



J-Spring

16 april 2008 Spant! - Bussum



## Temporale data en Hibernate

Johan Eikelboom



I T P R O F E S S I O N A L S



## Onderwerpen

- Begrippen.
  - Transactietijd, geldige tijd en bitemporale data
  - Tijdeenheden, Intervallen, tijdlijnen, (un)packed
- Begrijpen.
  - een HRM voorbeeld .
- Wat is er zo moeilijk aan?
  - Waarom uw project hierdoor kan mislukken.
  - Wat er mist in relationele databases.
- Hoe je met Hibernate en wat patterns kunt doen wat relationele databases niet kunnen.



## Geldige tijd, transactietijd en bitemporale data

- **Real-time – wat is het geval?**

Fred werkt **nu** voor de afdeling verkoop.

- ***Geldige tijd – wanneer is het gebeurd?***

Fred werkt sinds **1-1-2003** voor afdeling verkoop.

- **Transactietijd – wanneer hebben we wat vastgelegd?**

Ons systeem heeft op **1-1-2004** vastgelegd dat fred op de afdeling inkoop werkt.

- **Bitemporale data**

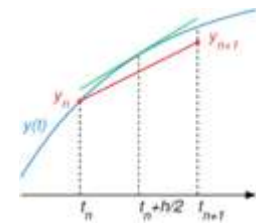
Ons systeem heeft op **1-1-2004** vastgelegd dat fred sinds **1-1-2003** op de afdeling inkoop werkt, en dit op **15-1-2004** gecorrigeerd omdat hij bij verkoop bleek te werken.



## Tijdeenheid



- Natuurkunde: continu variabele tijd.
- Transactietijd: fijnkorrelig tijdstempel.
- Geldige tijd: discrete tijdeenheid.
  - Datum, uur, seconde, jaar, maand week.
  - Begin en einde van de tijd BOT en EOT.
  - Eindig aantal eenheden.
  - Opvolger en voorganger.
  - Lineaire ordening:  $t_1 < t_2$  en  $t_2 < t_3 \rightarrow t_1 < t_3$ .
  - Bot, 1,2,3,4,5,6,7,8.. EOT.





## Sinds, gedurende en interval



- Sinds-vorm: sinds datum 5.
  - Fred werkt voor verkoop sinds 5.
- Gedurende-vorm: van 5 tot en met 7.
  - Fred werkte voor verkoop gedurende het interval  $[5,7]$
- Gesloten interval gebruik voor geldige tijd.
  - $[5,7]$   $\{5,6,7\}$
- Half-open interval voor transactietijd.
  - $[5,8)$   $\{5,6,7\}$



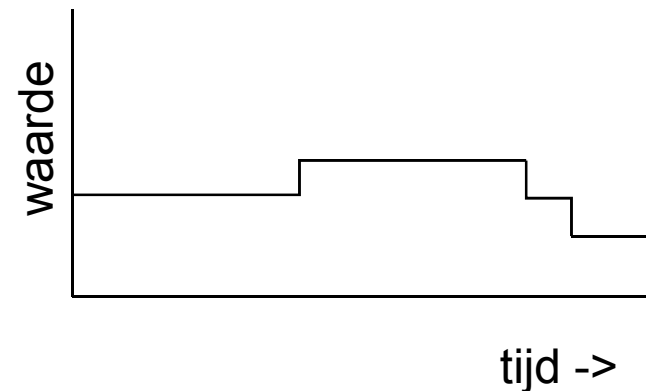
## Tijdslijn

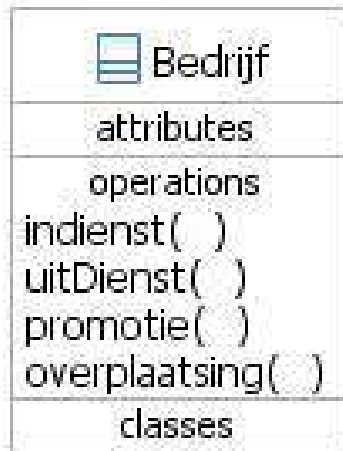


- Een tijdslijn is een aantal records waarin we het verloop in de geldige tijd van een gegeven weergeven.
- Op ieder moment één waarde.
- Waarde functioneel afhankelijk van tijd.

