

Üzemeltetési

Kézikönyv

Tartalomjegyzék

1	BEVEZETÉS	3
2	A RENDSZER TELEPÍTÉSÉT KÖVETŐ FELADATOK A SZERVEREN.....	4
2.1	A GÉP IP CÍMÉNEK BEÁLLÍTÁSA	4
2.2	A JELSZAVAK MEGVÁLTOZTATÁSA	4
2.3	HELYI SAJÁTOSSÁGOK BEÁLLÍTÁSA	4
2.3.1	A webes felület testreszabása.....	4
2.3.2	Katalógus testreszabása	5
2.4	KLIENS INSTALLÁLÁS	6
2.5	KLIENSEK KONFIGURÁLÁSA, EGYÉNI BEÁLLÍTÁSOK	6
2.5.1	JavaPAC	6
2.5.2	JavaCAT	10
2.6	BEÁLLÍTÁSOK AZ ADMIN MODULLAL	18
2.7	A MENTÉSEK AUTOMATIZÁLÁSA	18
2.7.1	Offline	18
2.7.2	Online	19
2.7.3	Export/Import	19
3	A RENDSZERGAZDA NAPI FELADATAI	19
3.1	FELHASZNÁLÓK ADMINISZTRÁLÁSA.....	19
3.1.1	Új felhasználók felvétele	19
3.2	SZABAD DISZK TERÜLET ELLENŐRZÉSE	20
3.3	MENTÉS LEFUTÁSÁNAK ELLENŐRZÉSE	20
3.4	BERAGADT USER PROCESSZEK ELLENŐRZÉSE, TAKARÍTÁSA.....	20
4	A RENDSZERGAZDA IDŐSZAKOSAN ELVÉGZENDŐ FELADATAI.....	22
4.1	ADATOK GYŰJTÉSE A SZERVER LETERHELTSÉGÉRŐL	22
4.2	SZABAD TÁBLATÉRTERÜLET ELLENŐRZÉSE.....	22
5	SEGÉDPROGRAMOK.....	23
5.1	MARC-REKORDOK EXPORTÁLÁSA.....	23
5.2	A DBLOAD HASZNÁLATA	23
5.3	A CHECKMARC HASZNÁLATA	24
6	RENDSZERMONITOROZÁS - HIBAFELDERÍTÉS	24
6.1	SPECIÁLIS FELHASZNÁLÓK SZEREPE.....	24
6.2	A UNIX, AZ ORACLE ÉS A CORVINA NAPLÓ-FÁJLJAI.....	24
6.2.1	A UNIX napló-fájljai.....	24
6.2.2	Az Oracle napló-fájljai	24
6.2.3	A Corvina napló-fájljai.....	25
6.3	PROCESSZEK LEÁLLÍTÁSA, ÚJRAINDÍTÁSA, MEGLÉTÜK ELLENŐRZÉSE.....	25
6.3.1	A processzek leállítása.....	25
6.3.2	A processzek újraindítása	26
6.3.3	Processzek futásának ellenőrzése	26
6.4	GYAKRAN FELTETT KÉRDÉSEK	26

1 Bevezetés

A *Corvina* rendszer használata a rendszergazda normál teendőihez képest nem jelent nagymértékű feladatmódosulást. Ez a kézikönyv abban szeretne segíteni, hogy mely feladatok elvégzése szükséges a rendszer helyes működéséhez, és milyen időközönként végezzük azt.

2 A rendszer telepítését követő feladatok a szerveren

2.1 A gép IP címének beállítása

A szerver IP címe, ha a telepítés az e-Corvina Kft telephelyén történt, a helyi teszhálózatnak megfelelően van kiadva. Ezt a helyi IP tartománynak megfelelő címre kell átállítani a /etc/hosts fájlban, majd a gépet újra kell indítani. Újraindítás után célszerű ellenőrizni, hogy minden szükséges komponens elindult-e. Ezek:

Oracle RDBMS (6 processz ora_...)

Oracle listener (tnslsnr)

Apache webszerver (httpd)

Online update (adatbázisonként egy updrec)

2.2 A jelszavak megváltoztatása

A telepítés során a

root

corvina

opac

...felhasználók jelszava kizárólag a telepítéshez használt, megváltoztatásuk hiánya nagymértékű biztonsági rés. A telepítést követően a rendszergazdának a helyi konvenciók betartásával kell megváltoztatnia. Ahhoz, hogy a későbbiekben a rendszergazda távolléte alatt bekövetkező hiba javítása elvégezhető legyen, vagy helyettesről, vagy a jelszavak elérhető helyen való létbe helyezésről kell gondoskodni.

2.3 Helyi sajátosságok beállítása

A /corvina/init könyvtárban található állományok módosítása mindenképpen a helyi adottságok alapján történik, az idő folyamán változhat, így ez részben a rendszergazda feladata.

liblogo.gif:

A könyvtár saját logója, ez jelenik meg a webes elérésű felületeken.

2.3.1 A webes felület testreszabása

A keresés webes megjelenését a /corvina/webpac/opac/html/ könyvtárban található fájlok adják. Ezek kisebb módosítása egyedivé teheti az arculatot, azonban vigyázzunk a fájlok lényegi részének változatlanlanságára, ellenkező esetben működési zavarokat okozhatunk.

A webes Reports modul a /corvina/perl/reports/reports, illetve a /corvina/perl/reports/lib/-ben található perl szkriptek segítségével állítja elő az induló, illetve az ebből származtatott oldalakat.

Az induló oldal struktúráját a program automatikusan generálja a /corvina/perl/reports/corvina/ könyvtárban létrehozott fájlstruktúra alapján. A riport lekérdezések *perl* szkriptekben a .rep kiterjesztésű fájlokban vannak implementálva.

