

## Hagyományos adatbázis lecserélése LAMP-ra

Néhány hagyományos adatbázis-alkalmazás szinte kínálja magát, hogy cseréljük le LAMP alapú webes alkalmazásra. Richard Hulse elmeséli, hogy az Új-Zélandi Rádió hogyan hajtott végre egy ilyen átállást.

© Kiskapu Kft. Minden jog fenntartva

■ Az *Új-Zélandi Rádió* egy nyilvános rádióadó, és mint minden ilyen, hatalmas zene- és műsortárral rendelkezik. 1987-ban egy új, számítógépes könyvtár-katalogizáló rendszerre bízunk a könyvtárakatok tárolását, ez volt a *BRS*. A *BRS*-t konkrétan a hanglemezek – és később a CD-lemezek –, a szalagok, az élő koncertfelvételek, az interjúk és a klasszikus zenei gyűjtemény adatainak tárolására használtuk. A rendszert eleinte egyszerű terminálokon, később PC-ken futó terminálemulátorokon keresztül értük el. Munkatársaink ugyancsak ezt a rendszert használták a *Concert FM* – a cég klasszikus zenei hálózata – műsorainak szerkesztésére és követésére. A *BRS* egy zárt katalógusprogram volt, a *Maxwell Online Inc.* árulta. *UNIX*-on futott, és hosszú életet élt meg. 16 év alatt átesett néhány hardverbővítésen, míg szoftveres oldalon néhány újabb adatbázistábla hozzáadásával elégtettük ki az újabb típusú adatok tárolására vonatkozó igényeket. A *BRS* – minden ezzel ellentétes várakozástól függetlenül – a legkisebb gond nélkül élte túl a 2000-ik évet, és valamikor 2003-ban kezdtünk komolyan foglalkozni lecserélésének gondolatával.

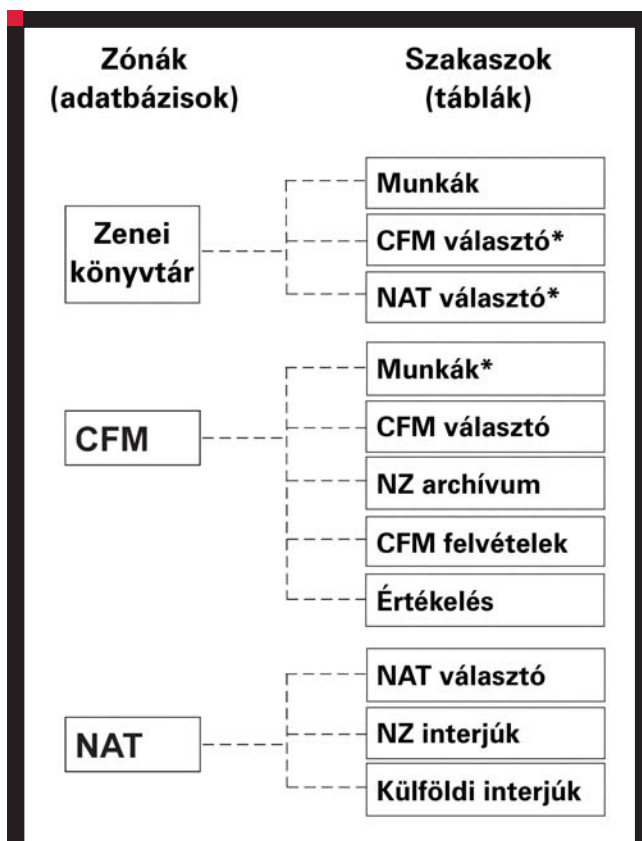
### Csináld magad!

Korábban egy ilyen jellegű átállási feladat végrehajtását alighanem külső cégre bízta volna. Tapasztalati tény ugyanakkor, hogy ilyenkor az eredmény egy zárt forrású, egyedi alkalmazás, valamint a módosítások és a frissítések terén a kötöttség valamely céghez. Amikor aztán ezek a cégek megszűnnek, alkalmazottaik pedig szétszélednek, a nélkülözhetetlen alkalmazás árván marad, az adatokat pedig roppant nehéz egy új alkalmazás alá áttelepíteni. A saját kivitelezésű munkák viszont nem mindig eredményesek, ezért gondosan mérlegeltünk minden szempontot. Az adatok és az alkalmazás kritikus szerepe, illetve a házon belüli erőforrások rendelkezésre állása alapján végül úgy döntöttünk, hogy magunk vágunk bele.

