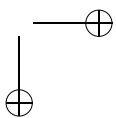
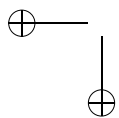
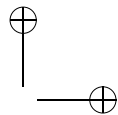
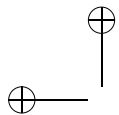


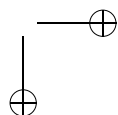
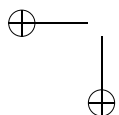
J2EE útikalauz Java programozóknak

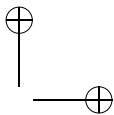
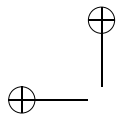




Szerkesztő
Nyékyné Gaizler Judit

Szerzők
Balázs Iván József
Csizmazia Balázs
Csontos Péter
Fornai Péter
Kégli Zoltán
Kozsik Tamás
Lakatos Attila
Mika Péter
Porkoláb Zoltán
Varga Tibor Ádám

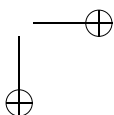
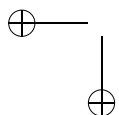


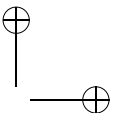
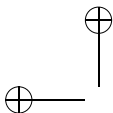


J2EE

útikalauz Java programozóknak

ELTE TTK Hallgatói Alapítvány
Budapest, 2002





Copyright © 2002 Balázs Iván József, Csizmazia Balázs, Csontos Péter, Fornai Péter, Kégli Zoltán, Kozsik Tamás, Lakatos Attila, Mika Péter, Nyékyné Gaizler Judit, Porkoláb Zoltán, Varga Tibor Ádám

Minden jog fenntartva.

Első kiadás, 2002. augusztus

Lektorálta: Fornai Péter, Csizmazia Balázs, Lakatos Attila

A Szerzők és a Kiadó e könyv tartalmi és formai összeállítása során a legjobb tudásuk szerint jártak el. Ennek ellenére nem zárható ki, hogy a könyvben előfordulnak hibák. A könyv, illetve a benne levő ismeretanyag használatából közvetlen vagy közvetett módon származó károkért sem a Szerzők, sem pedig a Kiadó nem vállal felelősséget.

Ezen kiadvány egészének vagy egy részének a másolása, reprodukálása (bármilyen formában), valamint lefordítása más nyelvekre kizárólag a Kiadó előzetes írásbeli hozzájárulásának birtokában engedélyezett.

A UNIX az X/Open bejegyzett védjegye. A Java a Sun Microsystems, Inc. védjegye. A Windows a Microsoft, Corp. védjegye.

A könyvben előforduló más védjegyek nevei nagy kezdőbetűkkel vannak írva, ha a Kiadónak tudomása van az adott védjegyről.

A szerzők megköszönik az ELTE TTK Informatikai Tanszékcsoportnak, hogy a kézirat elkészítéséhez rendelkezésükre bocsátotta az infrastruktúráját.

ISBN 963 463 578 4

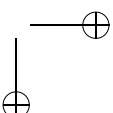
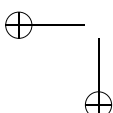
A könyv szerkesztése és szedése \LaTeX ben készült
Tipográfia, szedés és tördelés: Lőrentey Károly

Kiadó: ELTE TTK Hallgatói Alapítvány, Budapest
Nyomtatta és kötötte az Alfaprint Nyomda

Felelős vezető: Barabás Gábor

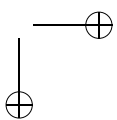
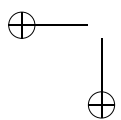
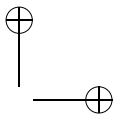
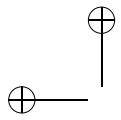
Borítóterv: Ruszkai Gábor

Példányszám: 1000



Áttekintés

Bevezetés	xix
1. Üzleti alkalmazások (Csizmazia Balázs)	1
2. A WWW objektumainak elérése (Csizmazia Balázs)	13
3. Java szervletek (Csizmazia Balázs)	65
4. JavaServer Pages (Mika Péter)	115
5. Távoli metódushívás (Csizmazia Balázs)	163
6. Java névleképezés- és katalógusinterfész (JNDI) (Varga Tibor Ádám)	197
7. Tranzakciók, a JDBC és a JTA (Kozsik Tamás)	225
8. Adatbázis-programozás (Lakatos Attila)	273
9. A CORBA és a Java kapcsolata (Porkoláb Zoltán)	325
10. A CORBA interfészgyűjteménye (Csizmazia Balázs)	361
11. Java Message Service (Csontos Péter)	399
12. Enterprise JavaBeans (Kégli Zoltán)	413
13. XML (Fornai Péter)	527
14. JavaMail (Balázs Iván József)	553
15. A Java Multimédia Framework (Csizmazia Balázs)	573
Irodalomjegyzék	667
Tárgymutató	679