2.3.2 Katalógus testreszabása

A katalógus beállításai, paraméterei a /corvina/etc könyvtár alatt található az alábbi fájlokban:

dbnév.cfg: indexelési előírások. A módosítások csak a későbbiekben elmentett rekordoknál érvényesülnek.

authctl/rc/search.info: authority kontroll beállítási paraméterek

Szerkesztése nem ajánlott, kivéve

UPDATE_NON_LOCAL_MATCH: TRUE

Ha ez az érték FALSE-ra van állítva, akkor a háttér authority rekordok átemelése le van tiltva.

common/article.cfg: névelők listája, böngésző index generálásakor használja

common/database.cfg: távoli adatbázisok listája, normál működéshez nem kell. A lokális adatbázis ne legyen felsorolva, mert problémát okozhat.

common/display.cfg: rekord megjelenítési formátumok.

common/fixtrans.cfg: könyvtár kód feloldás

common/help.cfg: CCL helpje, csak direkt használatnál érvényesül

common/location.cfg: lelőhelyek feloldása

common/magyar.cfg: karakter konverziós táblázat

common/notes.cfg: megjelenítésnél mezők szerinti formátumok, címkék

common/status.cfg: Dokumentum kölcsönzési státuszok értékei megjelenítéshez

common/stopword.cfg: Stopszavak listája.

2.4 Kliens installálás

A rendszerhez mellékelt CD-ről a JavaClients könyvtárat a c:\corvina könyvtárba kell másolni. Ha máshova kerül, akkor az egyes modulokhoz tartozó indító .bat fájlokban a path-t módosítani kell.

2.5 Kliensek konfigurálása, egyéni beállítások

2.5.1 JavaPAC

2.5.1.1 Szöveges üzenetek

HELY	kliens
FÁJL	c:\corvina\JavaClients\JavaPAC\cfg\messages
STRUKTÚRA	msgid = "«ANGOL NYELVŰ ÜZENET»" msgstr = "«MAGYAR NYELVŰ ÜZENET»"
HASZNÁLAT	A megváltoztatni kívánt szöveget keresse meg a fájlban! Ügyeljen a teljes, betűnkénti egyezésre! Az így talált sorban az időzőjelek közti magyar nyelvű szöveget cserélje tetszőleges, Önnek megfelelő szövegre!
Fontos!	Az angol nyelvű üzenetet ne módosítsa, csak a magyar nyelvű üzenetet!
Fontos!	Ellenőrizze, hogy a fájl utolsó sorában három csillag (***) szerepeljen!

2.5.1.2 Katalógusok

HELY	kliens
FÁJL	c:\corvina\JavaClients\JavaPAC\cfg\database.cfg
STRUKTÚRA	«szerver» «port» «a katalógus kódja» «a katalógus neve» [DEFAULT]
HASZNÁLAT	Az új adatbázis felvételéhez szükséges az adatbázis neve, kódja és a szerverének neve vagy IP-száma. Válassza ki, hogy a listában hol szerepeljen az új katalógus! A katalógusválasztó-listában (ld. JavaPAC felhasználói kézikönyv) a katalógusok sorrendje megegyezik azzal a sorrenddel, amelyet ebben a fájlban lát. Hozzon

létre új, üres sort és írja be a szerver címét (lehet ez teljes név – pl. giant.lib.klte.hu – vagy IP-szám – pl. 193.6.198.1 – is), a portot, amelyen a katalógushoz tartozó keresőprogram fut (ez általában 5000) és a katalógus megnevezését. Ha azt szeretné, hogy a JavaPAC indításakor automatikusan ehhez a katalógushoz kapcsolódjék, írja a sor végére a DEFAULT kulcsszót!

Fontos! Ügyeljen rá, hogy ez a szó csak egy sor végén szerepeljen a fájlban.

Katalógus listából való törléséhez a hozzátartozó sort kell a fájlból törölni.

Fontos! Egy sorban csak egy katalógus szerepelhet.

Fontos! Ellenőrizze, hogy a fájl utolsó sorában három csillag (***) szerepeljen!

2.5.1.3 Keresőfelület

HELY	kliens
FÁJL	c:\corvina\JavaClients\JavaPAC\cfg\opac.ini
STRUKTÚRA	<p>MAIN «főindexek száma»</p> <p>[+] «főindex kódja» «megnevezése»</p> <p>LIMIT «korlátozó index kódja» «megnevezése»</p> <p>- mindegy</p> <p>«lehetséges érték kódja» «megnevezése»</p> <p>END</p>
HASZNÁLAT	<p>A MAIN kulcszót követően adhatja meg, hogy hány főindex legyen látható a keresőfelületen. (A főindex egy beviteli mezőből és egy hozzátartozó legördülő listából áll. Az utóbbiban választhatja ki a felhasználó, hogy az előbbiben megadott értéket a rendszer a rekordok mely részében keresse – pl. címek vagy személyek közt.) A MAIN kulcszót követő sorokban azokat az indexeket kell felsorolni, amelyekre keresést akar megfogalmazni, ám lehetséges értékei számossága nem teszi lehetővé, hogy azokat konfigurációs fájlban felsoroljuk (pl. cím). Ha lehetővé teszi (pl. rekordtípus), akkor a korlátozó indexek közé érdemes felvenni és a lehetséges értékek felsorolásával (ld. alább) megkönnyíteni a felhasználónak a keresési feltételek megfogalmazását.</p>

Fontos! Az indexek mindig négybetűs kódból és felhasználó számára érthető megnevezésből álljanak. Ez mind a főindexre, mind a korlátozó indexre vonatkozik.

Az index elé írjon plusz (+) jelet, ha az index böngészhető.

Fontos! Legalább egy böngészhető indexnek kell a fájlban szerepelnie.

A korlátozó indexek sorának elején LIMIT kulcsszónak kell szerepelnie. Minden korlátozó indexet egy – mindegy tartalmú sor, majd lehetséges értékeinek felsorolása követ. Az értékek külön sorokban álljanak, elől a tényleges érték, ahogy az a rekordokban előfordulhat, aztán a felhasználó számára érthető formában (kibővítve). A kettő meg is egyezhet.

Fontos! A tényleges érték egy szó lehet. A második szótól kezdődően a program a felhasználó számára írt magyarázatként értelmezi.

A korlátozó indexek felsorolását END kulcsszó és a fájl végén három csillag (***) zárja.