### Helló, BRAD

Informatikai részlegünk vezetője, *Bruce Intemann* lett a tervezet vezetője, ő állított össze egy kísérleti rendszert egy asztali PC-n, melyen *Red Hat Linux 8*, *Apache* webkiszolgáló, *MySQL* és *PHP* (együttesen *LAMP*) futott. *Bruce*

kitalálta, hogyan lehet *BRS* alól egyszerű szöveges formátumba kimenteni az adatokat, valamint készített egy egyszerű, a *MySQL* teljes szöveges indexére alapuló keresőt. Az adatok egy kisebb részét kézzel átalakította. A hozzáférést egy normál webkiszolgálón keresztül biztosítottuk. Nagyjából ekkor fejeztem be egy *PHP*-s, webes munkát a cég egy másik részlegének, és felajánlottam segítségemet az új tervezet végigviteléhez. Amikor elértünk oda, hogy nevet kellett adni a rendszernek, arra gondoltunk, hogy a *B* és az *R* betűt meg kellene tartani, ezzel emlékeztetve a „szülők” nevének kezdőbetűire. Feleségem a *BRAD* névvel állt elő, munkatársaim egyike pedig kitalálta, hogy a név bizonyára a *Bruce and Richard's Audio Database (Bruce és Richard zenei adatbázisa)* elnevezés rövidítése. A név ezzel meg is ragadt. A próbarendszert hamarosan elfogadtuk, ezért írtam egy rövid *Perl* parancsfájlt, melynek feladata az összes adat feldolgozása – körülbelül kétszáz ezer rekordról van szó – és a *MySQL* adatbázisba való beillesztése volt. A feladat meglehetősen összetett volt, mert a globális keresések segítése céljából a kisebb adatbázisok közül sokat beolvasztottak a fő, *Works* nevű adatbázisba. Szerencsére egy külön mezővel jelölték meg az eredeti adatok helyét, forrását. A *BRS* által tárolt adatok formátumára az első kódreszlet mutat példát. Miután sikerült áttöltenünk az adatokról egy teljes pillanatfelvételt, objektumorientált *PHP*-ben újraírtam *Bruce* kódját. Felhasználtam a másik tervezethez készített kereső osztályomat is, amit hírek helyett zenei adatok megjelenítésére módosítottam. A nagyjából kész bemutató rendszert egy fejlesztési kiszolgálóra telepítettük, majd megkértük a többieket, hogy mondják el róla véleményüket. Válaszaik alapján úgy döntöttünk, hogy a visszajelzések szerint folyamatosan fogjuk fejleszteni a már üzembe helyezett rendszert. A kettősséggel biztosítható a munkatársak hozzáférése a működő rendszerhez, valamint lehetővé válik a két megoldás által adott keresési eredmények összehasonlítása. Szintén fontos volt, hogy az új rendszer elnyerje a személyzet bizalmát. Mivel az adatokat az eredeti csoportokra akartuk szétosztani, egy összetettebb parancsfájlt is készítettünk, mely kihámozta az eredeti adatforrásokat a *BRS*-ből. A kapott halmazokat különböző adatbázisokba és táblákba



1. ábra A BRAD adatforrásainak elrendezése. A csillaggal jelölt tábla egy a jelenlegi zónán kívül található táblához vezető álnévet jelez.

