Software Ingeniaritza

3. Gaia: Diseinua

3.3 Maila anitzeko software arkitekturak

A. Goñi, J. Ibáñez, J. Iturrioz, J.A. Vadillo

informatika fakultatea
facultad de informática

eman ta zabal zazu
Universidad del País Vasco
Euskal Herriko Unibertsitatea

Aurkibidea

- Sarrera
- Maila anitzeko softwarearen arkitektura logikoa:
  aurkezpena, negozio logika eta datuak.
- Bi mailako arkitektura fisikoa:
  bezero gizena / zerbitzari mehea.
- Bi mailako arkitektura fisikoa:
  bezero mehea / zerbitzari gizena.
- Hiru (edo gehiago) mailatako arkitektura fisikoa.

Sarrera

- Aplikazio batzuk modu kokurrente, seguru, fidagarri eta eraginkorrean exekutatu behar dira.

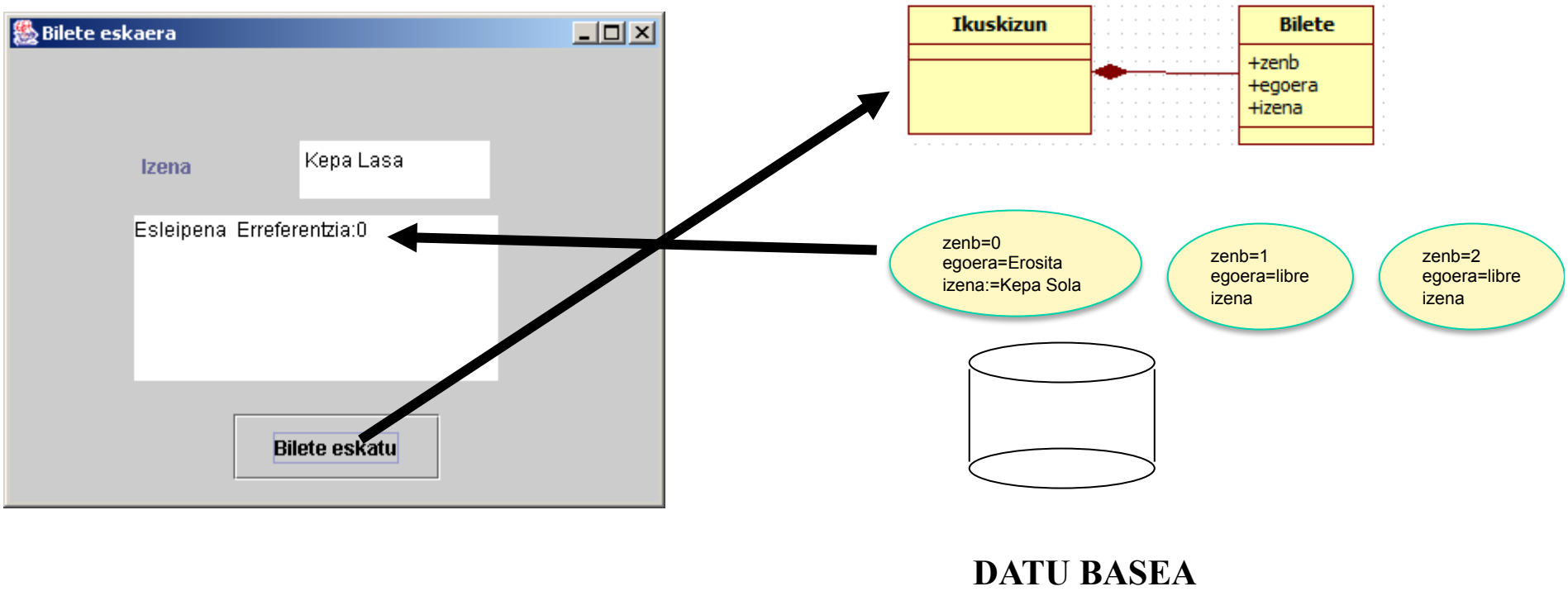
- Adibideak:
  - Zerbitzari zentral batekin konektatuta dauden kutxazain automatikoak.
  - Bidai erreserba bulegoak.
  - Espektakuluetarako sarrerak erosteko terminalak.

- Kasu hauetan erabilgarria da **zerbitzariaren aldean osagaien hedaketa duen arkitektura** (*server side component architecture*) bat izatea.
  - Osagaia: Interfaze multzo ezagun bat implementatzen duen kodea.

Sarrera

Adibidea: Bilete erreserba sistema

Bezeroak bere izena sartzen du, eta sistemak libre dagoen lehendabiziko biletea esleitzen dio, bere egoera “erosita” aldatuz.



Aplikazioaren Kodea

```
public class BileteakEskatu3Maila extends JFrame {
// Nota: EZ DAGO OSORIK !!
   JTextfield jTextField1=new JTextField();
   JLabel jLabel1 = new JLabel("Izena:");
   JButton jButton1 = new JButton("Bilete eskatu");
   ObjectContainer db;
public BileteakEskatu3Maila () {
  db=DB4oManager.getContainer();
}
void jButton1_actionPerformed(ActionEvent e) {
Ikuskizun ikuskizun=Ikuskizun.getIkuskizun("Oscar 2011");
 Bilete b=ikuskizun.getBileteLibre();
 int zenb=b.erosi(jTextField1.getText());
 db.store(b);
}

public static void main (String []arg) {
 BileteakEskatu3Maila b = new BileteakEskatu3Maila ();
    b.setVisible(true);}}
```

```
public class BileteakEskatu3Maila extends JFrame
// Nota: EZ DAGO OSORIK !!
   JTextfield jTextField1=new JTextField();
   JLabel jLabel1 = new JLabel("Izena:");
   JButton jButton1 = new JButton("Bilete eskatu");
   ObjectContainer db;
public BileteakEskatu3Maila () {
  db=DB4oManager.getContainer();
}
void jButton1_actionPerformed(ActionEvent e) {
Ikuskizun ikuskizun=Ikuskizun.getIkuskizun("Oscar 2011");
 Bilete b=ikuskizun.getBileteLibre();
 int zenb=b.erosi(jTextField1.getText());
 jTextArea1.append("Esleipena Erreferentzia: "+zenb+"\n");
 db.store(b);

}

public static void main (String []arg) {
 BileteakEskatu3Maila b = new BileteakEskatu3Maila ();
    b.setVisible(true);}}
```

