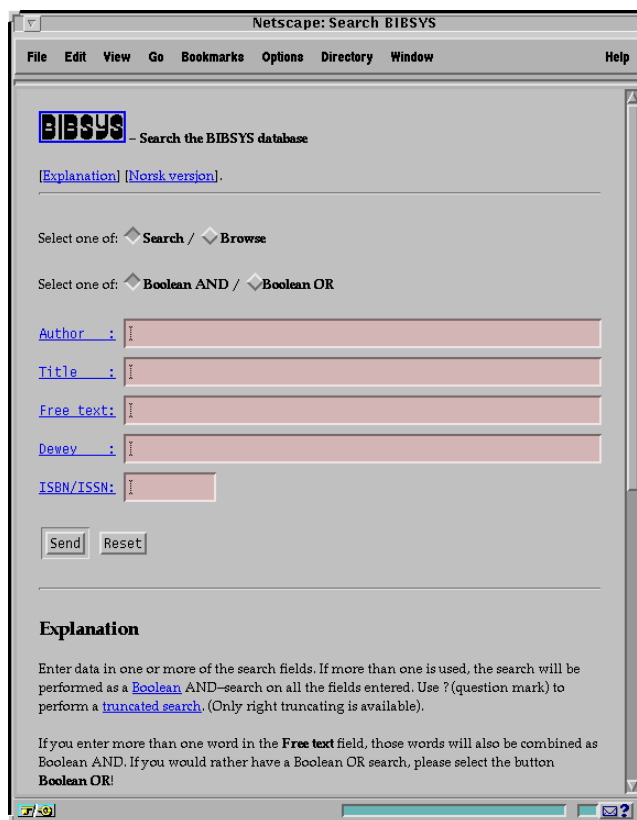


KÖNYVTÁRI KATALÓGUSOK KÖZPONTI LEKÉRDEZÉSE

Balázs László, lbalazs@giant.lib.klte.hu
 KLTE Egyetemi és Nemzeti Könyvtár, Informatika Osztály

1. Bevezetés

Az elmúlt években világossá vált, hogy szükség van arra, hogy több azonos gyűjtőkörű, illetve egymáshoz közel lévő könyvtár lekérdezhető legyen egy terminálról. Így az érdeklődőknek nem kell végigjárni a könyvtárakat, hanem csak abba a könyvtárba kell elmennie, ahol a könyv biztosan megvan. Még jobb megoldás az, ha egy kérdéssel, egyszerre lekérdezhető a rendszerbe bekapcsolt összes könyvtár, illetve azok egy kiválasztott csoportja. Így nem kell a felhasználónak különböző könyvtári rendszerek kezelését megtanulnia, elég a központi lekérdező rendszer - nevezzük brókernek - kezelését megtanulnia. Ilyen rendszer például a norvég BIBSYS:



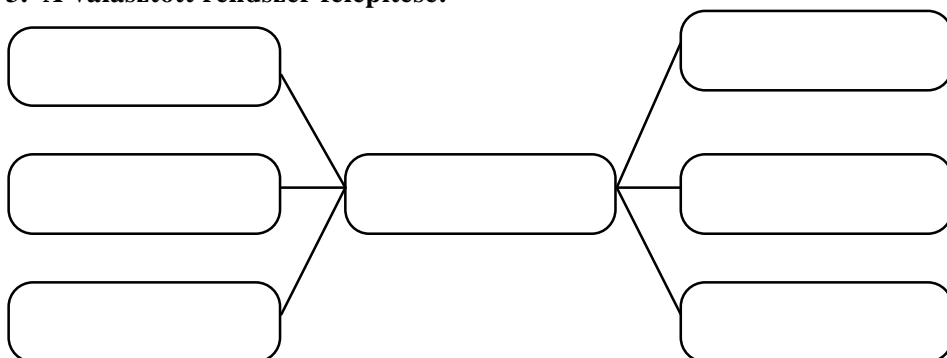
2. A központi lekérdezés lehetséges módozatai:

1. Union Catalog vagy egyesített katalógus
 Ebben az esetben az adatokat egy központi katalógusban tárolják és a keresés is ott történik. Előnye, hogy a keresés nem terheli a könyvtárak katalógusát, nem lehet vonal hiba, nem romolhat el egyetlen katalógus. Mindezek miatt megbízhatóbban működik, azonban különböző könyvtári rendszerek egyesített katalógusa csak a könyvtári rendszerek összehangolt átírásával valósítható meg.
2. Központi lekérdező rendszer és különálló katalógusok