Tartalomjegyzék

Bevezetés	xix
1. Üzleti alkalmazások (Csizmazia Balázs)	1
1.1. A komponens-korszak küszöbén	2
1.2. Az elemek	3
1.2.1. A WWW elérésére használható osztályok	3
1.2.2. Java szervletek – a webszerveroldali Java	4
1.2.3. A Java Server Pages technológia	4
1.2.4. Az XML technológia	5
1.2.5. A JDBC technológia	5
1.2.6. Távoli metódushívás (RMI)	5
1.2.7. Java és a CORBA kapcsolata	6
1.2.8. Névszolgáltatók elérése: a JNDI csomag	7
1.2.9. Enterprise JavaBeans modell	7
1.2.10. Tranzakciók kezelése Javában	8
1.2.11. A JMS: üzenet alapú infrastruktúra	9
1.2.12. A JavaMail infrastruktúra	9
1.3. Többrétegű alkalmazásmodell	9
2. A WWW objektumainak elérése (Csizmazia Balázs)	13
2.1. Egységes erőforrásnevek, nevek rendszere	14
2.2. Hálózati erőforrások URL-azonosítói	16
2.2.1. URL-azonosítók ábrázolása	17
2.2.2. Abszolút és relatív URL-azonosítók	18
2.3. URL-azonosítók általános formája	18
2.3.1. Az FTP protokoll URL-azonosítói	20
2.3.2. A HTTP (és a HTTPS) protokoll URL-azonosítói	20
2.3.3. A Gopher protokoll URL-azonosítói	21
2.3.4. Levezési cím URL-azonosítója	23
2.3.5. A USENET News objektumait azonosító URL-formák	23
2.3.6. TELNET URL-azonosítók	23
2.3.7. Fájlokat megnevező URL-azonosítók	24
2.3.8. Egyéb URL-azonosítók	24
2.4. A HTTP, HTTPS protokollok és alkalmazásaik	25
2.4.1. A MIME-szabvány	26
2.4.2. MIME tartalomtípusok paraméterezése, példák	28
2.4.3. A HTTP protokoll architektúrája	32
2.4.4. HTTP-fejlécmezők	34
2.4.5. Példák HTTP-alapú kliens-szerver kapcsolatokra	39

viii Tartalomjegyzék

2.5.	A WWW elérését támogató Java osztályok	41
2.5.1.	WWW erőforrások elérése az URL osztállyal	41
2.5.2.	Az URLConnection osztály alkalmazása	45
2.5.3.	A HttpURLConnection osztály	51
2.5.4.	Az elért WWW erőforrások tartalmának értelmezése	55
2.5.5.	Protokollkezelő osztályok felépítése	56
2.5.6.	A Java környezet bővítése új protokollkezelőkkel	58
2.5.7.	Tartalomkezelő osztályok	59
2.5.8.	A Java környezet bővítése új tartalomkezelőkkel	62
2.5.9.	Példa: FINGER protokollkezelő implementációja	63
3.	Java szervletek (Csizmazia Balázs)	65
3.1.	A HTML-űrlapok és működésük	66
3.2.	A Java szervletek	69
3.2.1.	A Java szervletek szerkezete	70
3.2.2.	A kliens kérés és szervlet válasz absztrakciója	72
3.2.3.	Szervletek inicializációja	77
3.2.4.	Szervletek származtatása a GenericServlet osztálytól	79
3.2.5.	HTTP protokoll alapú szervletek	81
3.2.6.	Szervletek szinkronizációja	87
3.3.	Állapotinformációk tárolása a szerveren	87
3.3.1.	Állapotinformációk kezelése cookie-k segítségével	90
3.3.2.	Állapotinformációk tárolása a session-mechanizmus segítségével	92
3.4.	A RequestDispatcher szerepe	95
3.5.	Példa HTTP-alapú szervlet alkalmazásokra	97
3.5.1.	A visszhang szervlet	98
	– Használat a JSDK 1.x változattal	101
	– Használat JSDK 2.x vagy Tomcat környezettel	102
3.5.2.	A felhasználó-bejegyzési szervlet	103
3.5.3.	Egy kliensoldali Java alkalmazás a fenti űrlapunkhoz	105
3.5.4.	Egy egyszerű szervlet a cookie-k bemutatására	107
3.5.5.	Egy cookie objektumok használatát szemléltető szervlet	109
3.5.6.	Egy látogatási számláló munkafolyamat-figyeléssel	111
3.5.7.	Egy webszerver-szerű szerver modellje	112
4.	JavaServer Pages (Mika Péter)	115
4.1.	A JSP és előnyei	116
4.2.	A JSP és a J2EE	119
4.3.	A JSP elemei	120
4.4.	Direktívák	121
4.4.1.	A <code>page</code> direktíva	121
4.4.2.	Az <code>include</code> direktíva	123
4.4.3.	A <code>taglib</code> direktíva	123
4.5.	Scriptelemek	124
4.5.1.	Deklarációk	124
4.5.2.	Script-részletek	124
4.5.3.	Kifejezések	126
4.5.4.	Implicit objektumok	126