2.5.1.4 Megjelenítés

HELY	szerver
FÁJL	/corvina/etc/opac/ccl/display.cfg
STRUKTÚRA	<code>\$ORDER «kijelzés típus»</code> <code>\$START «megnevezés»</code> <code>«mezőkód»=«almezőjel»«almezőjel»«almezőjel»</code> <code>\$END</code> <code>\$END</code>
HASZNÁLAT	Keresse meg azt a kijelzéstípust, amelyet változtatni kíván. A típus angol nyelvű megfelelőjét kell keresnie: SHORT (rövid), NORMAL (normál), LONG (hosszú), CARD (cédula), EXPORT (export), STATUS (státusz). Ha új mezővel kívánja bővíteni a kijelzéstípust, keresse meg azt a megnevezést, amely fedí az új mező tartalmát! Ha talál ilyet, új sorban adja meg a mező kódját, majd egyenlőségjelet követően a megjelenítendő almezőit. Ha megfelelő megnevezést nem talál a fent látható szerkezetben hozzon létre új, \$START kulcsszóval kezdődő és

\$END kulcsszóval végződő fejezetet, a \$START kulcsszó után írjon megfelelő megnevezést, és külön sorban adja meg a mező kódját, majd egyenlőségjelet követően a megjelenítendő almezőit.

Fontos! Ügyeljen arra, hogy minden \$START és \$ORDER kulcsszóhoz tartozzon \$END kulcsszó.

Új almező felvételéhez csak a megfelelő mező sorában az egyenlőség jel után be kell szűrni az almező jelét.

Fontos! A könnyebb áttekinthetőség és kezelhetőség érdekében tartsa be a mezők numerikus, illetve az almezők alfanumerikus sorrendjét.

Törléshez csupán a megfelelő sort (mezőtörlés esetén) vagy karaktert (almező törlésekor) kell a fájlból törölni.

Fontos! A fájl azon részeit, amelyről itt nem volt szó, hagyja változatlanul!

2.5.1.5 Címkék

HELY	szerver
FÁJL	/corvina/etc/opac/ccl/notes.cfg
STRUKTÚRA	\$START «mezőkód» “«indikátor»” = “ «szöveg»” \$END
HASZNÁLAT	Ebben a fájlban azt a szöveget adhatjuk meg, amely egy adott mező előtt szerepel normál, illetve hosszú kijelzésben. Ez a szöveg változhat annak függvényében, hogy a mező indikátorainak mi az értéke. A szöveg megváltoztatásához keresse meg a kérdéses mezőt, és ellenőrizze, hogy az indikátorainak kombinációi megfelelően szét vannak-e választva. Változtassa meg a szöveget vagy vegyen fel új indikátorkombinációt és adjon meg hozzá új szöveget! A csillag (*) bármilyen indikátort jelenthet.

2.5.1.6 Csatolt anyagok megjelenítése

HELY	kliens
FÁJL	-
STRUKTÚRA	-
HASZNÁLAT	A csatolt anyagok megjelenítéséhez a Corvina az operációs rendszer alapértelmezett

internetböngészőjét használja. Az alapértelmezett internetböngésző beállításáról az operációs rendszer súgójában vagy felhasználói leírásaiban találhat útmutatást.

Magának a csatolt fájlnek a megjelenítését a beállított internetböngésző végzi el. Az internetböngésző súgójában nézzon utána a fájlok megjelenítésével kapcsolatos tudnivalóknak!

2.5.1.7 Egyéni alapértelmezések

HELY	kliens
FÁJL	c:\corvina\JavaClients\JavaPAC\cfg\«felhasználónév»\opac.ini
STRUKTÚRA	<p>MAIN «főindexek száma»</p> <p>[+] «főindex kódja» «megnevezése» = «alapértelmezett érték»</p> <p>LIMIT «korlátozó index kódja» «megnevezése» «lehetséges érték kódja» «megnevezése» - mindegy</p> <p>END</p>
HASZNÁLAT	<p>Ha egyéni keresőfelület kíván létrehozni alapértelmezett keresőkifejezéssel, másolja át az opac.ini fájlt a c:\corvina\JavaClients\JavaPAC\cfg alkönyvtárban létrehozott, saját felhasználói nevével megegyező nevű alkönyvtárba. Állítsa össze saját keresőfelületét (ld. fent)! Alapértelmezett keresőfeltételt úgy hozhat létre, hogy (1.) a megfelelő főindex sorában egyenlőségjelet követően megadja az alapértelmezett értéket vagy (2.) a korlátozó index lehetséges értékeinek felsorolását nem a – mindegy értékkel, hanem tetszőleges másik értékkel kezdi.</p>

2.5.2 JavaCAT

2.5.2.1 Szöveges üzenetek

HELY	kliens
FÁJL	c:\corvina\JavaClients\JavaCAT\cfg\messages
STRUKTÚRA	<p>msgid = “«angol nyelvű üzenet»”</p> <p>msgstr = “«magyar nyelvű üzenet»”</p>
HASZNÁLAT	A megváltoztatni kívánt szöveget keresse meg a fájlban! Ügyeljen a teljes, betűnkénti egyezésre!

Az így talált sorban az időzőjelek közti magyar nyelvű szöveget cserélje tetszőleges, Önnek megfelelő szövegre!

Fontos! Az angol nyelvű üzenetet ne módosítsa, csak a magyar nyelvű üzenetet!

Fontos! Ellenőrizze, hogy a fájl utolsó sorában három csillag (***) szerepeljen!

2.5.2.2 Katalógusok

HELY	kliens
FÁJL	c:\corvina\JavaClients\JavaCAT\cfg\database.cfg
STRUKTÚRA	\$START «adatbázisnév» SERVER «szerver neve vagy IP-száma» VCAT «vcat port» VCIRC «vcirc port» VERSION 3.1 \$END
HASZNÁLAT	Ha új adatbázis felvételéhez szükséges az adatbázis neve, kódja és a szerverének neve vagy IP-száma. Válassza ki, hogy a listában hol szerepeljen az új katalógus! A katalógusválasztólistában (ld. JavaCAT felhasználói kézikönyv) a katalógusok sorrendje megegyezik azzal a sorrenddel, amelyet ebben a fájlban látunk. Hozzon létre új, üres struktúrát, majd adja meg a szerver címét (lehet ez teljes név – pl. giant.lib.klte.hu – vagy IP-szám – pl. 193.6.198.1 – is), azt a portot, amelyen a katalógushoz tartozó katalógizáló-szerverprogram fut (ez általában 5020) és azt a portot, amelyen a kölcsönző-szerverprogram fut (általában 5021).
Fontos!	Ügyeljen rá, minden katalógust \$START kulcsszó vezessen be és \$END kulcsszó zárjon!
Fontos!	Ellenőrizze, hogy a fájl utolsó sorában három csillag (***) szerepeljen! Az adatbázisok adatainak módosítását a JavaCAT Beállítások/Adatbázisok menüpont alatt is elvégezheti.