helyeztük, ez látható az 1. ábrán. A BRAD alatt a cég minden részlege egy-egy zónának számít, továbbá minden adatforrást külön részként kezelünk. Bármelyik zóna rendelkezhet álnévvel a más zónákba tartozó részekhez, illetve lehetőséget kaphat tetszőleges táblákban való keresésre, függetlenül azoknak a rendszerben elfoglalt helyétől. A BRS adatbázisa egyszintű, nem relációs volt, és az évek során sokféle személy sokféle formátumú adatot rögzített benne. Amikor átnéztem az új rendszerbe kerülő pillanatfelvételeket, a Perl parancsfájl módosításával megszüntettem néhány, elsősorban a dátum mezőket illető rendellenességet. Előfordult például, hogy az utolsó frissítés dátumának szöveges mezőjét valaki kézzel múltbéli időpontra módosította – a BRAD alatt ez egy a rendszer által karbantartott datetime, típusú mező. A rekordokat újabb, a létrehozás időpontját tároló mezővel is bővítettük.

### A BRAD és a nyílt forrás találgája

A BRAD fejlesztését egy LAMP-ot futtató kiszolgálón végeztük, és magától értetődőnek tűnt, hogy a fejlesztés során nyílt forrású PHP osztályokat használjunk. Az adatbázis elérésére, az űrlapok előállítására és feldolgozására, valamint az alapszintű hibakezelésre PHP Extension and Application Repository (PEAR) modulokat alkalmaztunk. Egy meglévő hibakezelő osztályt úgy módosítottunk, hogy figyelmeztessen a hibákra, de a teljes hibaüzenetet ne jelenítse meg a felhasználóknak.

1. kódrészlet Egy BRS alól származó rekord eredeti szöveg formátumban (Original Text Format)

```

*** BRS DOCUMENT BOUNDARY ***
..Document-Number:
    000080019
..TI:
    Ode for the centenary of Trinity College
    Dublin, Great parent, hail to thee (Z327)
..MA:
    Hyperion
..CA:
    CDA 66476
..ME:
    cd
..RA:
    The King's Consort
..CF:
    vocal - ode
..CP:
    PURCELL
..CD:
    Robert King
..SO:
    Gillian Fisher, Evelyn Tubb (sopranos),
    James Bowman, Nigel Short (counter-tenors),
    Rogers Covey-Crump (high tenor), John Mark
    Ainsley (tenor), Michael George, Charles
    Pott (basses)
..ST:
    T12-21
..AT:
    Purcell - Complete odes and welcome songs
    vol 5
..DU:
    002419
..PT:
..RD:
    2-4 Jan 1991
..RE:
..AD:
..SR:
..RO:
..CY:
    1694
..LI:
    Nahum Tate
..OR:
..LN:
..PU:
..RI:
..ED:
..LP:
..LQ:
    cp
..LQ:
..NO:
..IS:
..LD:
    921012
..LU:
    Teander
*** BRS DOCUMENT BOUNDARY ***
    
```

© Kiskapu Kft. Minden jog fenntartva

1. táblázat A BRAD által használt nyílt forrású modulok

PEAR::DBM	Adatbázis-hozzáférés
PEAR::HTML:QUICKFORM	A szerkesztő felület űrlapjai
PATUSER	Felhasználókezelés és a szerkesztői hozzáférések szabályozása
Hibakezelő osztály	Erősen módosítottuk, hiba-üzenetek helyett az oldal elsődleges tartalmát jeleníti meg.
Lapozó	Az eredmények lapozását teszi lehetővé. Módosításával lehetővé tettük az URL osztályon belüli feldolgozását.

Amikor szükségem volt valamilyen függvényre, először mindig megpróbáltam egy nyílt forrású modult keresni, csak ezt követően fogtam bele saját kód írásába. Ezzel a módszerrel rendkívüli mértékben felgyorsult a fejlesztés. (A BRAD által használt modulok listáját az 1. táblázat tartalmazza.)

### Várakozások és eredmények

Mivel a BRS-t hosszú időn keresztül megtartottuk, munkatársaink komoly rutint szereztek a használatában. A BRS sokoldalú és gyors keresésre volt képes. Képes volt például bármely szó bármely a felhasználó által megadott mezőben végzett keresésére. Voltak persze nyugjei, például bizonyos szavakat nem indexelt. Emiatt előfordult, hogy együttesek teljes nevét figyelmen kívül hagyta, ilyen volt például a *The Who* nevű társulat. Az ilyen esetekben, ha például a *The Who* névéhez kötődő anyagokat akartuk kikeresni, akkor valami mást is tudnunk kellett róluk, például ismerünk kellett az egyik tag nevét (*Pete Townsend*), vagy valamilyen általuk írt dolgot (*Tommy*). Természetesen egyik megoldás sem volt tökéletesen megbízható.