4.6.	Akciók	128
4.6.1.	A <code><jsp:useBean></code> akció	129
4.6.2.	A <code><jsp:setProperty></code> akció	131
4.6.3.	A <code><jsp:getProperty></code> akció	132
4.6.4.	A <code><jsp:include></code> akció	133
4.6.5.	A <code><jsp:forward></code> akció	133
4.6.6.	A <code><jsp:plugin></code> akció	134
4.6.7.	A <code><jsp:param></code> akció	135
4.7.	Elemkönyvtárak használata és készítése	135
4.7.1.	Elemkönyvtárak használata	135
4.7.2.	Egyszerű elemek készítése	140
4.7.3.	Scriptváltozók bevezetése	149
4.7.4.	A törzsüket is feldolgozó elemek készítése	152
4.7.5.	Egymásbaágyazott elemek készítése	155
4.7.6.	Elemkönyvtárak validálása és a futásidejű kivételkezelés	158
5.	Távoli metódushívás (Csizmazia Balázs)	163
5.1.	A távoli metódushívás alapjai	163
5.2.	A távoli metódushívás modellje	165
5.3.	Távoli metódushívás Java környezetben	166
5.4.	Egy távoli objektum implementációja	167
5.5.	Paraméterátadás	172
5.6.	A Java RMI viselkedése kommunikációs hibáknál	173
5.7.	A kliens- és a szervercsonk összekapcsolása	174
5.7.1.	Távoli objektumok megnevezése	174
5.7.2.	Távoli referenciák elérése	175
5.7.3.	A registry implementációja	177
5.8.	A példaprogramunk befejezése	178
5.9.	Távoli objektumok osztott szemétygyűjtése	179
5.10.	Osztott szemétygyűjtés a szerveroldalon	180
5.11.	Távoli osztályok elérése	181
5.12.	A példaprogramunk tesztelése	184
5.13.	Visszatekintés a fejlesztés menetére	186
5.14.	Távoli metódushívás hálózati tűzfalak mögött	187
5.15.	Távoli metódushívás az IIOP protokoll felett	188
5.15.1.	Új szoftvereszközök	189
	– Az új <code>rmic</code> fordító	189
	– Az <code>idlj</code> fordító	190
5.15.2.	Objektumok IIOP-képessé alakítása	190
	– Mire kell vigyáznunk az RMI/IIOP használatakor	190
	– Az RMI/IIOP használata távoli szerverobjektumoknál	191
	– Az RMI/IIOP használata távoli kliensobjektumoknál	192
5.16.	CORBA objektumok érték szerinti átadása	193
6.	Java névleképezés- és katalógusinterfész (JNDI) (Varga Tibor Ádám)	197
6.1.	A JNDI megoldás	198
6.1.1.	Fogalmak áttekintése	199
6.1.2.	A JNDI mint a J2EE része	200