2.5.2.3 Mezőmagyarázatok

HELY	kliens
FÁJL	c:\corvina\JavaClients\JavaCAT\cfg\notes.cfg

StruktúraField«mezőkód» = “«mezőmagyarázat»”
 Field«mezőkód»Subf«almezőjel» =
 “«almezőmagyarázat»”

HASZNÁLAT A mezőmagyarázó szöveg a szerkesztőfelület bal oldalán, az egyes sorok elején szerepel. Az almezőkhöz tartozó szövegek, a lista almezőbeszúráshoz menüpont bekapcsolt állapota esetén legördülő menüben jelennek meg, ezzel segítve a felhasználót megfelelő almezőjel kiválasztásában. Mindkét szövegtípus megjelenik a szerkesztőfelület alsó sorában, ha a kurzor az adott mezőkódon vagy almezőjelen halad át.

Új mezőmagyarázat felvételéhez pozícionáljon a felvenni kívánt mezőkód előtti legnagyobb mezőkódhoz tartozó utolsó sorra és ezután hozzon létre új üres sort.

Fontos! Az áttekinthetőség és a kezelhetőség megtartása érdekében őrizze meg a mezőkódok numerikus sorrendjét! Ez a program hatékonyságát is növeli.

Az új sorba Field kulcsszót követően, vele egybeírva adja meg a mező kódját, majd egyenlőségjel után, időzőjelek közé írja be a szerkesztőfelületen látni kívánt magyarázó szöveget. A következő sorokban sorolja fel a mezőhöz tartozó almezőket a fenti szerkezetben. Ha csak almezővel szeretné bővíteni a fájlt, keresse meg az adott mezőt és csak az almezőnek hozzon létre új sort. A téves szövegeket (az időzőjelek között) javíthatja, a feleslegeseket törölheti.

Fontos! Ellenőrizze, hogy a fájl utolsó sorában három csillag (***) szerepeljen!

2.5.2.4 Tiltott almezők

HELY	kliens
FÁJL	c:\corvina\JavaClients\JavaCAT\cfg\disable.cfg
STRUKTÚRA	«mezőkód» [«almezőjel»]
HASZNÁLAT	Soroljuk fel külön sorokba szedve a felhasználó módosítások előtt letiltani kívánt mezőket, illetve almezőket. Az előbbi esetben az adott mező minden almezője tiltott lesz, az utóbbiban csak a megjelölt almező(ke)t nem szerkesztheti.

Fontos! Ellenőrizze, hogy a fájl utolsó sorában három csillag (***) szerepeljen!

2.5.2.5 Rekordellenőrzés

HELY	kliens
FÁJL	c:\corvina\JavaClients\JavaCAT\compile.bat
STRUKTÚRA	... «ellenőrzőosztály»Check.java
HASZNÁLAT	Ez a fájl fordítja le az ellenőrzőosztályt (ld. alább). Csak akkor kell megváltoztatni, ha az alapértelmezett ellenőrzőosztály helyett másikat hozunk létre (ld. alább). Ekkor az újonnan létrehozott osztály nevét kell a fájl harmadik sorának végére írni: «új név»Check.java !

Fontos! A fájlt többi részét ne változtassa meg!

HELY	kliens
FÁJL	c:\corvina\JavaClients\JavaCAT\cfg\JavaCAT.ini
STRUKTÚRA	CHECK «ellenőrzőosztály»
HASZNÁLAT	Az ellenőrzőosztály kis méretű alkalmazás, amely ellenőrzéseket végez a rekordon mentés előtt. Ha ilyen ellenőrzést nem kíván, hagyja üresen a sort a CHECK kulcsszó után. Telepítéskor a DefaultCheck alkalmazást tartalmazza a rendszer, ennek ellenőrzési mechanizmusát megváltoztathatja (ld. alább) vagy más alkalmazásra cserélheti le.

Fontos! Utóbbira csak akkor van szükség, ha az ellenőrzési mechanizmusokat váltogatni akarja. Ellenkező esetben elegendő a DefaultCheck.java fájlt módosítani (ld. alább).

Ha mégis új alkalmazás létrehozás mellett dönt, készítsen másolatot a DefaultCheck.java fájlról ugyanabban az alkönyvtárban és adjon neki tetszőleges nevet! Használja a könyvtár vagy saját felhasználói nevét és a Check kulcsszót! Az új fájl kiterjesztése is .java legyen (pl. KLTECheck.java). Módosítsa az új fájl 3. sorát! Ehelyett:

```
public class DefaultCheck extends RecordCheck {  
    ez szerepeljen  
    public class «új név»Check extends RecordCheck  
    {
```

Fontos! Módosítsa (ld. fent), majd futassa le a compile.bat fájlt.

Fontos!	A fájlt többi részét ne változtassa meg! A JavaCAT.ini-ben a CHECK kulcsszó után adhatja meg a használni kívánt ellenőrzőosztály nevét: «név»Check ! Ugyanezt megteheti a JavaCAT Beállítások/Szerkesztőbeállítások menüpontja alatt is.
HELY	kliens
FÁJL	c:\corvina\JavaClients\JavaCAT\DefaultCheck.java
STRUKTÚRA	<pre> else if («feltétel») { errString = "«hibaüzenet»"; return false; } </pre>
HASZNÁLAT	<p>Ebben a fájlban ellenőrzéseket definálhatunk, amelyeket a program a rekordok mentése előtt elvégez. Ha a feltétel teljesül a program hibaüzenet kíséretében megakadályozza a rekord elmentését. Ha csak a mentés során olvasható hibaüzenetet kívánja megváltoztatni az idézőjelek közti szöveget módosítsa megfelelően. Új ellenőrzés megfogalmazásához szúrjon be egy fenti struktúrát a fájl végén látható return true; sor elé!</p> <p>A feltétel megfogalmazásához az alábbi függvényeket használhatja.</p>

has(«mezőkód»)