### WERKTALS

BADGENR	GEDURENDE	FNCCODE
101	[2:3]	ADMN
101	[4:7]	TSTR





medewerkers

functiehouders

werktBij

werktAls

afdeling

functie

[0..\*]

[0..\*]



## HRM systeem voorbeeld



# Data HRM systeem

## Functie

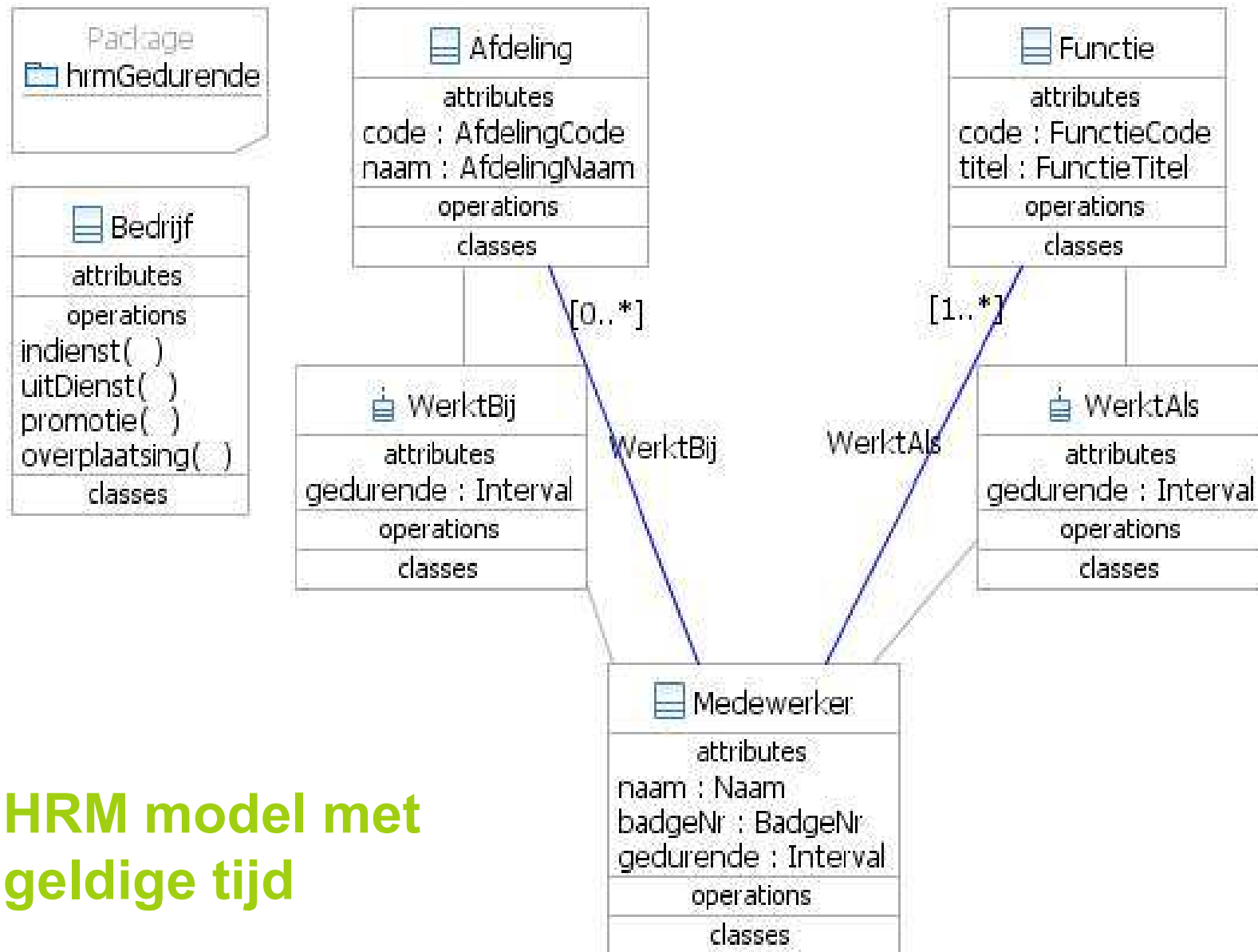
FNCCODE	TITEL
JPRG	Java programmeur
VKPR	Verkoper
INKP	Inkoper
TSTR	Tester
ADMN	Administrateur

## Afdeling

AFDCODE	AFDELINGNAAM
OS	Ontwikkelstraat
QA	Kwaliteitscontrole
BK	Boekhouding

## Medewerker

BADGENR	NAAM	FNCCODE	AFDCODE
101	Fred	TSTR	QA
102	Gerda	JPRG	OS
103	Bart	ADMN	BK



## HRM model met geldige tijd



## Medewerker Fred met tijdlijnen

- Methode: gedurende met EOT als sinds.
- Alternatief: aparte attributen voor sinds.

### MEDEWERKER

BADGENR	GEDURENDE	NAAM
101	[2:7]	Fred
102	[4:EOT]	Gerda

### WERKTALS

BADGENR	GEDURENDE	FNCCODE
101	[2:3]	ADMN
101	[4:7]	TSTR

### WERKTBIJ

BADGENR	GEDURENDE	AFDCODE
101	[2:5]	BK
101	[6:7]	QA

Kolommen zijn nu aparte tijdlijnen geworden.