x *Tartalomjegyzék*

6.2.	Példák a fejezetben	201
6.3.	A JNDI rendszer konfigurációja	201
6.3.1.	Konfiguráció környezetobjektum segítségével	202
6.3.2.	Rendszerpropertyk használata a konfigurációra	204
6.3.3.	Appletparaméterek használata a konfigurációra	204
6.3.4.	Erőforrásállományok használata a konfigurációra	205
6.3.5.	Környezetfelépítési algoritmus	206
6.4.	Kontextusobjektumok	206
6.4.1.	Nevek	207
6.4.2.	A lookup metódus	208
6.4.3.	Szinkronizáció	210
6.4.4.	Név felépítése	211
6.4.5.	URI használata név helyett	212
6.4.6.	Kötések	212
6.4.7.	Hivatkozások	213
6.4.8.	Átírányítások	213
6.4.9.	Enterprise JavaBeans	215
	– Enterprise beanek megkeresése	215
	– Enterprise beanek paramétereinek lekérdezése	215
6.4.10.	A J2EE környezet egyéb osztott komponensei	216
6.5.	Katalóguskontextus objektumok	216
6.5.1.	Tulajdonságok	217
6.5.2.	Keresések	217
6.5.3.	Séma	218
6.6.	Események	218
6.7.	LDAP v3 bővítések	220
6.7.1.	Bővített műveletek	220
6.7.2.	Kontrollok	221
	– Kontextuskontrollok	221
	– Kapcsolatkontrollok	221
6.7.3.	Aszinkron visszajelzések	221
6.8.	Biztonság	222
6.9.	Szolgáltatói illesztőfelület	222
6.10.	Standard JNDI környezeti változók teljes listája	223
7.	Tranzakciók, a JDBC és a JTA (Kozsik Tamás)	225
7.1.	A tranzakciók ismérvei	226
7.2.	Tranzakciók határának kijelölése	227
7.2.1.	Programozott tranzakció-demarkáció	228
	– JDBC tranzakciók	230
	– JTA tranzakciók	234
7.2.2.	Deklaratív tranzakció-demarkáció	236
	– A tranzakciós attribútum lehetséges értékei	237
	– A példánk deklaratív tranzakció-demarkációval	239
	– A konténer közbelép	241
	– Mikor melyik tranzakciós attribútumértéket használjuk?	244
7.2.3.	A telepítési leírás	250
	– <code>transaction-type</code>	251
	– <code>container-transaction</code>	251
	– <code>trans-attribute</code>	253
7.2.4.	A tranzakció-demarkáció megoldásainak összehasonlítása	254

7.3.	Időtúllépés – timeout	255
7.4.	Tranzakciók izolációja	255
7.4.1.	Izolációs szintek	256
7.4.2.	Az izolációs szint beállítása	259
7.5.	Lokális és globális tranzakciók	259
7.5.1.	Lokális és globális tranzakciók a J2EE platformban	261
7.5.2.	JTA – Java Transaction API	262
7.5.3.	JTS – Java Transaction Service	263
7.5.4.	A JTA előnyei és hátrányai	265
7.5.5.	Optimalizálási lehetőségek	265
7.6.	Tranzakciók visszavonása	266
7.6.1.	A <code>SessionSynchronization</code> interfész	267
7.6.2.	Kompenzáló tranzakciók	268
7.7.	Tranzakciók különböző J2EE komponensekben	269
8.	Adatbázis-programozás (Lakatos Attila)	273
8.1.	JDBC meghajtóprogramok	275
8.1.1.	Két- és háromrétegű adatbázis-elérési modell	275
8.1.2.	JDBC-nek megfelelő meghajtóprogramok	276
8.2.	ODBC-JDBC	276
8.3.	Továbbfejlesztési tervek	277
8.4.	A JDBC használata	277
8.4.1.	JDBC használata szervletekben	277
8.4.2.	Fordítás	278
8.4.3.	Futtatás	278
8.5.	Meghajtóprogramok kezelése	278
8.5.1.	A meghajtóprogram kiválasztása	278
8.5.2.	Meghajtóprogramok regisztrálása	278
8.6.	Kapcsolattartás az adatbázissal	279
8.6.1.	Adatbázis-URL-ek	279
8.6.2.	Kapcsolat felvételének hagyományos módja	280
8.6.3.	Kapcsolat felvételének ajánlott módja	282
8.6.4.	Kapcsolatok újrahasznosítása	283
8.6.5.	Tranzakciókezelés	283
	– Tranzakció-izolációs szintek	284
	– Globális tranzakciók	285
8.6.6.	Információ a kapcsolatról	285
8.6.7.	Hibakezelés	286
8.6.8.	Kapcsolat lezárása	286
8.7.	JDBC típusok	286
8.7.1.	A JDBC típus használata	287
8.7.2.	SQL-Java típusmegfeleltetés	287
	– CHAR, VARCHAR és LONGVARCHAR	287
	– DECIMAL és NUMERIC	289
	– BINARY, VARBINARY és LONGVARBINARY	289
	– BIT, BOOLEAN	289
	– Egész számok	289
	– Lebegőpontos számok	289
	– Dátum/idő	290
	– Hivatkozás	290