AURKEZPENA

```
public class BileteakEskatu3Maila extends JFrame {
// Nota: EZ DAGO OSORIK !!
   JTextfield jTextField1=new JTextField();
   JLabel jLabel1 = new JLabel("Izena:");
   JButton jButton1 = new JButton("Bilete eskatu");
   ObjectContainer db;
public BileteakEskatu3Maila () {
   db=DB4oManager.getContainer();
}
void jButton1_actionPerformed(ActionEvent e) {
Ikuskizun ikuskizun=Ikuskizun.getIkuskizun("Oscar 2011");
 Bilete b=ikuskizun.getBileteLibre();
 int zenb=b.erosi(jTextField1.getText());
  db.store(b);

}

public static void main (String []arg) {
 BileteakEskatu3Maila b = new BileteakEskatu3Maila ();
    b.setVisible(true);}}
```

DATU

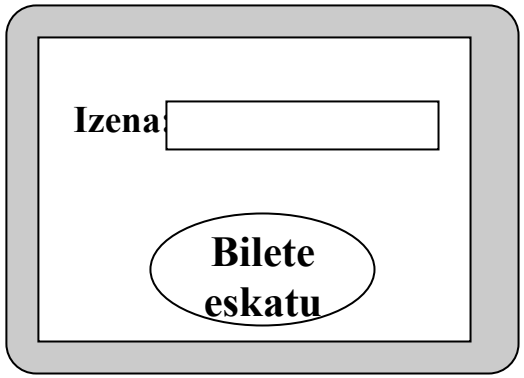
ATZIPENA

```
public class BileteakEskatu3Maila extends JFrame {
// Nota: EZ DAGO OSORIK !!
   JTextfield jTextField1=new JTextField();
   JLabel jLabel1 = new JLabel("Izena:");
   JButton jButton1 = new JButton("Bilete eskatu");
   ObjectContainer db;
public BileteakEskatu3Maila () {
  db=DB4oManager.getContainer();
}
void jButton1_actionPerformed(ActionEvent e) {
Ikuskizun ikuskizun=Ikuskizun.getIkuskizun("Oscar 2011");
 Bilete b=ikuskizun.getBileteLibre();
 int zenb=b.erosi(jTextField1.getText());
 db.store(b);

}

public static void main (String []arg) {
 BileteakEskatu3Maila b = new BileteakEskatu3Maila ();
    b.setVisible(true);}}
```

NEGOZIOAREN LOGIKA

3 Mailako software arkitektura logikoa

Aurkezpen maila



Erabiltzailearen interfaze grafikoa:

Frame edo Applet

Negozioaren logikaren maila

```
public class Ikuskizun
   implements TxartelenKudeatzailea{
  ...
  public int getBileteLibre (){
   ...
  }
```

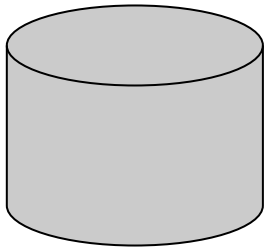
Negozioaren eragiketa propioak dituzten klaseak

- hasieratu
- getBileteLibre
- erosi

Hemen negozioaren arauak aplikatu daitezke
(erositako 10 txartelengatik bat oparitzen da, etab...)

Datuen maila

```
...
...
```



Datu-Basea

3 mailako software arkitektura logikoa

- **Aurkezpen maila**
  - Erabiltzaile interfazeak eta interakzioak inplementatzen dituzten osagaiak.

- **Negozioaren logikaren maila**
  - Negozioaren arazoak ebazten dituzten osagaiak.
  - Negozioaren arau propioak inplementatzen dituzten zerbitzu eta eragiketaz osatua dago.

- **Datuen maila**
  - Persistentzia lortzeko negozioaren logika mailak erabilita.
  - Datu-Base bat (edo gehiago) da.

3 mailako software arkitektura logikoa

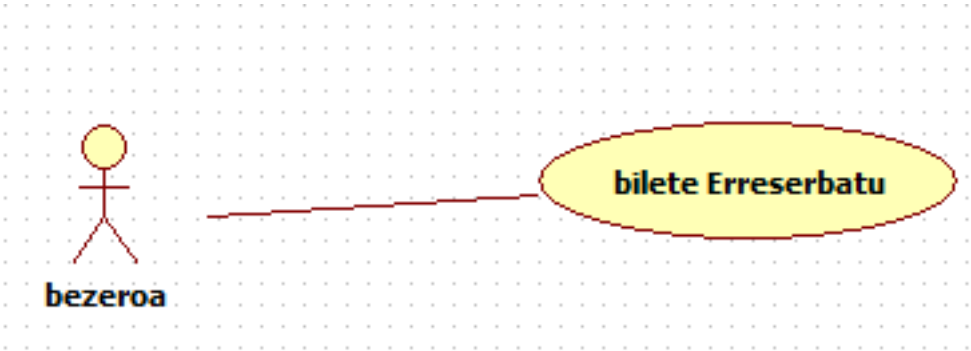
- Abantaila: Maila bat besteetatik isolatzea
  - Maila batean aldaketak egin daitezke, beste mailetan aldaketa gutxi eginik.

