 - خط اریه نودها یا اشیا (NodeType { Fields })
- 4 - فیلدها (در تعریف یک گره به صورت بالا ، مقادیری که بین {...} قرار میگیرد ، توصیف آن گره است و چگونگی تفسیر یا نمایش یک شیئی را تعیین می نماید.)

قالب یک شیئی در VRML :

```
Separator {
  Transformations Definition
  Surface properties
  Shapes
  Children
}
```

□ بنیان ساختمان داده اشیاء در VRML درخت معکوس است که سلسله مراتبی از گره ها را تعریف میکند. (دایره : separator – قش : ارث بری خصوصیات – مربع قش : children)

تبدیلات گره :

- 1 - **center** : نقطه‌های را در فضای سه‌بعدی تعریف میکند که عملیاتی نظیر چرخش اشیاء حول آن انجام خواهد شد.
- 2 - **scalingFactor** : ضرائب انقباض یا انبساط اشیاء را در راستای سه محور مختصات تعریف می کند .
- 3 - **scaleOrientation** : با این تبدیل ابتدا محور مختصات به اندازه a حول مرکز پیش فرض میچرخد و سپس در سه محور مختصات ، با ضرائب $x y z$ منقبض یا منبسط میشود.
- 4 - **Rotation** : این تبدیل اشیاء را به اندازه a رادیان حول بردار $(x y z)$ می چرخاند .
- 5 - **translation** : انتقال اشیا به اندازه x و y و z داده شده صورت می گیرد .

```
Transform {
  Center x y z
  scalingFactor x y z
  scaleOrientation x y z a
  rotation x y z a
  translation x y z
}
```

اشکال هندسی در VRML :

الف (کره :

```
Sphere {
  radius r }
```

ب (مخروط :

```
Cone {
  bottomRadius r
  height h
}
```

ج (استوانه :

```
Cylinder {
  bottomRadius r }
```

د) مکعب مستطیل :

Cube {
width w
height h
depth d }

material : برای رنگ آمیزی سطح اشیا در VRML می توان از گره MATERIAL استفاده کرد . با استفاده از material علاوه بر رنگ سطوح می توان ویژگی های درخشندگی و شفافیت و خصوصیات دیگر سطح اشکال را توصیف کرد .

Material {

Diffuscolor R G B }

الگوهای آدرس دهی در اینترنت : پروتکل آدرس نویسی در اینترنت را پروتکل اینترنت یا IP می نامیم . این پروتکل يك الگوي آدرس دهی الحاقی است . با استفاده از این پروتکل می توان ساختاري داشت که ساختار آن امکان ایجاد سیستم ارتباطی توسعه پذیری نظیر سیستم پستی را فراهم می کند .

IPv4

تعداد کامپیوتر	بیت	بایت
	8	1
	16	2
	24	3
2	32	4

آدرس IP : اطلاعات ارزشمندی را در مورد نقشه شبکه و محل يك ماشین در شبکه اینترنت با خود حمل می کند . مسیریابیها بر اساس این آدرس به سرعت موقعیت ماشین مقصد را یافته و بسته را به سمت آن هدایت می کنند . ساختار يك آدرس Ip سلسله مراتبی است .

آدرس ماشین میزبان / آدرس زیر شبکه / آدرس شبکه

IANA & Internic : برای اطمینان از یکتا بودن آدرس های IP برای ارتباطات عمومی مرکز Internic کنترل و نظارت بر روی آدرس های IP را بر عهده دارد . IANA نیز قدرت اجرایی و تصمیم گیری برای اختصاص آدرس های IP منحصر به فرد را فراهم کرده است .

Network Number	Host Number
----------------	-------------

با ارزشترین بیت در آدرس IP بیت سمت چپ است که مشخص کننده کلاس IP می باشد .

کلاس های آدرس IP :

1 - **کلاس A :** پرارزشترین بیت از آدرس ، مقدار صفر دارد و این بیت کلاس A را از دیگر کلاسها متمایز

میکند؛ 7 بیت بعدی " مشخصه آدرس شبکه " و سه بایت باقیمانده ، آدرس ماشین میزبان را تعیین میکند .

(1-0-0-1 تا 126-255-255-254)

2 - **کلاس B :** دو بیت پرارزش از آدرس IP مقدار 10 داشته باشد و 14 بیت باقیمانده از 2 بایت سمت چپ ،

آدرس شبکه را تعیین میکند و دو بایت اول از سمت راست (16 بیت) آدرس ماشین میزبان خواهد بود .

(128 تا 191)

- 3 - **کلاس C** : مناسب ترین و پرکاربردترین کلاس از آدرسهای IP است . در این کلاس ، سه بیت پرارزش دارای مقدار 110 است و 21 بیت بعدی از سه بایت سمت چپ برای تعیین آدرس شبکه مورد نظر بکار رفته است . در این کلاس میتوان حدود دو میلیون شبکه را در جهان آدرس دهی کرد و هر شبکه میتواند تا 254 عدد ماشین میزبان تعریف نماید . برای تشخیص آدرس ها کلاس C به عدد سمت چپ از آدرس IP که به صورت دهدهی نوشته شده است نگاه کنید . اگر عدد بین 192 تا 223 بود آن آدرس از کلاس C خواهد بود .
- 4 - **کلاس D** : در این کلاس ، چهار بیت پرارزش دارای مقدار 1110 است و 28 بیت باقیمانده از کل آدرس برای تعیین آدرسهای "چند مقصده"(آدرسهای گروهی) است . از این آدرسها برای ارسال یک دیتاگرام به طور همزمان برای چندین ماشین میزبان کاربرد دارد و بمنظور عملیات رسانه ای و چند پخش می رود (رقم سمت چپ بین 224 تا 239)
- 5 - **کلاس E** : فعلاً این دسته از آدرسها که پنج بیت پرارزش آنها در سمت چپ 11110 است کاربرد خاصی ندارند و برای استفاده در آینده بدون استفاده رها شده اند . (رقم سمت چپ بین 240 تا 256)

آدرسهای خاص :

