

# IP-GeoTracer

---

Autor: *Iván Baranda Molina*  
Treball Final de Carrera - Xarxes de Computadors  
Enginyeria Tècnica d'Informàtica de Sistemes

Consultora: *Maria Isabel March Hermo*

8 de gener del 2006

Esta página te la mereces. Toda para ti, Vanessa.

Gracias por ayudarme, por cuidarme, por apoyarme, por aguantar los fines de semana sin salir, mis ratos de mal humor (como todos los tenemos, no?), mis despistes, la falta de atención en ciertos momentos y ... resumiendo, gracias por estar ahí. ; - \*

Resum	1
1. Introducció	2
2. Objectius generals	3
3. Enfocament i metodologia	4
4. Planificació	5
4.1. Etapes de desenvolupament	5
4.2. Planificació inicial	5
5. Productes obtinguts	8
6. Resum de continguts	9
7. Capítol 1. Sobre localització geogràfica	10
7.1. Reverse DNS lookup (consulta inversa de nom de domini)	10
7.2. LOC DNS lookup (consulta del registre LOC al DNS)	11
7.3. Consultes WHOIS	11
7.4. Traceroute	12
7.5. Servei d'hora	12
8. Capítol 2. Anàlisi i disseny	13
8.1. Recollida de requisits	13
8.1.1. Model del negoci	13
8.1.2. Diagrama de casos d'ús	13
8.1.3. Guions	14
8.1.4. Diagrama d'objectes	14
8.1.5. Glossari del model de negoci	14
8.1.6. Documentació textual dels casos d'ús	16
8.1.7. Perfils d'usuari	18
8.1.8. Documentació de tasques futures	18
8.2. Anàlisi	19
8.2.1. Revisió dels casos d'ús	19
8.2.2. Paquets d'anàlisi i serveis	19
8.2.3. Classes d'entitats i els seus atributs	19
8.2.4. Relacions	19
8.2.5. Identificació de classes: frontera, control i operacions	20
8.2.6. Diagrama estàtic d'anàlisi	22
8.2.7. Anàlisi de la interfície d'usuari	25
8.3. Disseny	28
8.3.1. Disseny dels casos d'ús	28
8.3.2. Revisió del diagrama estàtic de disseny	29

8.3.3. Disseny de la persistència	31
8.3.4. Revisió del disseny de la interfície d'usuari	33
9. Capítol 3. Implementació	34
9.1. Arquitectura	34
9.2. Mòdul 1. Tecnologia de cerca i inferència.	34
9.3. Mòdul 2. Gestió de base de dades	37
9.4. Mòdul 3. Presentació de resultats	38
10. Capítol 4. Manual bàsic d'instal·lació i ús de IP-GeoTracer	41
11. Capítol 5. Millores per a futures versions	44
12. Capítol 6. Conclusions	46
Glossari	47
Bibliografia	50

## Resum

---

Des dels inicis d'Internet com a xarxa basada en el protocol IP, tot sistema connectat a aquesta xarxa i amb pretensions de comunicar-se amb qualsevol altre sistema també connectat, ha necessitat d'una adreça IP pública (“visible”) a Internet, per poder transmetre o rebre la informació desitjada. Aquesta IP, pot estar directament configurada al sistema emissor/receptor<sup>1</sup> o bé al sistema que ens faci el servei de porta d'enllaç a Internet<sup>2</sup>, que finalment serà l'adreça amb la qual sortim a Internet.

Tot i així, mai ha existit una associació directa entre una adreça IP i la seva localització geogràfica, de manera que quan el nostre sistema telemàtic comunica amb altre sistema similar fent ús de la xarxa Internet, no sempre tindrem coneixement de la ubicació d'aquest segon sistema, ni tan sols la dels nodes gràcies als quals arribem al destí fixat.

Com s'ha esmentat, no existeix una relació explícita entre una adreça d'Internet i la seva localització física, però existeixen diversos mètodes que, en conjunció, poden permetre delimitar la zona a la qual pertany, d'una manera bastant fiable.

De vegades permeten, com a màxim, deduir el país o estat, mentre que en alguns casos s'arriba a obtenir la província, la ciutat i fins i tot l'adreça.

El projecte que ens ocupa ha tingut com a objectiu oferir com a servei principal la localització geogràfica d'una adreça d'Internet mitjançant l'ús del subconjunt de tècniques més eficaces de les existents a l'actualitat.

El resultat és una aplicació independent i multi-plataforma que s'ocupa de la recerca necessària per assignar a cada node de la ruta la seva localització física, de representar gràficament el conjunt resultant i d'oferir altra informació rellevant relacionada amb l'adreça IP de destí fixada.

---

<sup>1</sup> Per exemple la connexió d'un PC per MODEM o la d'un enrutador configurat amb una IP estàtica o dinàmica, etc.

<sup>2</sup> Com pot ser el cas d'un PC connectat a Internet mitjançant una xarxa LAN.

# 1. Introducció

---

Moltes vegades ens pot resultar necessari conèixer la localització geogràfica d'una adreça d'Internet (IP) i/o la dels nodes que permeten que la informació arribi des del nostre punt fins a la adreça donada.

Les possibles utilitats d'una eina com la descrita podrien ser:

- Conèixer la nacionalitat dels visitants d'una web.
  - Per dirigir certs anuncis publicitaris.
  - Per fer la traducció automàtica del site.
- Permetre construir un mapa de les vies de comunicació d'una empresa distribuïda.
- Ajudar a localitzar la ubicació de l'inici d'una activitat fraudulenta a la xarxa.
- Avaluar la qualitat de la ruta segons la relació distància/temps de resposta.

L'aplicació que sorgeix a raó d'aquest projecte, aconseguix la informació necessària per localitzar amb certa precisió la situació geogràfica real dels sistemes d'enrutament que travessa un paquet per arribar al seu destí (adreça IP donada) i té a més la capacitat de representar gràficament, sobre un mapa del món, aquest trajecte fins al destí final.

Per aconseguir-ho l'aplicació ha estat dotada de diverses tècniques que treballant en equip arriben a deduir la ubicació de l'adreça donada:

- Reverse lookup de l'adreça IP i comparació de l'extensió del domini obtingut amb els TLD representants de països amb presència a Internet.
- Consultes WHOIS a servidors apropiats segons el TLD de l'adreça i altres factors.
- Un sistema propi de caché per ubicacions segons rangs d'IP.

La metodologia i tècniques emprades per l'aplicació es tractaran amb més profunditat a capítols posteriors.

El programari resultant es compon d'un únic mòdul de pes molt assequible (~1,7MB) per a descàrregues des d'Internet i totalment lliure, tant pel que fa als binaris com al codi font (open source).

La programació s'ha portat a terme en la seva totalitat en llenguatge Java (JDK 1.4.2\_01-b06) i, gràcies a això, al tamany resultant i a la unificació de l'aplicació en un únic mòdul, s'aconsegueix un alt grau de portabilitat que permet una instal·lació senzilla a gairebé qualsevol tipus d'estació de treball. Tot i això, les proves han estat realitzades només a plataformes Windows (Windows 2000 y Windows XP).

## 2. Objectius generals

---

L'objectiu principal del projecte és, mitjançant les tècniques i els coneixements adquirits amb les assignatures cursades, dissenyar i desenvolupar una aplicació capaç de representar gràficament el punt geogràfic aproximat d'una adreça IP donada i la dels nodes travessats per arribar-hi des de la nostra ubicació.

La qualitat esperada del resultat, pel que fa a l'aproximació geogràfica, passarà per obtenir, amb un alt grau de fidelitat, el país al qual pertany la IP. En els casos que ho permetin, obtindrem fins i tot la regió (província, estat, etc.) i la ciutat.

Tenint en compte que s'ha escollit desenvolupar una aplicació totalment lliure; no només el codi, els executables i la base de dades pròpia seran a l'abast de tothom, sinó que també els recursos utilitzats (consultes a servidor WHOIS, resolucions inverses de IP, etc.) seran sempre d'àmbit públic i global.

Per poder aconseguir la fita d'aquest projecte, prèviament s'ha hagut de realitzar un treball d'investigació per concretar, entre altres, les següents qüestions:

- Quines dades, directa o indirectament relacionades amb una IP, ens poden ajudar a ubicar-la.
- Quines eines o tècniques existents poden facilitar-nos la tasca d'aconseguir aquesta informació.
- Avaluar si convé que l'aplicació treballi en mode *online* o *offline*<sup>3</sup>, sobretot tenint en compte la possibilitat de que la informació necessitada pot ser molt variable en funció del temps.
- Avaluar la necessitat d'una base de dades pròpia que entre altres dades contindria informació geogràfica que permetés la confecció del plànol gràfic.

En resum, els objectius del projecte són:

Desenvolupar una aplicació capaç de deduir, segons diverses fonts d'informació, la localització geogràfica real d'una adreça IP fixada i la dels nodes que configuren el camí per arribar-hi. A més, ha de facilitar tota la informació trobada relativa a la esmentada IP de destí. També s'ha de procurar un baix manteniment de la base de dades i utilitzar només recursos de caràcter lliure o públic.

Nota: Donada la intenció de desenvolupar una solució totalment lliure, aquesta memòria no consta de cap apartat amb un estudi econòmic de costos o valoració econòmica final del producte.

---

<sup>3</sup> Per la seva natura, l'eina no pot donar cap resultat si s'executa en un entorn sense sortida a Internet. L'ús del terme *offline* en aquest context, tracta de diferenciar entre un aplicació que ha de consultar tota la informació a servidors externs (*online*) i una que només necessiti consultar la informació bàsica (*offline*).

### 3. *Enfocament i metodologia*

---

Tenint com a objectiu el desenvolupament d'una aplicació com l'esmentada, el primer pas que s'ha de donar és el d'investigar tot allò inherent a la localització física d'adreces IP. Com a tasca inicial, vaig haver d'avaluar les possibles fonts d'informació, arribant a la conclusió que allò que havia d'examinar en primer lloc, era la forma en que altres solucions lliures o disponibles al mercat resolien el problema.

D'aquesta fase vaig extreure algunes de les principals tècniques que finalment utilitzaria: consultes whois i caché de rangs de IP associats a països.

El següent pas consistia en aprofundir en el coneixement d'aquestes tècniques; protocol WHOIS, cerca de bases de dades lliures amb informació relativa a localització, etc.

Tot seguit vaig cercar documentació de mètodes addicionals als oferts per altres eines. Llavors vaig trobar el document "Hacking truths. What they don't teach in manuals." escrit per Ankit Fadia de la India, que va obrir-me els ulls en certs aspectes: a més dels que ja coneixia, mostrava diverses tècniques bàsiques que podien facilitar i accelerar la cerca d'aquesta informació. Els que resultaven especialment interessants pel projecte són: reverse DNS lookup i consultes d'hora a ISPs.

Un cop assolida la informació necessària per conèixer l'abast del problema a resoldre, entrem en la fase d'anàlisi i disseny, basada en el paradigma de l'orientació a objectes pel que fa al programari, i les bases de dades relacionals pel disseny de la persistència.

Amb el disseny del sistema enllestit, vaig començar per construir la base de dades i seguidament l'aplicació, seguint un mètode iteratiu i incremental per mòduls fixats per avançat, de manera que inicialment s'assolissin els objectius bàsics de l'aplicació per, a continuació, anar complementant-la amb millores funcionals i de presentació, entre d'altres.

El producte obtingut ofereix el servei descrit anteriorment d'una manera molt intuïtiva, amb resultats visuals fàcils d'interpretar i amb una precisió gens menyspreable.



## 4. Planificació

---

### 4.1. Etapes de desenvolupament

L'estructuració del temps i els recursos per a la consecució de l'objectiu s'ha repartit en sis tasques o fases de realització seqüencial:

1. Investigació i recollida de requisits.
2. Anàlisi i disseny de tot el sistema: aplicació i base de dades pròpia.
3. Desenvolupament iteratiu-incremental.
4. Proves i depuració.
5. Documentació del producte.
6. Confecció de la memòria del projecte.

### 4.2. Planificació inicial

A continuació es mostra en detall el pla proposat inicialment per a cadascuna de les fases:

<b>Tasca 1</b>	
<i>Temporalització</i>	21/09/05 – 02/10/05
<i>Descripció</i>	Recollida d'informació rellevant pel projecte i documentació de requisits. Entre la informació potencialment útil pel disseny del programari, destacar: <ul style="list-style-type: none"> <li>▪ Estudi d'altres aplicacions similars, ja existents.</li> <li>▪ Estudi de les possibles tècniques, tecnologies i eines necessàries (p.e: <i>whois</i>, <i>traceroute</i>)</li> <li>▪ Estudi d'idoneïtat de les diferents possibilitats de desenvolupar el programari: <i>online/offline</i>.</li> <li>▪ Estudi dels protocols i llenguatges necessaris. P.e per la consulta de certa informació a Internet.</li> </ul>
<i>Objectius</i>	Conèixer amb la màxima profunditat possible tot l'entorn en que s'ha de moure l'aplicació per afrontar l'anàlisi i el disseny amb la major facilitat i garantia d'èxit, tractant d'utilitzar les tècniques i els recursos més adients.
<i>Fites</i>	<ul style="list-style-type: none"> <li>▪ Informe de la funcionalitat i tecnologia base del programari existent amb una funció similar.</li> <li>▪ Obtenir una base en quant a tecnologies, eines i altres recursos pel desenvolupament del programari.</li> <li>▪ Documentació de requisits que ha de complir el programari</li> </ul>

### **Tasca 2**

<i>Temporalització</i>	03/10/05 – 23/10/05
<i>Descripció</i>	Fase d'Anàlisi i Disseny
<i>Objectius</i>	<ul style="list-style-type: none"> <li>▪ Obtenir l'anàlisi en funció dels requisits i funcionalitats desitjades.</li> <li>▪ Obtenir el disseny en funció de l'anàlisi trobada.</li> <li>▪ Obtenir l'anàlisi i disseny de la base de dades pròpia si finalment fos necessària.</li> </ul>
<i>Fites</i>	<ul style="list-style-type: none"> <li>▪ Obtenció de l'anàlisi i disseny del programari i l'estructura de dades necessària.</li> </ul>

### Tasca 3

<i>Temporalització</i>	24/10/05 – 27/11/05
<i>Descripció</i>	Fase de desenvolupament del programari
<i>Objectius</i>	<ul style="list-style-type: none"> <li>▪ Desenvolupament de l'aplicació en base al disseny orientat a objectes obtingut a la fase anterior. Aquesta fase es basarà en un sistema iteratiu i incremental que permetrà l'obtenció de mòduls amb funcions concretes però totalment fiables i funcionals. La unió final d'aquests mòduls ens aproparà a l'objectiu fixat.</li> </ul>
<i>Fites</i>	<ul style="list-style-type: none"> <li>▪ Obtenir la primera versió plenament funcional del programari.</li> <li>▪ Documentació referent al programari amb detall de la tecnologia i tècniques emprades.</li> </ul>

### Tasca 4

<i>Temporalització</i>	28/11/05 – 04/12/05
<i>Descripció</i>	Fase de proves i depuració de l'aplicació
<i>Objectius</i>	<ul style="list-style-type: none"> <li>▪ Provar a fons l'aplicació per trobar possibles errors de funcionalitat i que s'adapti als requeriments.</li> <li>▪ Implementar la versió definitiva de l'aplicació.</li> </ul>
<i>Fites</i>	<ul style="list-style-type: none"> <li>▪ Obtenir la versió definitiva del programari.</li> <li>▪ Obtenir la versió definitiva de la documentació referent al desenvolupament del programari.</li> </ul>

### Tasca 5

<i>Temporalització</i>	05/12/05 – 28/12/05
<i>Descripció</i>	Documentació del producte
<i>Objectius</i>	<ul style="list-style-type: none"> <li>▪ Reflectir les característiques, funcionalitats i modes d'ús del programari.</li> </ul>
<i>Fites</i>	<ul style="list-style-type: none"> <li>▪ Obtenció del manual de l'usuari.</li> </ul>

### Tasca 6

<i>Temporalització</i>	29/12/05 – 09/01/06
<i>Descripció</i>	Memòria i presentació
<i>Objectius</i>	<ul style="list-style-type: none"> <li>▪ Confeccionar la memòria del projecte.</li> <li>▪ Confeccionar la presentació del projecte.</li> </ul>

*Fites*

- Memòria i presentació del projecte.

