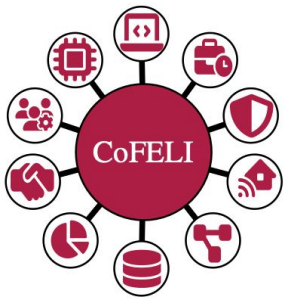


Collectif francophone pour l'enseignement libre de l'informatique

Bases de données historicisées

Modélisation

CoFELI:BDH_10

Christina KHNAISSER *(christina.khnaisser@usherbrooke.ca)*

—

CoFELI/Scriptorum/BDH_10-Modele-historicise (v100), version 1.0.0.a, en date du 2024-11-02

— document de travail, ne pas citer —

Sommaire

Introduction à la modélisation d'une base de données (entrepôt de données) historicisées.

Mise en garde

Le présent document est en cours d'élaboration ; en conséquence, il est incomplet et peut contenir des erreurs.

Historique

diffusion	resp.	description
2024-11-13	CK	Ébauche initiale.

Table des matières

Introduction

Le présent document a pour but de présenter principes de modélisation d'une base de données (entrepôt de données) historicisées.

La présentation repose sur une connaissance en SQL.

Contenu des sections

- La section 1 présente la roblématique générale et son contexte.
- La section 2 présente un exemple qui servira de fil conducteur aux sections subséquentes.
- La section 3 présente le modèle du temps le plus utilisé en informatique, fortement inspiré du modèle utilisée pour l'observation astronomique.

Évolution du document

La première version du document a été établie sur la base des travaux publiés par Adamson.

Travail en cours ou projeté

- DONE 2024-11-02 (CK) : rédaction à partir de diverses notes de cours.

1. Mise en contexte

Les bases de données transactionnelles ont pour vocation de refléter la réalité au moment de leur consultation (l'état courant).

Les bases de données temporelles ont pour vocation de refléter la réalité au moment décrit par une annotation temporelle.

Les bases de données historicisées ont pour vocation de refléter l'évolution de la réalité au cours d'une période selon le point de vue d'un agent.

Gérer l'évolution des données :

- Structurer les données en fonction de leur *sémantique* temporelle
- Maintenir la cohérence/Détecter l'incohérence temporelle

2. Problématique par l'exemple

2.1. Calendrier

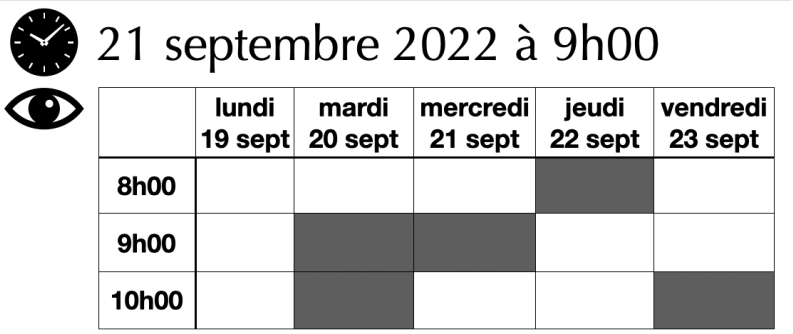
21 septembre 2022 à 7h00

	lundi 19 sept	mardi 20 sept	mercredi 21 sept	jeudi 22 sept	vendredi 23 sept
8h00					
9h00					
10h00					

no	date rdv debut	date rdv fin
1	20-09-2022 09:00	20-09-2022 10:00
2	20-09-2022 10:00	20-09-2022 11:00
3	21-09-2022 09:00	21-09-2022 10:00
4	21-09-2022 10:00	21-09-2022 11:00
5	22-09-2022 08:00	22-09-2022 09:00

Nous sommes le 21 septembre 2022 à **7h00** (la date du système):