- 1 - **آدرس 0.0.0.0** : هر ماشین میزبان که از آدرس IP خودش مطلع نیست این آدرس را بعنوان آدرس خودش فرض میکند . البته از این آدرس فقط به عنوان آدرس مبدا و برای ارسال یک بسته میتوان استفاده کرد و گیرنده بسته نمیتواند پاسخی به مبدا بسته برگرداند .
- 2 - **آدرس 0.Host ID** : زمانی به کار میرود که ماشین میزبان ، آدرس مشخصه شبکه ای که بدان متعلق است را نداند . در این حالت در قسمت NetID مقدار صفر و در قسمت HostID شماره مشخصه خود را قرار میدهد .
- 3 - **آدرس 255.255.255.255** : برای ارسال پیامهای فراگیر برای تمامی ماشینهای میزبان بر روی شبکه محلی که ماشین ارسال کننده به آن متعلق است .
- 4 - **آدرس NetID-255** : برای ارسال پیامهای فراگیر برای تمامی ماشینهای یک شبکه راه دور که ماشین میزبان فعلی متعلق به آن نیست . آدرس شبکه مورد نظر در قسمت NetID تعیین شده و تمامی بیتهای قسمت مشخصه ماشین میزبان 1 قرار داده میشود . البته بسیاری از مسیر یابها برای مصون ماندن شبکه از مزاحمتهای بیرونی ، چنین بسته هایی را حذف میکنند .
- 5 - **127.xx.yy.zz** : بعنوان " آدرس بازگشت " شناخته میشود و آدرس بسیار مفیدی برای اشکالزدایی از نرم افزار میباشد . به عنوان مثال اگر بسته ای به آدرس 127.0.0.1 ارسال شود ، بسته برای ماشین تولیدکننده آن بر خواهد گشت 1 ؛ در این حالت اگر درست و بدون اشکال نصب شده باشد فرستنده بسته باید آنرا TCP/IP نرم افزارهای مجدداً دیافکت کند .

محدوده های IP های شخصی :

دامنه	کلاس
10-0-0-0 to 10-255-255-255	A
172-16-0-0 to 172-31-255-255	B
169-254-0-0 to 169-254-255-255	B
192-168-0-0 to 192-168-255-255	C

ساختار يك بسته IP : 32 بیتی است :

<i>Version</i>	<i>IHL</i>	<i>Type of Service</i>	<i>Total Length</i>
<i>Idenfication</i>			<i>Fragment Offset</i>
<i>TTL</i>	<i>Protocol</i>		<i>Header Check Sum</i>
<i>آدرس مبدا</i>			
<i>آدرس مقصد</i>			
<i>Option</i>			
<i>Pay Load</i>			

- 1 - Version : اولین فیلد 4 بیتی در قسمت شناسه يك بسته IP نسخه پروتکل IP که براساس آن سازماندهی و ارسال شده است آورده می شود . بدیهی است اگر در این فیلد عدد 0100 آورده شود مشخص کننده استفاده ما از IP نسخه 4 می باشد و اگر از عدد 0110 استفاده شود مشخص کننده IP نسخه 6 است .
- 2 - (IP Header Length) IHL : این فیلد هم 4 بیتی است و طول کل سرآیند بسته را بر مبنای کلمات 32 بیتی مشخص می نماید . اگر به ساختار يك بسته IP دقت شود به غیر از فیلد option که اختیاری است وجود تمام فیلدهای شناسه الزامی است . از آنجاکه طول قسمت اجباری شناسه 20 بایت می باشد حداقل عددی که در این قسمت می تواند قرار گیرد عدد 5 (0101) می باشد و حداکثر عددی که می تواند در این قسمت قرار گیرد عدد 15 (1111) می باشد که مشخص کننده 60 بایت برای قسمت شناسه ip می باشد که با توجه به 20 بایت اجباری برای قسمت اول 40 بایت به قسمت option تعلق دارد . در قسمت اختیاری شناسه اطلاعاتی مثل آدرس هاس مسیرهای پیچیده شده آورده می شود .
- 3 - Type of service : این فیلد 8 بیتی توسط ماشین میزبان تقاضای سرویس ویژه ای را برای ارسال يك دیتاگرام می نماید . به عنوان مثال ممکن است ماشین میزبان بخواهد دیتاگرام صدا یا تصویر برای ماشین مقصد ارسال نماید در چنین شرایطی از زیر شبکه تقاضای ارسال سریع و به موقع اطلاعات را دارد و نه

قابلیت اطمینان 100 % و اگر بسته ارسالی مثل يك نامه الكترونيكي باشد در اینجا اطمینان 100 % از زیر شبکه مهم است و سرعت تاثیر چنداني بر کیفیت کار ندارد . ساختار type of service به صورت زیر قابل بیان است :