- Maila anitzeko softwarearen arkitektura logikoak

laguntzen du

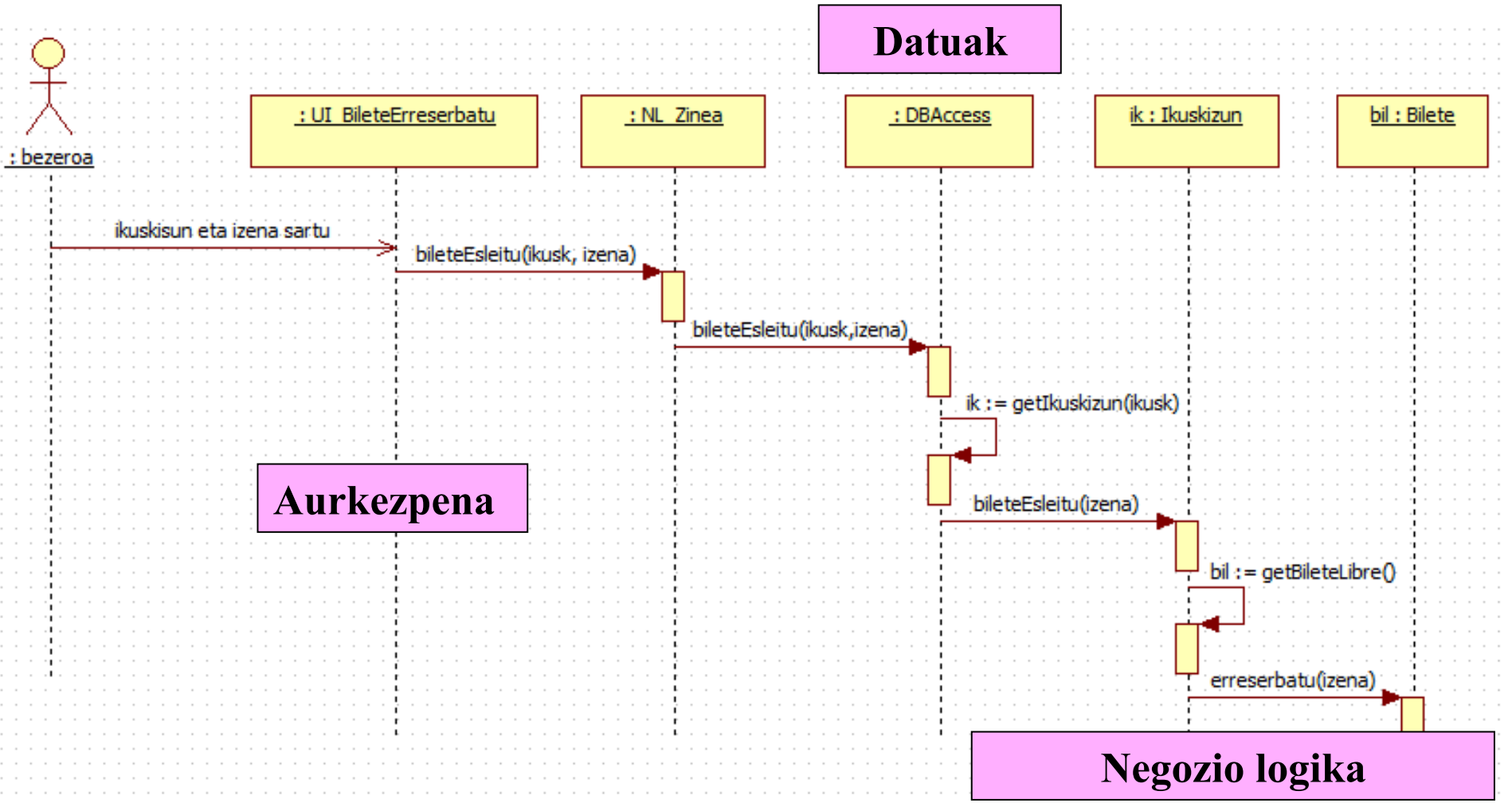
softwarearen hedagarritasuna eta berrerabilpena

Erabilpen kasuen analisi eta disenua maila anitzeko filosofia jarraitzen dute



Gertaeren fluxua: BILETE ERRESERBATU
****
Erabiltzailea bere izena ematen du
Sistemak bileteak libre ahal dauden begiratzen du
Bileteak badaude esleitzen zaio, izena gordetzen da eta bilete zenbakia itzultzen du

Sekuentzia diagramak ere….



Hiru mailako software arkitektura logikoa

- **Aurkezpen maila**
  - Klaseen bidez eraikitzen da (Frame-ak edota Applet-ak).
- **Negozioaren logikaren maila**
  - Negozioaren logikaren zerbitzu edota eragiketak dituzten Java klaseen bidez eraikitzen da.
- **Datuen maila**
  - Datu-Basea da.

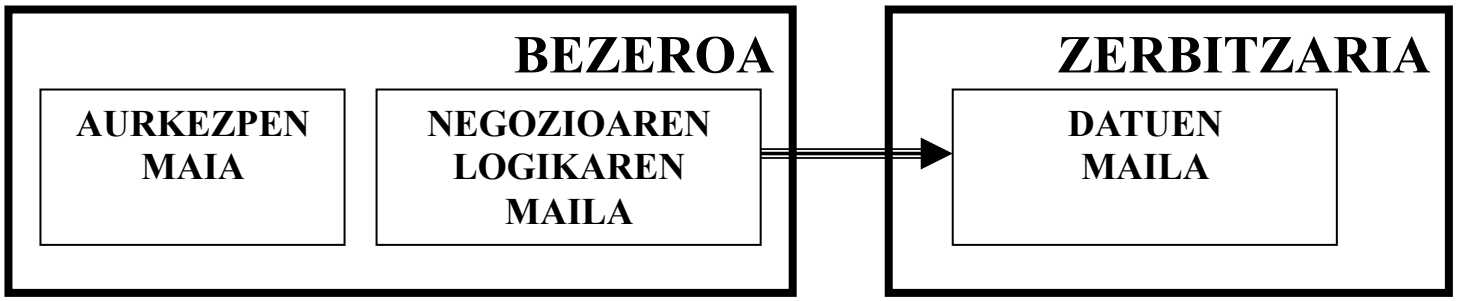
Hiru mailako software arkitektura fisikoa

- 3 Mailako banaketa logikoa maila fisikoetan:
  - 2 mailako arkitektura
    - Bi maila, nodo batean biltzen dira. Bi aukera:
      - negozioaren logika eta aurkezpena biltzen da, edo
      - negozioaren logikaren zati bat datuekin biltzen da.
  - 3 mailako (edo gehiagoko) arkitektura
    - maila bakoitza, gutxienez, nodo ezberdin batean.

Bi mailatako arkitektura fisikoa:

bezero gizena / zerbitzari mehea

- Aurkezpen maila eta negozioaren logikaren maila nodo batean biltzen dira.
- Beste nodoan datuen maila gelditzen da.