La planificació mostrada, finalment no s'ha pogut seguir al peu de la lletra, degut principalment als problemes trobats en el camí. Ha sofert certes desviacions en quant a la temporalització parcial de certes fases, que ha pogut provocar un efecte dominó a les etapes següents, en començar-les fora del temps previst.

Finalment, però, s'ha aconseguit la conclusió del treball a temps, gràcies sobretot a un increment en la inversió d'hores, concentrades majoritàriament a les fases finals.

## 5. Productes obtinguts

---

Com a resultat del projecte que ens ocupa hem obtingut un únic mòdul autònom que a més d'acomplir els requisits fixats inicialment, gaudeix de les següents característiques:

- **Independència.**  
No es tracta d'una solució client-servidor, de manera que no depèn de cap sistema servidor remot ni local. Les úniques comunicacions que efectua amb altres sistemes són per portar a terme consultes whois, resolucions de nom, i altres serveis àmpliament representats arreu del món.
- **Alta portabilitat.**  
La programació s'ha portat a terme totalment en un llenguatge multi-plataforma com és Java. A la teoria hauria de ser possible executar l'aplicació a qualsevol sistema operatiu, però a la pràctica no és possible sense fer una nova entrada a la base de dades, referent al S.O. Inicialment només està preparat pels sistemes Windows 2000 y Windows XP. En executar per primer cop l'aplicació en un altre sistema operatiu, crea aquesta entrada.
- **Lliure.**  
S'ha desenvolupat de manera que tot allò que incorpora sigui de lliure distribució i amb codi obert. A més, aquells serveis externs que fa servir són també públics i a l'abast de qualsevol que disposi de connexió a Internet.
- **Lleuger.**  
La versió comprimida del programa té un tamany total de menys de 2MB, que permetria una descàrrega ràpida des de qualsevol servidor d'Internet que el contingui.
- **Fàcilment instal·lable.**  
Donada la seva lleugeresa i atomaticitat, el procés d'instal·lació resulta molt senzill i ràpid. Veure apartat: Capítol 4. Manual bàsic d'instal·lació i ús de IP-GeoTracer.
- **Baix manteniment.**  
El manteniment és gairebé innecessari i opcional. Si s'escull realitzar aquest manteniment, només seria convenient portar al dia la llista de servidors whois de la base de dades i els dominis TLD als quals estan associats.  
El manteniment és opcional, perquè els principals servidors whois utilitzats porten molt de temps donant servei i són gestionats per organismes que constitueixen una part important de l'administració d'Internet. Tot i així, portar un manteniment eventual de la base de dades garanteix un resultat més acurat i fiable del sistema.
- **Interfície d'usuari senzilla i intuïtiva.**  
Gràcies a la funció tan concreta que realitza el programa, ha estat possible dissenyar classes frontera molt senzilles i fàcils d'utilitzar. Els resultats es mostren de forma que no hi ha lloc a confusions.

## 6. Resum de continguts

---

### ***Capítol 1. Sobre localització geogràfica***

Aquest capítol fa un repàs introductorí sobre les possibilitats actuals en l'àmbit de la ubicació física d'adreces IP. Es parla de les tècniques que poden ajudar a donar solució a aquest problema i dels obstacles que podem arribar a trobar.

### ***Capítol 2. Anàlisi i disseny***

Comprèn tota la documentació referent a les que probablement són les dues fases més importants del procés global de desenvolupament de qualsevol projecte: anàlisi i disseny.

Tenint en compte que aquestes fases formen part del procés inicial, no acaben descrivint amb exactitud el resultat final, perquè el disseny, tal com es presenta en aquest capítol, ha sofert diverses variacions superficials, amb el propòsit de salvar obstacles o aspectes imprevistos trobats pel camí i millorar l'aplicació respecte a la visió inicial.

### ***Capítol 3. Implementació***

Es detalla l'arquitectura global de la solució, fent especial èmfasi en els algorismes i mètodes de localització emprats. Es descriuen també certs detalls de la implementació dels mòduls de presentació de resultats i de gestió de dades.

### ***Capítol 4. Manual d'instal·lació i ús de IP-Geotracer***

Es tracta simplement d'un manual d'usuari molt bàsic, donat les limitades funcions del programari.

### ***Capítol 5. Millores per a futures versions***

Al capítol 3 es parlava de la tecnologia utilitzada al programa, però tot i que podria ser una de les eleccions més encertades d'entre les possibles, es deixa clar que no és pas un sistema tan exacte com seria desitjable. En aquest capítol es proposen certes ampliacions que podrien dotar l'aplicació d'una major fiabilitat, resolució i agilitat.

# Capítol 1

## 7. Sobre localització geogràfica

---

Tal com es deixava veure al resum d'aquesta memòria, les adreces d'Internet no porten associades, ni directa ni indirectament, un registre real de la seva ubicació física; que vol dir que tampoc existeix un mètode estàndard i acceptat que permeti obtenir aquesta informació només amb una adreça del protocol IP.

Per tant, entra en joc la imaginació d'aquelles persones que s'han proposat donar una solució a aquest problema per tal de desenvolupar tècniques que permetin arribar a obtenir alguna associació.

Avui dia existeixen diversos mètodes que permeten aconseguir certa aproximació, però no existeix cap mètode que destaquí sobre la resta i permeti desenvolupar una solució senzilla. Es tracta més aviat de combinar intel·ligentment les porcions d'informació que obtenim de les diferents tècniques, tant per confirmar la veracitat de la informació com per que, gràcies a la seva complementació, obtinguem una ubicació més o menys fiable.

Tot seguit passem a explicar la majoria de les diferents tècniques existents a dia d'avui amb aquest propòsit:

### ***7.1. Reverse DNS lookup (consulta inversa de nom de domini)***

Si en una consulta de nom de domini habitual (DNS lookup) preguntem quina és l'adreça IP que correspon a un FQDN concret, en una consulta inversa de nom de domini preguntarem pel FQDN associat amb una IP. Per a més detalls en quant al funcionament del sistema DNS i les diferents variants de consultes DNS: [How Reverse-Lookup Works](#), [How DNS Works](#)

Aquesta opció es basa en la resolució inversa de noms de domini com a sistema per obtenir el TLD (.net, .com, .es, .jp) al qual pertany la IP. A partir del TLD obtingut, si aquest representa l'extensió de domini d'un país, es pot donar per fet que aquesta IP pertany al país referenciat pel TLD. D'aquesta manera obtenim una resolució a nivell de país. Moltes vegades però, aquest nivell de resolució no és suficient, i amb la seguretat de tenir el país correcte, es pot combinar aquesta solució amb altres que permetin aprofundir una mica més.

Val a dir que per a TLDs que representin dominis internacionals com poden ser: net, com, biz, edu, org. No serem capaços de deduir cap localització, però si que ens podrà servir per escollir quina altra tècnica aplicar. Per exemple, ens pot ajudar a seleccionar un servidor whois per consultar.

Una altra possibilitat que ens ofereix la resolució inversa és la de la inferència en funció de les pistes que podem trobar a alguns noms complets. Per exemple:

<code>newdehli-00.backbone.vsnl.net.in</code>
---

ens proporciona les següents dues pistes:

- País: Índia (in)
- Regió: Nova Dehli (newdehli)

Aquest mètode però, no es gaire fiable, perquè ens podem trobar amb el cas que una part del nom (que no sigui l'extensió) coincideixi amb el codi d'un país (p.e. es), però que sigui per casualitat i nosaltres pensem que aquest node pertany al país deduït.

## 7.2. LOC DNS lookup (consulta DNS LOC)

En una consulta DNS normal podem preguntar per un cert registre d'un domini. Aquests registres poden ser de correu (MX), d'adreça IP (PTR), servidors de nom (NS), etc. Existeix també un tipus de registre que conté informació sobre la localització, en forma de coordenades (LOC), de tal manera que si preguntéssim pel registre LOC d'una adreça IP o un FQDN, ens hauria de retornar les coordenades de la situació real del sistema configurat amb aquesta IP.

Aquesta seria una molt bona solució pel problema que ens plantegem. El que la converteix en una opció poc útil és el fet que només un percentatge molt baix dels responsables encarregats del manteniment dels registres DNS porten a terme la inclusió d'aquest tipus de registre.

## 7.3. Consultes WHOIS

Les consultes WHOIS es basen en l'aprofitament de la informació que es registra quan un RIR delega un rang d'IPs a un ISP per a la seva explotació.

Per entendre bé aquest concepte és necessari tenir clar el funcionament bàsic de l'assignació d'adreces de Internet:

*Als usuaris finals se'ls assignen adreces IP gràcies als ISP (Internet Service Providers), que serà l'estament que a la vegada els proporcionarà accés al mitjà (xarxa Internet).*

*A la seva vegada, els ISP obtenen concessions de blocs complets d'adreces pels organismes encarregats de la gestió d'espais de números de Internet. Aquests poden ser registres locals (LIR), registres nacionals (NIR) o regionals. Els anomenats Regional Internet Registry (RIR). Aquests a la seva vegada han de demanar concessions d'adreces al principal organisme gestor d'adreces de Internet: IANA (Internet Assigned Numbers Authority).*

*Per conèixer amb més profunditat el mecanisme d'assignació de IPs, consulta la web de IANA a [www.iana.org](http://www.iana.org)*

Segons aquest sistema d'assignació d'adreces, són els registres d'Internet els que mantenen una base de dades amb les concessions a ISP de blocs o rangs d'adreces. Al mateix cop solen oferir un servei públic de consulta d'informació anomenat WHOIS, que ens permet rebre informació de contacte i altres dades en relació amb una IP.

Aquest servei pot resultar útil a l'hora de trobar informació sobre la localització de la IP, però donat que els registres s'efectuen per blocs d'IP, el nivell de resolució que obtindrem tampoc serà gaire alt. Normalment a nivell de país i de vegades a nivell de regió com a divisió d'un país, segons quin servidor whois ens resolgui la consulta.

#### 7.4. Traceroute

El concepte de traceroute contempla la resolució de la ruta per la qual passa el tràfic d'una comunicació entre dos sistemes connectats a una xarxa IP. El fet d'obtenir els noms (FQDN) dels nodes que interconnecten aquests dos sistemes, ens pot ajudar a deduir la localització, si a més ho combinem amb la d'inferència segons el nom associat a la IP (vegeu "Reverse DNS lookup" en aquest mateix apartat). Veiem un exemple:

```
Tracing route to 203.94.12.54 over a maximum of 30 hops
  0  abc.netzero.com (232.61.41.251)  2 ms  1 ms  1 ms
  1  xyz.Netzero.com (232.61.41.0)    5 ms  5 ms  5 ms
  2  232.61.41.10 (232.61.41.251)   9 ms  11 ms 13 ms
  3  we21.spectranet.com (196.01.83.12) 535 ms 549 ms 513 ms
  4  isp.net.ny (196.23.0.0) 562 ms 596 ms 600 ms
  5  196.23.0.25 (196.23.0.25) 1195 ms 1204 ms
  6  backbone.isp.ny (198.87.12.11) 1208 ms 1216 ms 1233 ms
  7  asianet.com (202.12.32.10) 1210 ms 1239 ms 1211 ms
  8  south.asinet.com (202.10.10.10) 1069 ms 1087 ms 1122 ms
  9  backbone.vsnl.net.in (203.98.46.01) 1064 ms 1109 ms 1061 ms
 10  newdelhi-01.backbone.vsnl.net.in (203.102.46.01) 1185 ms 1146 ms 1203 ms
 11  newdelhi-00.backbone.vsnl.net.in (203.102.46.02) 1159 ms 1073 ms
 12  mtnl.net.in (203.194.56.00) 1052 ms 642 ms 658 ms
```

amb aquest resultat de la comanda tracert (del sistema Windows) podem assegurar que el node de destí es troba a la Índia, i gairebé posar la mà al foc assegurant que a la regió de Nova Delhi, donat que el node anterior (ISP) té aquesta ubicació.

#### 7.5. Servei d'hora

Molts ISPs ofereixen el servei d'hora local. Una manera de saber almenys la zona horària del ISP que ofereix servei al destí fixat és provar a preguntar la hora al ISP fent-hi simplement una connexió TCP al port 13.

Tot i que com a mètode de localització no és gaire precís, donat la quantitat de països que pot haver-hi en una mateixa zona horària, sí que podria fer servei per donar més pes a la sospita d'una localització concreta.