			D	T	R	-	-
--	--	--	---	---	---	---	---

- بیت های P_1 & P_2 & P_3 (بیت های حق تقدم) :** این سه بیت در سمت چپ اولویت بسته IP را تعیین می کنند . اگر در این سه بیت صفر قرار گرفته شود بسته اطلاعاتی از نوع معمولی تلقی می شود و اگر عدد 111 قرار گیرد این بسته دارای بالاترین حق تقدم می باشد و معمولاً برای ارسال بسته های اطلاعاتی که به منظور تنظیم و بیکربندی شبکه استفاده می شود مورد استفاده قرار می گیرد .
- بیت های R.T.D :** بیت D به معنای تاخیر و R به معنای قابلیت اطمینان و T به معنای ظرفیت خروجی خط است و ماشین میزبان با قرار دادن يك در این بیت ها انتظارش را از زیر شبکه بیان می کند .
- 4 - Total Lenght :** در این فیلد که 16 بیتی است طول کل بسته IP که شامل مجموع اندازه قسمت شناسه و ناحیه داده می باشد تعیین می گردد . مبنای طول بر حسب بایت می باشد و بنابراین حداکثر طول کل بسته IP ارسالی می تواند 64Kb باشد ولی در عمل اینگونه نیست و اندازه اغلب بسته ها زیر 1500b می باشد .
- 5 - Indenfication :** برخی از مواقع مسیریابها یا ماشین های میزبان مجبورند يك دیتاگرام را به قطعات کوچکتر شکسته و ماشین مقصد مجبور است آنها را بازسازی کند بنابراین وقتی يك دیتاگرام واحد شکسته می شود باید شناسه ای داشته باشد تا در هنگام بازسازی آن در مقصد بتوان قطعات آن دیتاگرام را از بقیه جدا و سر هم نمود .
- 6 - Fragment offset :** دارای سه قسمت زیر است
- الف) بیت DF :** با يك شدن این بیت در يك بسته IP هیچ مسیریابی حق ندارد آنرا قطعه قطعه کند چرا که مقصد به هر دلیل قادر با بازسازی دیتاگرام های تکه تکه شده را ندارد اگر این بیت به يك تنظیم شده باشد و يك مسیریاب به دلیل بزرگی اندازه بسته نتواند آنرا انتقال دهد لاجرم حذف خواهد شد .
- ب) بیت MF :** این بیت مشخص می کند آیا بسته IP آخرین قطعه از يك دیتاگرام محسوب می شود یا نه . در آخرین قطعه از يك دیتاگرام بیت MF صفر خواهد بود و در بقیه الزاماً يك است .
- ج) بیت fragment offset :** این زیر فیلد 13 بیتی شماره ترتیب هر قطعه را در دیتاگرام شکسته شده مشخص می کند . با توجه به 13 بیتی بودن این زیر فیلد يك دیتاگرام حداکثر می تواند به 8192 تکه تقسیم شود .
- 7 - TTL (time to live) :** این فیلد 8 بیتی در نقش يك شمارنده طول عمر بسته را مشخص می کند . طول عمر يك بسته به طور ضمنی به طور ضمنی به زمانی اشاره می کند که يك بسته IP می تواند روی شبکه سرگردان باشد .
- 8 - Protocol :** دیتاگرامی که در فیلد داده از يك بسته IP حمل می شود با ساختمان داده خاص از لایه بالاتر تحویل پروتکل IP شده تا روی شبکه ارسال شود بنابراین فیلد شماره پروتکلی است که در لایه بالاتر تقاضای ارسال يك دیتاگرام را کرده است (8 بیتی)
- 9 - Header check sum :** این فیلد 16 بیتی به منظور کشف خطاهای احتمالی در شناسه هر بسته IP استفاده می شود .

تعریف الگوریتم مسیریابی : الگوریتم مسیریابی يك فرمول یا پروسه ریاضی است که با اعمال آن روی جداول مسیریابی میزان مطلوبیت و یا میزان بهینگی مسیرهای مختلف محاسبه و بر اساس آن مطلوبترین و بهینه ترین مسیر بدست می آید .

الگوریتم های مسیریابی : ترسیم گراف – انتخاب نوع الگوریتم

توپولوژی شبکه : مجموعه مسیریابهای و لینک های فیزیکی مابین آنها در زیرساخت ارتباطی يك شبکه توپولوژی آن شبکه است .

ترافیک شبکه : به تعداد متوسط بسته های اطلاعاتی که در واحد زمان روی يك لینک ارسال یا دریافت می شود .

Hop (گام) : به عبور يك بسته از يك مسیریاب گام و به تعداد مسیریابهایی که يك بسته در طی مسیر از مبدا به مقصد می پیماید .

ازدحام : زمانی که تعداد متوسط بسته های ورودی به يك مسیریاب از تعداد متوسط بسته های خروجی آن بیشتر شود ازدحام رخ داده است و تاخیر ارسال بسته ها در آن مسیریاب شروع به افزایش خواهد کرد .

انواع الگوریتمهای مسیریابی :

- 1 - از دیدگاه روش تصمیمگیری و میزان هوشمندی الگوریتم : ایستا و پویا
- 2 - از دیدگاه چگونگی جمعآوری و پردازش اطلاعات زیرساخت ارتباطی شبکه : سراسری متمرکز (LS) و غیرمتمرکز توزیع شده (DV)

ایستا : هیچ اعتنایی به شرایط توپولوژیکی و ترافیک لحظه ای شبکه نمی شود. معمولاً در این الگوریتم ها برای هدایت يك بسته ، هر مسیریاب از جداولی استفاده میکند که در هنگام برپایی شبکه تنظیم شده و در طول زمان ثابت است. در هنگام وقوع هرگونه تغییر در توپولوژی زیرساخت شبکه ، این جداول باید توسط مسئول شبکه بصورت دستی مجدداً تنظیم شود. اگرچه این الگوریتم ها بسیار سریعند ولی چون ترافیک لحظه ای شبکه متغیر است ، نمیتوانند بهترین مسیرها را انتخاب نمایند و هرگونه تغییر در توپولوژی زیرساخت ارتباطی شبکه ، یک مشکل عمده و جدی ایجاد خواهد کرد.

پویا : مسیریابی بر اساس آخرین وضعیت توپولوژیکی و ترافیک شبکه لحاظ می شود و جداول مسیریابی در این نوع الگوریتم ها هر T ثانیه یکبار به هنگام می شود . این الگوریتم ها بر اساس وضعیت فعلی شبکه تصمیم گیری می نمایند ولی ممکن است پیچیدگی این الگوریتم ها به قدری زیاد باشد که زمان تصمیم گیری برای انتخاب بهترین مسیر ، طولانی شده و منجر به تاخیرهای بحرانی شده و نهایتاً به ازدحام بیانجامد .

سراسری متمرکز : هر مسیریاب باید اطلاعات کاملی از زیرساخت ارتباطی شبکه داشته باشد. یعنی هر مسیریاب باید تمامی مسیریابهای دیگر ، ارتباطات بین آنها و هزینه هر خط را دقیقاً شناسایی نماید. سپس با جمع آوری این اطلاعات " ساختمان داده " مربوط به گراف زیرساخت شبکه را تشکیل بدهد.

الگوریتم LS : الگوریتم هایی که برای مسیریابی به اطلاعات کاملی از زیرساخت شبکه و هزینه ارتباط بین هر دو مسیریاب نیازمندند .

غیر متمرکز توزیع شده : مسیریاب اطلاعات کاملی از زیرساخت شبکه ندارد بلکه فقط قادر است هزینه ارتباط با مسیریابهایی که بطور مستقیم و فیزیکی با آنها در ارتباط است محاسبه و ارزیابی نماید. سپس در فواصل زمانی منظم ، هر مسیریاب جدول مسیریابی خود را برای مسیریابهای مجاور ، ارسال می نماید.