– igaz, ha van a rekordban adott mezőkódú mező.

has(«mezőkód», \$«almezőjel»)

- igaz, ha van a rekordban adott mezőkódú mező és annak van adott almezőjelű almezője.

is(«mezőkód», \$«almezőjel», "«érték»")

- igaz, ha van a rekordban adott mezőkódú mező, annak van adott almezőjelű almezője és ennek az almezőnek az értéke a megadott értékkel egyenlő.

is(«mezőkód», \$«almezőjel», «értékek»)

- igaz, ha van a rekordban adott mezőkódú mező, annak van adott almezőjelű almezője és ennek az almezőnek az értéke a megadott értékek valamelyikével egyenlő. Az értékeket előzőleg (a fájl elején) definálni kell:

String[] «értékek» = { "«érték»", "«érték»", ... }

is(«mezőkód», ind, "«érték»")

- igaz, ha van a rekordban adott mezőkódú mező és annak indikátora a megadott értékkel egyenlő. Az érték helyén használhatunk joker karaktereket is: ? (nem üres), * (bármilyen, lehet üres is). Az üres helyet pont (.) jelzi.

is(«mezőkód»,ind,«értékek»)

- igaz, ha van a rekordban adott mezőkódú mező és annak indikátora a megadott értékek egyikével egyenlő. Az értékeket előzőleg (a fájl elején) definálni kell:

String[] «értékek» = {“«érték»”,“«érték»”,...}

Az érték helyén használhatunk joker karaktereket is: ? (nem üres), * (bármilyen, lehet üres is). Az üres helyet pont (.) jelzi.

is(“«mező afejlécben»”,“«érték»”)

- igaz, ha a rekordban fejlécének adott mezőjében a megadott érték szerepel.

is(“«mező afejlécben»”,«értékek»)

- igaz, ha a rekordban fejlécének adott mezőjében a megadott értékek valamelyike szerepel. Az értékeket előzőleg (a fájl elején) definálni kell:

String[] «értékek» = {“«érték»”,“«érték»”,...}

A feltétel megfogalmazásához a fenti függvényeket kombinálja logikai műveletekkel, használhatja a && (és), || (vagy) valamint ! (nem) operátorokat és zárójeleket tetszőleges számban és kombinációban. Fogalmazza meg a hibaüzenetet, majd mentse el a fájlt.

Fontos! Bármit módosított, futassa le a compile.bat fájlt.

Fontos! A fájlt többi részét ne változtassa meg!

A nevezett fájlban részletesebb, példákkal illusztrált leírás olvasható a hibafeltételek megfogalmazásáról.

2.5.2.6 Fájl csatolása

HELY	kliens
FÁJL	c:\corvina\JavaClients\JavaCAT\cfg\JavaCAT.ini
STRUKTÚRA	DIGIT «digitalizáló osztály» TRANS «átviteli osztály»
HASZNÁLAT	Az digitalizáló és az átviteli osztályok kis méretű alkalmazások, amelyek a csatolandó fájl

digitalizálásáról, illetve a már csatolt fájl tárolási helyére juttatásáról gondoskodik. Ha nem kíván fájlokat csatolni a bibliográfiai rekordokhoz, hagyja üresen a sort a DIGIT és a TRANS kulcsszó után. Telepítéskor a SimpleDigitAppl illetve a WebServerTransAppl alkalmazást tartalmazza a rendszer. Az előbbi alkalmas tetszőleges digitalizáló alkalmazás illesztéséhez, az utóbbi pedig tetszőleges WEB-szerverre juttatja a csatolt fájlokat. Az előbbit ne módosítsa, az utóbbit is csak akkor, ha a Corvinához ObjektumSzerver is tartozik. Ebben az esetben a TRANS kulcsszó után írja be az ObjServerTransAppl kulcsszót.

2.5.2.7 Digitlizálás

HELY	kliens
FÁJL	c:\corvina\JavaClients\JavaCAT\cfg\digit.cfg
STRUKTÚRA	APPL «digitalizáló alkalmazás» dest «célkönyvtár»
HASZNÁLAT	A digitalizáló alkalmazás tetszőleges program teljes elérésiútvonala lehet a szükséges indítási paraméterekkel együtt (pl. szkennert). A célkönyvtár az a könyvtár, amelybe a digitalizáló alkalmazás a digitalizált anyagokat elhelyezi. A digitalizáló alkalmazás sorát az APPL kulcsszót követően üresen hagyhatjuk, akkor a felhasználónak “kézzel” kell a célkönyvtárba másolnia a csatolandó fájlokat.
Fontos!	Célkönyvtárat mindig meg kell adni, s írhatónak kell lennie.
Fontos!	Ellenőrizze, hogy a fájl utolsó sorában három csillag (***) szerepeljen!

2.5.2.8 Fájlserver

HELY	kliens
FÁJL	c:\corvina\JavaClients\JavaCAT\cfg\ftp.cfg
STRUKTÚRA	HOST «szerver» PREFIX «a fájlnev elé illesztendő rész» DESTDIR «célkönyvtár»
HASZNÁLAT	Ezt a fájlt akkor használja a rendszer, ha a JavaCAT.ini fájlban a TRANS kulcsszó sorában a WebServerTransAppl kulcsszó szerepel. A HOST

kulcsszó után adja meg a szerver nevét vagy IP-számát! A DESTDIR kulcsszó sorában annak az alkönyvtárnak kell szerepelnie, ahol a csatolt anyagok tárolásra kerülnek. Ennek a Web-szerver html-oldalait tartalmazó (általában HTDOCS nevű) alkönyvtár alatt kell lennie. A PREFIX kulcsszó után azt a prefixumot kell megadni, amelyet a fájlnev elé illesztve az internetböngésző megtalálja az adott fájlt. Általános esetben ez a Web-szerver címéből és a célkönyvtár Web-szerver alapkönyvtárt követő részéből áll össze.