xii *Tartalomjegyzék*

8.7.3.	SQL3 adattípusok	290
	– BLOB/CLOB	290
	– ARRAY	290
	– REF	291
	– Az SQL NULL kezelése	291
8.7.4.	Felhasználói SQL adattípusok	291
	– Paraméteres (DISTINCT) felhasználói SQL típus	291
	– Strukturált (STRUCT) felhasználói SQL típus	292
	– Felhasználói SQL adattípusok Java megfeleltetése	292
8.7.5.	Adatbázisbeli Java objektumok	296
8.8.	SQL utasítások végrehajtása	296
8.8.1.	A <code>Statement</code> interfész	296
	– <code>Statement</code> létrehozása	297
	– <code>Statement</code> végrehajtása	297
	– <code>Statement</code> megszakítása	298
	– <code>Statement</code> befejeződése	298
	– Escape-szintaxis használata	299
8.8.2.	A <code>PreparedStatement</code> interfész	301
	– <code>PreparedStatement</code> létrehozása	301
	– <code>PreparedStatement</code> végrehajtása	301
	– Bemenő paraméterek típusának lekérdezése	302
	– Bemenő paraméterek értékeinek megadása	302
	– Nagyméretű bemenő paraméterek megadása	302
8.8.3.	A <code>CallableStatement</code> interfész	302
	– <code>CallableStatement</code> létrehozása	303
	– <code>CallableStatement</code> végrehajtása	303
	– Kimenő paraméterek értékeinek lekérdezése	303
8.9.	SQL utasítások kötegelt végrehajtása	304
8.9.1.	Kötegelt végrehajtás előkészítése	304
8.9.2.	Kötegelt végrehajtás indítása	304
8.9.3.	Hibakezelés kötegelt végrehajtás esetén	304
8.9.4.	Példa kötegelt végrehajtásra	305
8.10.	Eredménytáblák kezelése	306
8.10.1.	Eredménytábla típusok	306
8.10.2.	Eredménytábla létrehozása	306
8.10.3.	Eredménytábla elérésének optimalizálása	307
8.10.4.	Navigálás az eredménytáblában	307
8.10.5.	Értékek beolvasása	309
8.10.6.	Értékek módosítása	310
8.10.7.	Új sor beszúrása	310
8.10.8.	Sor törlése	310
8.10.9.	Változások észlelése	310
8.10.10.	Eredménytábla lezárása	312
8.11.	Önálló eredménytáblák	313
8.11.1.	Adatforrás megadása	313
8.11.2.	Eredménytábla feltöltése	313
8.11.3.	Eredménytábla használata	314
8.11.4.	Eredménytábla események	314
8.12.	Példa	315

9. A CORBA és a Java kapcsolata (Porkoláb Zoltán)	325
9.1. Az ORB	326
9.2. CORBA szolgáltatások	328
9.2.1. Névszolgáltatás	328
9.2.2. Biztonsági szolgáltatás	329
9.2.3. Eseménykezelés	329
9.2.4. Externalizáció	329
9.2.5. Perzisztencia	329
9.2.6. Életciklus	330
9.2.7. Konkurencia-kezelés	330
9.2.8. Tranzakció-kezelés	330
9.3. Az Interface Definition Language (IDL)	331
9.4. Az IDL-Java leképezés	333
9.4.1. Nevek	333
9.4.2. Alaptípusok	333
9.4.3. Literálok	334
9.4.4. Konstans	335
9.4.5. Struktúra	336
9.4.6. Unió	336
9.4.7. Felsorolási típus	338
9.4.8. Tömb	339
9.4.9. Sztring	339
9.4.10. Sorozat	340
9.4.11. Any	341
9.4.12. Kivétel	342
9.4.13. Paraméterátadás	342
9.4.14. Kliens környezet	344
9.5. Példa a CORBA használatára	345
9.5.1. Az IDL forrás és fordítása	345
9.5.2. Generált állományok	346
– Java interfészek	347
– Helper osztályok	347
– Holder osztályok	348
– A kliens oldali csomópontok	348
– Szerver oldali váz	349
9.5.3. A szerver megvalósítása örökléssel	351
9.5.4. A szerver megvalósítása delegálással	354
9.6. CORBA névszolgáltatás	358
9.7. További lehetőségek	359
10. A CORBA interfészgyűjteménye (Csizmazia Balázs)	361
10.1. Az IR legalapvetőbb interfésze	363
10.2. A CORBA Any és a típuskódok	364
10.3. Az IR egyszerű és összetett komponensei	367
10.4. Az IR alkotóelemeiről részletesebben	372
10.5. Interfészek visszakeresése az interfészgyűjteményből	381
10.6. Az interfészgyűjtemény elemeinek azonosítása	381
10.7. A típuskódok és az Any	384
10.8. Az ORB objektumról bővebben	388
10.9. Interfészgyűjteményelemek és típuskódok kapcsolata	394