روش ارسال سیل آسا : این روش که برای ارسال بسته های همگانی (فراگیر) کاربرد دارد سریعترین الگوریتم برای ارسال اطلاعات به یک مقصد در شبکه به شمار میرود. طریقه ارسال در این روش آنست که هر مسیریاب با دریافت اینگونه بسته ها موظف است آنرا روی تمامی مسیرهای خروجی خود (به غیر از مسیری که بسته را از آن دریافت کرده است) ارسال نماید.

مشکلات روش سیل آسا : اگر قاعده بر این باشد که همه مسیر یابها یک بسته نوع فراگیر را روی تمامی خروجیهای خود ارسال نمایند ، ممکن است پس از چند لحظه خودشان آن بسته را دریافت کرده و چون مجدداً آنرا روی خروجیهای خود ارسال میکنند این عمل تا بی نهایت ادامه خواهد یافت و در یک دور باطل کل شبکه از این بستهها پر شده و روند ارسال هیچگاه متوقف نمیشود و عملاً شبکه از کار خواهد افتاد .

راه حل مشکل روش سیل آسا :

1 - قرار دادن شماره شناسایی برای هر بسته : در این روش برای هر بسته فراگیر ، یک شماره منحصر به فرد و یکتا درج میشود و مسیریابی که بستههای را یکبار دریافت کند شماره آنرا در جدولی ثبت مینماید . با دریافت یک بسته فراگیر ، شماره شناسایی آنرا در جدول جستجو میکند و در صورت وجود ، آنرا حذف مینماید .

2 - قرار دادن طول عمر برای بسته ها : در این روش یک فیلد شمارنده در سرآیند بسته قرار داده میشود و به ازای عبور بسته از هر مسیر یاب یکی از آن کم شده و وقتی این شماره به صفر رسید آن بسته از شبکه حذف خواهد شد .

مراحل الگوریتم LS :

- 1 - مسیر یابهای مجاور خود را که بصورت فیزیکی به آنها متصل است شناسایی کرده و آدرس آنها را بدست آورد. (ارسال بسته hello pack و دریافت آدرس مسیر یابهای مجاور)
 - 2 - تاخیر (یا بطور کلی هزینه) مسیر یابهای مجاور خود را اندازه گیری نماید. (یافتن هزینه مسیر یابهای مجاور)
 - 3 - یک بسته بسازد و تمام اطلاعاتی که از مسیر یابهای مجاور خود دارد در آن قرار بدهد. (تشکیل بسته LS)
 - 4 - بسته ساخته شده را به روش " سیل آسا " برای تمامی مسیر یابهای شبکه ارسال نماید و همچنین بسته هایی را که از مسیر یابهای دیگر میرسد دریافت و ذخیره کند. (ارسال به روش سیل آسا)
 - 5 - با استفاده از الگوریتمی مناسب ، بهینه ترین مسیر را بین هر دو مسیر یاب در شبکه ، پیدا نماید (حل الگوریتم به کمک روش دایجسترا)
- بسته LS :** هر مسیر یاب موظف است پس از جمع آوری اطلاعات لازم از مسیر یابهای مجاور خود بسته ای از این اطلاعات تشکیل بدهد؛ به این بسته ، " بسته LS " گفته می شود . در حالت کلی این بسته باید شامل اطلاعات زیر باشد:
- (الف) آدرس جهانی مسیر یاب تولید کننده بسته
- (ب) یک شماره ترتیب (تا بسته های تکراری از بسته های جدید تشخیص داده شوند)
- (ج) طول عمر بسته (تا اطلاعات بسته ، زمان انقضای اعتبار داشته باشد)
- (د) آدرس جهانی مسیر یابهای مجاور و هزینه تخمینی .

آدرس مسیر یاب مبدا	
شماره ترتیب	
Age	
	هزینه هر مسیر یاب

مراحل حل يك الگوریتم دایجسترا :

- فرض کنید الگوریتم بخواهد بهترین مسیر بین گره‌های 1 و 2 را پیدا کند :
- 1 - گراف زیر شبکه را تشکیل داده و گره‌های 1 و 2 را مشخص کنید. فرض کنید گراف زیر شبکه با استفاده از "ماتریس همجاری 1" نشان داده شود.

```
int dist[MAX_NODES][MAX_NODES];
```

در این ماتریس عنصر $a[i,j]$ هزینه رسیدن از گره i به گره j است. اگر هیچ مسیر مستقیمی بین دو گره نباشد مقدار ماتریس در درایه مورد نظر بی نهایت در نظر گرفته می شود.

2 - برای هر گره از گراف یک "رکورد حالت" ایجاد کنید که شامل سه فیلد باشد:

```
Stract state {
    Int predecessor;
    int length;
    enum {permanent, tentative} label;
} State[MAX_NODES];
```

- 3 - برای تمامی گره‌ها، متغیرهای حالت را بصورت زیر مقداردهی اولیه نمایید:

```
State[i].predecessor=NULL
State[i].length=INFINITY;
State[i].label=tentative;
```

- 4- در این مرحله ابتدا گره ابتدایی را به حالت دائم در می آوریم و سپس عملیات مقایسه زیر را انجام می دهیم :

```
if (State[t].length+dist[t][i]<State[i].length) {
    State[i].predecessor=t;
    State[i].length= State[t].length+dist[t][i]; }
```

- 5- از بین تمام گره‌هایی که بر چسبشان موقتی است گرهی که کمترین هزینه را تا 1 دارد پیدا کرده و به عنوان نقطه کار جدید انتخاب می کنیم.

- 6 - اگر 1 به گره مقصد نرسیده باشد مجدداً به مرحله 4 بر می گردیم.
- 7 - اگر 1 به گره مقصد رسید از آن شروع کرده و گره ماقبل آنرا از رکورد حالت استخراج کرده و این کار را تکرار می کنیم تا به گره مبدا برسیم.

اصول کار الگوریتم DV :

- 1 - هر مسیریاب موظف است هزینه خطوی را که بصورت فیزیکی با مسیریابهای دیگر دارد، محاسبه کرده و در جدول خود درج نماید. هزینه خطوی که مسیریاب با آنها در ارتباط مستقیم نیست، در این جدول بی نهایت در نظر گرفته میشود.