## Capítol 2

### 8. Anàlisi i disseny

#### 8.1. Recollida de requisits

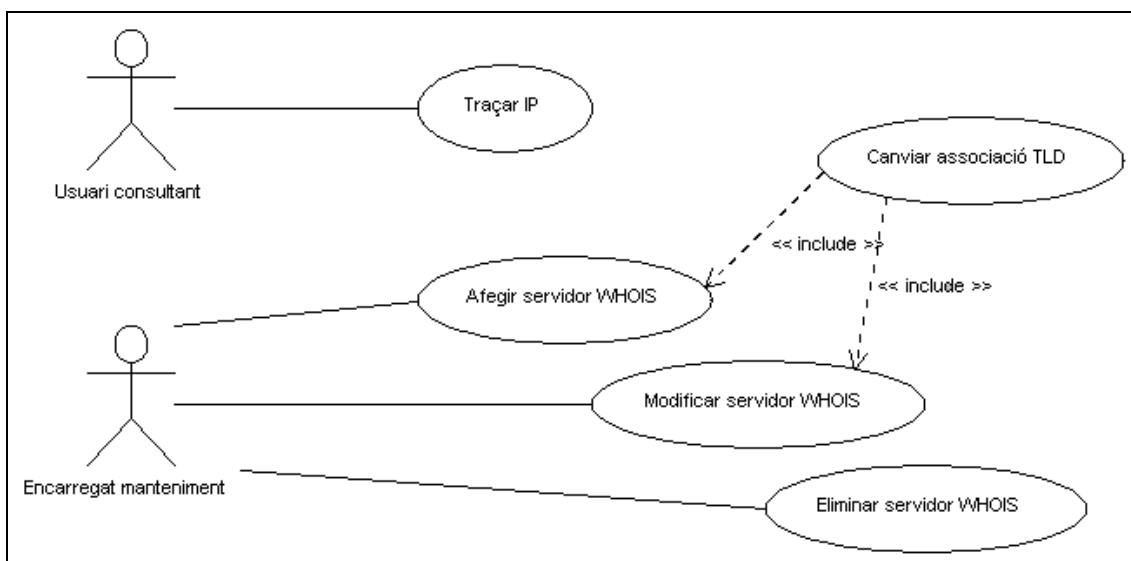
##### 8.1.1. Model del negoci

El programari té la funció principal de proporcionar informació relativa a una IP donada per l'usuari consultor. Entre aquesta informació és especialment rellevant la ruta geogràfica que s'ha hagut de seguir per arribar a la IP de destí. Aquesta ruta estarà conformada per diferents nodes, cadascun amb una adreça IP, situats entre l'origen i el destí (ambdós considerats també nodes).

Per tal d'aconseguir tota la informació possible d'una IP, el programari ha d'executar diverses funcions. Una de les funcions més importants que ha de poder realitzar és una consulta WHOIS. Aquest tipus de consulta es realitza a un servidor especialitzat i retorna informació relativa a una IP concreta. El servidor al qual es destina la consulta, es tria en funció de l'extensió del domini o TLD (.com, .net, .es, etc.), donat que normalment els servidors WHOIS no contenen informació de tots els rangs d'IP, sinó només dels corresponents a certs TLD.

A més, pot passar que un servidor WHOIS deixi d'existir, o que aparegui un altre nou o que canviïn els TLD pels quals pot oferir informació. Per tant l'aplicació haurà d'oferir la possibilitat de donar d'alta, modificar i eliminar servidors WHOIS de la base de dades i també canviar les possibles associacions d'aquests amb els TLD existents. Aquesta funció l'haurà de realitzar un usuari amb certs coneixements del sistema.

##### 8.1.2. Diagrama de casos d'ús



### 8.1.3. Guions

#### **Usuari consultant:**

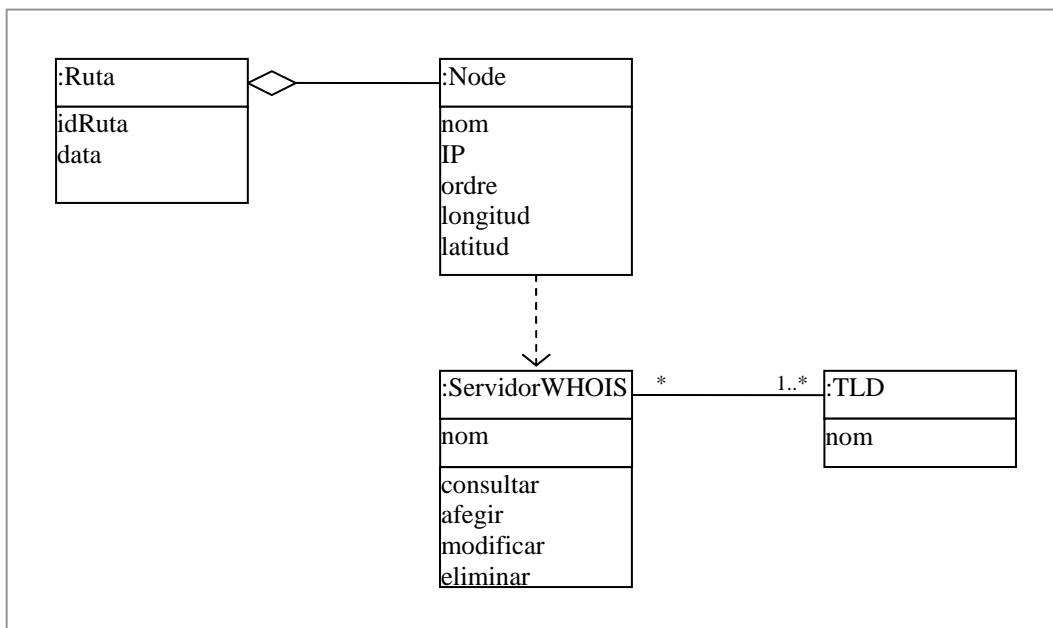
La única funció de l'usuari és la de consultar informació relativa a una IP o nom de domini complet (FQDN), que també ha d'incloure la ruta geogràfica per on s'arriba a la IP donada i per tant la localització física d'aquesta adreça destí. En l'execució d'aquesta funció es prou important el paper de les consultes que ha de fer el programari a servidors WHOIS de Internet.

#### **Encarregat manteniment:**

Els servidors WHOIS que fan servir el programari per aconseguir informació sobre una IP a vegades deixen de funcionar, o no serveixen per donats dominis (TLD), o deixen d'existir o fins i tot poden apareixer de nous. Per tant, l'encarregat del manteniment ha de procurar que la base de dades de la aplicació contingui només servidors vigents i associats només amb aquells TLDs pels quals contenen informació.

Per tal de poder portar a terme aquesta funció consta de les següents operacions: afegir servidor WHOIS, modificar servidor WHOIS i eliminar servidor WHOIS. Les associacions a TLD les pot portar a terme optativament en el moment de realitzar una modificació o en afegir un servidor nou.

### 8.1.4. Diagrama d'objectes



### 8.1.5. Glossari del model de negoci

**Adreça IP:** Adreça de xarxa segons el protocol IP, que permet localitzar un host qualsevol dins una xarxa IP, com pot ser Internet.

**Destí:** Adreça IP que representa el node final/destí d'una ruta.

- Encarregat mant.:* Usuari encarregat d'actualitzar la base de dades del programari portant al dia el llistat de servidors WHOIS vigents i els TLD pels quals serveixen informació.
- FQDN:* *Full Qualified Domain Name.* Nom de domini complet. Nom de construcció alfabètica que equival a una adreça de Internet (IP). Un nom de domini complet ha de contenir en si mateix tots aquells dominis o subdominis als quals pertany separats per punts. De manera que a la dreta de tot figura el domini de nivell màxim (TLD) de la jerarquia de dominis, el següent per la dreta serà un subdomini d'aquest, i així successivament, fins arribar a la paraula de davant del primer punt, que dona nom a la màquina que té assignada la IP que equival al FQDN.
- Node:* Elements que componen una ruta, identificats per la adreça IP que tenen assignada.  
Anomenem nodes tant als dispositius d'origen i destí fixats com als intermitjos que permeten unir-los.  
En el cas dels nodes intermitjos: Sistemes de nivell 3 del model OSI (xarxa) que permeten arribar al destí fixat o bé apropar-se un node més, gràcies a la seva capacitat d'enrutar.  
En el cas dels nodes extrems: Sistemes originador i receptor que aprofiten l'estructura de nodes de la xarxa IP Internet per comunicar-se.
- Origen:* Adreça IP que representa el node inicial/origen d'una ruta.
- Ruta:* Representació gràfica de la trajectòria entre els nodes pels quals s'ha de passar per arribar al node destí.
- Servidor WHOIS:* Servidor amb presència a Internet que proporciona informació relativa a una adreça o rang IP. Entre la informació que s'obté, sol constar el país on s'ha assignat aquesta IP, i a vegades fins i tot la ciutat i altres dades de contacte amb el propietari de la IP o rang d'adreces IP.
- TLD:* *Top Level Domain.* Domini de nivell màxim dins la jerarquia constituïda pels noms de domini. Hi consten només aquells dominis que pegen directament de l'arrel de l'arbre, administrada per INIC (*Internet Network Information Center*).  
Dins un nom de domini complet (FQDN), representa la part del nom situada després de l'últim punt.
- Usuari consultant:* Usuari del programari que necessita obtenir informació referent a una IP, la seva localització geogràfica i la ruta a seguir per arribar-hi.

**8.1.6. Documentació textual dels casos d'ús****Cas d'ús número 1: “Traçar IP”**

Resum de a funcionalitat	Aconseguir tota la informació possible relativa a una adreça IP i traçar gràficament la ruta per arribar-hi.
Paper dins el treball de l'usuari	Cas únic i per tant principal per l'actor.
Actors	<b>Usuari consultant.</b>
Casos d'ús relacionats	– cap –
Precondició	– cap –
Postcondició	Obtenim les dades referents a la IP i la ruta geogràfica per arribar-hi.
Descripció	L'usuari introdueix una adreça d'Internet per la qual vol obtenir la informació referida. Aquesta adreça pot estar expressada en forma d'adreça IP o com a nom de domini complet. P.e: <a href="http://www.uoc.edu">www.uoc.edu</a>
Alternatives de procés i excepcions	Pot passar que el nom de host donat no existeixi (no figuri al servidor DNS i per tant no tingui cap IP associada) o que no existeixi informació relacionada amb la IP donada.

**Cas d'ús número 2: “Afegir servidor WHOIS”**

Resum de a funcionalitat	Dona d'alta un nou servidor WHOIS a la base de dades.
Paper dins el treball de l'usuari	Freqüent per l'actor, però poc freqüent pel programari.
Actors	<b>Encarregat manteniment.</b>
Casos d'ús relacionats	<u>Canviar associació TLD</u>
Precondició	El servidor no existeix a la base de dades.
Postcondició	El servidor s'ha incorporat a la base de dades.
Descripció	L'encarregat de manteniment introdueix la URL del servidor d'Internet i per a quins dominis TLD serveix.
Alternatives de procés i excepcions	<ul style="list-style-type: none"> <li>▪ Si el servidor ja existeix es cancel·la l'operació.</li> <li>▪ Si es relaciona amb algun TLD, aquesta ha d'extir, i l'operació d'associació es realitza segons el cas d'ús <u>Canviar associació TLD</u>.</li> </ul>

**Cas d'ús número 3: “Modificar servidor WHOIS”**

Resum de a funcionalitat	Modifica un servidor WHOIS a la base de dades, permetent a la vegada canviar les associacions amb TLD.
Paper dins el treball de l'usuari	Poc freqüent per l'actor i encara menys pel conjunt de l'aplicació.
Actors	<b>Encarregat manteniment.</b>
Casos d'ús relacionats	<u>Canviar associació TLD</u>
Precondició	El servidor existeix a la base de dades i a més té com a mínim una associació amb algun TLD.
Postcondició	El servidor continua existint a la base de dades, però amb certs canvis. Conservarà com a mínim una associació.
Descripció	L'encarregat de manteniment pot canviar el nom del servidor o les associacions amb TLD.
Alternatives de procés i excepcions	Si es tracta d'eliminar totes les associacions del servidor, s'imprimirà per pantalla un avís dient que tot servidor WHOIS n'ha de conservar almenys una, en cas contrari s'ha de donar de baixa (eliminar) el servidor.

**Cas d'ús número 4: “Eliminar servidor WHOIS”**

Resum de a funcionalitat	Dona de baixa un servidor WHOIS existent a la base de dades.
Paper dins el treball de l'usuari	Freqüent per l'actor, però poc freqüent pel programari.
Actors	<b>Encarregat manteniment.</b>
Casos d'ús relacionats	- cap -
Precondició	El servidor existeix a la base de dades.
Postcondició	El servidor s'ha donat de baixa de la base de dades.
Descripció	L'encarregat de manteniment selecciona el servidor a donar de baixa i amb aquesta operació escindeix també les possibles relacions amb TLD.
Alternatives de procés i excepcions	- cap -

### **8.1.7. Perfils d'usuari**

El potencial usuari consultant està mínimament familiaritzat amb el funcionament de la xarxa Internet i coneix els conceptes necessaris per entendre el resultat de l'aplicació.

L'encarregat de manteniment també està familiaritzat amb la xarxa Internet i a més amb el funcionament dels servidors WHOIS i el resultat retornat per aquests.

En teoria els usuaris no tenen experiència en cap aplicació semblant.

La funció amb major índex d'utilització serà, evidentment, la de consulta d'informació sobre una IP. Les operacions de manteniment són, en general, poc habituals.

### **8.1.8. Documentació de tasques futures:**

#### ***Traçar IP***

L'usuari consultant simplement introdueix una adreça IP o un nom de domini complet i espera a rebre el resultat per pantalla.

#### ***Afegir servidor WHOIS***

L'encarregat de manteniment accedeix a l'apartat de manteniment de la base de dades i una vegada dins escull la opció d'afegir servidor WHOIS.

Seguidament introdueix el nom complet del servidor i selecciona d'entre els existents, els TLD pels quals pot servir consultes (veure tasca Canviar associacions TLD). Guarda els canvis i si ha de continuar donant d'alta servidors, torna a repetir el procés, sinó, surt de l'apartat de manteniment de la base de dades.

#### ***Modificar servidor WHOIS***

L'encarregat de manteniment accedeix a l'apartat de manteniment de la base de dades i una vegada dins escull la opció de modificar servidor WHOIS.

Seguidament escull el servidor a modificar i canvia el seu nom si cal o canvia les associacions amb TLD (veure tasca: Canviar associacions TLD). Guarda el canvis y surt de l'apartat de manteniment de la base de dades.

#### ***Eliminar servidor WHOIS***

L'encarregat de manteniment accedeix a l'apartat de manteniment de la base de dades i una vegada dins escull la opció d'afegir servidor WHOIS.

Seguidament escull el servidor a donar de baixa i confirma l'operació. Automàticament s'eliminen les associacions del servidor amb TLDs.