- Bezeroa eta Zerbitzariaren arteko komunikazioa db4o bidez.
- APIak behar dira (db4o driver-ak).
- Bezero guztietan DBaren DRIVER-ak instalatu behar dira.

BEZERO

Aurkezpena

```
public class BileteakEskatu2MailaBezeroGizena extends JFrame {
BileteakEskatu2MailaBezeroGizena bileteKud;
void jButton1_actionPerformed(ActionEvent e) {
  int ema = bileteKud.getBilete(jTextField1.getText()).getZenb();
   if (ema<0) jTextArea1.append("Errorea bilete esleitzerakoan");
   else jTextArea1.append("Esleipena. Erref: "+ema+"\n");} }
```

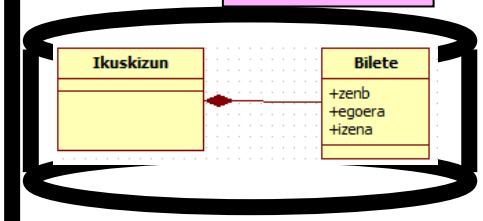
Negozio Logika

```
public class BileteKudDB
      implements BileteKud2MailaBezeroGizena {
   public BileteKudDB () {
   db=DB4oManager.getContainer();
}
  public Bilete getBilete(String izena) {
    Ikuskizun ikuskizun=Ikuskizun.getIkuskizun(izena);
    Bilete b=ikuskizun.getBileteLibre();
    int zenb=b.erosi(izena);
    db.store(b);
 return b;}
```

KLASEA INSTALATU com.db4o.DB4oManager

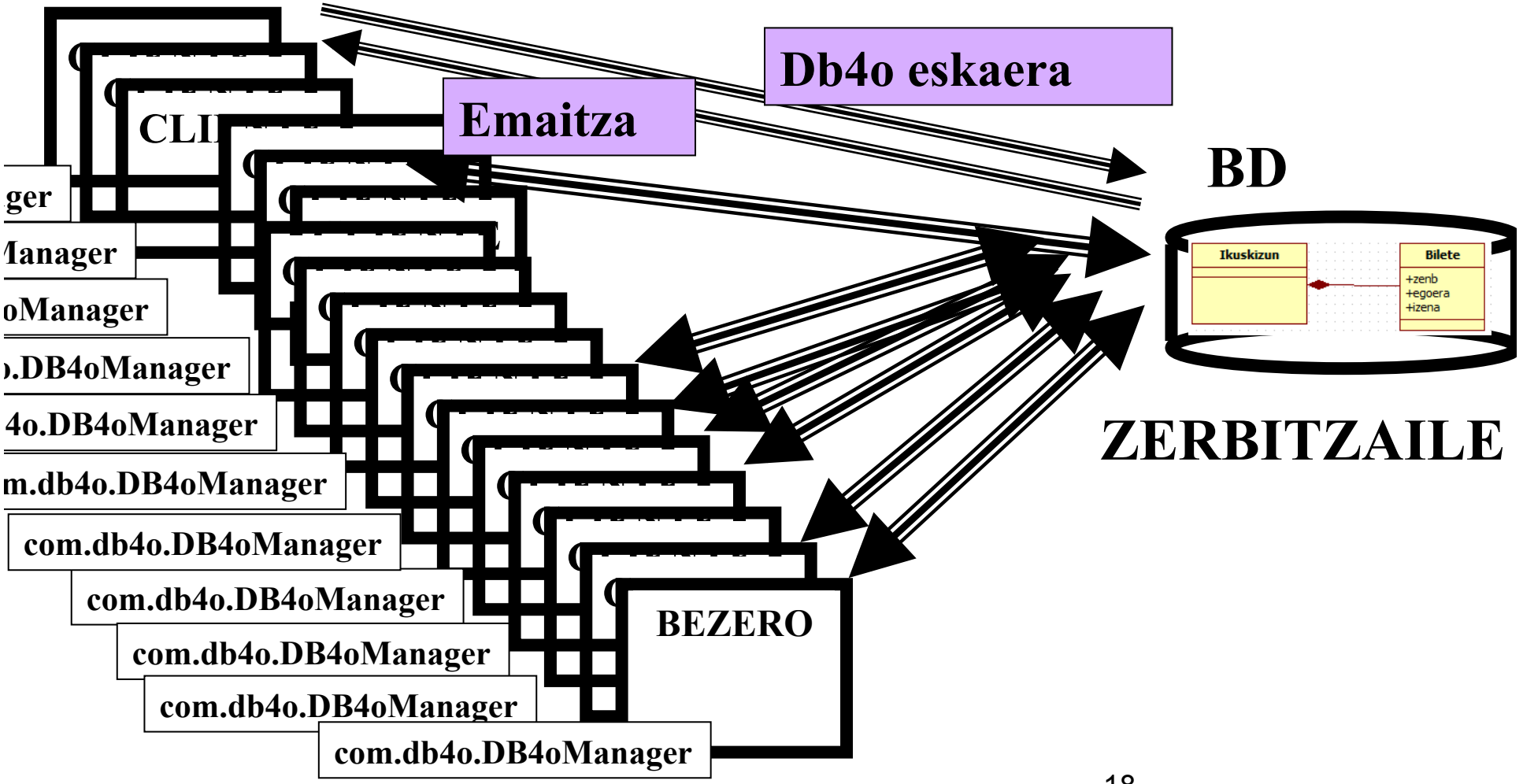
DB

Datuak



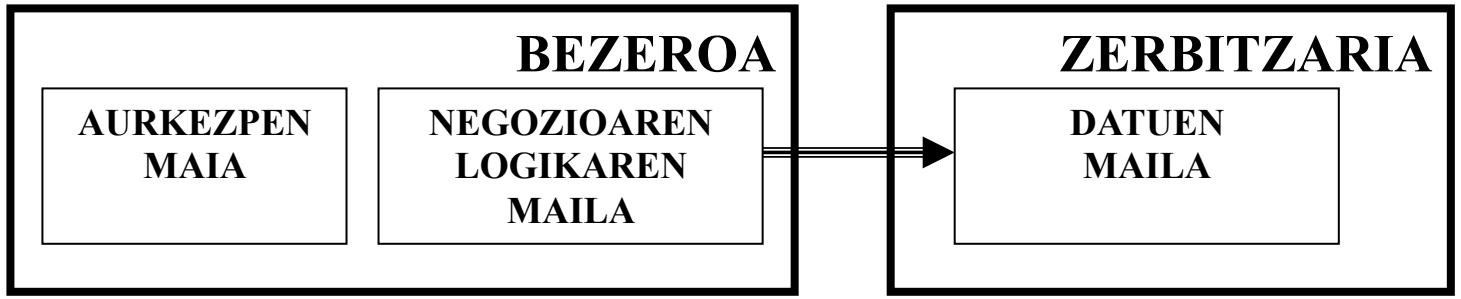
ZERBITZAILE

Bezero bakoitzak konexio bat DB-arekin



Bi mailako arkitektura fisikoa:

bezero gizena / zerbitzari mehea

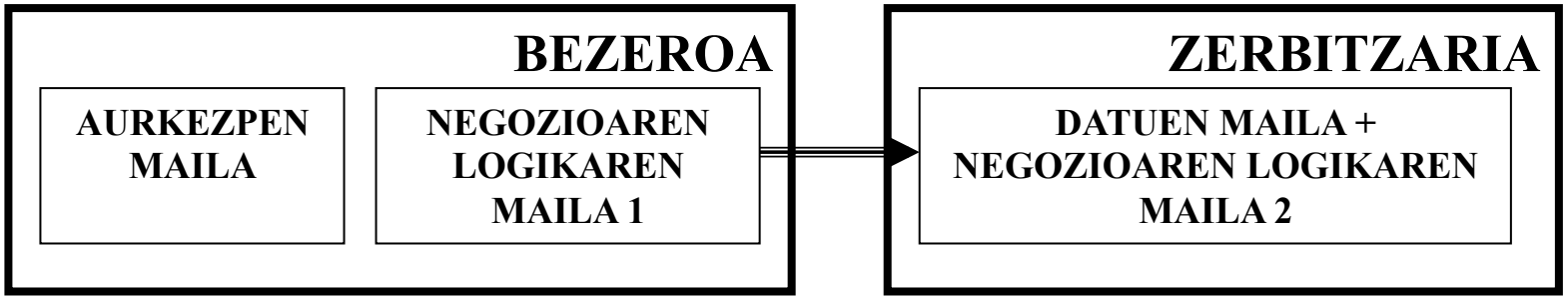


- Aplikazioaren hedaketa garestia da:
  driverrak instalatu eta bezero guztiak konfiguratu behar dira.
- DBKSa aldatzeak bezero guztietan berrinstalatzea suposatzen du.
- DBaren eskema aldatzeak bezero guztiei eragin diezaieke.
- Negozioaren logika aldatzeak birkonpilatzea eta bezero guztietan hedatzea erakartzen du.
- DBarekin konexioa garestia da. Bezero bakoitzak konexio bat du.
- Sarea gainkargatu daiteke, sententzi bakoitzak sarea erabiltzen baitu.

Bi mailatako arkitektura fisikoa:

bezero mehea / zerbitzari gizena

- Negozioaren logikaren zati bat datuen mailarekin konbinatzen da.



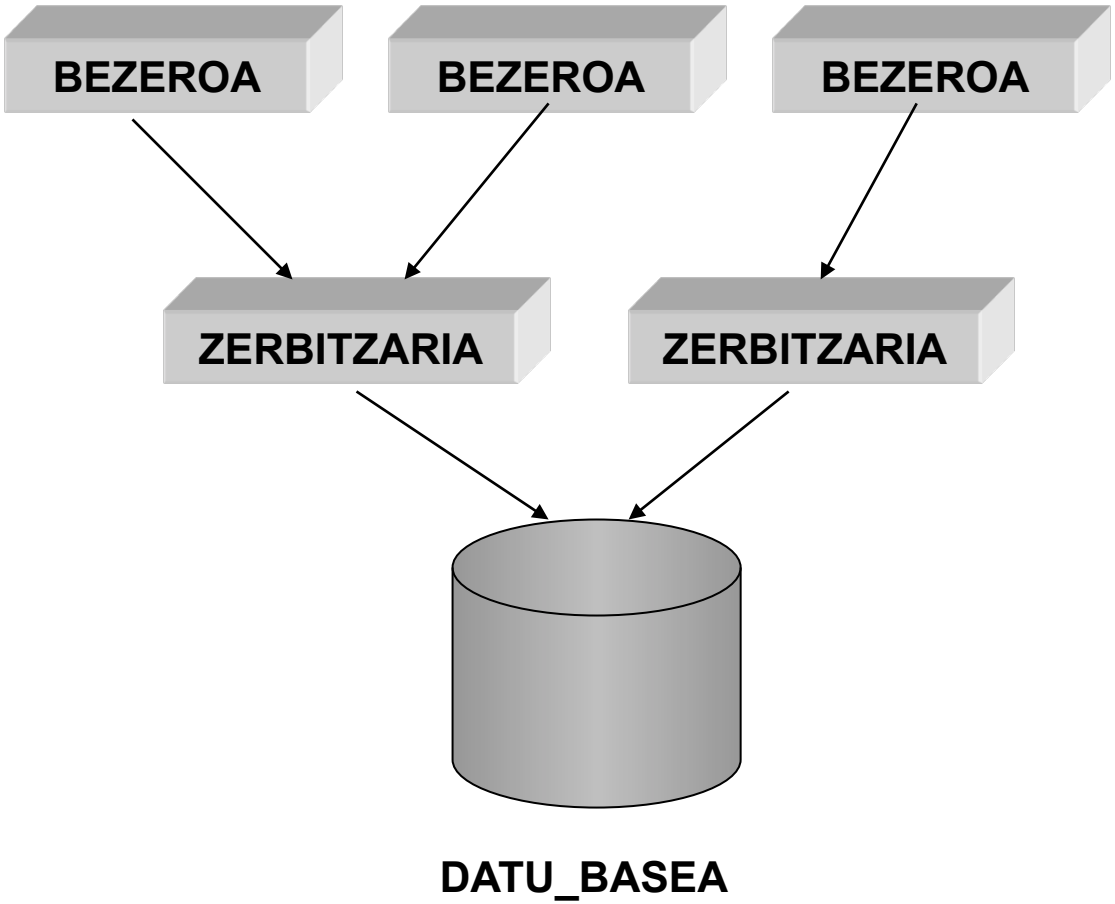
• DBan biltegiratutako prozedurak (stored procedures) erabiltzen dira. Biltegiratutako prozedura batek SQL sententzia multzo bat egikaritzeko balio du.