- 2 - هر مسیریاب موظف است در بازه های زمانی مشخص، ستون هزینه از جدول مسیریابی خودش را برای مسیریابهای مجاور ارسال نماید، ("یعنی فقط برای مسیریابهایی که با آن در ارتباط است نه تمام مسیریابها")

- 3 - هر مسیریاب موظف است پس از دریافت جداول مسیریابی از مسیریابهای مجاور، جدول خود را طبق یک الگوریتم بسیار ساده به هنگام نماید.

□ الگوریتم DV حداکثر تا 30 روتر پاسخگو است در تعداد روتر های بیشتر پیچیده می شود.

جدول مسیر یابی :

در الگوریتم های DV جدول هر يك از مسیریابها در حافظه خودنگهداری می کند که به جدول مسیریابی معروف است. در این جداول به ازای هر مسیریاب در زیر شبکه يك رکورد وجود دارد و هر رکورد دارای دو فیلد مجزا است که عبارتند از :

- الف) فیلد مسیر :** این فیلد خروجی مناسب برای رسیدن به يك مسیریاب خاص را در شبکه مشخص می کند.
- ب) فیلد مقدار تقریبی :** این فیلد هزینه تقریبی يك بسته تا مسیریاب مقصد را مشخص می کند. معیار هزینه می تواند تاخیر یا تعداد گام باشد. در چنین حالاتی واحد تاخیر میلی ثانیه و واحد hop تعداد خواهد بود.

□ در الگوریتم LS که حداکثر دارای n مسیریاب و هر مسیریاب دارای حداکثر k کانال ورودی خروجی باشد. در بدترین حالت به فضاي nk رکورد جهت ذخیره سازی اطلاعات نیاز مندیم. اما در الگوریتم DV به ازای حداکثر n مسیریاب نهایتاً نیاز به ذخیره سازی n رکورد می باشد.

مشکل الگوریتم DV : عدم همگرایی سریع جداول مسیریابی . هنگامی رخ می دهد که یک مسیریاب و یا یک کانال ارتباطی خراب شود . (شمارش تا بی نهایت)
راه حل مشکل الگوریتم DV : وقتی یک مسیریاب میخواهد اطلاعاتی را به همسایه‌هایش بدهد هزینه رسیدن به آنهایی را که قطعاً باید از همان مسیریاب بگذرند را اعلام نمیکند تا بی نهایت اعلام میکنند .

مسیریابی سلسله مراتبی : تعدادی از مسیریابها را به دسته ها و نواحی مختلف تقسیم بندی می کنیم تا از بروز تاخیرهای بحرانی دریافتی مسیر بهینه جلوگیری شود . در این روش کل شبکه به دسته های زیر تقسیم بندی می شود :

- 1 - کل شبکه به چندین دسته تقسیم می شود
- 2 - هر دسته به چند ناحیه تقسیم می شود
- 3 - هر ناحیه به تعدادی مسیریاب تقسیم بندی می شود

رکورد = از تعداد دسته ها یکی کم کرده و با تعداد مسیریابها جمع می کنیم

لایه انتقال (tcp) : وظیفه لایه انتقال در شبکه ، " فراهم آوردن خدمات سازماندهی شده ، مطمئن و مبتنی بر اصول سیستم عامل ، برای برنامه های کاربردی در لایه بالاتر است ، بگونهای که مشکلات و ناکارآمدی لایه IP جبران و ترمیم شود . وظیفه ای را که لایه انتقال بر عهده دارد ، میتوان با وظایفی که "سیستم مدیریت فایل" به عنوان بخشی از سیستم عامل بر عهده دارد ، قیاس کرد .

کاستی های لایه IP :

- 1 - تضمینی وجود ندارد وقتی بسته های برای یک ماشین مقصد ارسال میشود آن ماشین آماده دریافت آن بسته باشد و بتواند آنرا دریافت کند.
- 2 - تضمینی وجود ندارد وقتی چند بسته متوالی برای یک ماشین ارسال میشود به همان ترتیبی که بر روی شبکه ارسال شده اند ، در مقصد دریافت شوند.
- 3 - تضمینی وجود ندارد که وقتی بسته ای برای یک مقصد ارسال میشود ، به دلیل دیر رسیدن مجدداً ارسال نشود و در چنین حالتی ممکن است بسته های به اشتباه دو بار در مقصد دریافت شود .
- 4 - لایه IP هیچ وظیفهای در قبال توزیع بسته ها بین پروسه های مختلفی که بر روی یک ماشین واحد اجرا شده اند ندارد.
- 5 - لایه IP هیچ وظیفهای در قبال تنظیم سرعت تحویل بسته ها به یک ماشین ندارد . مثلاً ممکن است یک ماشین با سرعت بسیار زیاد بسته هایی را تولید کرده و تحویل لایه IP بدهد ولی ماشین مقصد قادر نباشد بسته ها را با این سرعت دریافت کند و بسته ها در مقصد به دلیل عدم توانایی در دریافت ، از بین بروند.

راه کارهای TCP برای جبران کاستی های لایه IP :

- 1 - A یک بسته خاص را به عنوان درخواست برای ارتباط ، به آدرس ماشین B می فرستد و منتظر می ماند
- 2 - B درخواست ارتباط را دریافت کرده و بر حسب شرایط ، آمادگی یا عدم آمادگی خود را به A اعلام می کند
- 3 - در صورتی که A در یک مهلت زمان مشخص ، پاسخ مثبت مبنی بر آماده بودن B دریافت نماید میتواند به ارسال داده ها اقدام نماید.

پروتکل های اتصال گرا : به پروتکل هایی که قبل از مبادله داده ها سعی در برقراری یک ارتباط و ایجاد هماهنگی قبلی می نمایند

آدرس سوکت: آدرس IP يك ماشين يكتا را در كل شبکه مشخص مي كند . شماره پورت نيز از بين پروسه هاي اجرا شده بر روي ماشين يكي از آنها را به عنوان مبدا يا مقصد تعيين مي كند بنابراین زوج آدرس IP و آدرس پورت مي توانند يك پروسه يكتا و واحد را روي هر ماشين در دنيا مشخص كنند .