11. Java Message Service (Csontos Péter)	399
11.1. Bevezetés	399
11.1.1. Üzenetkezelés	399
11.1.2. A köztes réteg (middleware)	399
11.1.3. A bábeli zűrzavar és ami megszünteti	400
11.1.4. A JMS-ről röviden	400
11.1.5. A JMS és a J2EE többi részének kapcsolata	401
11.1.6. A fejezet szerkezete	401
11.2. Architektúra	401
11.2.1. Kapcsolódás az üzenetkezelő rendszerekhez	401
11.2.2. Az üzenetek kezelésének fajtái	402
11.2.3. Perzisztencia	404
11.3. A JMS eszköztára	404
11.3.1. Inicializációs eszközök	404
11.3.2. Az üzenetek küldése	406
11.3.3. Az üzenetek fogadása	408
– Aszinkron üzenetfogadás	408
– Szinkron üzenetfogadás	409
11.4. JMS-kompatibilis termékek	410
12. Enterprise JavaBeans (Kégli Zoltán)	413
12.1. EJB Bevezető	414
12.1.1. Komponens Architektúra	414
12.1.2. EJB mint komponens architektúra	416
12.1.3. EJB és JavaBeans	417
12.1.4. EJB készítés szereplői	417
– A bean gyártó	417
– Az alkalmazás összeállító	417
– Az EJB telepítő	418
– A rendszerüzemeltető	418
– A konténer és szerver gyártó	419
– A Perzisztencia Menedzser gyártó	419
– A fejlesztő eszköz gyártó	419
12.2. EJB architektúra áttekintése	419
12.2.1. Enterprise bean és EJB konténer	420
12.2.2. Átlátszatlanság felsőfokon	420
12.2.3. Enterprise bean típusok	421
12.2.4. Enterprise bean felépítése	421
– Üzleti logika avagy az enterprise bean osztály és segédei	422
– Programozástechnikai megkötések	423
– EJB objektum és távoli interfész	424
– Lokális interfész, azaz enterprise bean a szomszédban	425
– Otthon objektum és otthon interfész	426
– Lokális otthon objektum és lokális otthon interfész	427
– Telepítési leíró	428
– Alkalmazáserver specifikus fájlok	429
– Ejb-jar fájl	430
12.2.5. Az enterprise bean kliens oldali nézete	431
12.2.6. Kivételekről röviden	432

