

Tentamen Databases

2 september 2005

09:00 - 12:00, Educatorium Alfa

- Scheur de antwoordvellen doormidden.
- Maak elke vraag op een ander vel.
- Vermeld op elk vel je naam en studentnummer. Indien één van deze zaken ontbreekt, wordt het vel niet nagekeken.
- Toon bij het inleveren je collegekaart.
- Schrijf en formuleer duidelijk.
- Bij elke vraag wordt verwacht dat je laat zien hoe je aan het antwoord komt (tenzij anders wordt vermeld).
- Je mag ten hoogste één A4 met aantekeningen raadplegen.
- Gebruikte afko's:
 - 2PL = two-phase locking
 - 3NF = derde normaalvorm
 - BCNF = Boyce-Codd normaalvorm
 - DP = dependency preserving
 - FD = functional dependency
 - RA = relationele algebra
 - SQL = SQL (Structured Query Language)
- **Succes!**

Puntentelling:

- 1: 16
- 2: 24
- 3: 16
- 4: 24
- 5: 20

1 Algemeen

Bij de onderstaande vragen zijn er 0 of meerdere antwoorden correct. Je dient alle correcte antwoorden aan te geven. Een toelichting is niet noodzakelijk.

Vraag 1:

Welke van de onderstaande uitspraken zijn waar?

A: Elk relatieschema heeft minimaal één candidate key.

B: Indien een relatieschema meerdere candidate keys heeft, hebben deze altijd hetzelfde aantal attributen.

C: De attributen van een candidate key vormen altijd een strikte deelverzameling van een superkey.

D: Een foreign key is een primary key in een andere relatie.

E: Een primary key kan niet uit meer dan twee attributen bestaan.

Vraag 2:

Welke van de onderstaande uitspraken zijn waar?

A: Elke seriële schedule wordt geaccepteerd door een 2PL scheduler.

B: Elke schedule die geaccepteerd wordt door een 2PL scheduler is serieel.

C: Twee operaties op hetzelfde data-element geven altijd een conflict.

D: Twee read-operaties geven nooit een conflict.

E: Volgens het 2PL-protocol mag elke transactie niet meer dan 2 locks tegelijkertijd hebben.

Vraag 3:

Welke van de onderstaande afleidingsregels voor FD's zijn geldig?

A: $X \rightarrow Y, YZ \rightarrow U \Rightarrow XZ \rightarrow U$

B: $X \rightarrow Y, X \rightarrow Z \Rightarrow Y \rightarrow Z$

C: $XY \rightarrow ZU, YU \rightarrow X \Rightarrow Y \rightarrow Z$

Vraag 4:

Stel we hebben een I/O-mechanisme volgens de principes *deferred update* en *flush forced*. Welke van de onderstaande uitspraken zijn waar?

A: AFTER-images zijn van belang bij een recovery.

B: Het schrijven van een update naar het externe geheugen vindt mogelijkerwijs plaats na de commit.

C: Het regelmatig draaien van een checkpoint is niet zinvol.

D: BEFORE-images zijn van belang bij een recovery.

2 Queries

Een energiebedrijf heeft een zonne-energiesysteem geïnstalleerd. Het systeem bestaat uit een groot aantal zonnepanelen. Deze zogenaamde fofovoltaïsche panelen zetten energie uit zonlicht om in 24 Volt gelijkstroom. Elk zonnepaneel is aangesloten op een transformator die de 24 Volt gelijkstroom omzet naar 230 Volt wisselstroom. Elke transformator heeft een bepaalde gegeven efficiëntie. Deze efficiëntie ligt in de regel tussen de 85% en 95%. Het systeem bestaat uit een beperkt aantal transformatoren. Eén transformator is verbonden met meerdere panelen, maar een paneel is slechts op één transformator aangesloten. De transformatoren zijn aangesloten op het elektriciteitsnet. Als er licht is, produceert het systeem energie (ook bij bewolking) en deze energie wordt in het elektriciteitsnet gevoed. Uiteraard geldt: hoe meer licht, hoe hoger het geleverde vermogen. Wolken verminderen dus het geleverde vermogen van het systeem.

Om de prestaties van elk individueel zonnepaneel te monitoren, zijn de transformatoren uitgerust met een netwerkkaart die het mogelijk maakt op afstand de details van elk aangesloten paneel uit te lezen. De voor elk paneel beschikbare informatie bestaat uit een uniek paneelidentificatienummer, het huidige vermogen (Watt) en de huidige stroomsterkte (Ampère). Een pc verzamelt periodiek - eens per 6 seconden - voor elk paneel de informatie van de transformatoren. Deze gegevens worden opgeslagen in een database. Deze database heeft het volgende schema:

T(trafoïd, ttype, tmerk, efficiëntie)

P(paneelid, ptype, pmerk, locatie, trafoïd)

U(paneelid, uitleesdatum, uitleestijd, vermogen, stroomsterkte).

(T staat hier voor transformator, P voor paneel en U voor uitlezing.)

We hebben de volgende queries:

Q1: Geef het paneelid en de locatie van de panelen die aangesloten zijn op een transformator met een efficiëntie van tenminste 90%.

Q2: Geef het paneelid en het gemiddelde vermogen op 20 augustus 2005 van het paneel dat het hoogste gemiddelde vermogen had op 20 augustus 2005.

Q3: Geef het type en merk van de zonnepanelen waarvoor geen enkele uitlezing bestaat op 20 augustus 2005.

Hieronder volgen expressies in de RA of in SQL. Geef aan welke queries corresponderen met welke expressies. De relatie tussen queries en expressies is many-to-many en optioneel.