ساختار بسته TCP :

<i>Source Port</i>		<i>Destination Port</i>	
<i>Sequence Number</i>			
<i>Acknowledgment Number</i>			
<i>THL</i>	<i>Unused</i>	<i>Flag</i>	<i>Window Size</i>
<i>Check Sum</i>		<i>Urgent Pointer</i>	
<i>Option</i>			
<i>Data</i>			

1 - *Source Port* : در اين فيلد يك شماره 16 بیتی به عنوان آدرس پورت مبدا كه اين بسته را جهت ارسال توليد كرده قرار خواهد گرفت .

2 - *Destination Port* : در اين فيلد آدرس پورت پروسه مقصد كه آنرا تحويل خواهد گرفت تعيين خواهد شد .

3 - *Sequence Number* : اين فيلد 32 بیتی شماره ترتيب آخريين بايتي را كه در فيلد داده از بسته جاري قرار دارد نشان مي دهد . در پروتكل TCP شماره ترتيب بر حسب شماره آخريين بايتي است كه در بسته جاري قرار دارد قرار گرفته و ارسال شده است .

4 - **Acknowledgment Number** : شماره ترتیبی است که فرستنده بسته منتظر دریافت آن است . این فیلد 32 بیتی است .

5 - **THL** : عددی که در این فیلد قرار می گیرد طول سرآیند بسته TCP را بر مبنای کلمات 32 بیتی تعیین می کند . با توجه به بسته TCP قسمت ثابت و اجباری آن 20 بایت است لذا کمترین عددی که در این قسمت قرار خواهد گرفت عدد 5 خواهد بود و بقیه این ظرفیت متعلق به option می باشد . (ظرفیت ای که برای قسمت option در نظر گرفته می شود بایستس ضریبی از 4 باشد)

6 - **Unused** : بدون استفاده است و حجم آن 6 بیت است .

7 - **Flag** : 6 بیت بعدی در بسته TCP هر کدام نقش يك بیت پرچم را که معنا و کاربرد خاصی دارند بازی می کند .

الف (**URG**) : این بیت در صورتی که مقدار يك داشته باشد معین می کند که در فیلد urgent pointer معتبر و قابل استفاده می باشد یا خیر . در صورتی که این بیت صفر باشد مشخص کننده این است که مقادیری که در قسمت urgent pointer آورده شده قابل استفاده نمی باشد .

ب (**ACK**) : اگر در این بیت مقدار يك قرار گرفته باشد نشان می دهد که عددی که در این فیلد قرار گرفته (Acknowledgment Number) معتبر می باشد یا خیر .

ج (**PSH**) : اگر در این بیت عدد يك قرار گرفته باشد فرستنده اطلاعات از گیرنده تقاضا می کند که داده های موجود را بافر نکند و در اصرع وقت آن را جهت پردازش های بعدی تحویل برنامه کاربردی آن بدهد . در حالتی که درخواست ما از برنامه های کاربردی در حد چند بایت باشد و انجام این درخواست اضطراری و سریع باشد . با يك نمودن بیت PSH این درخواست در واقع نگه داری نشده و به سرعت اجرا می شود .

د (**RST**) : اگر در این بیت مقدار يك قرار گیرد ارتباط به صورت يك طرفه و ناتمام قطع خواهد شد . بدین معنا که به هر دلیل اشکالی در یکی از طرفین ارتباط صورت پذیرد به معنای خاتمه ارتباطات است و همچنین بیت RST به معنای عدم پذیرش از یکی از طرفین تبادل داده به کار می رود .

ه (**SYN**) : این بیت نقش اساسی را در برقراری يك ارتباط بازی می کند .

و (**FIN**) : اگر یکی از طرفین داده دیگری برای ارسال نداشته باشد در هنگام ارسال آخرین بسته خود این بیت را يك می کند و در حقیقت ارسال اطلاعات را از طرف خودش يك طرفه قطع می کند . در این حالت اگر چه ارسال اطلاعات از طرف او قطع شده باشد ولیکن طرف مقابل هنوز ممکن است با ارسال اطلاعات مشغول برقراری ارتباط باشد .

8 - **Window Size** : مقدار قرار گرفته در این بیت مشخص می کند که فضای بافر گیرنده چند بایت دیگر فضای خالی دارد یعنی به طرف مقابل اعلام می کند که مجاز است از بایت با شماره ترتیبی که در فیلد Acknowledgment Number مشخص شده حداکثر به اندازه مقداری که در این فیلد درج شده است ارسال شده باشد و در غیر این صورت فضای کافی برای دریافت داده ها وجود نداشته و ناگزیر دور ریخته خواهد شد .

9 - **Check Sum** : این فیلد 16 بیتی است و کشف خطا در آن قرار دارد .

10 - **Urgent Pointer** : در این فیلد يك عدد به عنوان اشاره گر قرار می گیرد که موقعیت داده های اضطراری را درون بسته TCP معین می کند . تولید این داده ها زمانی اتفاق می افتد و ارسال می شوند که عملی شبیه وقوع وقفه ها در هنگام اجرای يك برنامه کاربردی رخ بدهد بدون آنکه ارتباط قطع شود .

روش برقراري ارتباط در TCP (دست تكاني 3 مرحله اي):

1 - طرف شروع كننده ارتباط ، يك بسته TCP خالي از داده كه در آن بيت $syn=1$ و بيت $ack=0$ است و درون

فيلد شماره ترتيب (seq no) عدد x را قرار مي دهد . اين عدد يك عدد تصادفي است و به جاي ارسال داده

ها از صفر - از اين شماره شروع به ارسال داده مي شود .