#### ***Canviar associacions TLD***

L'encarregat arriba a aquest apartat des de "Afegir servidor WHOIS" o "Modificar servidor WHOIS". L'usuari selecciona d'una llista aquells TLD que vol associar o dissociar del servidor en tractament. En acabar guarda els canvis i retorna a l'apartat de manteniment de la base de dades.

## 8.2. Anàlisi

### 8.2.1. Revisió dels casos d'ús

No cal fer revisió dels casos d'ús. La versió obtenida a la recollida de requisits ja és correcta.

### 8.2.2. Paquets d'anàlisi i serveis

No cal fer divisió. El projecte és suficientment senzill per no haver de fer ús d'aquesta tècnica.

### 8.2.3. Classes d'entitats i els seus atributs

Classe Node: nom(text), adrecaIP(enter), ordre(enter), longitud(real), latitud(real).

Classe Ruta: idRuta(integer), data(data).

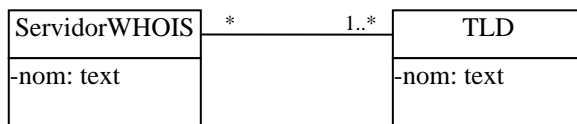
Classe ServidorWHOIS: nom(string), patroConsulta(string)

Classe TLD: nom(string), descripcio(string)

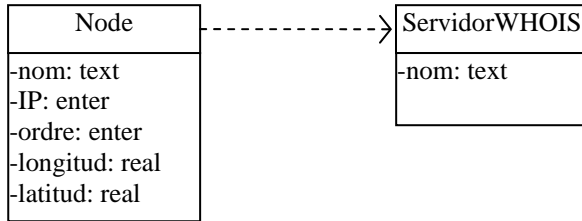
### 8.2.4. Relacions

No existeixen relacions d'herència de cap tipus.

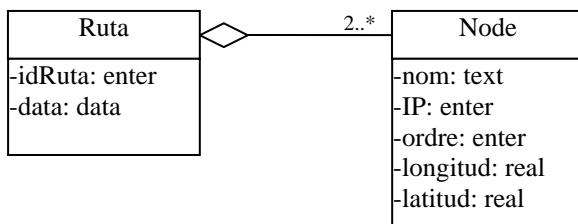
De relacions associatives trobem l'existència entre les classes ServidorWHOIS i TLD que denoten que un objecte de la classe ServidorWHOIS només serveix informació referent als objectes TLD als quals està relacionat.



Troben també una relació de dependència entre les classes Node i ServidorWHOIS, fruit de les consultes, a un servidor o servidors, que ha de portar a terme un node per aconseguir informació sobre l'adreça IP que té assignada.

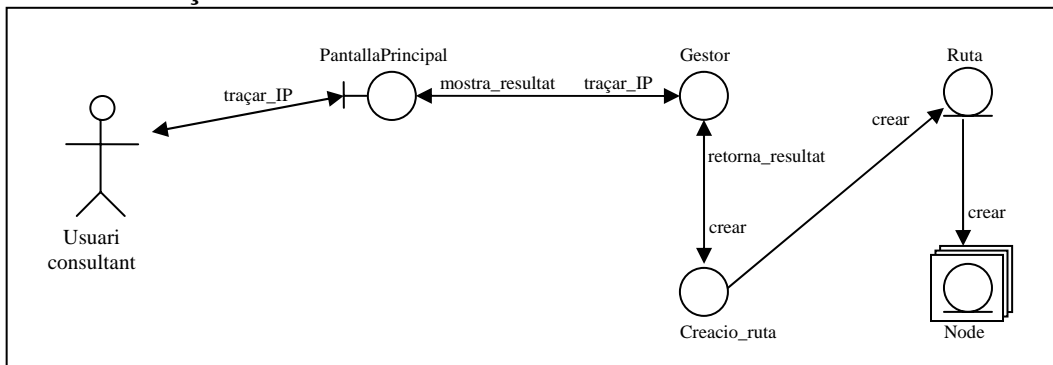


D'agregacions tenim la de les classes Ruta i Node, donat que un objecte Ruta es compon de com a mínim dos nodes: origen i destí. Com que la desaparició d'una ruta no comporta la desaparició dels seus nodes i a més aquests poden pertànyer a una altra ruta, no es tracta d'una composició.



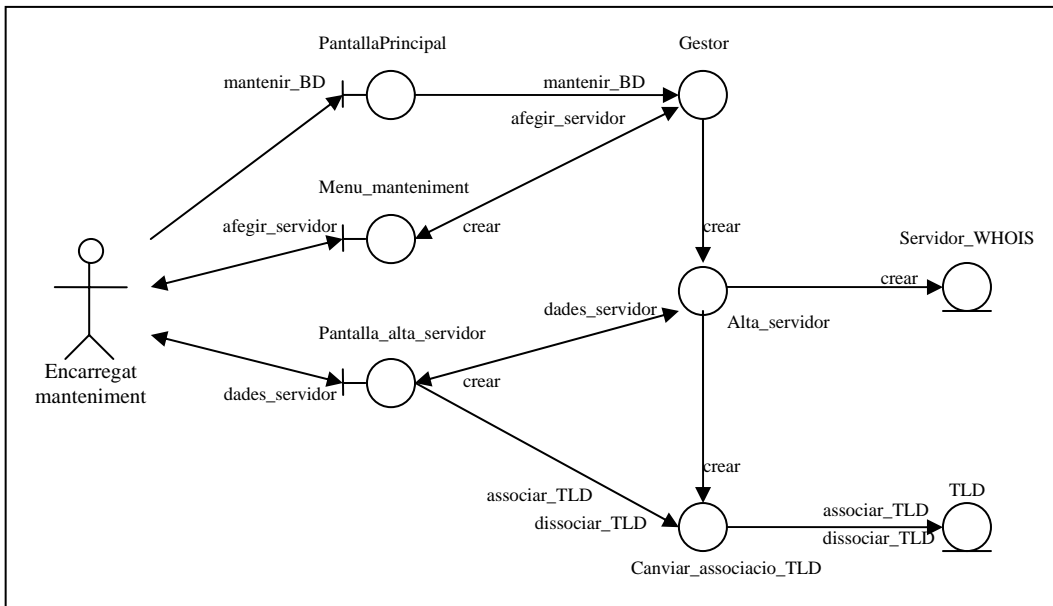
**8.2.5. Identificació de les classes de frontera, les classes de control i de les operacions**

**Cas d'ús "Traçar IP":**

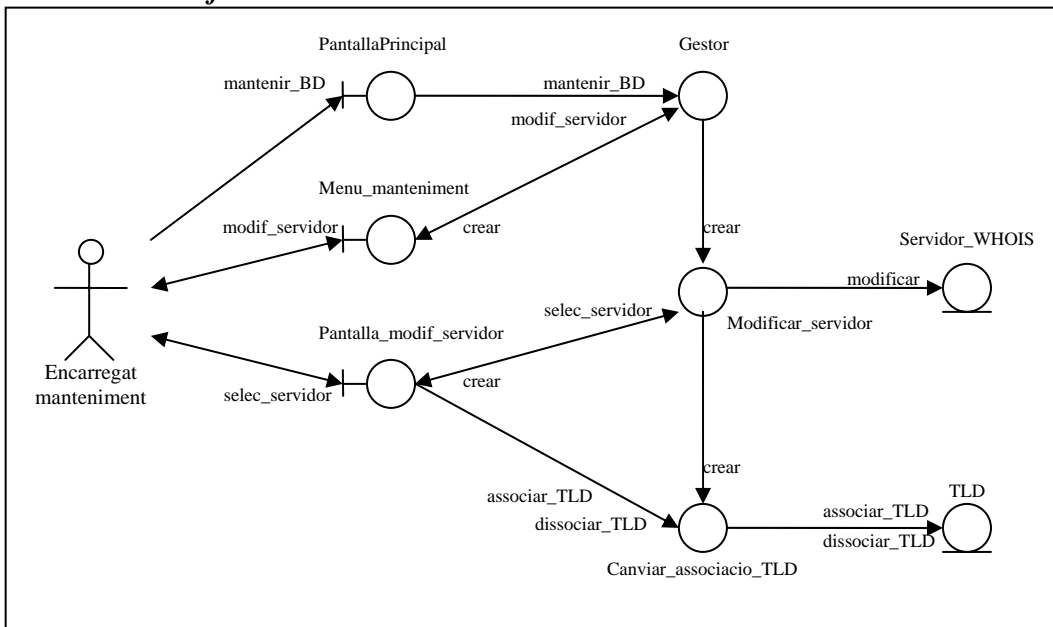


**Cas d'ús "Afegir servidor WHOIS":**

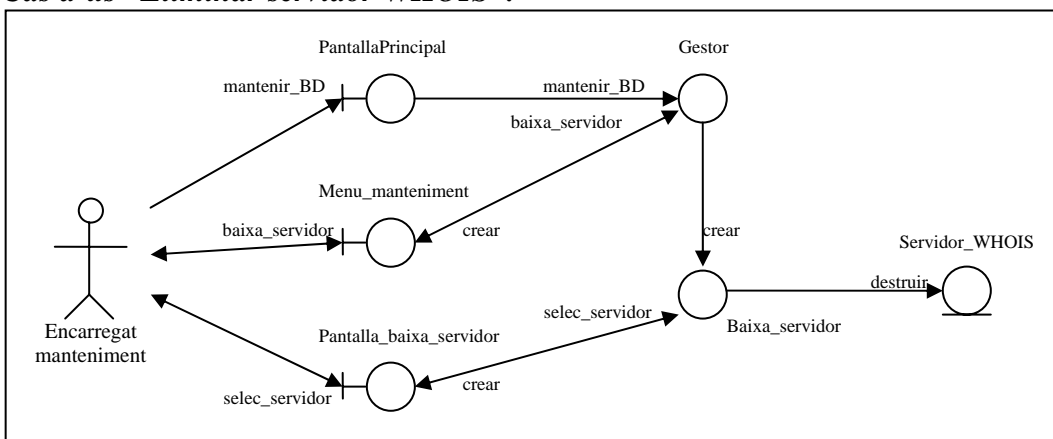




**Cas d'ús "Modificar servidor WHOIS":**



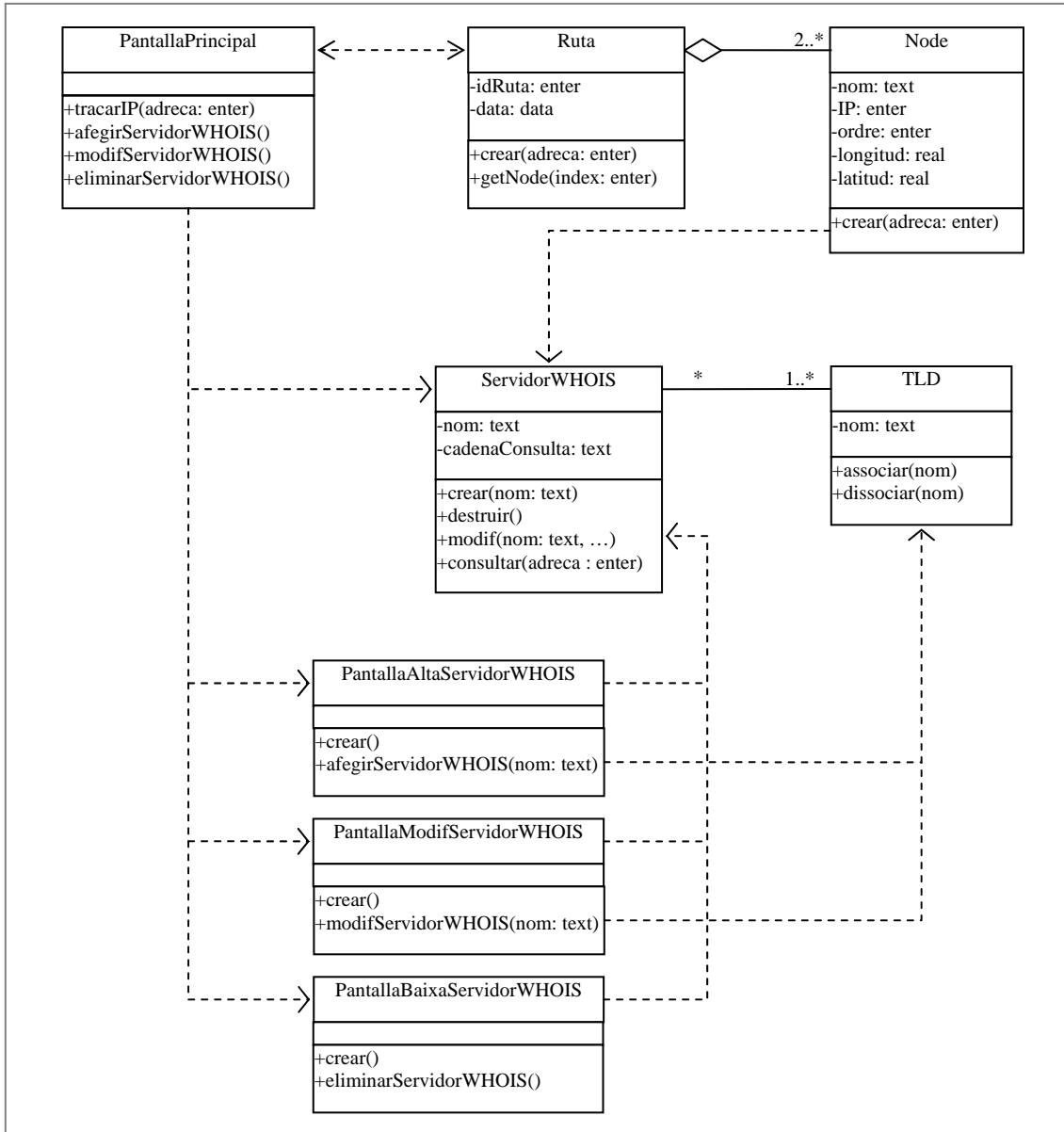
**Cas d'ús "Eliminar servidor WHOIS":**



### 8.2.6. Diagrama estàtic d'anàlisi

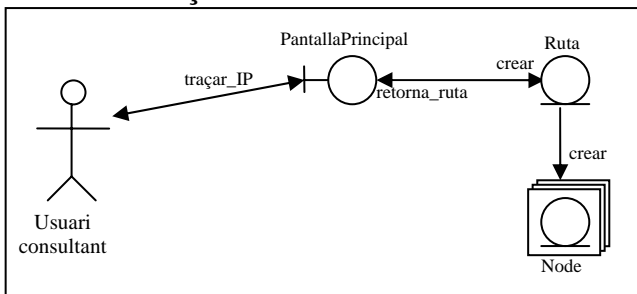
Per a l'elaboració del diagrama, inicialment només es tindrà en compte la classe Gestor, com a classe de control. La resta de classes de control es poden resumir com a operacions pròpies d'aquesta classe.