A parancssoros felület időnként kiszámíthatatlan viselkedése és használatának körülményessége azt eredményezte, hogy munkatársaink jó része a zenekönyvtárost kérte meg az általa kívántak kikeresésére, egyszerűen egy kézzel írt listát adva át neki a kért anyagokról. Az új rendszerrel szembeni elvárások között tehát szerepelt a legalább egyenértékű, de inkább jobb keresési lehetőség, valamint a felületet is úgy kellett elkészíteni, hogy minimális betanulás után bárki használni tudja.

A BRS egyik leghasznosabb szolgáltatása az volt, hogy a keresési kifejezéseket adott mezőkre lehetett korlátozni. Például:

```
Mozart.cp. zongora
```

A fenti keresés minden olyan anyagot kiad, melynek szerzője (cp, composer) Mozart, rekordjának bármely más mezőjében pedig szerepel a zongora. Úgy határoztunk, hogy a BRAD alatt is megtartjuk ezt az írásmódot, így a felhasználók a korábban megszokott módon végezhetik el a kereséseket. Eleinte egy speciális keresési oldal is

akartunk készíteni a BRAD-hez, ám a fenti írásmód annyira rugalmasnak bizonyult, hogy végül a külön oldal szükségtelessé vált.

### Túllépve a korlátokon

A tervezet megvalósítása során számos kihívással szembesültünk. A legtöbb kapcsán elvárásaink szerint kellett módosítani a *MySQL* alapértelmezett működését. Az első probléma az volt, hogy a *MySQL*-t rá kellett vennünk az adatokban túlságosan gyakori előfordulásuk miatt nem indexelt szavak figyelembe vételére. Esetünkben ugyanis minden egyes szó fontos.

*MySQL* alatt a tiltott szavakat úgy távolíthatjuk el, hogy – még mielőtt bármit is bevinnénk az adatbázisba – az alábbi sort adjuk a *MySQL* beállító fájljához:

```
ft_stopword_file = ""
```

A második gond az volt, hogy a *MySQL* által megszabott négy karakteres határérték alatti hosszúságú szavak alapján végzett kereséseket is lehetővé kellett tenni. A BRS rendszer minden szót indexelt, függetlenül annak hosszúságától, kivéve persze a tiltott szavakat, és a tiltott szavak eltávolításával a megadott feltételeknek jobban megfelelő keresési eredményeket kaphattunk.

Ezt a kérdést két lépésben oldottuk meg. Először az alábbi sort a beállító fájlhoz adva három karakterre csökkentettük az indexszavak hosszát:

```
set-variable = ft_min_word_len=3
```

A kezelt adatmennyiséget is figyelembe véve ez a beállítás még elfogadható teljesítményvesztéssel járt. Második lépésként készítettünk egy intelligens, a lekérdezést a megadott legrövidebb szótól függően átalakító, majd a *MySQL*-nek továbbküldő keresőmotort. Segítségével a keresési kifejezések hosszától teljesen függetlenül lehet teljes szöveges keresést végezni.

Utolsó teendőnk az volt, hogy minden keresést alapesetben is *ÉS* kereséssé alakítsunk. A *MySQL* teljes szöveges boolean módja, ha nem használunk módosítókat, *VAGY* keresést végez. Alapesetben egy + karakterrel tudunk *ÉS* keresést indítani, tehát a lekérdező motorra bízunk, hogy – egyéb módosító hiányában – beillessze a szükséges + karaktereket.