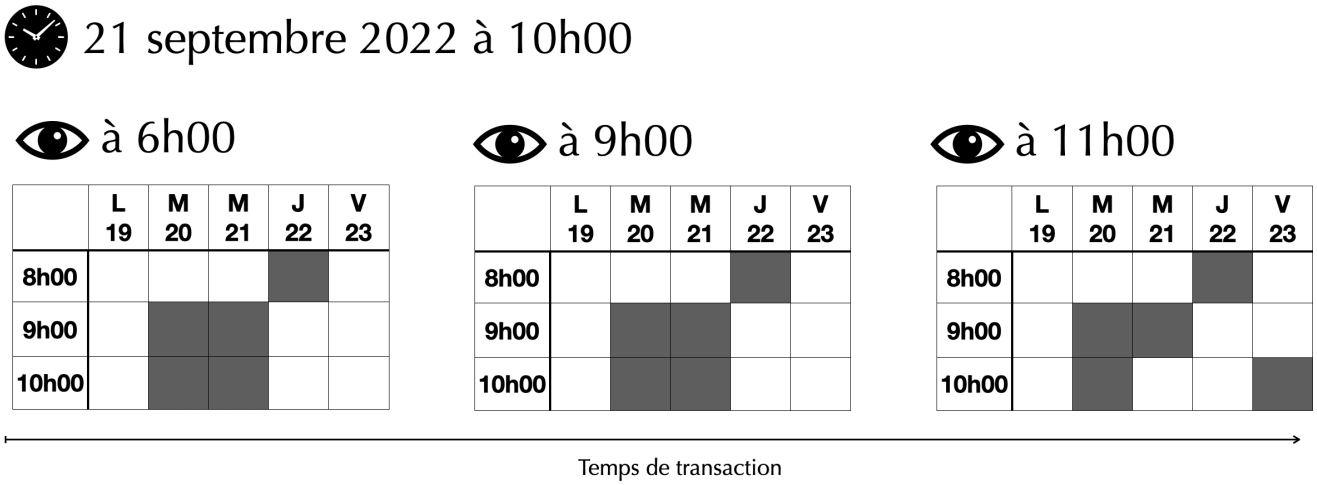
- Les faits qui représentent l'état courant de mon calendrier sont :
- Les faits qui représentent l'état courant de mon calendrier sont :
  - J'avais deux rendez-vous le mardi 20 septembre un de 9h00 à 10h00 et un de 10h00 à 11h00.
  - J'ai deux rendez-vous, un de 9h00 à 10h00 et un de 10h00 à 11h00.
  - J'aurai un rendez-vous le jeudi 22 septemnbre de 8h00 à 9h00.



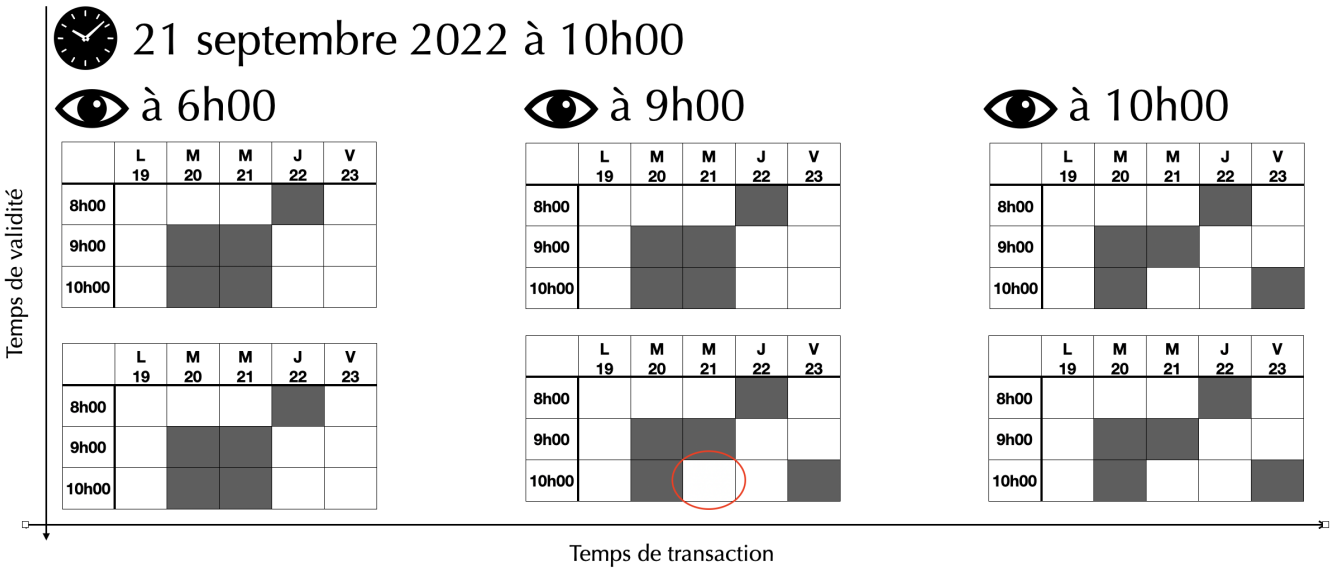
no	date rdv debut	date rdv fin
1	20-09-2022 09:00	20-09-2022 10:00
2	20-09-2022 10:00	20-09-2022 11:00
3	21-09-2022 09:00	21-09-2022 10:00
4	23-09-2022 10:00	23-09-2022 11:00
5	22-09-2022 08:00	22-09-2022 09:00