Mentre anem fent el diagrama estàtic veiem que la majoria de crides que suporta la classe que controla la majoria d'accions del programari: *Gestor*, també les ha de incorporar la classe frontera PantallaPrincipal com a intermediària i que la part referent al manteniment de la base de dades podria simplificar-se fent les crides afegirServidor, modificarServidor i eliminarServidor directament, sense haver de cridar la classe MenuManteniment, (p.e. mitjançant un sistema de menú amb l'opció "manteniment BD" i les corresponents sub-opcions: "afegir servidor", "modificar servidor", "eliminar servidor"). Per tant, un millor anàlisi permet simplificar el projecte encara més i eliminar per complet les classes de control i la classe frontera MenuManteniment.

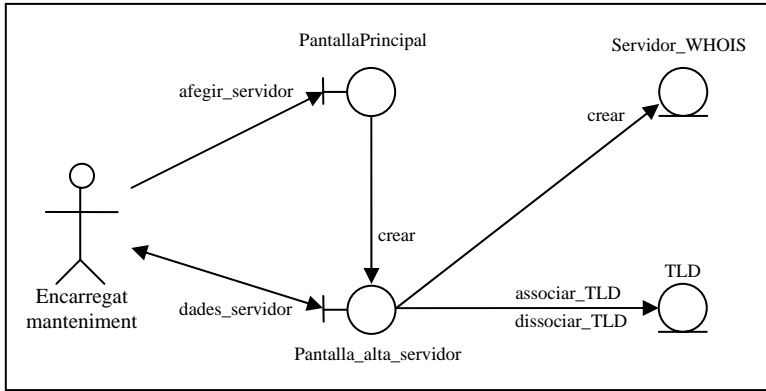


Donats els canvis que ha sofert l'estructura, s'han de redisenyar els diagrames de col·laboració:

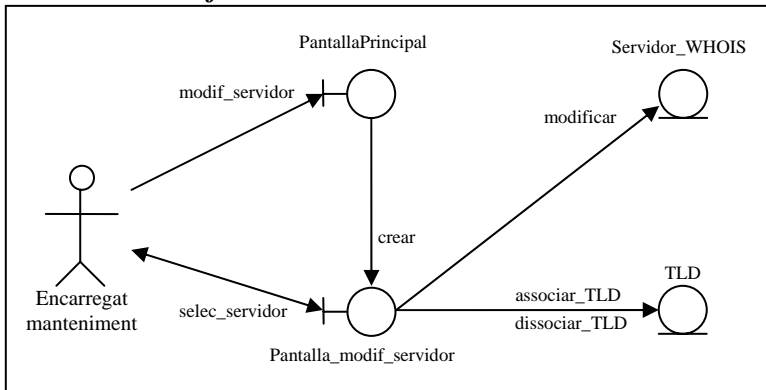
**Cas d'ús "Traçar IP":**



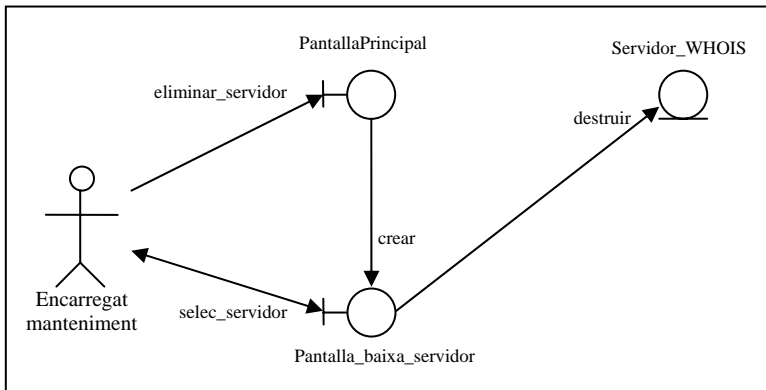
**Cas d'ús "Afegir servidor WHOIS":**



**Cas d'ús "Modificar servidor WHOIS":**



**Cas d'ús "Eliminar servidor WHOIS":**



8.2.7. Anàlisi de la interfície d'usuari

**PantallaPrincipal:**

**Manteniment BD**

Adreça IP  
o nom de host:

**Ruta geogràfica:**

(Espai per presentar gràficament la ruta fins la IP de destí. Representant cada node pel que s'ha de passar)

**Informació del node:**

(Informació textual sobre el node seleccionat. Inicialment el de

El menú “Manteniment BD” una vegada desplegat tindria un aspecte semblant a aquest:

**Manteniment BD**

- Afegir servidor WHOIS
- Modificar servidor WHOIS
- Eliminar servidor WHOIS

**Ruta geogràfica:**

(Espai per presentar gràficament la ruta fins la IP de destí. Representant cada node pel que s'ha de passar)

**Informació del node:**

(Informació textual sobre el node seleccionat. Inicialment el de

***PantallaAltaServidorWHOIS:***

**Alta de Servidors WHOIS**

**Servidors WHOIS:**

Nom (FQDN)	Cadena de consulta

Afegir

**TLD associats:**

Nom

Dissociar

**TLD dissociats:**

Nom

Associar

Sortir

***PantallaModifServidorWHOIS:***

**Manteniment de Servidors WHOIS**

**Servidors WHOIS:**

Nom (FQDN)	Cadena de consulta

Modificar

**TLD associats:**

Nom

Dissociar

**TLD dissociats:**

Nom

Associar

Sortir

***PantallaBaixaServidorWHOIS:***

**Baixa de Servidors WHOIS**

**Servidors WHOIS:**

Nom (FQDN)	Cadena de consulta

### 8.3. Diseny

#### 8.3.1. Disseny dels casos d'ús

##### *Cas d'ús “Traçar IP”*

Quan l'usuari introdueix la IP o nom de domini complet a l'espai indicat i prem el botó “Traçar IP”, corresponent a la crida del mètode `tracarIP`. Aquesta crida al constructor de la classe `Ruta`, passant-li per paràmetre l'adreça de destí d'aquesta ruta. En la instanciació se li assigna a l'objecte un `idRuta`, la data de càlcul d'aquesta i seguidament comença el procés de cerca i creació dels nodes que la componen: inicialment es crida un mètode privat de `Ruta` (similar a `traceroute`) que retorna una llista de les adreces dels nodes pels quals s'ha de passar per arribar a la IP de destí donada. Amb aquesta llista, crea un `Node` per cada IP trobada assignant-li la IP que li correspon, i al mateix procés de creació, el propi node fa ús de mètodes privats per aconseguir informació sobre la seva IP.

Finalitzat tot aquest procés, la crida `TracaIP` feta des de `PantallaPrincipal` rep l'objecte `Ruta` creat i l'utilitza per representar-lo gràficament amb l'ajuda de mètodes interns de `dibuixarNode` i `dibuixarSegment`.

##### *Cas d'ús “Afegir servidor WHOIS”*

L'usuari selecciona l'opció “Afegir servidor WHOIS” del menú “Manteniment BD” a `PantallaPrincipal`. Aquest últim instancia `PantallaAltaservidorWHOIS`, que demana les dades del servidor a afegir. L'usuari pot en tot moment cancel·lar l'operació prement el botó `Sortir`, o introduir les dades del servidor i premer `Afegir` per confirmar l'alta.

1. Si l'usuari ha optat per cancel·lar, es tanca i destrueix la instància de `PantallaAltaServidor` i per tant torna a `PantallaPrincipal`.
2. Si l'usuari confirma l'alta, es crea un objecte `ServidorWHOIS` i s'actualitza la pantalla per tal de reflectir l'alta al llistat. A partir d'aquí l'usuari té l'opció de `Sortir` o començar a assignar-li TLDs:
  - a. Si tria `Sortir`, es tanca i destrueix la instància de `PantallaAltaServidorWHOIS` i torna a `PantallaPrincipal`.
  - b. Si tria assignar algun TLD, l'usuari selecciona de la llista de TLDs dissociats aquell que vol associar i prem el botó “Associar”. Es crida el mètode `associar` de l'objecte `TLD` referenciat, que s'encarrega de crear la relació entre el servidor i el TLD.

##### *Cas d'ús “Modificar servidor WHOIS”*

L'usuari selecciona l'opció “Modificar servidor WHOIS” del menú “Manteniment BD” a `PantallaPrincipal`. Aquest últim instancia `PantallaModifServidorWHOIS`, que demana a l'usuari que seleccioni el servidor a modificar. L'usuari pot en tot moment cancel·lar l'operació prement el botó “Sortir”, o introduir les noves dades del servidor a l'espai indicat i premer “Modificar” per confirmar l'operació, o bé canviar les associacions d'un servidor seleccionat.

1. Si l'usuari ha optat per cancel·lar, es tanca i destrueix la instància de `PantallaAltaServidor` i per tant torna a `PantallaPrincipal`.



2. Si l'usuari confirma la modificació, es crida el mètode `modif` amb els paràmetres dels nous atributs de l'objecte `ServidorWHOIS` seleccionat. S'actualitza la pantalla per tal de reflectir els canvis.
3. Si tria assignar o dissociar algun TLD, l'usuari selecciona de la llista de TLDs dissociats aquell que vol associar i prem el botó "Associar" i/o viceversa. Es crida el mètode `associar/dissociar` de l'objecte TLD referenciat, que s'encarrega de crear/destruir la relació entre el servidor i el TLD.

### ***Cas d'ús "Eliminar servidor WHOIS"***

L'usuari selecciona l'opció "Eliminar servidor WHOIS" del menú "Manteniment BD" a PantallaPrincipal. Aquest últim instancia `PantallaBaixaServidorWHOIS`, que demana a l'usuari que seleccioni el servidor a donar de baixa. L'usuari pot en tot moment cancel·lar l'operació prement el botó "Sortir", o prémer "Eliminar" per confirmar l'operació.

1. Si l'usuari ha optat per cancel·lar, es tanca i destrueix la instància de `PantallaAltaServidor` i per tant torna a `PantallaPrincipal` sense destruir l'objecte.
2. Si l'usuari confirma la baixa, es crida el mètode `destruir` de l'objecte `ServidorWHOIS` seleccionat. S'actualitza la pantalla per tal de reflectir els canvis i s'espera per una nova baixa o sortir.

### ***8.3.2. Revisió del diagrama estàtic de disseny***

Per tal de simplificar els noms, les classes frontera canviaran el seu, simplement abreujant la paraula "Pantalla" que les precedeix amb "P", de manera que la classe `PantallaPrincipal` quedarà de la manera: `PPrincipal`, i així amb la resta.

Pel que fa a la reutilització de classes, no és possible, donat que no disposem de cap llibreria fins el moment.

Tampoc és necessària l'adaptació d'herència, donat que no hi ha cap cas d'herència entre classes al projecte.

En quant a la millora del rendiment, l'agrupació de classes que es va realitzar pel disseny del diagrama estàtic d'anàlisi, aconsegueix una bona optimització en aquest sentit.

La classe inicial del programa serà la classe `PPrincipal`, que es crea amb la pròpia execució de l'aplicació. Tant bon punt s'instancia, queda a l'espera d'ordres de l'usuari. El tipus de TAD escollit a la classe `Ruta` per emmagatzemar els seus components, els nodes; és un simple vector (array) o llista amb punt d'interès. Els motius d'aquesta elecció són:

- Ha d'ésser una estructura seqüencial.
- No es portaran a terme operacions de inserció pel mig ni esborrat d'elements.
- El nombre d'elements serà molt baix.
- Només ha de suportar les operacions: inserció i consulta seqüencial (cost constant per ambdues).

La resta de canvis respecte al diagrama estàtic d'anàlisi queden clarament reflectits al diagrama següent:

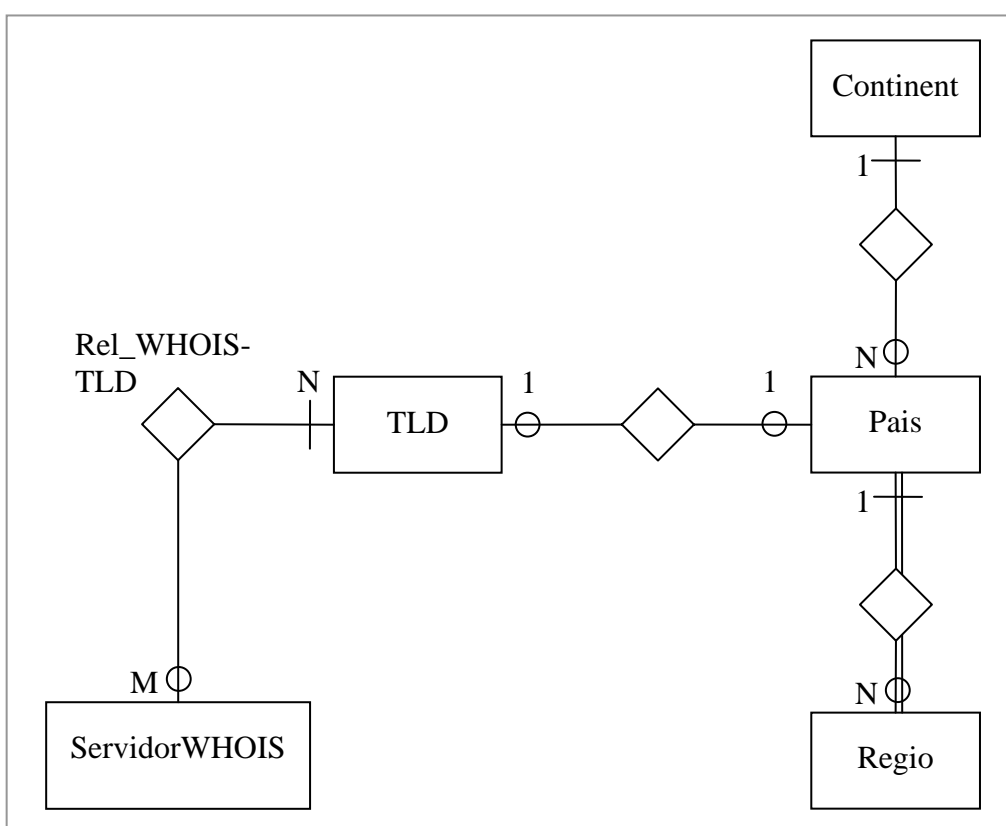


### 8.3.3. Disseny de la persistència

#### Disseny de la base de dades

Els objectes que necessitaran que la informació que contenen perduri després de la finalització del procés que els cridi són: *ServidorWHOIS*, *TLD*, *Regio*, *Pais* i *Continent*. Les entitats frontera i les entitats de classe *Ruta* i *Node* no necessiten pas conservar el valor dels seus atributs entre diferents execucions del programa.