2 - صرف مقابل با دريافت تقاضاي فوق اگر مایل به برقراري ارتباط نباشد با ارسال يك بسته خالي كه در آن بيت

$rst=1$ است اين تقاضا را رد مي كند ولي اگر متمایل به برقراري ارتباط باشد يك بسته خالي از داده با

مشخصات زير توليد مي كند :

$syn=1 \ \& \ ack=1 \ \& \ seq \ no=y \ \& \ ack \ no=x+1$

3 - شروع كننده ارتباط با قرار دادن مقادير زير شروع ارتباط را تصديق مي كند :

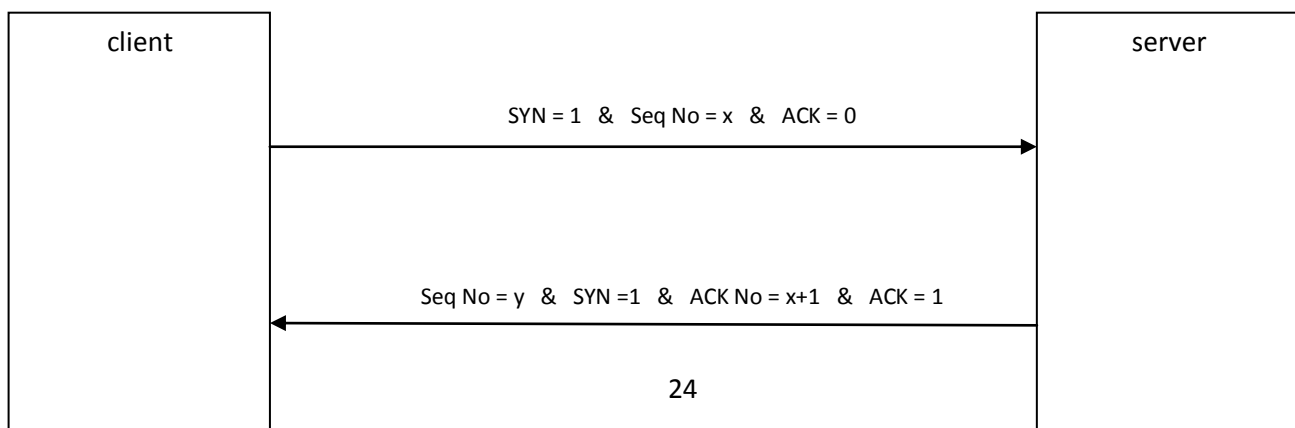
$syn=1 \ \& \ ack=1 \ \& \ seq \ no=x+1 \ \& \ ack \ no=y+1$ براي خاتمه ارتباط روند زير صورت مي گيرد :


طرفي كه داده هاش براي ارسال تمام شده است يك بسته TCP ارسال مي نمايد كه در سرآيند آن بيت FIN را يك

قرار داده است. طرف مقابل اين درخواست را دريافت ميكند و با ختم يك طرفه آن موافقت ميكند. ولي چون ارتباط

بصورت يکطرفه ختم ميشود طرف مقابل ميتواند تا جائيكه داده دارد ، آنها را ارسال كند و نهايتاً در آخرين بسته ، بيت

FIN را يك بگذارد تا پس از تصديق آن ، ارتباط به صورت دو طرفه ختم شود .



$$\text{SYN} = 1 \ \& \ \text{Seq No} = x+1 \ \& \ \text{ACK} = 1 \ \& \ \text{ACK No} = y+1$$


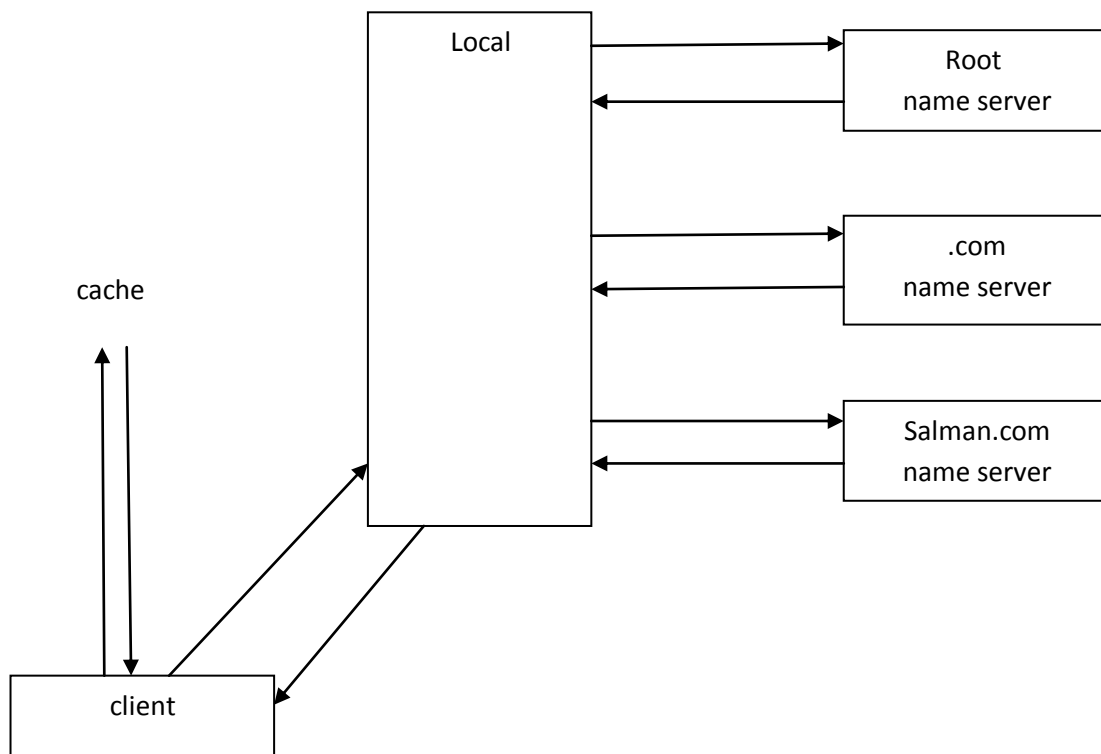
روشهای جستجو در سرویس دهنده های نام :

الف (پرس و جوی تکراری: در پرس وجوهای تکراری قسمت اعظم تلاش برای تبدیل یک نام بر عهده سرویس دهنده محلی است؛ این DNS حداقل به آدرس ماشین Root به عنوان نقطه شروع نیاز دارد. وقتی یک تقاضای ترجمه آدرس به سرویس دهنده محلی ارسال میشود در صورتی که قادر به ترجمه نام به معادل IP آن باشد ، معادل آدرس IP نام مورد نظر را به تقاضا کننده برمیگرداند. در غیر این صورت سرویس دهنده محلی خودش یک تقاضا برای DNS سطح بالا ارسال میکند .