Fontos! A sikeres átvitelhez szükséges, hogy a felhasználó írni tudja a megadott célkönyvtárat. Ezt a szerver operációs rendszerének szintjén lehet beállítani.

Fontos! A fájl többi részét ne módosítsa!

Fontos! Ellenőrizze, hogy a fájl utolsó sorában három csillag (***) szerepeljen!

2.5.2.9 Objektumszerver

HELY	kliens
FÁJL	c:\corvina\JavaClients\JavaCAT\cfg\objserv.cfg
STRUKTÚRA	HOST «szerver» PORT «port» ARCH [«archívum» «archívum» ...]
HASZNÁLAT	Ezt a fájlt akkor használja a rendszer, ha a JavaCAT.ini fájlban a TRANS kulcsszó sorában a ObjServerTransAppl kulcsszó szerepel. A HOST kulcsszó után adja meg a szerver nevét vagy IP-számát! A PORT kulcsszó sorában annak a portnak kell szerepelnie, amelyen az objektumszerver fut (általában 80). Az ARCH kulcsszó után az objektumszerveren használt archívumokat kell felsorolni (ld. az objektumszerver felhasználói leírását).
Fontos!	A fájl többi részét ne módosítsa!
Fontos!	Ellenőrizze, hogy a fájl utolsó sorában három csillag (***) szerepeljen!

2.5.2.10 Szerkesztőfelület

HELY	kliens
FÁJL	-
STRUKTÚRA	-

HASZNÁLAT a JavaCAT Beállítások/Szerkesztőbeállítások menüpontja alatt konfigurálható a szerkesztőfelület színei, a betűméret, a magyarázó szövegek hossza, a sorok hossza. Ld. erről részletesebben a JavaCAT online helpjét és felhasználói leírását.

2.6 Beállítások az Admin modullal

A használat előtt a felhasználókat el kell látni jogosultsággal. Ezt a JavaADM klienssel tehetjük meg. Alap helyzetben egy szerepkör van definiálva: "mindenes", amihez minden funkció hozza van rendelve, ezt akár minden felhasználó megkaphatja, de ha szükséges lehet definiálni eltérő szerepköröket, és ezeket rendelni az egyes felhasználókhoz. A kölcsönzéshez és gyarapításhoz szükséges alapadatokat szinten a JavaADM-mel lehet megadni: példány és olvasótípusok, kölcsönzési politika, szállítók, lelőhelyek, stb.

2.7 A mentések automatizálása

A mentés folyamata lehet kézi, illetve automatizált. Amennyiben rendelkezésre áll elegendő hely automatikusan elérhető médián (pl. második merevlemez meghajtó), akkor az utóbbi módszer javasolt.

Ha két merevlemez meghajtó található a gépben, akkor célszerű az egyiket az operációs rendszer és az adatbázis tárolására, a másikat pedig a mentések IDEIGLENES tárolására használni. Fontos, hogy a diszken lévő mentéseket bizonyos időközönként külső médiára másoljuk. Ennek a fő oka, hogy a szerver egyes hardver elemeinek meghibásodása mindkét belső meghajtón tönkretetheti az adatokat.

Oracle adatbázisok mentése a különböző visszaállítási szolgáltatások tekintetében az alábbi típusúak lehetnek.

2.7.1 Offline

Ekkor egy (normal v. immediate módon) leállított unmount állapotban lévő adatbázis-példány adat-, online napló és vezérlőállományait lementjük (pl. mágnesszalagra vagy az eredeti adathordozótól független médiára). Ezután azonban bármilyen adatbázis meghibásodás esetén csak a mentés pillanatát tükröző teljes adatbázis állítható vissza. Az adatfájlok mentése történhet operációs rendszer segédprogrammal (tar, cp, stb.) vagy Oracle RMAN utility-vel, illetve harmadik gyártó média-kezelő szoftverével.

2.7.2 *Online*

Ebben az esetben az adatbázis szolgáltatások megszakítása és korlátozása nélkül készíthető mentés a teljes adatbázisról vagy annak egy részéről (egy vagy több táblaterről). Az adatbázist alkotó adatfájlok mentésére ugyancsak több lehetőség van (tar, cp, RMAN), azonban lemásolásuk előtt és után néhány speciális SQL utasítás végrehajtására is szükséges (alter tablespace ... begin backup, stb.). A helyreállítás tekintetében a legtöbb szolgáltatás ennél a mentési típusnál vehető igénybe (teljes, részleges, idő alapú), de a végrehajtása megalapozott Oracle ismereteket igényel. Online mentés esetén az adatbázist ún. archiveolog üzemmódban kell futtatni.

Amennyiben az Oracle telepítését az e-Corvina Kft végezte, úgy az \$ORACLE_HOME/admin/backup/ob_save.sh script ütemezett futtatásával a mentés megoldható. Futtatás előtt a scriptben az OB_TAPE_N környezeti változót be kell állítani.

2.7.3 *Export/Import*

Egy Oracle segédprogrammal (exp dbname/dbname) az adatbázis struktúráját és adatait fájlba menthetjük, amit egészében vagy részben (pl. egy táblát) egy másik segédprogrammal (imp dbname/dbname) visszatölthetünk egy meglévő üres adatbázisba. Előnye, hogy a mentés ideje alatt az adatbázis elérhető, továbbá, hogy visszatölthető egy másik adatbázis példányba is (még másik platformon futó Oracle esetén is), hátránya, hogy a helyreállítás lassú folyamat és csak a mentéskori állapot állítható vissza. Többnyire csak kiegészítő mentésként szokták használni.

A mentés és helyreállítás technikáinak részletes leírása az Oracle Server online dokumentáció „Backup and Recovery” fejezetében található.

3 A rendszergazda napi feladatai

3.1 *Felhasználók adminisztrálása*

3.1.1 *Új felhasználók felvétele*

UNIX és Oracle felhasználók definiálására nincs szükség, a Corvina rendszer felhasználói „corvina” userként dolgoznak a szerveren.

A Corvina felhasználók adminisztrálása, és a szükséges jogosultságok beállítása a JavaAdmin program segítségével történik.