Al procés de disseny de la persistència no ha sigut necessària cap tècnica de supressió d'herència, donat que no hi ha cap cas d'aquest tipus de relació entre les classes. El diagrama entitat-relació de la base de dades és el següent:



L'entitat *Regio* esdevé dèbil perquè el codi de cada regió només identifica la regió dins del país al qual pertany. Per tant, necessita de la relació amb *Pais* per tal de poder identificar de manera única cada regió.

El sistema gestor de base de dades (SGBD) escollit pel l'aplicació és el model relacional Microsoft Access 2002.

Fent la transformació del model ER al model relacional, tenim que la declaració de les taules de la base de dades és la següent:

***TServidorWHOIS***  
Nom: text

cadenaConsulta: text

### ***TTL***

nom: text

descripcio: text

### ***TRel\_WHOIS-TLD***

servidorWHOIS: text

clau forana de servidorWHOIS

TLD: text

clau forana de TLD

### ***TRegio***

codi: text

codiPais: text

clau forana de Pais

nom: text

latitud: real

longitud: real

### ***TPais***

codi: text

codiContinent: text

clau forana de Continent

nom: text

capital: text

latitud: real

longitud: real

### ***TContinent***

codi: text

nom: text

## ***Gestors de disc***

Tindrem els gestors de disc següents: GServidorWHOIS, GRel\_WHOIS-TLD, GTLD, GRegio, GPais, GContinent.

***GServidorWHOIS***: S'instancia en engegar el sistema per mitjà de l'operació crear.

L'operació llegirFila llegeix una fila de la taula TServidorWHOIS identificada per la clau primària.

L'operació filaAObjecte crea un objecte ServidorWHOIS i l'omple amb els valors d'una fila de TServidorWHOIS; també consulta a GRel\_WHOIS-TLD per trobar amb quins TLD té relació i els instancia.

El mètode objecteAFila copia un a un els valors dels atributs d'un objecte a la fila seleccionada de TServidorWHOIS, però també ho ha de fer per cadascuna de les noves relacions amb TLD creades, mitjançant les operacions corresponents dels gestor GRel\_WHOIS-TLD.

***GRel\_WHOIS-TLD***: S'instancia en engegar el sistema per mitjà de l'operació crear.

L'operació llegirFila llegeix una fila de la taula TRel\_WHOIS-TLD identificada per la clau primària.

L'operació consulta és polimòrfica i pot actuar de dues maneres: pot consultar quins servidors WHOIS serveixen informació per un TLD donat, o obtenir la llista de dominis TLD pels quals serveix un servidor donat; tot en funció del paràmetre passat. Per a qualsevol dels dos casos l'operació realitza la consulta a la taula TRel\_WHOIS-TLD i crea els objectes indicats pel resultat, per mitjà del seu gestor de disc.

L'operació altaRelacio, guarda a la taula TRel\_WHOIS-TLD una nova relació ServidorWHOIS-TLD.

L'operació baixaRelacio esborra la fila corresponent a una relació.

**GTLD:** S'instancia en engegar el sistema per mitjà de l'operació crear.

L'operació llegirFila llegeix una fila de la taula TTLTLD identificada per la clau primària.

L'operació filaAObjecte crea un nou objecte TLD amb els valors de la fila.

**GRegio:** S'instancia en engegar el sistema per mitjà de l'operació crear.

L'operació llegirFila llegeix una fila de la taula TRegio identificada per la clau primària.

L'operació filaAObjecte crea un nou objecte Regio amb els valors de la fila.

**GPais:** S'instancia en engegar el sistema per mitjà de l'operació crear.

L'operació llegirFila llegeix una fila de la taula TPais identificada per la clau primària.

L'operació filaAObjecte crea un nou objecte Pais amb els valors de la fila.

**GContinent:** S'instancia en engegar el sistema per mitjà de l'operació crear.

L'operació llegirFila llegeix una fila de la taula TContinent identificada per la clau primària.

L'operació filaAObjecte crea un nou objecte Continent amb els valors de la fila.

#### **8.3.4. Revisió del disseny de la interfície d'usuari**

El resultat obtingut a l'apartat d'anàlisi de la interfície d'usuari ja és suficientment vàlid, donat que ja s'han tingut en compte tots els aspectes de disseny necessaris.

## Capítol 3

### 9. Implementació

---

#### 9.1. Arquitectura

Donada la natura de lliure distribució del programari resultant, s'ha optat des d'un principi per construir una solució unificada i lleugera. Això ens ha obligat a sacrificar certa precisió a l'hora de representar alguns nodes, donat que, tenint la possibilitat de fer-ho, s'ha descartat l'emmagatzematge d'informació sobre localització de ciutats o qualsevol altra divisió menor de la de les regions d'un país, que com a mínim hauria multiplicat per vint el pes total del producte final. Per tant el màxim de resolució que obtindrem en el millor dels casos serà de la regió d'un país (província o estat, segons aquest) pel que fa a representació gràfica de cada node. En molts casos si és possible la deducció de la ciutat, però només a nivell informatiu. Es a dir, l'aplicació no serà capaç de representar el node segons les coordenades d'una ciutat (pel motiu de tamany de la BD esmentat al principi del paràgraf), i ho farà segons les coordenades de la seva regió.

Amb la finalitat de descriure amb facilitat el conjunt del programari el podríem seccionar en tres mòduls dels quals parlarem tot seguit:

1. **Tecnologia de cerca i inferència.**  
Amb la finalitat de cercar i estudiar tota la informació al seu abast per tal de poder deduir la localització de certa adreça.
2. **Base de dades.**  
Encarregat de portar a terme la part de persistència de l'aplicació i de facilitar diverses funcions que veurem tot seguit.
3. **Presentació de resultats.**  
Que té com a funció mostrar la informació processada en tres camps diferenciats: L'arbre de la ruta de nodes fins el destí, tota la informació relativa al node de destí i la representació gràfica sobre un mapa de la ruta esmentada.

#### 9.2. Mòdul 1. Tecnologia de cerca i inferència.

##### *Guió procedimental bàsic del mòdul:*

Quan el mòdul rep l'adreça de destí de la ruta, efectua les següents accions bàsiques:

1. Calcula la ruta per arribar-hi. Obtenim els nodes.
2. Fa un recorregut dels nodes de la ruta per:
  - a. Obtenir el nom de host de cada node.

- b. Aplicar “l’algorisme principal” per obtenir la localització en forma de les classes País o Regió per cada node.
  - c. Obtenir informació addicional gràcies a consultes WHOIS.
3. Cedeix el control al mòdul de presentació dels resultats.

### ***Descripció del mòdul:***

Aquesta part es compon de la tecnologia que permet al conjunt obtenir les dades necessàries per poder prendre la decisió sobre la localització d’una adreça d’Internet. Disposa de diversos algorismes per tractar de portar a terme aquesta tasca. De fet, aquests algorismes formen part d’un algorisme principal encarregat de decidir quina combinació dels seus membres és la més profitosa, segons cada cas; i aplicar-la per obtenir els resultats desitjats.

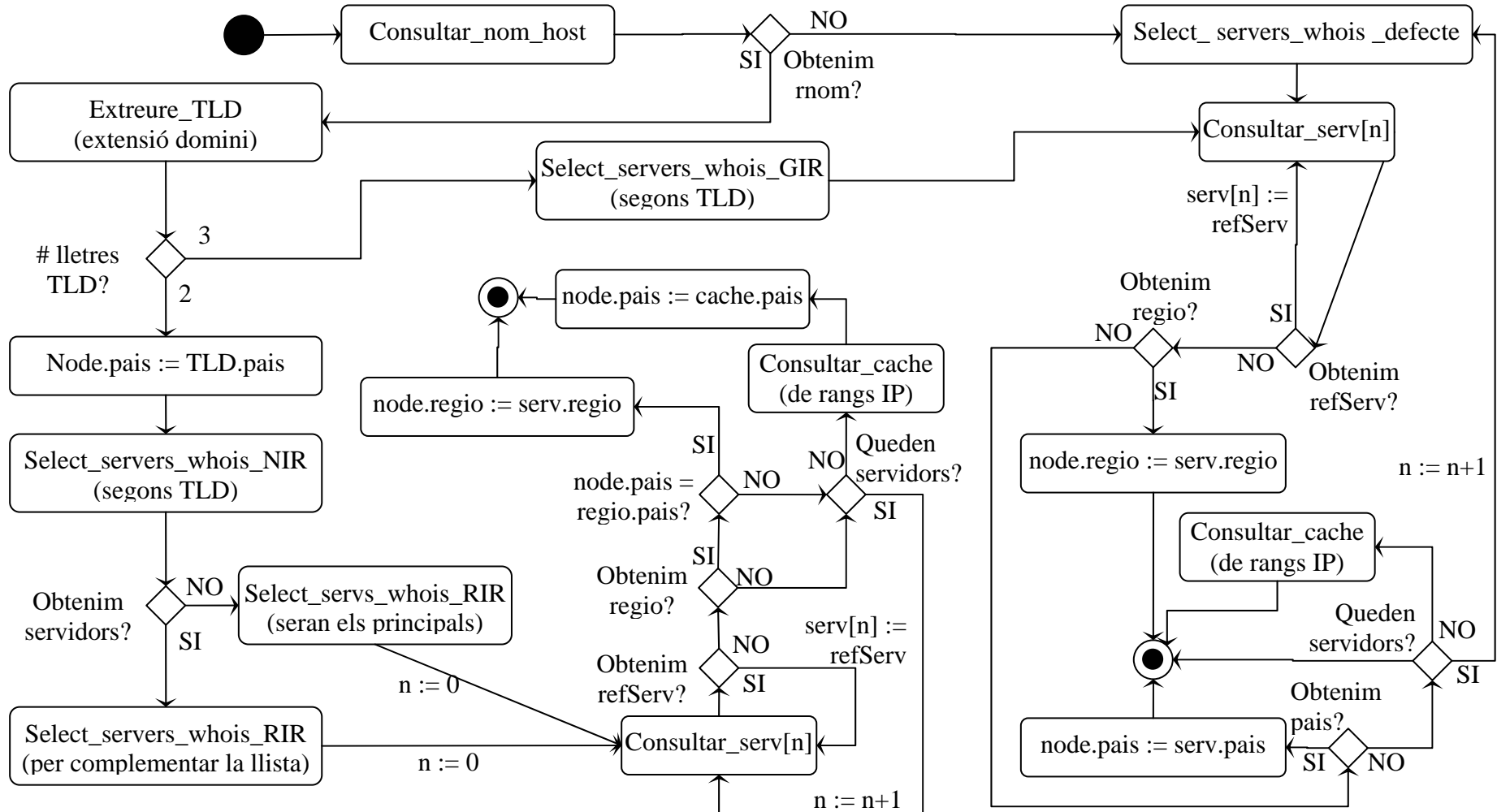
Dels mètodes citats a l’apartat “Sobre localització geogràfica”, els escollits per formar part del motor de cerca i deducció del programa han estat:

- Traceroute.
- DNS reverse lookup.
- Consultes WHOIS.

A més, s’ha dotat al sistema d’un últim recurs per casos en que no s’aconsegueix informació que permeti deduir la localització d’una IP. Es tracta d’una taula-caché amb més de 74.000 registres, que guarda la relació de blocs de IPs i la nacionalitat associada per cadascun.

La política de l’algorisme principal no és pas aplicar una o altra tècnica en funció del cas, sinó que combina les qualitats de cada mètode amb la intenció d’apropar-se a dos objectius prioritaris: reduir al màxim la possibilitat d’errada i aproximar al límit la localització de cada node.

A continuació es detalla el funcionament d’aquest algorisme principal:





Com es veu al diagrama, el primer pas és el de descobrir els nodes que formen part de la ruta. Aquesta funció acostuma a fer-se imitant el comportament d'una eina present a gairebé tots els sistemes operatius amb funcions de xarxa: *traceroute*.

Traceroute té el seu origen al sistema operatiu Linux, però les implementacions fetes per altres sistemes han seguit el mateix principi:

*L'enviament de paquets de petició d'eco amb temps de vida curts. El primer surt amb un valor de 1, que va augmentant a cada nova petició enviada. Tot amb la intenció de rebre paquets ICMP amb el missatge "temps de vida esgotat" de cadascun dels nodes intermitjos i el missatge final d'eco per part del node de destí.*

Degut al llenguatge escollit pel desenvolupament (Java), i la seva limitació en quant al tractament de missatges ICMP, m'he vist forçat a escollir entre dues solucions que restaven portabilitat al programa:

- Implementar una versió de traceroute amb codi natiu (per exemple amb C) i cridar-la des del codi Java.
- Cridar des de el codi Java la comanda de sistema traceroute (o tracert per Windows) i llegir el resultat.

La primera opció tancava les portes a tota possibilitat d'execució a altre sistema que no fos l'utilitzat pel desenvolupament. Hauria estat necessari implementar una versió de la funció per cada sistema operatiu potencial.

La segona opció, tot i suposar també una limitació de portabilitat, resultava més senzilla d'implementar i de dotar-la de portabilitat. De tal manera que si el software es vol executar en un sistema concret, només hem d'informar a l'aplicació de la comanda que aquest sistema disposa per fer un traceroute. Quan el programa s'executa en un sistema operatiu pel que no té associada cap comanda traceroute, pregunta a l'usuari per la sintaxi de la comanda i l'emmagatzema a la BD associada al sistema operatiu, de manera que no haguem de respondre aquesta pregunta a la següent execució.

Per defecte l'aplicació només està configurada per funcionar en sistemes Windows i Línux, tot i que només s'ha provat en un entorn Windows.

### **9.3. Mòdul 2. Gestió de base de dades.**

#### ***Descripció del mòdul:***

Tot i tractar-se d'una solució autònoma, compta amb una base de dades local del tipus relacional que ha estat dissenyada i implementada per tal d'accelerar al màxim les possibles consultes. Conté la informació necessària per poder portar a terme les següents funcions:

- Traducció d'un país o regió a coordenades geogràfiques.
- Possibilitat de deducció del país al qual pertany una regió i el continent al qual pertany un país.
- Deducció dels servidors WHOIS adients segons el TLD.
- Disposició d'una taula-caché com a últim recurs per a la ubicació, gràcies a una relació de rangs d'IP amb la nacionalitat a la qual pertanyen.