12.3.	Session beanek	433
12.3.1.	Állapotmentes session bean	434
12.3.2.	Példa állapotmentes session beanre	434
12.3.3.	Példányfarm és az állapotmentes session bean életciklusa	435
12.3.4.	Állapottal rendelkező session bean	436
12.3.5.	Passzíválás-aktiválás	436
12.3.6.	Példa állapottal rendelkező session beanre	437
	– Távoli interfész	437
	– Otthon interfész	438
	– Enterprise bean osztály	438
	– Teszt kliens	440
	– Telepítési leíró	442
	– Alkalmazáserver specifikus fájl	442
12.4.	Entitás beanek	443
12.4.1.	Entitás beanek életciklusa és a példányfarm	445
12.4.2.	Az elsődleges kulcs szerepe	447
12.4.3.	BMP entitás beanek	448
	– Távoli interfész	449
	– Otthon interfész	450
	– Elsődleges kulcs	452
	– Enterprise bean osztály	453
	– Segédosztályok	463
	– Telepítési leíró	464
	– Alkalmazáserver specifikus fájl	466
	– Adatbázis DDL	466
	– Teszt kliens	466
12.4.4.	CMP entitás beanek	469
	– Távoli interfész	473
	– Otthon interfész	474
	– Elsődleges kulcs osztály	476
	– Enterprise bean osztály	477
	– Segédosztály	480
	– Telepítési leíró	481
	– Alkalmazáserver specifikus fájlok	484
	– Teszt kliens	484
12.4.5.	Kapcsolatok entitás beanek között	487
	– Kapcsolat multiplicitása	487
	– Lusta betöltés	493
	– Kaszkádolt törlés	493
12.4.6.	Az EJB lekérdező nyelv – EJB-QL	494
12.5.	Üzenetvezérelt bean	497
12.5.1.	Példa üzenetvezérelt beanre	498
	– Üzenetvezérelt bean osztály	498
	– Telepítési leíró	499
	– Alkalmazáserver specifikus fájl	500
	– JMS kliens	501
12.5.2.	Viszontválasz lehetőségek	502
12.6.	Enterprise beanek használata	503
12.6.1.	Lokális kontra távoli interfészek	503
12.6.2.	Tervezési minták	503
	– Session facade	503
	– Érték objektum	504

12.6.3.	CMP kontra BMP perzisztencia	504
12.6.4.	Állapottal rendelkező vagy állapotmentes session bean	504
12.6.5.	Állapottal rendelkező session bean kontra <code>HttpSession</code>	505
12.6.6.	EJB objektumok és EJB otthon objektumok kezelői	505
12.7.	Enterprise bean környezete	506
12.7.1.	EJB referencia	506
12.7.2.	Erőforrások kezelése	508
12.8.	Biztonság	510
12.8.1.	Autentikáció	510
12.8.2.	Jogosultság ellenőrzés	510
	– Alkalmazásszintű jogosultság ellenőrzés	511
	– Deklaratív jogosultság ellenőrzés	513
12.8.3.	Mikor melyik?	515
12.8.4.	Beanek egymás közt biztonságban	515
12.9.	Tranzakciók	517
12.9.1.	Tranzakció kezelés helyei	517
	– Tranzakció kezelés és entitás beanek	518
	– Melyiket válasszuk?	519
12.9.2.	Konténer-vezérelt tranzakciók	519
12.9.3.	Beanvezérelt tranzakciók	522
12.9.4.	Kliens által indított tranzakciók	523
12.9.5.	Állapottal rendelkező session beanek és tranzakciók	523
13. XML	(Fornai Péter)	527
13.1.	Bevezetés az XML-be	528
13.1.1.	Speciális jelek használata	529
13.1.2.	Feldolgozási utasítások	529
13.1.3.	CSS Példa, az XML megjelenítése	530
13.2.	DTD, az XML dokumentumok nyelvtanának megadása	530
13.2.1.	Az elemek definiálása	532
13.2.2.	Entitások definiálása	533
13.2.3.	Attribútumok definiálása	533
13.2.4.	Névtartományok	534
13.3.	XML Sémák	534
13.4.	Az XSL technológia	535
13.4.1.	Bevezetés az XSLT-be	536
13.4.2.	Példa az XSLT használatára	536
13.5.	Java és az XML	539
13.6.	A SAX használata Javából	539
13.6.1.	Példa a SAX használatára	540
13.6.2.	Hibakezelés SAX-szal	542
13.7.	A DOM használata Javából	542
13.7.1.	A DOM ábrázolás	544
13.7.2.	A DOM ábrázolás bejárása	546
13.7.3.	Névtartományok	548
13.7.4.	Párhuzamosság	548
13.8.	Az XSLT használata Javából	548
13.9.	XSLT kiterjesztése Javával	550
13.10.	Az XML jövője	551