## Problemen in tijdlijnen.

- De volgende dingen kunnen fout gaan in tijdlijnen:
  - !! Overlappende intervallen met dezelfde waarde.
  - !! Overlappende intervallen met conflicterende gegevens.
  - !! Aansluitende intervallen met dezelfde waarde.

### WERKTALS (conflicterend!)

BADGENR	GEDURENDE	FNCCODE
101	[2:3]	ADMN
<b>101</b>	<b>[2:6]</b>	<b>ADMN</b>
101	[5:6]	TSTR
<b>101</b>	<b>[7:7]</b>	<b>TSTR</b>



## Packed en Unpacked

- Packed formaat
  - Geen overlap tussen intervallen
  - Aansluitende intervallen met zelfde waarde samengenomen
- Unpacked
  - Intervallen met lengte één.
  - Niet overlappend

WERKTALS (packed op gedurende)

BADGENR	GEDURENDE	FNCCODE
101	[2:3]	ADMN
101	[4:7]	TSTR

WERKTALS (unpacked op gedurende)

BADGENR	GEDURENDE	FNCCODE
101	[2:2]	ADMN
101	[3:3]	ADMN
101	[4:4]	TSTR
101	[5:5]	TSTR
101	[6:6]	TSTR
101	[7:7]	TSTR



## Wat is er moeilijk aan?

- Intervallen  
logisch één attribuut,  
fysiek twee attributen (begin en einddatum)
- Wat is eigenlijk de primaire sleutel?  
badgenr?, gedurende? , begindatum?
- En hoe zit het met verwijssleutels?
- Hoe moet het met overlappende en aansluitende intervallen?
- Hoe combineer je de gegevens van de twee tijdlijnen?



## De gevolgen voor uw project



- In functiepunten is het maar een fractie meer als het een tabel is met toch al veel attributen.
- Een update op een tijdlijn om een willekeurige periode te muteren kost 10-15 SQL statements.
- En dan hebben we alleen nog maar de mutatie uitgeschreven, nog geen temporale businesslogica.
- En er is ook nog een transactietijd dimensie.
- Adding manpower to a late project .. Fred Brooks.



## Temporale data in het relationele model

- Het is mogelijk om u\_operaties te definiëren om temporale data te behandelen in het relationele model. (C.J. Date)
- Daarmee hebben we nog geen temporale database.
- Maar deze u\_operaties kunnen wel op een eenvoudige manier als patterns in het domeinmodel worden opgenomen.
- En zo kunnen we de complexiteit reduceren.



## Using (gedurende) join (u\_join)

Using (gedurende)

WERKTBIJ

BADGENR	GEDURENDE	AFDCODE
101	[2:5]	BK
101	[6:7]	QA

JOIN

WERKTALS

BADGENR	GEDURENDE	FNCCODE
101	[2:3]	ADMN
101	[4:7]	TSTR

=

ARBEIDSHISTORIE

BADGENR	GEDURENDE	AFDCODE	FNCCODE
101	[2:3]	BK	ADMN
101	[4:5]	BK	TSTR
101	[6:7]	QA	TSTR



Met unpacked data werkt een gewone join wel.

WERKTBIJ

BADGENR	GEDURENDE	AFDCODE
101	[2:2]	BK
101	[3:3]	BK
101	[4:4]	BK
101	[5:5]	BK
101	[6:6]	QA
101	[7:7]	QA

join

WERKTALS

BADGENR	GEDURENDE	FNCCODE
101	[2:2]	ADMN
101	[3:3]	ADMN
101	[4:4]	TSTR
101	[5:5]	TSTR
101	[6:6]	TSTR
101	[7:7]	TSTR

=

Arbeidshistorie

BADGENR	GEDURENDE	AFDCODE	FNCCODE
101	[2:2]	BK	ADMN
101	[3:3]	BK	ADMN
101	[4:4]	BK	TSTR
101	[5:5]	BK	TSTR
101	[6:6]	QA	TSTR
101	[7:7]	QA	TSTR



## Samenstelling van u\_join

using (gedurende) werktbij join werktals

=

1. Unpack tabel werktbij op gedurende
  2. Unpack tabel werktals op gedurende
  3. Join deze twee
  4. Pack het resultaat op gedurende
- En het goede nieuws: dit kan ook zonder de data fysiek te unpacken.