### A lekérdezőfordító

A BRAD lelke egy kifejezés-feldolgozó és egy lekérdezőfordító. A kifejezés-feldolgozó átveszi a lekérdezőt, feltördeleli, az összetevőket pedig egy-egy tömbbe helyezi. Minden tömb tartalmaz egy *MySQL* módosítót (+, -, <, >, ~); egy atomot, vagyis a lekérdezői kifejezés egy részét, mely lehet egy szó vagy egy mondat; továbbá egy elhagyható mezőnevet. A kifejezés-feldolgozó – ha nem talál egyéb módosítót – minden atomhoz önműködően hozzáadja a + karaktert, amivel alapesetben minden keresés *ÉS* kereséssé válik. Egyértelműen ez a jobb megközelítés, hiszen a felhasználók is azt várják a kereséstől, hogy minél több szót megadva egyre kifinomultabb eredményt kapjanak.

Az elhagyható mezőt speciális keresések megvalósítására használjuk, ezeknél meghatározott szavaknak meghatározott mezőben kell szerepelniük. BRAD alatt is megtartottuk

a már említett . karaktert, mint mezőkeresési jelölést. Amikor megkezdünk egy normál keresést, a lekérdezésfordító a keresés hatókörébe eső táblák mindegyikét lekérdezi, majd eredményként ezek teljes szöveges mezőit adja vissza. Ezek alapján kerül összeállításra egy az összes teljes szöveges mezőre kiterjedő lekérdezés.

A lekérdezésfordító teljes szöveges, általános kifejezéseket és nem teljes szöveges, mezőspecifikus kifejezéseket vegyesen is képes kezelni. A lekérdezésfordító segítségével a **BRAD** adatforrásai gyakorlatilag korlátok nélkül bővíthetők, az előállított lekérdezések pedig dinamikusan alakíthatók – szükségtelené válnak tehát az adatbázis valódi objektumait képviselő objektumosztályokhoz tartozó statikus alapelekérdezések. A felhasználók – egy normál weboldarról – a keresés minden jellemzőjét képesek szabályozni. (2. ábra) A felhasználók annyi zónát vagy táblát választhatnak ki, amennyit csak akarnak. A testreszabási lehetőségekkel maradéktalanul meg lehet felelni a cég és a felhasználók minden igényének.

### A BRAD bővítése

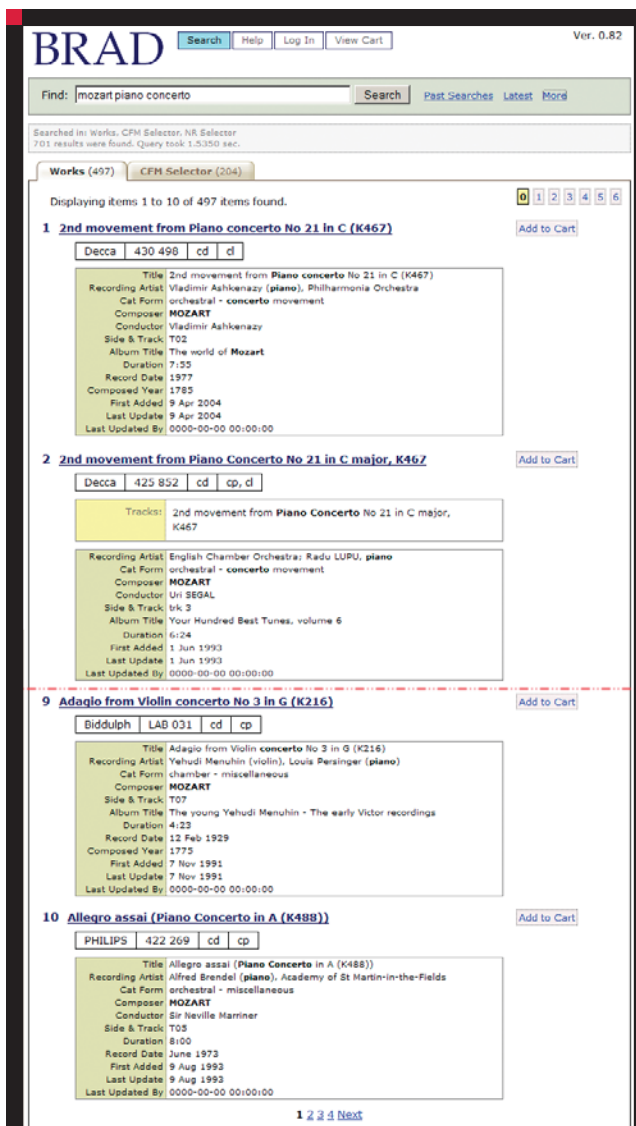
A **BRAD** fejlesztése során különös figyelmet fordítottunk a tetszőleges irányú bővíthetőségre. A keresések bármely típusú adata kiterjeszthetők, illetve az egyes adattípusokon speciális kereséseket is lehet végezni.