Le 21 Septembre 2022 à **9h00**

- Le rendez-vous de 10h00 est reporté à vendredi :
  - la rencontre de 10h00 est supprimée ;
  - un rendez-vous le vendredi 23 septembre à 10h00 est ajouté.



no	date rdv debut	date rdv fin	transaction debut	transaction fin
1	20-09-2022 09:00	20-09-2022 10:00	19-01-2022 08:00	
2	20-09-2022 10:00	20-09-2022 11:00	19-01-2022 08:00	
3	21-09-2022 09:00	21-09-2022 10:00	19-01-2022 08:00	
4	21-09-2022 10:00	21-09-2022 11:00	19-01-2022 08:00	21-01-2022 **10:00**
4	23-09-2022 10:00	23-09-2022 11:00	21-01-2022 **10:00**	
5	22-09-2022 08:00	22-09-2022 09:00	19-01-2022 08:00	



no	date rdv debut	date rdv fin	validité début	validité fin	transaction debut	transaction fin
1	20-09-2022 09:00	20-09-2022 10:00	19-01-2022 08:00		19-01-2022 08:00	
2	20-09-2022 10:00	20-09-2022 11:00	19-01-2022 08:00		19-01-2022 08:00	
3	21-09-2022 09:00	21-09-2022 10:00	19-01-2022 08:00		19-01-2022 08:00	
4	21-09-2022 10:00	21-09-2022 11:00	19-01-2022 08:00	21-01-2022 **09:15**	19-01-2022 08:00	21-01-2022 10:00
4	23-09-2022 10:00	23-09-2022 11:00	21-01-2022 **09:15**		21-01-2022 10:00	
5	22-09-2022 08:00	21-09-2022 09:00	19-01-2022 08:00		19-01-2022 08:00	

3. Historicisation

3.1. Attribut temporel

- Intervalle du domaine (*user-defined time*) : intervalle temporel qui fait référence à l'horloge associée au processus ayant engendré le tuple.
  - période de rendez-vous
  - date de fabrication, date d'expiration
  - date prise du médicament, date arrêt du médicament
- Intervalle de validité (*valid time*) [@V] : intervalle temporel qui fait référence à l'agent qui a constaté la proposition représentée par le tuple.
  - Sa valeur est fournie par l'agent.
  - Un intervalle de validité commence à l'instant où l'agent considère la proposition vraie et se termine lorsqu'il que la proposition devienne fausse.
  - Il peut y avoir plusieurs référentiels de validité dans le contexte où les agents ne partagent pas une même horloge.
- Intervalle de transaction (*transaction time*) [@T] : intervalle temporel qui fait référence à l'horloge du SGBD qui enregistre le tuple.
  - Sa valeur est établie par le SGBDR sur la base de l'instant de confirmation de la transaction (*commit time*).
  - Un intervalle de transaction débute par l'instant de la transaction d'insertion et l'instant de la transaction de suppression (ou mise à jour).
  - L'utilisateur ne peut pas la modifier directement.