• Bezeroa eta Zerbitzariaren arteko komunikazioa:
SQLz eta biltegiratutako prozedurekin.

• APIak behar dira (adibidez JDBC edota ODBC).

• Bezero guztietan DBaren DRIVER-ak instalatu behar dira.

Hiru mailako arkitektura fisikoa



AURKEZPEN MAILA

NEGOZIOAREN LOGIKAREN MAILA

(Aplikazio zerbitzailetan)

DATUEN MAILA

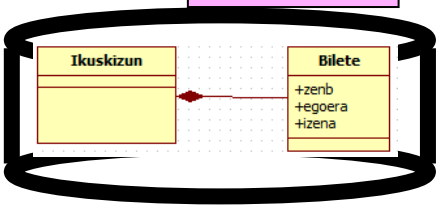
BEZERO

```
public class BileteEskatu extends JFrame {
BileteKud bileteKud;
void jButton1_actionPerformed(ActionEvent e) {
   int res = bileteKud.getBilete(jTextField1.getText()).getNum();
    if (res<0) jTextArea1.append(" Errorea bilete esleitzerakoan ");
    else jTextArea1.append(" Esleipena. \nErref: "+ema+"\n ");} }
```

Aurkezpena

DB

Datuak



DATU ZERBITZAILE

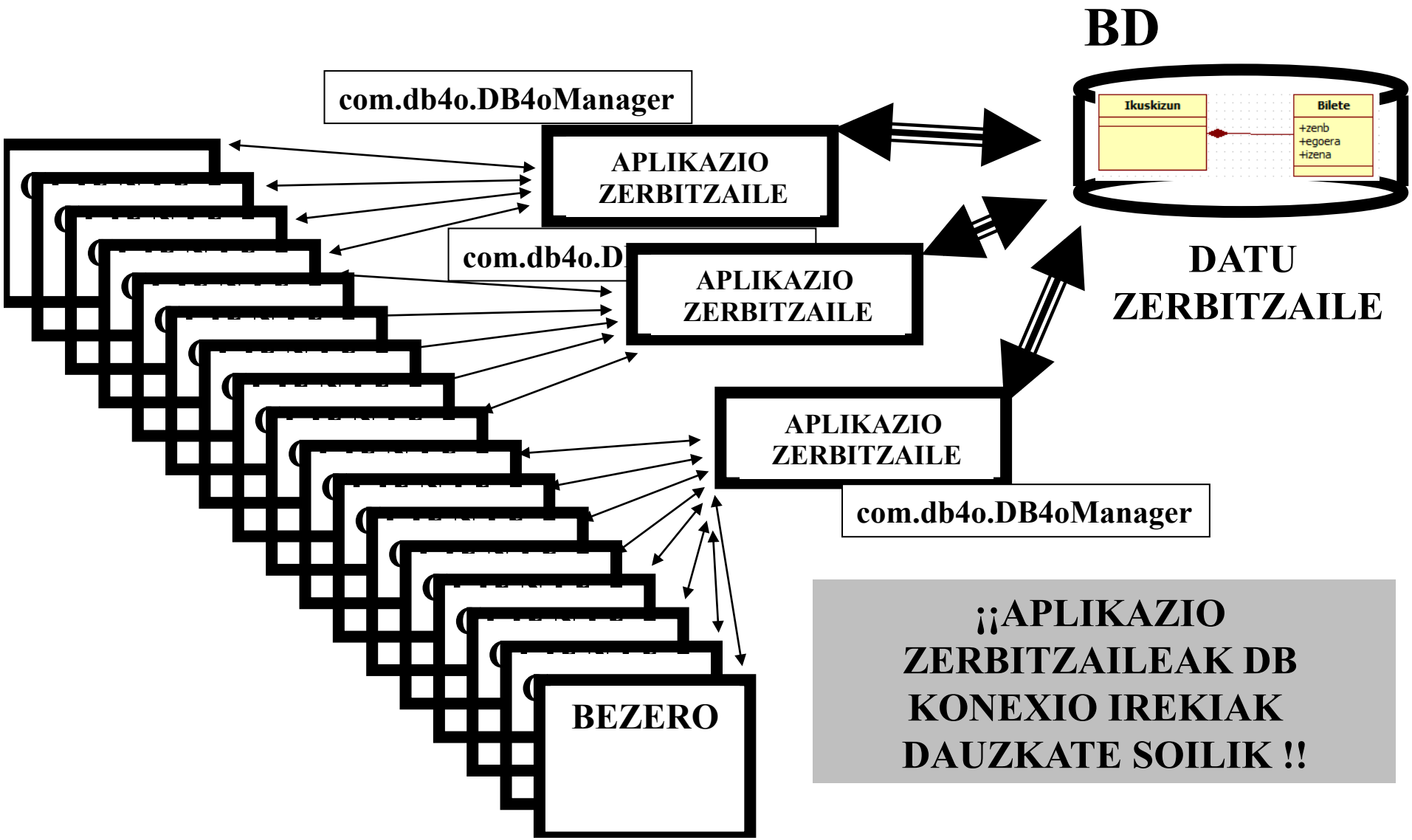
APLIKAZIO ZERBITZARIA

KLASEA INSTALATU com.db4o.DB4o Manager

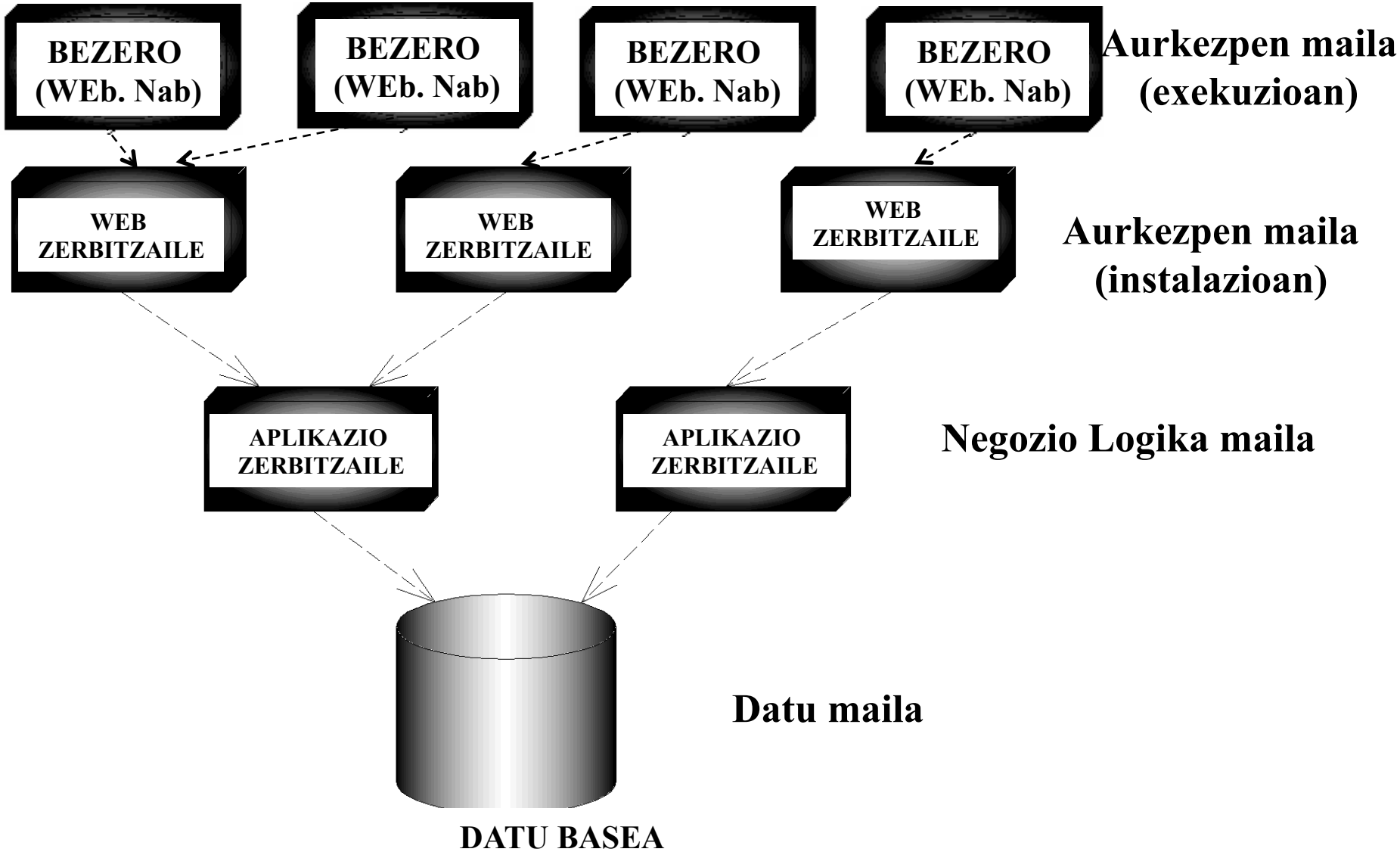
```
public class BileteKudDB  implements BileteKud2MailaBezeroGizena
{ public BileteKudDB () {
    db=DB4oManager.getContainer();
}
   public Bilete getBilete(String izena)
    {Ikuskizun ikuskizun=Ikuskizun.getIkuskizun(izena);
     Bilete b=ikuskizun.getBileteLibre();
     if (b!=null) { b.erosi(izena); db.store(b); return b;}
       else return new Bilete(-1,"");; } // Ez zegoen librerik}}}
```