Ez a megoldás sokkal egyszerűbben megvalósítható. Találnunk, vagy készítenünk kell egy közös interfészt a rendszerekhez, és egy központi lekérdező rendszert - egy brókert. Ilyen szabványos interfész lehet a ¹Z39.50-es szabványt megvalósító felület. A megoldás hátránya, hogy a különböző katalógusok különbözőképpen lehetnek indexelve különböző mezőket tartalmazhatnak, így nem biztos, hogy konzisztens lesz a keresés eredménye. Igen nagy előnye azonban, hogy a könyvtári rendszer átírása nélkül megvalósítható.

Mivel a másodikként leírt rendszer sokkal gyorsabban és olcsóbban megvalósítható, az IIF egy ilyen projectet indított. Ennek a megvalósításán dolgozunk néhány hónapja. A továbbiakban erről fogok részletesebben beszélni.

3. A választott rendszer felépítése:



Az adatbázis szerver:

Az adatbázis szerver alatt értjük a könyvtári rendszert és annak valamilyen szabványos interfészét. Ez végzi a keresést az adatbázisban.

A bróker:

A bróker, megfelelő formára alakítva továbbítja a klientsől származó kérdést a különböző adatbázis szervereknek. Összegyűjti, rendszerezzi esetleg konvertálja azok válaszait, és visszajuttatja a kliensnek.

A kliens:

Összeállítja a kérdést, megjeleníti a találatokat.

A bróker és a kliens kapcsolata:

A bróker és a kliens között terminálkapcsolatot is használhatunk, azonban sokkal használhatóbb, még inkább felhasználó barát lesz a rendszerünk, ha valamilyen grafikus klients használunk. A megvalósítást gyorsíthatjuk és szabványosabbá tehetjük, ha valamilyen már létező szabványos kliens - szerver kapcsolatot használunk. Ebben az esetben a legjobb választás a ²World Wide Web kapcsolat. Ez azt jelenti, hogy a kliens egy szabványos WWW kliens a bróker pedig egy ³CGI program, vagyis egy kiegészített WWW szerver. **Ebben az esetben nem csak szabványosabbá tettük megvalósításunkat, hanem használhatóvá is vált minden elterjedt platformon!**

A database server és a bróker kapcsolata:

Erre használhatnánk a szinte már szabványnak számító Z39.50 protokollt, azonban rá kell jönnünk, hogy a ma Magyarországon elterjedt könyvtári rendszerek csak ígérik ezt a szabványos interfészt, de egyik sem valósítja meg 100%-ban. A távlati megoldás azonban biztosan ez lesz.

Felvetődik azonban egy másik kérdés is: a magyar vonalakon érdemes-e használni egy párbeszédre épülő protokollt, nem jobb választás-e az egy kérdés egy válasz megoldás, amelyet a WWW rendszerek is használnak?

4. A megvalósítás közben felmerült kérdések:

¹ A szabvány részletes leírása megtalálható: URL: <ftp://ftp.loc.gov/pub/z3950/official/>

² A WWW rendszerről bővebben: URL: <http://www.w3.org/pub/WWW/WWW/>

³ A CGI programokról olvashatunk például: Christian Neuss, Johan Vromans: The Webmasters Handbook

Az adatbázisok szinkronizációja:

- Ki mit ért az alatt, hogy szerző, cím ... ?
- Kulcsszavas keresés vagy teljes mező illesztés? Lehetséges, hogy a különböző adatbázisok mind másképpen értelmezik a kérdést. Mi történjen ekkor?
- Mi történjen, ha olyan indexre történik a keresés, amelyik nincs minden adatbázisban?

Programozástechnikai kérdések:

- Ha nem Z39.50-es protokollal kommunikál a szerver és a bróker, akkor hogyan?
- Milyen legyen a protokoll? Milyen kérdésekre milyen válaszok adhatók? Mekkora legyen a csomag méret?
- Érdemes-e még egy szabványos felületet kialakítani, az adatbázis szerveren futó kommunikációs program és az adatbázis backend-je között?
- A bróker lekérdezze-e a teljes találati halmazt és fésülje-e össze, rendezze-e sorba? Ehhez tárolni kellene a teljes találati halmazt az összes adatbázisból és szépen adagonként továbbadni a kliensnek. Ekkor azonban szinte egy adatbázis kezelő rendszerre van szükség a brókerben is.
- A bróker egyszerre csak egy bizonyos részét kapja-e meg a találati halmaznak minden adatbázis szervertől? Ekkor azonban nem lehet sorba rendezni, vagy csak abban a részhalmazban lesz érvényes a sorba rendezés. A részhalmazok egymás között nem sorban következnek!