3.2 Szabad diszk terület ellenőrzése

df -k :Az egyes partíciókon lévő szabad területek méretét listázza kilobyte-ban. Amennyiben a szabad terület a root (/), az ideiglenes (/tmp) vagy a *Corvina* adatbázis és a csatlakozó területek felhasználható területe 20% alá csökken, haladéktalanul meg kell kezdeni a probléma elhárítását.

Megoldások:

- ✓ Már nem használt állományok törlése
- ✓ core fájlok törlése
- ✓ Szükséges, de mozgatható állományok (nincsenek folyamatos használatban, illetve nem kötött a fizikai elhelyezkedésük) más partícióra mozgatása
- ✓ Nagyobb meghajtó beszerzése. Ha a *Corvina* adatbázisa kerül az új meghajtóra, kérje szakembereink segítségét.

3.3 Mentés lefutásának ellenőrzése

A feladat két részből áll.

- ✓ Operációs rendszer szinten létezik-e az elmentett file, a mentésről készült log nem tartalmaz-e hibát.
- ✓ Adatbáziskezelő szinten a mentés helyes.

Ez utóbbi ellenőrzés folyamata a következő:

Adatbáziskezelő szinten a legegyszerűbb, ha a tesztadatbázisba visszatöltjük az adatokat. Amennyiben ez hibátlanul megtörténik, úgy a mentést helyesnek feltételezzük. Ez idő és gépteljesítmény igényes folyamat, végrehajtása a normál napi munkavégzés közben nem javasolt.

A mentésről készült log neve és helye egyedi, a mentés mikéntjétől függ. Automatikus mentés esetén jellemzően a root email-ben kapja a státuszinformációt.

3.4 Beragadt user processzek ellenőrzése, takarítása

A feladat a szokásos rendszergazdai tevékenységek közé tartozik, azzal a kiegészítéssel, hogy a nobody user nevében futtatott httpd processzekre külön oda kell odafigyelni.

A processzek manuális indításáról/leállításáról szóló fejezetben láthatjuk, melyik *Corvina*-val kapcsolatos felhasználó mely processzeket kell, hogy futtassa. Ezek felhasználók nevében semmi más nem futhat tartósan. Ha ezektől eltérő processzt találunk 1-2 napos futási idővel, az nagy valószínűséggel hibás, beragadt processz.

A szerver oldali backendek egy-egy *oracleCORVINA* processzt is elindítanak. Előfordulhatnak szülő-processz nélküli *oracleCORVINA* processzek, amelyeket sok erőforrást felemésztenek, ezeket csak az Oracle újraindításával lehet törölni.

4 A rendszergazda időszakosan elvégzendő feladatai

4.1 Adatok gyűjtése a szerver leterheltségéről

A rosszul felmért igények, vagy a rendszer használata során felmerülő terhelési csúcsok (pl. egyetemi könyvtár vizsgaidőszak környékén) alulméretezésből adódó elégtelen működéshez vezethetnek. Ezek időbeni felismerése fontos a folyamatos hatékony működéshez.

sar

A program alkalmas az erőforráshasználat igen sokrétű mérésére.

Javasolt mérések:

```
sar -A -o sar.log 10 > /dev/null
```

Ez tíz másodperc időtartamú teljeskörű mérést jelent. Performanciatesztre 1-2 napig futtassuk, hely hiányában csak a kérdéses opciókkal.

A log elemzése sar -A -f sar.log segítségével történik. Részinformációk a -A kapcsoló helyettesítésével kaphatóak.

vmstat

A virtuális memória használatával kapcsolatos jellemzőket méri, köztük a processz, virtuális memória, diszk, trap és CPU aktivitásokat.

Javasolt mérések:

```
vmstat 5 > vmstat.log
```

Ez ötmásodpercenként végzi a mérést, futtassuk a kérdéses időszakban 1-2 napig, majd a kapott logfájlt elemezzük.

iostat

Alkalmas terminál, diszk, stb. I/O műveleteinek mérésére, többek közt a műveletek CPU foglaltságának figyelésére.

Javasolt mérések:

```
iostat 5 > iostat.log
```

Ez ötmásodpercenként végzi a mérést, futtassuk a kérdéses időszakban 1-2 napig, majd a kapott logfájlt elemezzük.

4.2 Szabad táblaterület ellenőrzése

Az adatbázis táblaterületét az alábbi módon tudjuk lekérdezni:

```
sqlplus system/manager
```

```
select tablespace_name, sum(bytes) from dba_free_spaces group by
tablespace_name;
```

A lekérdezés byte-okban adja vissza az egyes táblateretek szabad területét. Ha az adat táblatér szabad területe 200MB alá csökken, akkor adatfájl hozzáadásával célszerű a területet növelni.

5 Segédprogramok

5.1 MARC-rekordok exportálása

A MARC-rekordok a dbmarc-segédprogrammal exportálhatók fájlba. A program a /corvina/sbin-alkönyvtárban található, paraméterei a következők:

```
/corvina/sbin/dbmarc -o output_file [-L] [-R] [-r
recnum_range] [-u uniqueid] [-v authority_file] database
-o          a fájl neve, amibe a MARC-rekordok kerülnek,
-L          hatására a helyi authority-ket exportálja,
-R          hatására a háttéauthority-ket exportálja,
-r m-n      hatására az m és n közé eső rendszerazonosítóval
             rendelkező rekordokat exportálja,
-u id       hatására az id bibliográfiai azonosítójú leírást
             exportálja
-v file      hatására az exportált bibliográfiai rekordokhoz
             kapcsolódó authority-ket is kiexportálja, ezek a file
             nevű fájlba kerülnek,
database    az adatbázis neve, amiből a rekordokat exportáljuk
```

A -L és -R paraméterek hiányában az adatbázisban található bibliográfiai rekordokat exportálja.