Així doncs, el contingut bàsic, sense entrar en detalls, de la base de dades seria el següent:

- Informació sobre regions: nom, longitud, latitud i país al qual pertany.
- Informació sobre països: nom, longitud, latitud i continent al qual pertany.
- Informació sobre continents: nom.
- Informació sobre servidors WHOIS: nom complet i paràmetres útils per la consulta.
- Informació sobre TLDs: nom.
- Informació sobre relacions entre servidors WHOIS i TLDs: servidor, TLD i prioritat.
- Informació sobre blocs d'IPs: IP inicial, IP final, país.
- Informació sobre configuració de l'aplicació: propietat, valor.

Tractant-se d'un mòdul passiu, no aplica la descripció de la seva activitat.

#### **9.4. Mòdul 3. Presentació de resultats.**

El mòdul de presentació de dades també admetria una subdivisió del seu sistema:

- Part de presentació de la ruta en format arbre.
- Part de presentació de la ruta en format gràfic.
- Part de presentació de la informació recollida sobre l'adreça de destí.

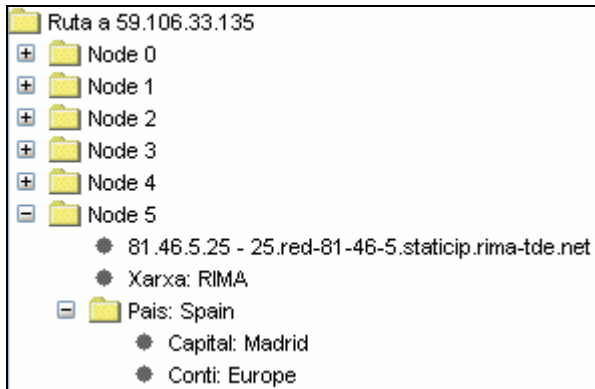
##### ***Guió procedimental bàsic del mòdul:***

Tan bon punt el mòdul de cerca i inferència ha cedit el control al mòdul de presentació, es desencadena la següent seqüència d'accions:

1. Recorregut dels nodes de la ruta per:
  - a. Determinar els segments, delimitats per dos nodes amb localització correcta. Per a cada segment, dibuixar al mapa:
    - i. una línia com a representació del segment.
    - ii. els nodes origen i destí del segment coincidint amb els extrems de la línia dibuixada al punt anterior.
    - iii. Dibuir de forma diferenciada (en forma i color) i sobre la línia del segment, els nodes intermitjos, que són els que no tenen localització associada.
  - b. Crear una nova branca a l'arbre de ruta per aquest node i:
    - i. Crear nova branca filla del node amb la IP i el nom de host.
    - ii. Crear nova branca filla del node amb la xarxa del node.
    - iii. Crear nova branca filla del node amb la ciutat del node.
    - iv. Crear nova branca filla del node amb Regió/País al qual pertany.
    - v. Crear nova branca filla de l'anterior amb País/Continent al qual pertany
  - c. Donar format HTML a la informació WHOIS obtinguda pel node de destí i mostrar-la en una secció de la finestra creada a tal efecte.

**Descripció de les diferents parts del mòdul:**

Pel que fa a la presentació de la ruta en forma d'arbre es tracta simplement de la construcció d'un arbre basat en la classe *JTree* de Java, que ens ha permès realitzar una representació comprimida, jeràrquica i fàcil de consultar de la informació relativa a cada node de la ruta.



En quant a la part de presentació gràfica, s'ha utilitzat un mapamundi a escala com a imatge de fons. Sobre una capa superior es dibuixen els nodes que componen la ruta i les línies fictícies que els uneixen.



Els objectes de localització (Regió o País) deduïts i assignats a cada node segons el mòdul "Tecnologia de cerca i inferència", incorporen les coordenades que permetran la representació de cada node al plànol.

La traducció de les coordenades geogràfiques reals a les del mapa sobre el qual es pretén representar, es realitza gràcies a les següents fórmules:

$$\text{longitud} = (\text{nodegetLongitud}() * \text{mapa.getWidth}() / 360) + \text{offsetX} - \text{nodeWidth} / 2;$$

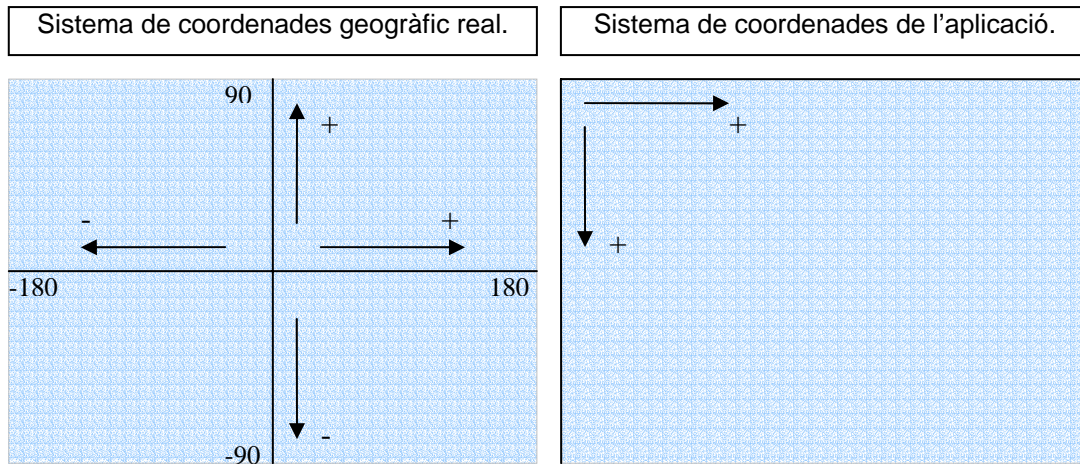
$$\text{latitud} = (\text{node.getLatitude}() * -\text{mapa.getHeight}() / 180) + \text{offsetY} - \text{nodeHeight} / 2;$$

on:

- `node.getLongitud()` és igual a la coordenada de longitud del node.
- `node.getLatitude()` és igual a la coordenada de latitud del node.
- `Mapa.getWidth()` és l'amplada de la imatge que representa el mapa base.
- `Mapa.getHeight()` és l'alçada de la imatge que representa el mapa base.
- `offsetX` és igual a: `mapa.getWidth() / 2`

- offsetY és igual a:  $\text{mapa.getHeight()} / 2$

La raó de ser de les variables offsetX i offsetY és la diferència de sistemes de coordenades, que es mostra gràficament a la següent figura:



La secció que s'encarrega de la representació de la resta d'informació aconseguida per cada node ho fa en format text/html. Bàsicament el que fa es introduir una capçalera que anuncia la ruta per la que es mostra informació i tot seguit afegeix una enumeració dels nodes amb la informació referent a cadascun.

La següent captura d'un exemple permet fer-se una idea més ràpidament:

```

Informació WHOIS:

Ruta amb destí: 59.106.33.135

Node 0 - 81.43.3.77:

Resultat WHOIS:
% This is the RIPE Whois query server #1.
% The objects are in RPSL format.
%
% Note: the default output of the RIPE Whois server
% is changed. Your tools may need to be adjusted. See
% http://www.ripe.net/db/news/abuse-proposal-20050331.html
% for more details.
%
% Rights restricted by copyright.
% See http://www.ripe.net/db/copyright.html

% Note: This output has been filtered.
% To receive output for a database update, use the "-B" flag.

% Information related to '81.40.0.0 - 81.43.86.255'

inetnum: 81.40.0.0 - 81.43.86.255
netname: RIMA
descr: Telefonica de Espana SAU
descr: Red de servicios IP
descr: Spain
country: ES
admin-c: LJP5-RIPE
tech-c: FLT14-RIPE
status: ASSIGNED PA
    
```

## Capítol 4

### 10. Manual d'instal·lació i ús d'IP-GeoTracer

---

#### **Manual d'instal·lació:**

VEURE ARXIUS: *instalacio.txt* i *execucio.txt*, situats a la carpeta *Aplicacio* del paquet que conté aquest document.

#### **Manual d'ús:**

La principal funció de l'aplicació és la de trobar tota la informació relacionada amb una IP donada i representar gràficament la seva ruta. Per poder realitzar aquesta tasca el primer que ens demana el programari és que introduïm una adreça IP vàlida o un nom de host complet (FQDN) a l'espai etiquetat amb "Adreça IP o nom de host:".

Petició amb nom de host:

Adreça IP o nom de host:	<input type="text" value="www.uoc.edu"/>	<input type="button" value="Traçar"/>
--------------------------	--	---------------------------------------

Petició amb adreça IP:

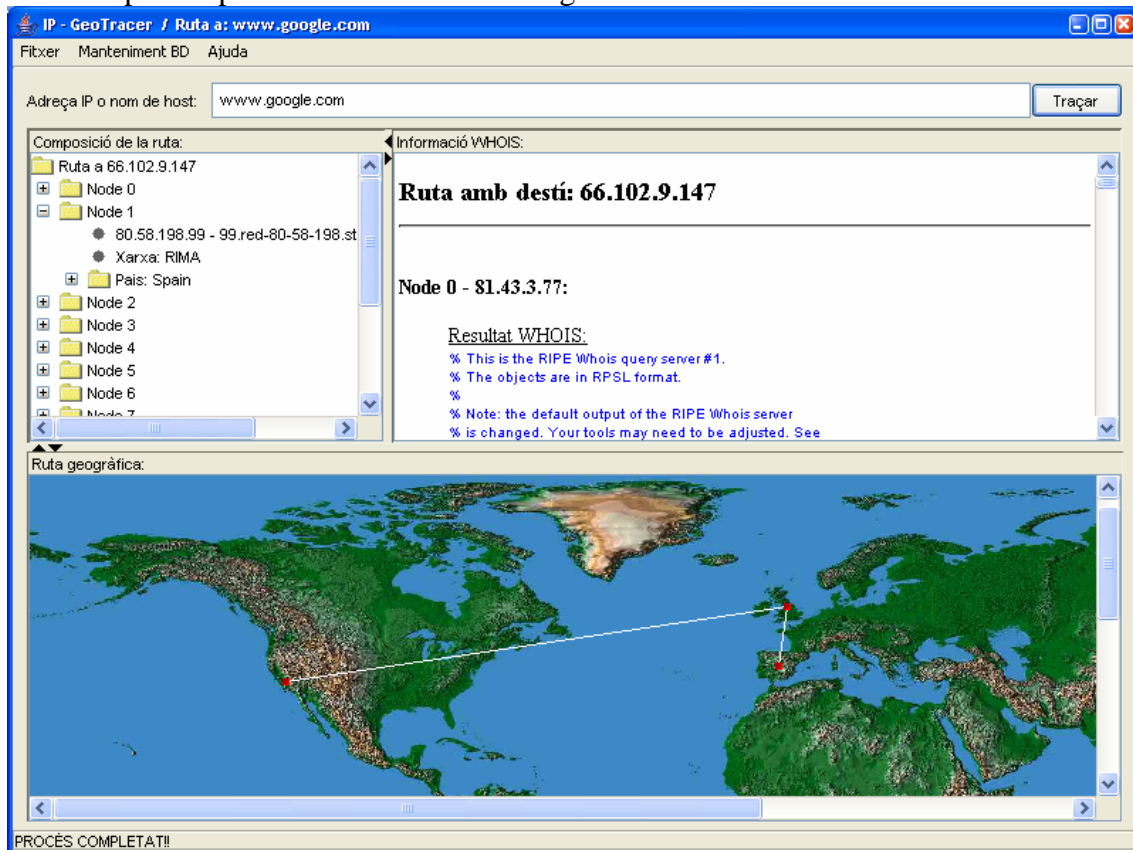
Adreça IP o nom de host:	<input type="text" value="80.39.77.254"/>	<input type="button" value="Traçar"/>
--------------------------	---	---------------------------------------

tot el que s'ha de fer a continuació és prémer el botó "Traçar" o polsar <Enter>. Immediatament començarà el procés de cerca i tractament de la informació. Aquest procés pot resultar una mica feixuc i arribar a mantenir el programa "congelat" durant uns minuts. La mitjana d'espera és d'aproximadament 5~6 segons/node. Per tant una ruta de 20 nodes ens pot mantenir a l'espera entre 100 i 120 segons.

Tan bon punt finalitzi la cerca i el tractament de les dades obtingudes es representarà a la mateixa pantalla el resultat del procés que quedarà dividit en les tres seccions de la finestra ja explicades a l'apartat "9.4. Mòdul 3. Presentació de resultats":

- Arbre de ruta
- Informació WHOIS de l'adreça donada
- Representació geogràfica de la ruta

Un exemple del possible resultat seria el següent:



### *Interpretació del resultat:*

La interpretació del resultat és molt intuïtiva, i de fet només requereix una breu explicació en quant a la simbologia utilitzada per dibuixar els nodes al mapa. Es realitza certa diferenciació entre nodes segons el següent criteri:

- Node amb localització concreta (país o regió).  
Es representa amb el símbol: ■
- Node amb localització concreta (regió), pel que s'han d'aproximar les coordenades a les del país per manca d'informació a la BD de les coordenades de la regió.  
Es representa amb el símbol: ■
- Node sense localització. Es tracta d'un node pel que no s'ha pogut aconseguir informació sobre la seva ubicació.  
Es representa amb el símbol: ●

### ***Possibilitats d'interacció amb el resultat:***

#### ***Zoom In / Zoom Out.***

És possible realitzar una ampliació o reducció de la vista del gràfic que representa la ruta. Resulta tan simple com fer doble clic a sobre d'aquesta.

A cada acció d'ampliació/reducció s'escala la imatge original a l'escala anterior + 0.5.

El màxim que es pot escalar la imatge és fins el doble del seu tamany natural (2.0).

L'escala predeterminada és 0.5. Es a dir, una reducció a la meitat de l'original.

Els valors possibles són doncs: 0.5; 1.0; 1.5; 2.0

#### ***Navegació per l'arbre de ruta.***

A la representació en format d'arbre jeràrquic de la ruta és possible veure la informació relacionada amb cada node si obrim la seva carpeta fent clic al símbol "+" a la seva esquerra o doble clic sobre el mateix.

Si el node que hem desplegat té assignada una localització, podrem actuar de la mateixa manera, desplegant la informació relativa a la regió o país, segons el cas, i així successivament.