A cégnél súlyos problémát jelentett, hogy a különböző feladatokra különböző adatbázisokat használtunk, és az adathalmazok különféle alkalmazásokba szóródtak szét. Két rádióhálózatunk a *Selector* nevű alkalmazást használja az adásba kerülő műsorok ütemezésére. A *Nemzeti Rádió* egy körülbelül tízezer zeneszámot tartalmazó adatbázissal dolgozik, a *Concert FM* pedig öt, együttesen körülbelül százezer anyagot tároló adatbázisra támaszkodik. A *Concert FM* adatbázisát azért kellett ötfelé osztani, mert a *Selector* csak korlátozott adatmennyiséget képes kezelni.

Ha valaki adott zeneszámra szeretett volna rákeresni, akkor mindhárom alkalmazással – *BRS*, a *Nemzeti Rádió Selector*a és a *Concert FM Selector*a – meg kellett próbálkozni. Mindhárom rendszerben nem lehetett egyszerre keresni. Szerencsére a *Selector*hoz volt egy az adatok XML formátumba való kimentésére alkalmas segédprogram. Igaz, leírás nem volt hozzá, és nem is sikerült szerezni, *Bruce*-nak sikerült rájönnie működésének módjára. Az adatokat így ki tudjuk menteni, illetve egy windowsos munkaállomásról *FTP*-n keresztül fel tudjuk tölteni a **BRAD** kiszolgálójára. A műveletet minden reggel elvégezzük, a **BRAD** kiszolgálóján pedig egy *Perl* parancsfájl gondoskodik az adatok beemeléséről. A *Concert FM* öt adatbázisát egyetlen táblába egyesítettük, ugyanis az adatok egyediek. Az eredeti keresőmodult kibővítettük, így már egynél több táblában is képes keresni, illetve a mezők számától és típusától függetlenül képes visszaadni az eredményeket. Az eredmény lapozós stílusban jelenik meg. (3. ábra) Az ábrán a *Works* táblából származó első találatok láthatók. A másik fülön a *Concert FM Selector* táblájából származó eredmények száma látható. A keresés hatókörétől és a találatok számától függően tetszőleges számú lap megjelenhet. A műsorszerkesztők így végre egyetlen felületen keresztül bármilyen típusú zenére rá tudnak keresni. Az első változat elkészítése óta munkatársaink kérésére számos további szolgáltatással bővítettük a rendszert. Ilyen például a keresési történet



2. ábra A BRAD keresőfelületének ismertetése (lásd: 2. táblázat)



3. ábra Az eredményeket tartalmazó lap egy részlete

és a kosár. A kosárba bármely tábla elemeit be lehet helyezni. A kosarakat el lehet menteni és elő lehet keresni, illetve létrehozás után a kosár számát el lehet küldeni elektronikus levélben a könyvtárosnak. Így nincs szükség hosszas listák nyomtatására vagy átküldésére, elég csupán a kosár számát közölni.

### Hazai zeneszámok keresése és hossz alapján végzett keresés

Legújabb szolgáltatásunk az álnév alapú keresés. Egy álnév egy gyakran használt, összetettebb kifejezést helyettesít. Ilyen például az *Új-Zéland*, vagyis az olyan zene, amelynek előadói vagy szerzői között új-zélandi művész is szerepel. Fontos jellemző, ugyanis mindkét adóhálózat meghatározott arányban hazai zeneszámokat sugároz.