Példák:

Bibliográfiai rekordok exportja:

```
/corvina/sbin/dbmarc -o bib.marc hpakdb
```

Helyi authority-k exportja:

```
/corvina/sbin/dbmarc -L -o auth.marc hpakdb
```

5.2 A dbload használata

A dbload-segédprogram adatbázisépítésre szolgál. Paraméterei:

```
/corvina/sbin/dbload [-m marc_file] [-a authority_file]
[-L] [-S] [-T sort_directory] [-N <first>-<last>] [-.
<dot-freq>] database
```

```
-m file      a bibliográfiai rekordokat tartalmazó fájl neve,
-a file      az authority-rekordokat tartalmazó fájl neve,
-L          az authority-rekordokat a háttéauthority-k közé tölti
```

-S	a szkeleton-authority-k létrehozását titltja,
-T	ideiglenes fájlok célkönyvtára
-N	a végrehajtandó lépések tartománya
-. m	a felhasználó m-rekordonként kap visszajelzést az egyes lépések végrehajtásakor
database	a létrehozandó adatbázis neve

5.3 A checkmarc használata

A checkmarc-segédprogrammal MARC-rekordokat tartalmazó fájlokat ellenőrizhetünk, illetve alakíthatunk szöveges formába.

Ellenőrzés

```
/corvina/bin/checkmarc <bib.marc
```

Szöveges formába alakítás

```
/corvina/bin/checkmarc -t 001 -f latintwo.cfg <bib.marc
```

6 Rendszermonitorozás - Hibafelderítés

6.1 Speciális felhasználók szerepe

corvina	:	A <i>Corvina</i> rendszer tulajdonosa
oracle:		Adatbázistulajdonos, ő tud műveleteket végezni a <i>Corvina</i> -val kapcsolatos adatbázisokon.
opac	:	A katalogizáló modul kiszolgálója

6.2 A UNIX, az Oracle és a Corvina napló-fájljai

6.2.1 A UNIX napló-fájljai

A *Corvina* Solaris 8 operációs rendszeren fut, jelen fejezet ezen operációs rendszer napló-fájljainak összegyűjtése.

```
/var/adm/log/
/var/cron/log/
/var/log/
/var/spool/lp/logs/ = /var/lp/logs/ = /etc/lp/logs/
/dt/
/dt/sessionlogs/
```

6.2.2 Az Oracle napló-fájljai

Cégünk az *Oracle 8i v* verzióját installálja a *Corvina* alapjaként.

Az Oracle általános naplófájljai a :

```
/orahome/admin/CORVINA/bdump
```

könyvtárban találhatók.

6.2.3 A Corvina napló-fájljai

A *Corvina* rendszer működéséről folyamatos naplózás készül, amit a `/corvina/var/log`, vagy másolatban a `/corvina/report` könyvtárakban találunk.

`update.log`: az online update logja, a hibajelentéseken kívül minden rekordmentésről is készül bejegyzés: időpont, rekord azonosító, státusz. A log-ból a sikertelen mentések oka visszakereshető.

`vcat.err`: VCAT backend hibák naplója

`vcat.log`: VCAT rekord naplója: mentések időpontja, felhasználó azonosító, rekord azonosító

`vcirc.err`: VCIRC hibák naplója

`report.log`: Report modul használatának logja

`ccl.log`: Keresések logja, ez a rekordfrissítésekről időpont és státuszinformációkat tartalmaz.

Lényegesek még a `bk` kiterjesztésű fájlok, ezek a törölt rekordokat tartalmazzák.

Egyéb *Corvina* naplófájlok:

`/corvina/local/etc/apache/logs/`

`/corvina/var/log/`

6.3 Processzek leállítása, újraindítása, meglétük ellenőrzése

6.3.1 A processzek leállítása

A *Corvina* a boot-folyamat során automatikusan elindul. Leállítása több lépésben történhet.

A webservert leállítása:

`/corvina/sbin/apachectl stop`

Az *Oracle* adatbáziskezelő leállítása:

`sqlplus internal`

password: manager

`shutdown`

Amennyiben futó processzek miatt a rendszer nem engedi leállítani az adatbázis-szervert, ez esetben:

`shutdown -force`

Ez utóbbi esetén azonban előfordulhat, hogy az adatbázis inkonzisztens állapotban marad. Ezt elkerülhetjük, ha megbizonyosodunk róla, senki nem dolgozik az adatbázissal a leállítás pillanatában.

Az *Oracle* leállítás az updrec modul leállítását is eredményezi.

6.3.2 A processzek újraindítása

Először az adatbáziskezelőt indítsuk el:

```
sqlplus internal  
password: manager  
startup
```

Az Oracle listener indítása:

```
lsnrctl start
```

A következő lépés a webszerver indítása:

```
/corvina/sbin/apachectl start
```

Az utolsó de NAGYON FONTOS lépés az updrec modul elindítása:

```
su - dba  
/corvina/sbin/on_upd &
```

6.3.3 Processzek futásának ellenőrzése

Az updrec modul futásának ellenőrzése:

```
ps -ef | grep updrec
```

A szerveren a CAT, CIRC és PAC modulok aktuális száma:

```
ps -ef | grep -c vcat  
ps -ef | grep -c vcirc  
ps -ef | grep -c vccl
```

6.4 Gyakran feltett kérdések

HIBAJELENSÉG	MEGOLDÁS
Az elmentett rekordok nem jelennek meg az OPAC-ban.	Ellenőrizze, hogy a rekordok nem kerültek-e Exception 11 munkafájlba. Ha igen, akkor javítani kell a rekordot és újra elmenteni.
Az elmentett rekordok Exception 10 munkafájlba kerülnek.	Időpont ütközés miatt a rekord nem menthető, az adatbázisban lévő rekordot kell módosítani és az Exception 10-ből a rekordot törölni.

Az elmentett rekordok nem jelennek meg az OPAC-ban, és munkafájlban sem.	Ellenőrizze, hogy fut-e, az online update.
Böngészésnél szokatlanul lassú a rendszer, és ez időben állandósul.	Adatbázis hangolás.
Az Oracle leállítása és újraindítása, valamint adatbázis betöltés után az updrec nem fut.	Tipikus hiba. Ezeknél a műveleteknél a démon hibával leáll. Újraindítását lásd az 6.3.2 fejezetben.
Nem lehet kapcsolódni a szerverhez.	Ellenőrizze a hálózati kapcsolatot, a kliensnél beállított szerver adatokat (név, ip, port, adatbázis név), és a futó Oracle processzek számát.
ACQ, SER és ADM kliensekkel nem lehet kapcsolódni az adatbázishoz.	Ellenőrizze az Oracle listener futását.