E1:

```
SELECT P.ptype, P.pmerk
FROM P
WHERE P.paneelid NOT IN
      (SELECT U.paneelid
       FROM U
       WHERE U.uitleesdatum = '20-08-2005')
```

E2:

```
SELECT U.paneelid, AVG(U.vermogen)
FROM U
WHERE U.uitleesdatum = '20-08-2005'
GROUP BY U.paneelid
HAVING AVG(U.vermogen) >= ALL
      (SELECT AVG(U.vermogen)
       FROM U
       GROUP BY U.paneelid)
```

E3:

```
SELECT U.paneelid, AVG(U.vermogen)
FROM U
WHERE U.uitleesdatum = '20-08-2005'
GROUP BY U.paneelid
HAVING MAX(AVG(U.vermogen))
```

E4:

```
SELECT U.paneelid, AVG(U.vermogen)
FROM U
WHERE U.uitleesdatum = '20-08-2005'
GROUP BY U.paneelid
HAVING AVG(U.vermogen) >= ALL
      (SELECT AVG(U.vermogen)
       FROM U
       WHERE U.uitleesdatum = '20-08-2005'
       GROUP BY U.paneelid)
```

E5:

```
SELECT P.ptype, P.pmerk
FROM P
WHERE P.paneelid NOT IN
      (SELECT U.paneelid
       FROM U
       WHERE U.paneelid = P.paneelid
       AND U.uitleesdatum = '20-08-2005')
```

E6:

```
SELECT P.paneelid, P.locatie
FROM T, P
WHERE T.trafoïd = P.trafoïd
AND T.efficientie >= 90
```

E7:

```
SELECT P.ptype, P.pmerk
FROM P
WHERE NOT EXISTS
  (SELECT U.paneelid
   FROM U
   WHERE U.uitleesdatum = '20-08-2005')
```

E8:

```
SELECT U.paneelid, AVG(U.vermogen)
FROM U
WHERE U.uitleesdatum = '20-08-2005'
GROUP BY U.paneelid
HAVING AVG(U.vermogen) >= ALL
```

E9:

```
SELECT P.paneelid, P.locatie
FROM P
WHERE P.paneelid NOT IN
  (SELECT P.paneelid
   FROM P, T
   WHERE P.trafoïd = T.trafoïd
   AND T.efficientie >= 90)
```

E10: $\pi_{\text{paneelid,locatie}}((\pi_{\text{paneelid}}(P) - \pi_{\text{paneelid}}(\sigma_{\text{efficientie} \geq 90}(P \bowtie T))) \bowtie P)$

E11: $\pi_{\text{ptype,pmerk}}(T \bowtie P \bowtie (\sigma_{\text{uitleesdatum} <> '20-08-2005'}(U)))$

E12: $\pi_{\text{ptype,pmerk}}((\pi_{\text{paneelid}}(P) - \pi_{\text{paneelid}}(\sigma_{\text{uitleesdatum} = '20-08-2005'}(U))) \bowtie P)$

E13: $\pi_{\text{ptype,pmerk}}(\sigma_{\text{uitleesdatum} <> '20-08-2005'}(T \bowtie P \bowtie U))$

E14: $\pi_{\text{paneelid,locatie}}((\pi_{\text{paneelid}}(P) - \pi_{\text{paneelid}}(\sigma_{\text{efficientie} < 90}(P \bowtie T))) \bowtie P)$

3 Functionele afhankelijkheden

We hebben een aantal attributen die betrekking hebben op personen.

- *paspoortnummer*
- *rijbewijsnummer*
- *sofnummer*
- *achternaam*
- *initialen*
- *geboortedatum*
- *geboorteplaats*
- *adres (inclusief woonplaats)*
- *postcode*
- *e-mailadres*

(Maak bij het beantwoorden van de vragen eventuele aannames expliciet door ze op te schrijven.)

- (i) Geef alle attributen die functioneel afhankelijk zijn van *sofnummer*.
- (ii) Geef alle attributen die functioneel afhankelijk zijn van *adres*.
- (iii) Geef alle attributen die functioneel afhankelijk zijn van *paspoortnummer*.
- (iv) Geef alle attributen die functioneel afhankelijk zijn van *e-mailadres*.

4 Normaalvormen

- (i) Stel we hebben een relatieschema $R(ABCDEFGG)$ en een verzameling FD's $F_R = \{E \rightarrow FB, E \rightarrow A, B \rightarrow C, A \rightarrow CD, C \rightarrow BD\}$.

Geef twee verschillende verliesvrije BCNF-decomposities van R . Laat zien welke methode je gebruikt.

- (ii) Stel we hebben een relatieschema $R(ABCDEFGHIJ)$ en een verzameling FD's $F_R = \{A \rightarrow BI, CB \rightarrow AD, I \rightarrow E, GB \rightarrow IDJ, G \rightarrow AC\}$.

- (a) Geef alle candidate keys voor R . Geef hierbij een korte toelichting.
 (b) Geef een verliesvrije DP 3NF-decompositie van R . Laat zien welke methode je gebruikt.

5 Concurrency

We beschouwen de volgende twee schedules.

- (i) Geef aan of deze schedules serialiseerbaar zijn of niet. Licht toe. Geef zo mogelijk de equivalente seriële schedules.
 (ii) Geef eveneens aan of de schedules getolereerd worden door een 2PL-scheduler. Geef hierbij een korte toelichting.

$S1$					$S2$				
T1	T2	T3	T4	T5	T1	T2	T3	T4	T5
w(y)								r(z)	
		w(z)				r(y)			
				r(x)	w(x)				r(y)
			w(y)			r(x)			
			r(x)		r(z)				
				r(z)			r(x)		
	r(y)			r(y)		r(y)			
			w(y)					w(x)	
r(z)							w(z)		

Einde