## Projecties

### ARBEIDSHISTORIE

BADGENR	GEDURENDE	AFDCODE	FNCCODE
101	[2:3]	BK	ADMN
101	[4:5]	BK	TSTR
101	[6:7]	QA	TSTR

u\_project van afdcode

u\_project van fnccode

### WERKTBIJ

BADGENR	GEDURENDE	AFDCODE
101	[2:5]	BK
101	[6:7]	QA

### WERKTALS

BADGENR	GEDURENDE	FNCCODE
101	[2:3]	ADMN
101	[4:7]	TSTR



## Andere u\_operators

- Temporale primary key en temporale foreign key.
- Temporale delete en update operaties.  
(terugwerkende kracht mutaties)

WERKTBIJ

BADGENR	GEDURENDE	AFDCODE
101	[2:5]	BK
101	[6:7]	QA

+

update

BADGENR	GEDURENDE	AFDCODE
101	[5:6]	OS

=

WERKTBIJ

BADGENR	GEDURENDE	AFDCODE
101	[2:4]	BK
101	[5:6]	OS
101	[7:7]	QA



## Zesde normaalvorm

- We hadden ook arbeidshistorie als entiteit kunnen kiezen in de database.
- Door projectie kunnen we dan werktbij en werktals weer terugkrijgen.
- Dit is te zien als een zesde normaalvorm.
- Met dezelfde criteria als andere normalisaties, redundantie versus sneller retrieval als join niet nodig is.



## Hoe dit te doen met Hibernate

1. Definieer een tijdeenheid.
2. Implementeer deze. Vermijd `java.util.Date` en `java.util.Calendar`.
3. Maak een interval met begindatum einddatum.
4. Definieer operaties op de intervallen:  
voor, raakt, overlapt, start, eindigt, omvat, gelijk



## Hoe dit te doen met Hibernate

- Maak mappings voor de associaties als `orderedSets`.
- Het wordt geordend op de intervallen:  
`compare = 0` als intervallen gelijk  
`a < b` als `a` before `b` of `a` raakt `b`  
`b > a` als `b` before of `b` raakt `a`  
anders: `krak` -> `NotPackedException`



## Tijddlijnen

- SortedSets als tijddlijn collectie.
- Elementen moeten vergeleken kunnen worden op interval (compare) en op inhoud (gelijk).
- Er moet een methode zijn om het interval te splitsen (copie of clone) voor mutaties.



## Tijdlijn methodes

- update (TijdlijnElement e)  
voegt e toe aan de collectie. Knijpt overlappende intervallen bij en voegt samen als het naast een element met zelfde waarde komt. Tijdlijn blijft packed.
- delete (Interval i)  
Knijpt overlappende stukken af, of splitst intervallen of verwijdert elementen.
- project, join, erase clip.
- Implementatie: subklasse of wrapper.



## Codevoorbeeld Test interval operatie voor()

```
@Test
public void testVoor() {
    assertTrue("" + i1, i1.voor(interval));
    assertFalse("" + i2, i2.voor(interval));
    assertFalse("" + i3, i3.voor(interval));
    assertFalse("" + i4, i4.voor(interval));
    assertFalse("" + i5, i5.voor(interval));
    assertFalse("" + i6, i6.voor(interval));
    assertFalse("" + i7, i7.voor(interval));
    assertFalse("" + i8, i8.voor(interval));
    assertFalse("" + i9, i9.voor(interval));
    assertFalse("" + i10, i10.voor(interval));
    assertFalse("" + i11, i11.voor(interval));
    assertFalse("" + i12, i12.voor(interval));
    assertFalse("" + i13, i13.voor(interval));
}
```



## Tijdlijn TestFixture met naam van de maand als waarde.

```
protected void setUp() throws Exception {  
    super.setUp();  
    ArrayList<TestHelper> lijn = new ArrayList<TestHelper>();  
    lijn.add(new TestHelper(20040101,20040131,"januari"));  
    lijn.add(new TestHelper(20040201,20040229,"februari"));  
    lijn.add(new TestHelper(20040301,20040331,"maart"));  
    lijn.add(new TestHelper(20040401,20040430,"april"));  
    lijn.add(new TestHelper(20040501,20040531,"mei"));  
    fixture = new Tijdlijn<TestHelper>(lijn, this);  
}
```



## Een tijdlijn test

```
public void testVerwerkRegel4b() {  
    TestHelper rgl = new TestHelper(20040310,20040410,"maart");  
    // Zo eenvoudig is een terugwerkende kracht mutatie:  
    fixture.verwerkRegel(rgl);  
    TestHelper[] voorsp = {  
        new TestHelper(20040101,20040131,"januari"),  
        new TestHelper(20040201,20040229,"februari"),  
        new TestHelper(20040301,20040410,"maart"),  
        new TestHelper(20040411,20040430,"april"),  
        new TestHelper(20040501,20040531,"mei")  
    };  
    controle(voorsp);  
}
```



## Conclusies

- Het is mogelijk complexiteit te reduceren door goede patterns voor temporale data gebaseerd op u\_operators.
- De operaties zouden in database producten beschikbaar moeten komen. Al 15 jaar weinig voortgang (TSQL).
- Er zijn diverse papers en dissertaties beschikbaar in de academische wereld. Nog niet alle patronen zijn volwassen.