© Kiskapu Kft. Minden jog fenntartva

## 2. táblázat A keresőfelület mezői

1a: Keresési terület	Ide kell beírni a keresési kifejezést.
1b: Search (keresés) gomb	Rákattintva indíthatjuk el a keresést.
2a: Past Searches (korábbi keresések)	Egy a korábbi kereséseket megjelenítő területet nyit meg. A megjelenő elemekre kattintva az egyes kereséseket újra el lehet végezni.
2b: Latest (utolsó)	A Works, a CFMS és a NATS adatforrások legutolsó bejegyzéseit jeleníti meg. Mind-egyik forrásból legfeljebb 250 találatot ad vissza, a lista elejére a legújabb bejegyzést helyezi.
2c: Fewer (kevesebb)	Ez a hivatkozás, illetve a következő sorban szereplő a More (több) és a Fewer (kevesebb) beállítás között vált; csökkentti vagy növeli a BRAD által megjelenített keresési lehetőségeket.
3a: Zone (zóna)	A BRAD a cég különféle részeihez kötődő zónákra osztja adatait. A zónáknak számos különböző adatforrásuk van. A zónaválasztó segítségével megszabhatjuk, hogy a cég melyik részére vonatkozzon a keresés.
3b: In (Ebben)	Segítségével kiválaszthatjuk, hogy a zóna melyik adatára szeretnénk rákeresni. A kereséseket általában egy-egy zóna összes adatára vagy egyetlen adattípusára végezzük.
4: Order by (Rendezési szempont)	–
5: Media (Adathordozó)	A BRAD képes az adott típusú adathordozón található anyagokra korlátozni a kereséseket.
6: Count (számláló)	Az oldalanként megjelenített tételek száma.
7: Display Mode (Megjelenítési mód)	A megjelenítési módok a régi BRS rendszerhez kötődnek, segítségével a felhasználó kiválaszthatja, hogy az eredményt milyen összegzésben szeretné látni. A megjelenítési módok testreszabhatók, ha tehát egy felhasználó valamilyen különleges formátumot igényel, akkor megoldható annak hozzáadása.
8: Show Results (eredmények megjelenítése)	Lapokon vagy listaként. A BRAD normál módja a keresések eredményét lapozós stílusú felületen jeleníti meg. Lista módban egy fejléccel ellátott listát jelenít meg. A lista mód főleg nyomtatásra vagy elektronikus levélben való továbbításra szánt listák előállítására használható.
9: Show Details in (részletek megjelenítési helye)	Segítségével a felhasználó megadhatja, hogy adott rekord teljes adattartalmát – a rekord hivatkozására kattintva – új vagy az aktuális ablakban szeretné látni.

Az évek során a cégnél megforduló emberek számtalan adatbevitelt végeztek, ennek során a fő *Works* adatbázisban különféle mezőket és azonosítókat használtak az új-zélandi-ság jelzésére. Az *NZ Music* (mint *New-Zealand*, új-zélandi zene) álnév – VAGY keresést eredményezve – automatikusan hozzáadja a lekérdezéshez a szükséges kifejezéseket és mezőket. Mindehhez egy új osztályt készítettünk, mely a kifejezés-feldolgozó felett helyezkedik el, és annak segítségével bontja ki a lekérdezésben szereplő álneveket. Ezután a feldolgozó hozzáadja a szükséges átadott értékeket a lekérdezésfordító által kezelt lekérdezésveremhez. Nézzünk néhány példát a BRAD álneveinek használatára. A Mozart @nza

lekérdezésnél minden Mozart vagy NZ tartalmú mezőt megkapunk. A tényleges lekérdezés így néz ki:

```
SELECT * FROM brs.works WHERE (cf REGEXP
↳ '[:<:~]local[:>:]' OR cf
REGEXP '[:<:~]nz[:>:]' OR lq REGEXP
↳ '[:<:~]nz[:>:]') AND MATCH
ti,ra,cf,cd,cp,so,at,notes,lq AGAINST ('+Mozart'
↳ IN BOOLEAN MODE) ORDER BY ti asc LIMIT 1000
```

A hossz alapján végzett keresés szintén a szerkesztők kérésére került be a rendszerbe, segítségével adott időtartamok közötti zeneszámokat lehet találni. Sokszor előfordul ugyanis, hogy valamely szerzőtől nagyjából meghatározott hosszúságú művet kell előkeresni. A *BRAD* alatt a szögletes zárójel közé illesztett számokat hosszmegszorításokként kezeljük. A *BRAD* képes a közelítő keresésre, illetve az adott hossztartományokon belüli keresésekre. További példák a széljegyzetben találhatók.