5. A project jelenlegi állása

A project tervezett lépései:

1. A jelenleg működő hasonló rendszerek tanulmányozása
2. Az elvárások megfogalmazása
3. Egy minél egyszerűbb, de minden tekintetben működő rendszer létrehozása
4. A rendszer tesztelése és ha szükséges az elvárások kiigazítása
5. Végleges rendszer elkészítése
6. Tesztelés

A project első fázisában két könyvtári rendszer a BRS és a Voyager vesz részt. A BRS kapcsolat kialakításán Burgermeister Zsolt (burgi@kunge.sk-szeged.hu) dolgozik, A Voyager integrálása a KLTE Egyetemi és Nemzeti Könyvtárban általam folyik.

Jelenleg a harmadik lépésnél tartunk, egy minél egyszerűbb, de működő, rendszer elkészítésén dolgozunk.

A kliens:

Grafikus WWW klienst használunk, így a felhasználó egy form-ot tölt ki.

A bróker:

A form miatt nincs szükség a kérdés szintaktikai ellenőrzésére, hiszen a felhasználó csak szintaktikailag helyes kérdést tud feltenni.

Az adatbázis szerverek lekérdezése párhuzamosan történik.

Jelenleg az egyes adatbázisok találati halmazait limitáljuk.

A bróker a visszaérkező találati sztringeket összefűzi és átalakítja HTML formára, ezt a WWW rendszer továbbítja a klienshez.

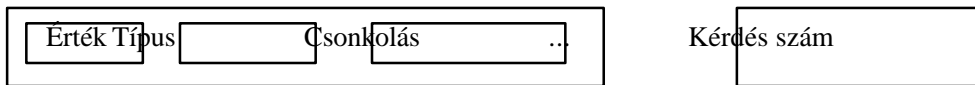
Az adatbázis szerver:

A szerver az maga a könyvtári rendszer, egy C programmal kiegészítve amelyik a kapcsolattartást valósítja meg. A kommunikáció socket-eken keresztül történik, egy általunk definiált protokollal, amely így néz ki fizikailag:

Hossz (long)

maximum 256 Byte hosszú szegmensek ...
--

Logikailag pedig így:



Így a különböző rendszerek esetében csak egy függvényt kell elkészíteni, amely lekérdezi az adatbázist, és visszaadja az eredményt.

Az adatbázisok szinkronizációja:

A jelenlegi fázisban nem várjuk, hogy a résztvevő könyvtárak megváltoztassák az adatbázisaik indexelését, vagy új mezőket vezessenek be. Így a kérdésre adandó választ jelenleg rábízunk az adott rendszerre, például, hogy kulcsszavas-e a keresés. Ha pedig valamilyen mező nem létezik az adott adatbázisban, akkor ezt jelezzük a felhasználó felé. Így nem lesz konzisztens a találati halmaz, de hogy ez mennyire zavaró azt majd a tesztelés során látjuk meg. Például: Tegyük fel, hogy Arany János nevű szerzőt keresünk:

1. Ha a keresés kulcsszavas, az eredmény tartalmazni fogja az összes olyan rekordot amelynek szerzői mezőjében vagy az Arany vagy a János előfordul.
2. Ha a keresés teljes illeszkedést vár el, akkor csak az Arany János-t fogja megtalálni. Ha ezek a módszerek keverednek elég furcsa eredményt kapunk.

A következő fázisban megpróbáljuk a lekérdezést úgy alakítani, hogy esetleg a kérdés átalakításával a válaszok egységesedjenek.

Előre láthatólag ősre elkészül egy mindenki által használható változat, amely az IIF nagygépen a Helkán működik majd. Előre is kérek mindenkit, hogy próbálja ki majd, és írja meg nekünk tapasztalatait. Minden véleményt figyelembe veszünk a végleges változat kidolgozásánál.