14. JavaMail (Balázs Iván József)	553
14.1. Protokollokról és szabványokról	554
14.2. Az API áttekintése	555
14.2.1. A <code>javax.mail</code> csomag	555
14.2.2. A <code>javax.mail.internet</code> csomag	556
14.2.3. A <code>javax.mail.search</code> csomag	557
14.2.4. A <code>javax.mail.event</code> csomag	557
14.3. Példaprogramok	558
14.3.1. Egy üzenet összeállítása, elküldése	559
14.3.2. Egy levél részeinek kiírása	563
14.3.3. Egy kiszolgálón tárolt üzenetek kilistázása	567
14.3.4. Keresés üzenetek között	569
14.3.5. Egy kiszolgálóra érkező üzenetek figyelése	570
15. A Java Multimédia Framework (Csizmazia Balázs)	573
15.1. Java média programozói elemek	574
15.2. Multimédia alapismeretek és a JMF	575
15.3. Néhány hangtárolási formátum	577
15.3.1. AIFF formátum	577
15.3.2. Sun-NeXT audió formátum	577
15.3.3. MIDI formátum	577
15.3.4. MP3 formátum	577
15.3.5. RealAudio formátum	578
15.3.6. Microsoft audió formátum	578
15.3.7. GSM audió formátum	578
15.4. Mozgóképfájl-tárolási formátumok	578
15.4.1. MPEG formátum	578
15.4.2. AVI formátum	579
15.4.3. QuickTime formátum	579
15.4.4. H.261 és H.263 formátum	579
15.5. Kilo és mega a gyakorlatban	579
15.6. A JMF által támogatott formátumok	580
15.7. Állóképtömörítés – a JPEG	583
15.8. Az MPEG-1 tömörítési algoritmus	588
15.8.1. A feldolgozandó videó formátuma és felbontása	589
15.8.2. A kódolási folyamat	589
15.8.3. I-kockák kódolása	590
15.8.4. P-kockák kódolása	590
15.8.5. B-kockák kódolása	591
15.8.6. Az MPEG hangtömörítése	591
15.9. Az MPEG-4 és videótömörítése	591
15.9.1. Az MPEG-4 szabvány főbb részei	591
15.9.2. Az MPEG-4 videótömörítése	592
15.9.3. Audióvizuális objektumok az MPEG-4-ben	593
15.9.4. Az MPEG-4 célkitűzései	593
15.9.5. Az MPEG-4 jelenet-fogalma	594
15.9.6. Az MPEG-4 jelenet-leírása	596
15.9.7. Az MPEG-4 systems és DMIF modellje	597

15.10. A MediaPlayer komponens	599
15.10.1. A MediaPlayer használata	600
15.10.2. A MediaPlayer testreszabása	601
15.10.3. Példaprogram a MediaPlayer használatára	603
15.10.4. Appletok biztonsága és a JMF	604
15.11. A JMF architektúrája	605
15.11.1. A Player interfész	607
15.11.2. A Controller interfész	608
15.11.3. A Clock interfész	608
15.11.4. Egy Player objektum létrehozása	609
15.11.5. A GainControl	609
15.11.6. Kapcsolódó osztályok és interfészek áttekintése	611
15.12. A Controller állapotai	613
15.12.1. Controller események	614
15.12.2. Controller események szinkron kezelése	619
15.12.3. Controller kivételek	621
15.12.4. Controller áttekintése	622
15.13. A Clock interfész és az idő	623
15.14. A Control interfész	626
15.15. Egyszerű médialejátszás a JMF-fel	627
15.16. Időfüggő médiák lejátszásának szinkronizálása	632
15.16.1. Elemi multimédia-szinkronizáció	632
15.16.2. A JMF szinkronizációs mechanizmusa	634
15.16.3. Példaprogram a szinkronizációra	635
15.17. A Processor interfész	638
15.18. Multimédia az Interneten (az RTP protokoll)	639
15.18.1. RTP-kapcsolatok jellemzői	640
15.18.2. RTP kapcsolat kezelése a programban	641
15.19. Multimédia az Interneten (az RTSP protokoll)	643
15.19.1. Az RTSP protokoll felépítése	643
15.19.2. Példa egy RTSP kapcsolatra	643
15.19.3. RTSP és a JMF-lejátszók	645
15.20. A multicast multimédia infrastruktúra	646
15.21. Elosztott multimédia-rendszerek	649
15.22. Multimédiakommunikáció az Interneten	653
15.22.1. Integrált szolgáltatási rendszer	654
15.22.2. Differenciált szolgáltatási rendszer	656
15.22.3. Szolgáltatás-minőségi paraméterek	657
15.23. A Java média stúdió	658
15.23.1. A JMStudio indítása	659
15.23.2. A menüpontok áttekintése	659
15.23.3. Média adatok lejátszása	660
15.23.4. Érkező média adat fájlba mentése	661
15.23.5. Média adat küldése a hálózaton RTP-vel	661
15.24. A multimédia szerverekről	662
15.25. Gyakorlati projekt	663
Irodalomjegyzék	667
Tárgymutató	679