3.2. Structure

- Relation non historicisée [R@N]
- Relation avec un intervalle de validité [R@V]
- Relation avec un intervalle de transaction [R@T]
- Relation bitemporelle avec un intervalle de validité et un intervalle transaction [R@VT]

3.2.1. Exemple

Tableau 1. Produit@N

noP	nom	couleur	prix
P1	Boite	noire	1.50$
P2	Chaise	bleue	65.00$
P3	Table	brun	200.00$
P4	Armoire	brun	150.00$

Tableau 2. Produit@V

noP	nom	couleur	prix	validité
P1	Boite	noire	1.50$	[1, 6)
P1	Boite	noire	2.50$	[6, 8)
P1	Boite	blanc	2.50$	[8, 10)
P1	Boite	blanc	3.50$	[10,)
P2	Chaise	bleue	65.00$	[2, 12)
P2	Chaise	bleue	115.00$	[12,)
P3	Table	brun	200.00$	[6,)
P4	Armoire	brun	150.00$	[6, 14)

Produit_nom@V

```
CREATE TABLE "Produit_nom@V" (
    noP      char(2),
    nom      varchar,
    validite int4range,
    CONSTRAINT produit_nom_cc00 PRIMARY KEY (noP, validite) );
```

noP	nom	validité
P1	Boite	[1,)
P2	Chaise	[2,)
P3	Table	[6,)
P4	Armoire	[6, 14)

Produit_couleur@V

```
CREATE TABLE "Produit_couleur@V" (
    noP      char(2),
    couleur  varchar,
    validite int4range,
    CONSTRAINT produit_couleur_cc00 PRIMARY KEY (noP, validite) );
```

noP	couleur	validité
P1	noire	[1, 8)
P1	blanc	[8,)

P2	bleue	[2,)
P3	brun	[6,)
P4	brun	[6, 14)

Produit_prix@V

```
CREATE TABLE "Produit_prix@V" (
    noP      char(2),
    prix      varchar,
    validite int4range,
    CONSTRAINT produit_prix_cc00 PRIMARY KEY (noP, validite) );
```

noP	prix	validité
P1	1.50$	[1, 6)
P1	2.50$	[6, 10)
P1	3.50$	[10,)
P2	65.00$	[2, 12)
P2	115.00$	[12,)
P3	200.00$	[6,)
P4	150.00$	[6, 14)

Tableau 3. Fournisseur@V

noF	noP	quantité	validité
F1	P1	200	[2, 6)
F1	P1	250	[8,)
F1	P2	100	[6,11)
F2	P3	400	[4,)
F2	P4	500	[4, 13)

Tableau 4. Produit_nom@V

noP	nom	validité
P1	Boite	[1,)
P2	Chaise	[2, 8)
P3	Table	[6,)
P4	Armoire	[6, 12)

3.2.2. Processus

Soit R{K, A} une relation en 5FN :

- K est l'ensemble des attributs clé,
- A{a1...an} l'ensemble des attributs .

L'historicisation de R consiste à ajouter un attribute temporel puis normaliser la relation en 6FN :

- Projection-Jointure : K et ai
- Restriction-Union :
  - Intervalle point début indéterminé
  - Intervalle point fin indéterminé
  - Intervalle déterminée
  - Intervalle indéterminée

3.3. Contraintes

Non-Redondance

La non-redondance est garantie lorsque les intervalles de deux tuples contenant les même données n'ont aucun point en commun.

```
ALTER TABLE "Produit_couleur@V"
  ADD CONSTRAINT produit_couleur_non-redondance
      EXCLUDE USING GIST (noP WITH =, couleur WITH =, validite WITH &&);
```

Non-Contradiction

La non-contradiction est garantie lorsque les intervalles de deux tuples ayant la même clé dans S et des données différentes n'ont aucun point en commun.

```
ALTER TABLE "Produit_couleur@V"
  ADD CONSTRAINT produit_couleur_non-contradiction
    EXCLUDE USING GIST (noP WITH =, couleur WITH <>, validite WITH &&);
```