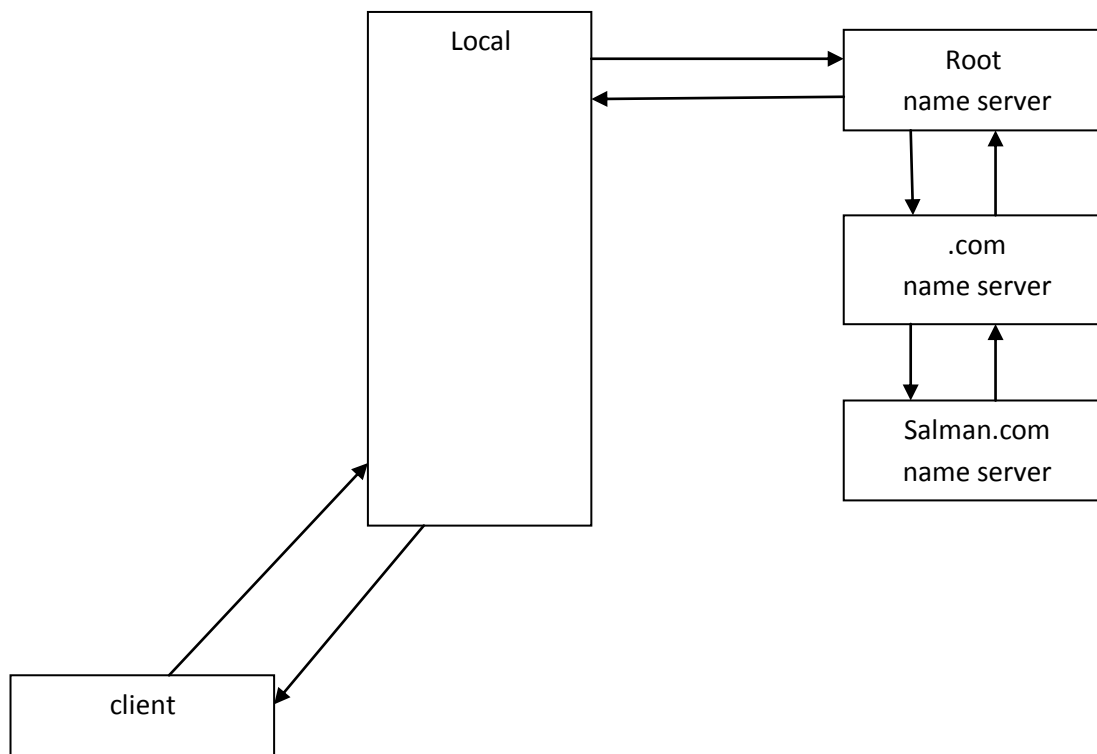
1 - برنامه کاربردی با فراخوانی "تابع تحلیل نام" ، تقاضای ترجمه آدرس را برای سرویس دهنده محلی ارسال کرده و منتظر می ماند

2 - سرویس دهنده محلی از سرویس دهنده Root آدرس ماشین یک DNS که متولی حوزه com. است را سؤال میکند.

- 3- آدرس سرویس دهنده مربوط به حوزه com. بر می گردد
- 4- سرویس دهنده محلی ، از ماشین معرفی شده در مرحله قبلی ، آدرس سرویس دهنده مربوط به حوزه Microsoft.com را در خواست می نماید
- 5- فهرستی از سرویس دهنده های DNS مربوط به Microsoft.com بر می گردد
- 6- سرویس دهنده محلی تقاضای ترجمه آدرس نمادین WWW. Microsoft.com را از DNS متعلق به حوزه Microsoft.com می کند .
- 7- معادل آدرس IP نام WWW. Microsoft.com بر می گردد 8- آدرس IP خواسته شده در اختیار برنامه کاربردی قرار میگیرد.



ب (پرس و موی بازگشتی: دقت کنید که در روش پرسوجوی تکراری نسبت به روش پرسوجوی بازگشتی ، حجم عمده عملیات برعهده سرویس دهنده DNS محلی است و مدیریت خطاها و پیگیری روند کار سادهتر خواهد بود و روش منطقی تری برای بکارگیری در شبکه اینترنت محسوب میشود. روش پرسوجوی بازگشتی برای شبکه های کوچک کاربرد دارد.



ج (پرس و جوی معکوس: روش کاربدین صورت است که سرویس‌دهنده محلی یک تقاضا برای DNS متناظر با شبکه ای که NetID آن باشد در آدرس IP ارسال می کند .

رتبه بندی و تحلیل لینک :

ماجول رتبه بندی : ماجولي که در تمام موتورهاي جستجوست که پس از غربال کردن نتایج بي ارزش یا کم ارزش آنها را بر حسب اهمیتشان رتبه بندی می کند تا آنچه را که کاربر دریافت می دارد فهرست مرتب شده اي از صفحات مرتب با کلید واژه هایش باشد .

معیار های رتبه بندی نتایج جستجو :

الف (اطلاعات مندرج درن صفحه :

- 1 - دفعات تکرار کلمات تکراري
- 2 - ترتیب و مجاورت کلمات کلیدی
- 3 - محل درج کلمات کلیدی
- 4 - درج کلمات کلیدی درون آدرس صفحه
- 5 - پررنگ بودن کلمات کلیدی و اندازه قلم
- 6 - بهره گیری از برچسب های توصیفی meta
- 7 - بهره گیری از بر چسب Alt Tag

ب (اطلاعات مندرج در بیرون صفحه

MDT برچسب های توصیفی متن : نوعی از برچسب های استاندارد و شناخته شده html هستند که شما می توانید به کمک آن يك خلاصه كوچك از محتوي سایتان را برای هر درون کاو فراهم کنید تا طبقه بندی و شاخص آنها به صورت صحیح انجام گیرد .

```

<meta name="key words" content = "      ">
<meta name="description" content = "    ">
  
```

<meta name="revised" content = " " />

فایل Root.txt : يك فایل متني كه درون سرویس دهنده وب و درون دایرکتوري اصلي هر وب سایت ذخیره می شود و تنظیمات گردش و جستجو در درون سایت را عرضه می نماید .