#### ***Copy & Paste***

Les seccions rotulades com: "Composició de la ruta:" i "Informació WHOIS:" tenen un contingut que es pot seleccionar, copiar i enganxar a altre lloc.

El contingut de "Informació WHOIS:" té format HTML, de manera que es pot copiar a un document amb format, sense que perdi el format propi.

Si volem iniciar una nova cerca, només necessitem tornar a escriure la nova adreça de destí com s'explicava a l'inici de l'apartat i tornar a prémer el botó "Traçar". En finalitzar el nou procés de cerca i tractament de dades els nous resultats substituiran els antics a tots tres sectors de la finestra.



## Capítol 5

### 11. Millores per a futures versions

---

Tot i que el projecte comprèn una aplicació finalitzada i totalment funcional, hi ha certs aspectes que el creador del projecte reconeix com a millorables. De fet algunes d'aquestes millores o ampliacions haurien estat implementades si no fos, principalment, per la limitació temporal. Per tant, serien millores a portar a terme en futures versions.

Algunes de les propostes apunten cap a optimitzacions funcionals o de rendiment, altres cap a una millora en la interacció de l'usuari amb el resultat, i altres simplement es preocupen per una millora estètica de la interfície gràfica. Aquí només es descriuran les que considero més importants: funcionalitat i rendiment.

#### ***Perfeccionar la implementació de la funció “traceroute”.***

Aquesta primera versió del projecte ha resolt el problema del descobriment dels nodes de la ruta mitjançant una crida a una comanda equivalent a traceroute del sistema operatiu, degut a dificultats del llenguatge Java (veure apartat “Mòdul 1. Tecnologia de cerca i inferència” del capítol “Implementació”).

Això obliga al procés a esperar la finalització de la comanda per poder examinar el resultat i extreure'n els nodes.

En el moment de desenvolupament d'aquesta part, no vaig contemplar la possibilitat d'implementar la funció amb l'enviament de datagrames UDP a un port de destí no utilitzat, en comptes de missatges ICMP. El que quedaria per avaluar és si els actuals JDKs de Java permeten el tractament del missatges ICMP rebuts, tot i no permetre l'enviament.

Si aquesta proposta fos factible, permetria optimitzar la implementació que fan els sistemes operatius del traceroute amb alguns canvis com:

- l'enviament paral·lel (contra el seqüencial) dels datagrames. En comptes d'esperar a rebre la resposta del datagrama amb temps de vida  $x$  per enviar el de temps de vida  $x+1$ , els enviaria tots a la vegada i esperaria les respostes de cadascun per separat.
- provar primer si rebem resposta del sistema de destí, abans de començar a descobrir la ruta,
- evitar l'espera per la resolució de nom de la IP de cada node, i llançar aquesta resolució com a fil (thread) paral·lel, per tal que no congelés el procés de cerca dels nodes.

Això no només milloraria molt notablement el temps de resposta del procés traceroute, sinó que també repercutiria directament en una correcció del temps de resposta de l'algorisme de cerca i inferència de localització dels nodes, donat que, per una banda, s'estalviaria haver de resoldre el nom canònic de cada node i, per altra, permetria avançar el llançament de les consultes WHOIS per cada node.

#### ***Augmentar el rendiment utilitzant threads concurrents i sincronitzats.***

Actualment es mostra el resultat total al final del procés y només llavors. Mitjançant l'ús de threads no només disminuiria el temps total de procés mitjançant tasques



paral·leles, sinó que també seria possible anar mostrant el resultat a mida que s'obtingués. Fins i tot seria possible fer una primera presentació basada en el contingut de la caché de rangs IP, i a continuació anar confirmant els nodes segons la informació que s'obtingués d'altres fonts.

El principal punt del programa on es podria fer ús de threads per disminuir els temps de resposta és al procés de cerca i inferència. Com he avançat al punt anterior, podríem utilitzar threads per la resolució de nom mentre el procés principal continua cercant els nodes de la ruta. També per consultes WHOIS, presentació gràfica dels nodes, presentació d'informació a l'usuari sobre el progrés del procés, etc.

Tot això significaria una millora del temps total de resposta de l'aplicació i suprimiria la sensació actual de procés "congelat".

### ***Migració del sistema de base de dades a altre sistema multiplataforma.***

El sistema gestor de base de dades utilitzat actualment és: *Microsoft Access*. Aquest sistema de dades obliga a tenir un driver específic a cada sistema operatiu on s'executi l'aplicació per aconseguir que aquesta funcioni correctament.

Si es migrés l'estructura i la informació d'aquesta BD cap a altre sistema com *JDataStore* o qualsevol altre gestor multiplataforma, aconseguiríem un major índex de portabilitat de l'aplicació.

### ***Manteniment de la base de dades des de la interfície d'usuari.***

Anteriorment s'ha parlat de que la base de dades no precisa de manteniment, i que en cas que se'n vulgui fer, és molt bàsic, donat que només s'hauria d'introduir, si es desitges, algun nou servidor whois, eliminar algun que ja no existís i canviar la preferència d'aquests o les seves relacions amb els TLD existents. El fet que parlem de baix manteniment, és perquè realment no són valors que sofreixin gaire variació. Si per exemple deixa de funcionar un servidor, el programa s'encarrega de buscar una altre, de la seva llarga llista, que li faci servei.

Tot i així, resultaria interessant poder portar a terme aquest manteniment bàsic, i encara més interessant poder-ho fer des de la mateixa aplicació.

Aquesta opció ja es va tenir en compte a l'anàlisi i el disseny, però no s'ha pogut implementar per la manca de temps ja esmentada.

### ***Administració intel·ligent de servidors whois.***

La consulta a servidors whois és una de les tècniques més utilitzades per l'algorisme principal de cerca i inferència de la localització.

Tenint en compte aquest fet, seria convenient dotar l'aplicació d'un motor d'aprenentatge intel·ligent en funció de l'avaluació dels resultats obtinguts per aquests. Segons aquest mecanisme, progressivament anirien quedant com a servidors preferents aquells que oferissin un millor servei. Els possibles valors a avaluar podrien ser:

- idoneïtat del resultat pel TLD consultat,
- percentatge d'èxit en la connexió,
- temps de connexió necessari,
- definició en la localització segons la seva resposta.

D'aquesta forma, aconseguiríem cada cop una aplicació més ràpida i eficaç, i a la vegada disminuiríem encara més el manteniment de la base de dades.

## Capítol 6

### 12. *Conclusions*

---

Com hem pogut veure, el problema de la localització geogràfica no és un problema fàcil de resoldre. Des de la meua modèstia i la millor intenció, he provat d'oferir una solució tan acurada com he pogut.

Tot i quedar bastant satisfet amb el treball fet, m'ha quedat "clavada una espina" per les possibles millores que podria haver incorporat; i que segurament farien d'aquest, un producte molt superior.

No puc negar que gràcies a aquest projecte he après molt i en sectors molt diversos, però a la vegada també molt interessants. Sobretot, si tenim en compte que actualment treballa com a tècnic administrador de sistemes i xarxes d'una mitjana empresa. Amb això vull dir que tot plegat ha servit per despertar una mica més la meua curiositat, i que potser no només faci una nova versió millorada de IP-GeoTracer, com deia fa uns instants, sinó que, és possible que aprofundeixi encara més i acabi fabricant les meves pròpies eines, que m'ajudin tant en el meu desenvolupament personal com professional.

## Glossari

---

- Adreça IP:** Una **adreça IP** és un número que identifica a una interfície d'un dispositiu (habitualment un ordinador) dins d'una xarxa que faci servir el protocol IP.
- Caché:** Un **caché** és un conjunt de dades duplicades d'altres originals, amb la propietat que les dades originals són costoses d'accedir, normalment en temps, respecte a la còpia en el caché. Quan s'accedeix per primera vegada a una dada, es fa una còpia en el caché; els accessos següents es realitzen a l'esmentada còpia, fent que el temps d'accés aparents a la dada sigui menor.
- Consulta DNS:** Consulta a un servidor de nom basat en el protocol TCP/IP i un sistema de pregunta/resposta que retorna l'adreça IP d'un host en preguntar pel nom d'aquest.
- Consulta WHOIS:** Veure: WHOIS.
- DNS:** El Domain Name System (**DNS**) és una base de dades distribuïda i jeràrquica que emmagatzema informació associada a noms de domini en xarxes com Internet. Encara que com a base de dades el DNS es capaç d'associar diferents tipus d'informació a cada nom, els usos més comuns són l'assignació de noms de domini a adreces IP i la localització dels servidors de correu electrònic de cada domini.
- Enrutador:** Veure: Router.
- FQDN:** Un **FQDN** (Fully Qualified Domain Name) es un nom comprensible per persones que inclou el nom de la computadora i el nom de domini associat a la mateixa. Per exemple, donada la computadora anomenada «serv1» i el nom de domini «bar.com», el FQDN serà «serv1.bar.com», i alhora un FQDN associat a Serv1 podria ser <<post.serv1.bar.com>>.
- ICMP:** El Protocol de Control de Missatges d'Internet (**ICMP** per les seves sigles en anglès *Internet Control Message Protocol*) és un dels protocols centrals de la suite de protocols d'Internet. Principalment l'utilitzen els Sistemes operatius de les computadores en una xarxa per enviar missatges d'error, indicant per exemple que un servei determinat no esta disponible o que un router o host no pot ser localitzat.
- IP:** El protocol d'Internet (**IP**, de les seves sigles en anglès *Internet Protocol*) és un protocol no orientat a connexió utilitzat tant per

l'origen com per el destí per a la comunicació de dades a través d'una xarxa de paquets commutats.

- ISP:** **ISP** és l'acrònim en anglès d'*Internet Service Provider* (Proveïdor de Serveis d'Internet), empresa dedicada a connectar a Internet la línia telefònica dels usuaris, xarxes diferents e independents, ambdues.
- LAN:** *Local Area Network* (en català "Xarxa d'Àrea Local), més comunament referida pel seu acrònim **LAN**, es refereix a les xarxes locals d'ordinadors.
- LIR:** **Local Internet Registry**. Organismes encarregats de realitzar concessions de blocs d'adreces d'Internet a nivell local. Concedeixen a proveïdors ISP.
- Módem:** Acrònim de les paraules **modulador/demodulador**. El mòdem actua com equip terminal del circuit de dades (ETCD) permetent la transmissió d'un flux de dades digitals a través d'una senyal analògica.
- NIR:** **National Internet Registry**. Organismes encarregats de realitzar concessions de blocs d'adreces d'Internet a nivell nacional. Concedeixen a LIR i proveïdors ISP.
- Node:** Sistema de nivell 3 del model OSI en el que conflueixen part de les connexions amb altres nodes que comparteixen les seves mateixes característiques. Tots aquests nodes s'interrelacionen entre ells d'una manera no jeràrquica i conformen el que en termes sociològics o matemàtics anomenem xarxa. En el context d'aquest projecte, la xarxa és Internet.
- Open Source:** Codi obert (**open source** en anglès) és el terme per el que es coneix al software distribuït i desenvolupat en una determinada manera. Aquest terme va començar a utilitzar-se en 1998 per alguns usuaris de la comunitat del software lliure, intentant fer-lo servir com a reemplaçament a l'ambigu nom original, en anglès, del software lliure (free software).
- RIR:** **Regional Internet Registry**. Són els organismes encarregats de realitzar concessions de blocs d'adreces d'Internet a nivell regional. Concedeixen a organismes NIR, LIR o a proveïdors ISP.
- Router:** El **router** (*enrutador* o *encaminador*) es un dispositiu hardware o software d'interconnexió de xarxes d'ordinadors/computadores que opera a la capa 3 (nivell de xarxa) del model OSI. Aquest dispositiu interconnecta segments de xarxa o xarxes senceres. Fa passar paquets de dades entre xarxes prenent com a base la informació de la capa de xarxa. El router pren decisions lògiques

respecte a la millor ruta per l'enviament de dades a través d'una xarxa interconnectada i després dirigeix els paquets fins al segment i el port de sortida adequats.

- Threads:* Molts llenguatges de programació (com Java), i altres entorns de desenvolupament suporten els anomenats fils (en anglès, **threads**). Els fils són similars als processos en que representen una seqüència simple d'instruccions executada en paral·lel amb altres seqüències. Els fils són una forma de dividir un programa en dues o més feines que corren simultàniament.
- TLD :* Domini d'Internet. Tecnicament, un domini d'Internet (o **TLD**, *Top-level Domain*) és cada node que descendeix del domini arrel ".", i representa una subxarxa nominal (classificada per noms, diferent d'una subxarxa protocolar classificada per adreces IP) dins del *Sistema de Noms del Domini* (DNS, per les seves sigles en anglès).
- Traceroute:* **Traceroute** és una eina de diagnòstic de xarxes que permet seguir la pista dels datagrames que van des d'un host (punt de xarxa) a un altre. S'obté a més una estadística de les velocitats de transmissió d'aquests paquets.
- UDP:* *User Datagram Protocol* (**UDP**) és un protocol del nivell de transport no orientat a connexió, basat en l'intercanvi de datagrames. Permet l'enviament de datagrames a través de la xarxa sense que s'hagi establert prèviament una connexió, ja que el propi datagrama incorpora suficient informació de direccionament a la seva capçalera.
- WHOIS:* **WHOIS** és un producte TCP basat en preguntes/respostes que s'utilitza per consultar una base de dades per determinar les dades del propietari d'un nom de domini o d'una adreça IP en la Internet.

## *Bibliografia*

---

Getting geographical Information using an IP Addr. Per Ankit Fadia  
[ankit@bol.net.in](mailto:ankit@bol.net.in)

The IP-to-Country Handbook  
<http://ip-to-country.webhosting.info/>

IANA IP Address Services  
Internet Addressing Numbers Authority  
[www.iana.org](http://www.iana.org)

RIPE Database Reference Manual  
Réseaux IP Européens  
[www.RIPE.net](http://www.RIPE.net)

American Registry for Internet Numbers  
[www.arin.net](http://www.arin.net)

Asia Pacific Network Information Center  
[www.apnic.net](http://www.apnic.net)

Latino America y Caribe Network Information Center  
[www.lacnic.net](http://www.lacnic.net)

Definicions de mots tècnics del sector informàtic  
[es.wikipedia.org](http://es.wikipedia.org)

EDUCYPEDIA - The educational encyclopedia.  
<http://users.pandora.be/educyclopedia/index.htm>

Consulta de FAQs  
[www.faqs.org](http://www.faqs.org)

Antionline IP Locator  
<http://www.antionline.com/index.php>

GeoBytes - IP Address Locator Tool  
<http://www.geobytes.com/Default.htm>