Compacité

La compacité garantit l'égalité temporelle entre deux relations.

```
with
  -- Division des intervalle par point
  T1 (start_ts, end_ts, ts, noP, validite, source) as (
    select 1, 0, lower(validite), noP, validite, 'Produit_couleur' as source
    from "Produit_couleur@V"
    union all
    select 0, 1, upper(validite), noP, validite, 'Produit_couleur' as source
    from "Produit_couleur@V"
    union all
    select 1, 0, lower(validite), noP, validite, 'Produit' as source
    from "Produit@V"
    union all
    select 0, 1, upper(validite), noP, validite, 'Produit' as source
    from "Produit@V"
  ),
```

```
  -- Groupement des points par clé
  T2 (crt_Total_ts, ts, noP, validite, source) as (
    select sum(start_ts) over w - sum(end_ts) over w as crt_Total_ts
         , ts
         , noP
         , validite
         , source
    from T1
    window w as (partition by noP order by ts, end_ts rows unbounded preceding)
  ),
```

```
  -- Construction d'intervalles contigus
  T3 as (
    select distinct noP, validite, source
        , int4range(ts , LEAD(ts) over (partition by noP order by ts )) as T
    from T2
    window w as (partition by noP order by ts)
  ),
```

```
  -- Identification des trous temporels
  anomalies as (
    select distinct noP
         , t32.source
         , t31.t
         , t32.validite
    from T3 as t31 join T3 as t32 using (noP)
    where t31.t <@ t32.validite
      and t31.t <> 'empty'
  )
select noP, t as gap, array_agg(distinct source) as source
from anomalies
group by noP, t
having array_length(array_agg(distinct source), 1) = 1;
```

Intégrité référentielle temporelle

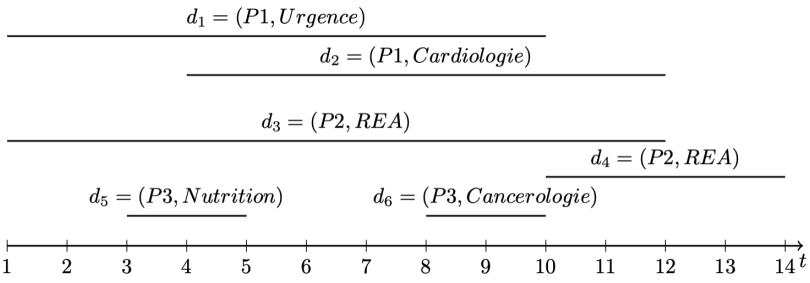
L'intégrité référentielle temporelle garantit l'inclusion temporelle entre deux relations.

```
-- même requête que la compacité.
-- remplacer la condition du regroupement par :
-- having array_agg(distinct source) = 'Produit'
```

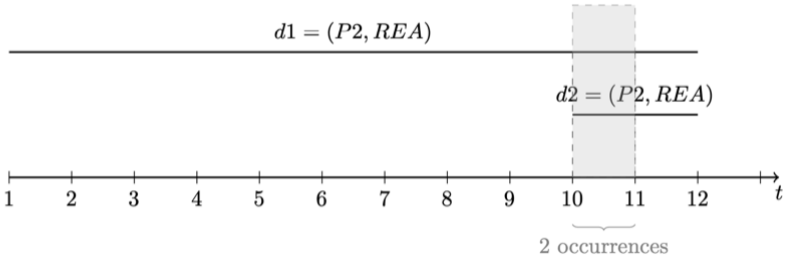
Exemple 1. Exemple Séjour d'un patient à l'hopital