Negozio Logika

Hiru mailako arkitektura fisikoa



Web aplikazioak maila gehiago eskeintzen dizkigute



BEZEROA

Web Nabigatzaile

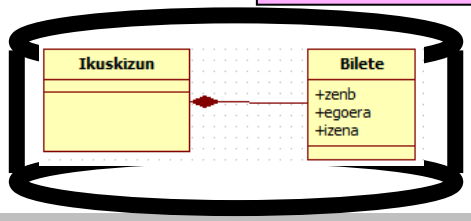
Aurkezpena exekutatu

WEB SERBITZAILE

HTML + JSP + Bean

Aurkezpena instalatu

DB Datuak



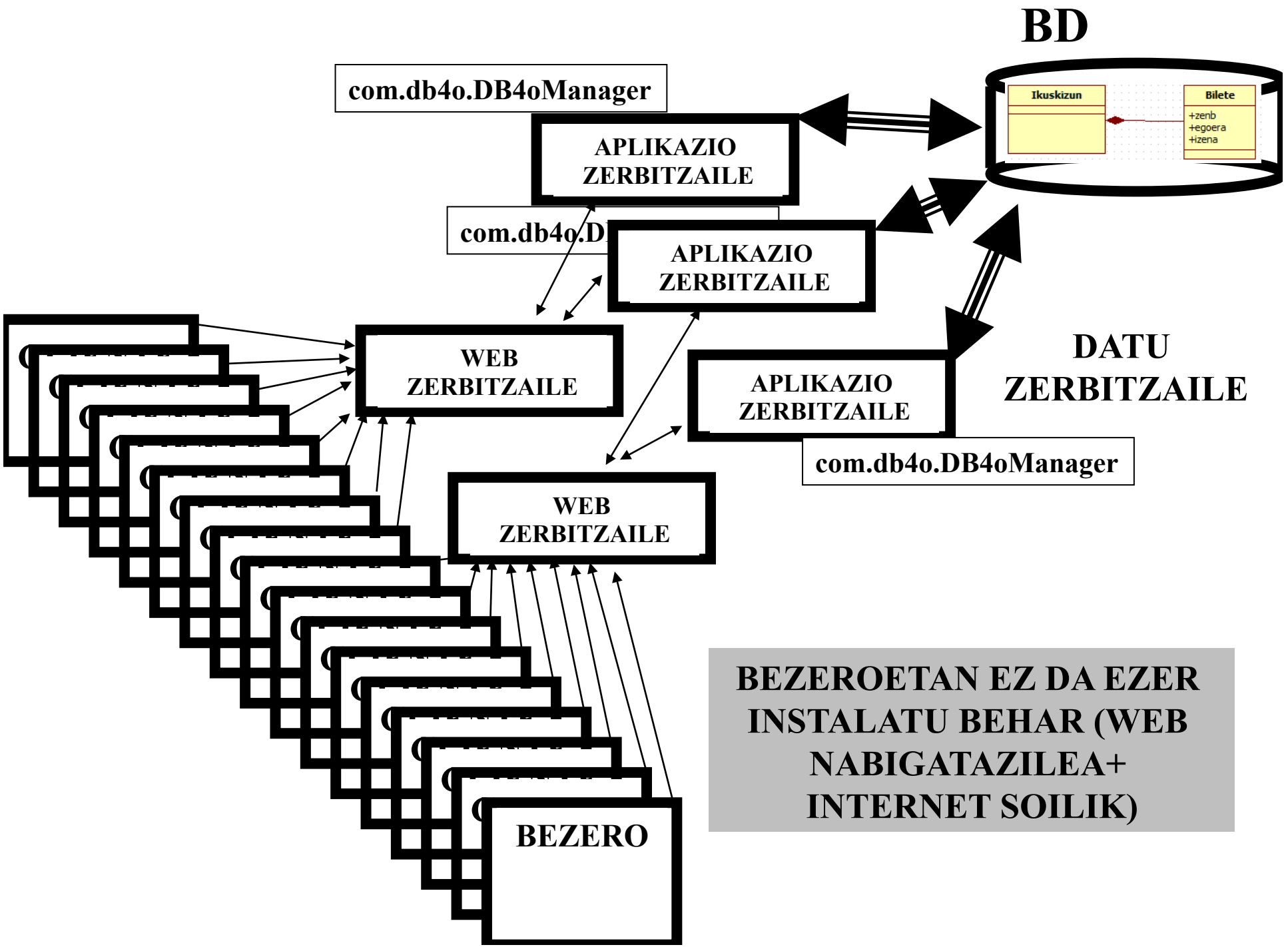
DATU ZERBITZAILE

APLIKAZIO ZERBITZARIA

KLASEA INSTALATU com.db4o.DB4o Manager

```
public class BiletеKudDB  implements BiletеKud2MailaBezeroGizena
{ public BiletеKudDB () {
    db=DB4oManager.getContainer();
}
   public Bilete getBilete(String izena)
    {Ikuskizun ikuskizun=Ikuskizun.getIkuskizun(izena);
     Bilete b=ikuskizun.getBiletеLibre();
     if (b!=null) { b.erosi(izena); db.store(b); return b;}
       else return new Bilete(-1,"");; } // Ez zegoen librerik}}}
```

Negozio Logika


BD
Ikuskizun
Bilete
+zenb
+egoera
+izena
com.db4o.DB4oManager
APLIKAZIO ZERBITZAILE
com.db4o.D
APLIKAZIO ZERBITZAILE
WEB ZERBITZAILE
APLIKAZIO ZERBITZAILE
DATU ZERBITZAILE
com.db4o.DB4oManager
WEB ZERBITZAILE
BEZERO
BEZEROETAN EZ DA EZER INSTALATU BEHAR (WEB NABIGATAZILEA+ INTERNET SOILIK)

Hiru mailako arkitektura fisikoa

- DBaren driverrak soilik negozioaren logika dagoen nodoetan (nodo zerbitzarietan) instalatu behar da.
- DBKSa edo DBaren eskema aldatzeak ez du suposatzen bezero guztietan berrinstalatzea. Soilik negozioaren logikakoak.
- Negozioaren logika aldatzeak ez du eragiten berkonpilatzea eta bezero guztietan hedatzea.
- DBarekin konexioa ez da hain garestia. Bezeroek ez dituzte DBarekin konexioak egiten, soilik negozioaren logika duten zerbitzariek.

Orokorrean hobetzen da
eraginkortasunean, **hedagarritasunean** eta **mantentzean**

3 mailatako arkitektura fisikoa

- Badago aplikazioak eraikitzeko teknologia, osagai eta objektu banatuen filosofia jarraituz (server-side components):
  - Enterprise JavaBeans (EJB):
    - Java 2 Enterprise Edition (J2EE) plataformarako osagaien arkitektura.
    - Sun Microsystems-ek definituta.
    - Aurkezpen maia gehiago zatitu daiteke Java Applet-ak, Servlet-ak edota JSP-ak erabiliz.
  - CORBA:
    - Object Request Broker (ORB) bidez objektu banatuen artean komunikaziorako arkitektura.
    - OMGk definitutako estandarra.
  - DCOM/COM+ eta .NET plataforma
    - Microsoft-ek garatutako tekonolgia baliokidea.

3 mailatako arkitektura fisikoa

- Baina Java-ren teknologia “erraza”rekin ere lortu daiteke:
  - **Aurkezpen maila** definitzeko:
    - Interfaze grafikoen eraikuntza Java-n (AWT eta SWING).
  - **Negozioaren logika** eta **datuen maila**ren arteko komunikazioa lortzeko:
    - OO datubase konexioa (db4o).
    - DB erlazionalekin konexioa (JDBC).
  - **Negozioaren logika**ren maila definitzeko eta **aurkezpen maila**tik deitzeko:
    - Objektu banatuekin konputazioa (RMI).

Galderak

- Zenbat mailatan gomendatzen da aplikazio bat zatitzea? Zeintzuk dira zatitze honen abantailak?
- Zer dira maila logikoak eta maila fisikoak?
- Zeintzuk dira 2 mailatako sistemen murriztapenak (bezero/zerbitzaria)?
- Sekuentzia diagrama bat emanda, zer ipiniko zenuke maila bakoitzean?
- Zer aldaketa egin beharko litzateke..
  - db4o=> DBKS Access
  - Negozio erregela berria: Pertsona batek ezin ditu 6 sarrera baino gehiago erosi
  - Emaitza beste leihoan agertzea