### Példák a BRAD alatti hosszra keresésre

Rövidebb mint:  
brahms [<20]

Között (hossztartományt is meg lehet adni). Az alábbi lekérdezéssel a *Mozarthoz* kötődő, 20 perc, 30 másodperc és 30 perc, 15 másodperc közötti hosszúságú zeneszámokat kapjuk meg:  
mozart [20:30-30:15]

Közelítő keresések – a megadott időtartam keresése +/- 10 százalékos eltéréssel. A c a „circa”, a körülbelül rövidítése:  
mozart [ c 24 ]

Időtartományt is hozzáadhatunk. Az alábbi lekérdezés például a 24 perc, plusz/mínusz 1 perc időtartamú zene-számokat adja vissza:

```
mozart [ c 24 r 1 ]
```

Összetett időtartam-keresések – az alábbi lekérdezéssel Beethoven 20 és 22 perc közötti időtartamú műveire kereshetünk rá:

```
beethoven.cp [20-22]
```

Az utolsó lekérdezéshez tartozó, lefordított lekérdezés a következő:

```
SELECT * FROM cfm.cfms WHERE (du <= 1320)
AND (du >= 1200)
AND MATCH ti,ca,ma, ra,cd,cp,so,at,notes AGAINST
('+beethoven' IN BOOLEAN MODE)
AND MATCH cp AGAINST ('+beethoven' IN BOOLEAN
MODE)
ORDER BY ti asc LIMIT 1000
```

A *Concert FM Selectorának* adatai közt – már említettem őket – mindenhol helyesen meg van adva az NZ előadó és az időtartam mező értéke, így ezeket az álneveket biztonságosan használhatjuk a teljes adatkészleten. Mivel a *Works* adataiban keverednek az elemtípusok, esetében csak az érvényes időtartammal rendelkezőkre terjed ki a keresés. Mivel korábban a *Works* bejegyzései között egyáltalán nem lehetett időtartamra keresni, ez is előrelépésnek mondható.

## A jövő

Írásom születésekor éppen megkértek, hogy illesszem bea *BRAD* alá a céges telefonkönyvet, illetve a hírrészleg számára nemrég készült egy kísérleti kiejtési útmutató.

## Összefoglalás

Tervezetünk révén újabb rendszerre tudtuk cserélni az *Új-Zélandi Rádió* kulcsfontosságú katalógusát, illetve újabb szolgáltatásokat bocsátottunk a munkatársak rendelkezésére – mindezt alacsony birtoklási költség mellett. Az új és a régi adatok közös tárolórendszerrel kaptak.

A továbbiakban valószínűleg meg fogjuk oldani, hogy a műsorszerkesztők az egy-egy szerzővel vagy művésszel kapcsolatos zeneműveket, interjúkat és archív anyagokat egyetlen lekérdezéssel megkereshessék. Az eredmények között valószínűleg szerepelni fog az illető telefonszáma és nevének helyes kiejtése is.

A *BRAD* minden bizonnyal fejlődő rendszer marad, mi pedig újabb és újabb alkalmazásokat fogunk kitalálni hozzá. Ez az egyik előnye a csináld magad mozgalomnak – a rendszer a sajátunk, akkor és úgy bővítjük vagy módosítjuk, ahogy és amikor csak akarjuk.

*Linux Journal* 2005. március, 131. szám

### Richard Hulse

Vezető hangmérnök az Új-Zélandi Rádiónál. Jelenleg több informatikai tervezeten is dolgozik, ezek egyike a rádió weboldalának továbbfejlesztése ([www.radionz.co.nz](http://www.radionz.co.nz)).