K ai Intervalle

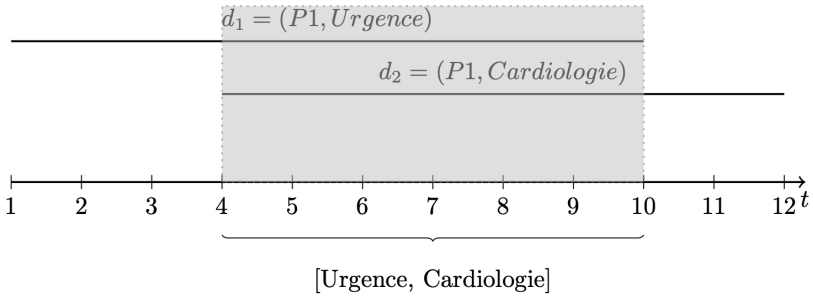
	PatId	Unité	T
d_1	P1	Urgence	[1,10)
d_2	P1	Cardiologie	[4,12)
d_3	P2	REA	[1,12)
d_4	P2	REA	[10,14)
d_5	P3	Nutrition	[3,5)
d_6	P3	Cancérologie	[8,10)



	PatId	Unité	T
d_1	P1	Urgence	[1,10)
d_2	P1	Cardiologie	[4,12)
d_3	P2	REA	[1,12)
d_4	P2	REA	[10,14)



	PatId	Unité	T
d_1	P1	Urgence	[1,10)
d_2	P1	Cardiologie	[4,12)
d_3	P2	REA	[1,12)
d_4	P2	REA	[10,14)



[Urgence, Cardiologie]

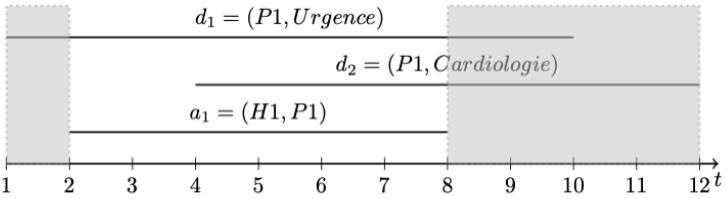
- Séjour

	PatId	Unité	T
d_1	P1	Urgence	[1,10)
d_2	P1	Cardiologie	[4,12)
d_3	P2	REA	[1,12)
d_4	P2	REA	[10,14)

- Hospitalisation

Identifiant

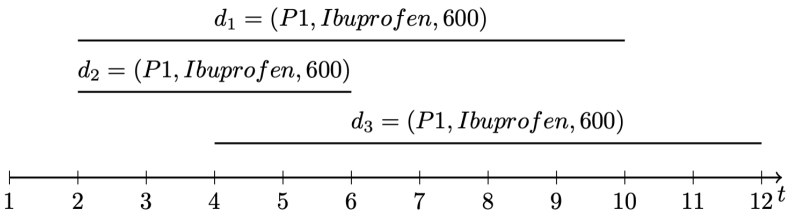
	HId	PatId	T
a_1	H1	P1	[2,8)
a_2	H1	P2	[1,12)



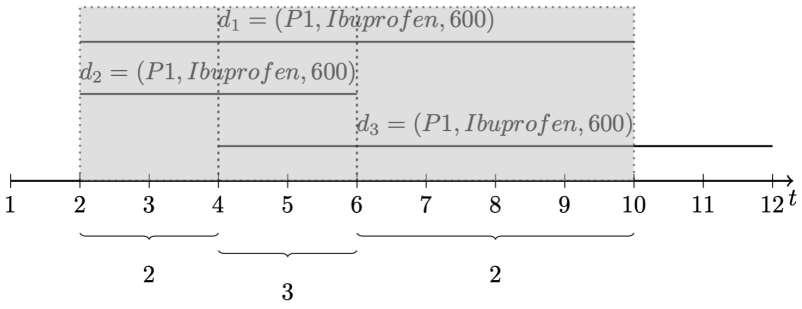
Exemple 2. Exemple médication d'un patient

