

BD50

Conception des Bases de Données

Examen Final – 24 juin 2017

Département Informatique

Conditions de réalisation :

Support de cours et travaux pratiques de BD50 autorisés

Durée : **3 heures**

Barème

Gestion des transport de la compagnie ATLANTIK

Optimisation du Modèle Physique de données	5 pts
Optimisation Oracle	4 pts (1 pt par question)
Rédaction d'une requête SQL	3 pts
Optimisation des requêtes	2 pts
Requête 1	1 pt
Requête 2	1 pt
Programmation PL/SQL	6 pts

Gestion du transport des voyageurs : optimisation du modèle

Présentation du contexte :

La compagnie Atlantik est une compagnie maritime implantée sur la côte ouest de la France. Ayant obtenu plusieurs contrats avec des conseils généraux, la compagnie assure la desserte maritime d'îles du littoral français par délégation de service public.

Parmi les îles desservies à partir du continent, on trouve, par exemple, Belle-Île-en-mer, Houat, Ile de Groix, Ouessant, Molène, Sein, Bréhat, Batz, Aix ou encore Yeu.

Pour son activité de transport de voyageurs, la compagnie Atlantik souhaite présenter toute l'information concernant les liaisons maritimes qu'elle assure sur un même portail internet. Les informations du site seront générées dynamiquement à partir d'une base de données. Ce portail doit notamment proposer aux internautes les services suivants :

- consultation des liaisons,
- consultation des tarifs,
- consultation des horaires des traversées,
- réservation en ligne.

Un extrait des liaisons proposées par secteur est présenté ci-dessous

Secteur	Liaison			
	Code Liaison	Distance en milles marin	Port de départ	Port d'arrivée
Belle-Ile-en-Mer	15	8.3	Quiberon	Le Palais
	24	9	Le Palais	Quiberon
	16	8.0	Quiberon	Sauzon
	17	7.9	Sauzon	Quiberon
	19	23.7	Vannes	Le Palais
	11	25.1	Le Palais	Vannes
Houat	25	8.8	Quiberon	Port St Gildas
	30	8.8	Port St Gildas	Quiberon
Ile de Groix	21	7.7	Lorient	Port-Tudy
	22	7.4	Port-Tudy	Lorient
etc...				

Les tarifs dépendent de la liaison, de la période du voyage, et du type du passager ou du véhicule transporté.

Les types sont classés en 3 catégories principales susceptibles d'évolution :