K ai Intervalle

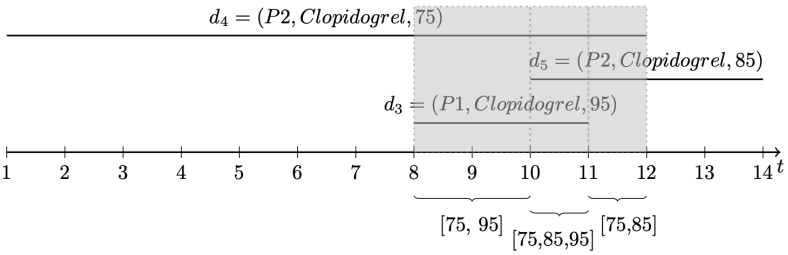
	PatId	DrugName	Dose	T
d_1	P1	Ibuprofen	600mg	[2,10)
d_2	P1	Ibuprofen	600mg	[2,6)
d_3	P1	Ibuprofen	600mg	[4,12)
d_4	P2	Clopidogrel	75mg	[1,12)
d_5	P2	Clopidogrel	85mg	[10,14)
d_6	P2	Clopidogrel	95mg	[8,11)



	PatId	DrugName	Dose	T
d_1	P1	Ibuprofen	600mg	[2,10)
d_2	P1	Ibuprofen	600mg	[2,6)
d_3	P1	Ibuprofen	600mg	[4,12)
d_4	P2	Clopidogrel	75mg	[1,12)
d_5	P2	Clopidogrel	85mg	[10,14)
d_6	P2	Clopidogrel	95mg	[8,11)

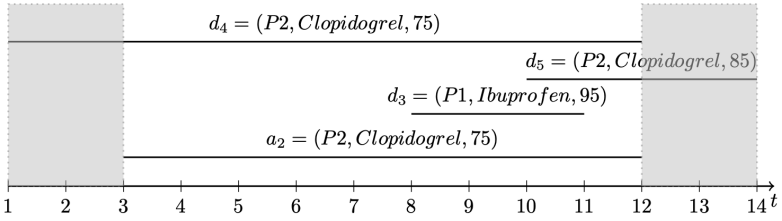


	PatId	DrugName	Dose	T
d_1	P1	Ibuprofen	600mg	[2,10)
d_2	P1	Ibuprofen	600mg	[2,6)
d_3	P1	Ibuprofen	600mg	[4,12)
d_4	P2	Clopidogrel	75mg	[1,12)
d_5	P2	Clopidogrel	85mg	[10,14)
d_6	P2	Clopidogrel	95mg	[8,11)



- Prescription d'une dose

	PatId	DrugName	Dose	T
d_1	P1	Ibuprofen	600mg	[2,10)
d_2	P1	Ibuprofen	600mg	[2,6)
d_3	P1	Ibuprofen	600mg	[4,12)
d_4	P2	Clopidogrel	75mg	[1,12)
d_5	P2	Clopidogrel	85mg	[10,14)
d_6	P2	Clopidogrel	95mg	[8,11)



- Administration d'une dose

	PatId	DrugName	Dose	T
a_1	P1	Ibuprofen	600mg	[2,12)
a_2	P2	Clopidogrel	75mg	[3,12)

Références

[Date2014a]
Chris J. DATE, Hugh DARWEN, Nikos A. LORENTZOS ;
Time and Relational Theory: Temporal Databases in the Relational Model and SQL ;
Morgan Kaufmann, Waltham (MA, US), 2014 ;
ISBN 978-0-12-800631-3.

[Manthey2018a]
Rainer MANTHEY ;
Temporal Information Systems ;
Institut für Informatik, Universität Bonn, 2018 ;
https://pages.iai.uni-bonn.de/manthey_rainer/TIS2018 (consulté le 2024-04-19).

[Khnaisser2019a]
Christina KHNAISSER ;
Construction de modèles de données relationnels temporalisés guidée par les ontologies ;
Université de Paris en cotutelle avec l'Université de Sherbrooke, 2019 ;
http://www.theses.fr/s177273

PG

https://www.postgresql.org/docs/current/rangetypes.html
https://www.postgresql.org/docs/current/ddl-constraints.html#DDL-CONSTRAINTS-EXCLUSION

Produit le 2025-09-18 13:06:40 -0400

Collectif francophone pour l'enseignement libre de l'informatique