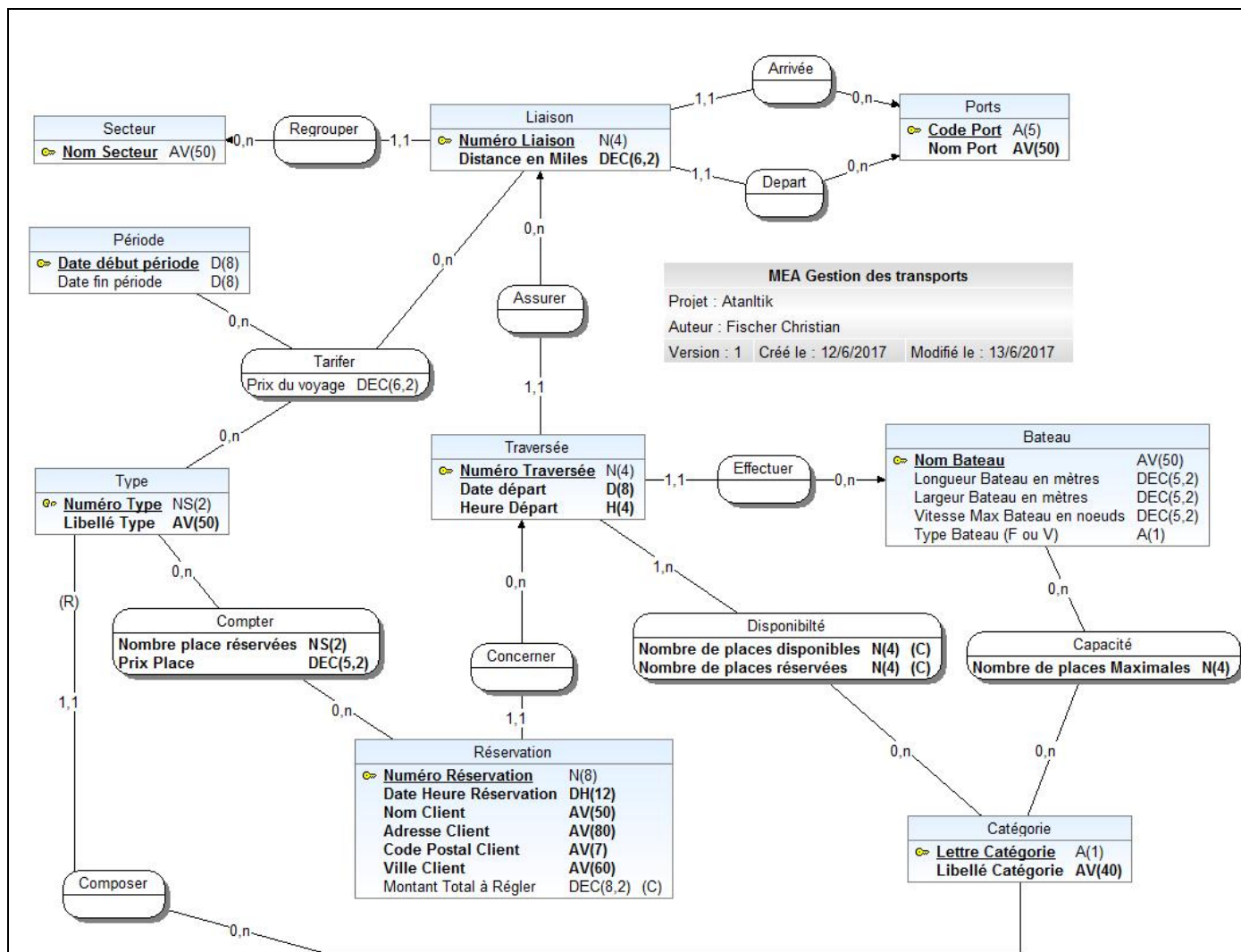
- "Passager" : tout passager, tout âge confondu,
- "Véh.inf.2m" : véhicules dont la hauteur est inférieure ou égale à 2 mètres,
- "Véh.sup.2m" : véhicules dont la hauteur est supérieure à 2 mètres.

Extrait du tarif

Compagnie Atlantik		Tarifs en euros		
Liaison 15 : Quiberon - Le Palais				
Catégorie	Type	Période		
		01/09/2010 15/06/2011	16/06/2011 15/09/2011	16/09/2011 31/05/2012
A Passager	A1 - Adulte	18.00	20.00	19.00
	A2 - Junior 8 à 18 ans	11.10	13.10	12.10
	A3 - Enfant 0 à 7 ans	5.60	7.00	6.40
B Véh.inf.2m	B1 - Voiture long.inf.4m	86.00	95.00	91.00
	B2 - Voiture long.inf.5m	129.00	142.00	136.00
C Véh.sup.2m	C1 - Fourgon	189.00	208.00	199.00

Le type de bateau est défini par les valeurs : F pour le fret ou V pour les voyageurs

Modèle entité association de la gestion des transports :



Travail à faire

1. A partir du MEA ci-dessus, fournir la représentation graphique du modèle du modèle physique de données optimisé (en justifiant vos choix).

2. Optimisation du modèle physique de données pour Oracle

Travail à faire

1. A partir du modèle physique de données optimisé indiquez les tables qui seront implantées en IOT (Index Organized Table).
2. Rédiger une instruction SQL de création d'une table organisée en index, de votre choix. Le tablespace de stockage de la clé primaire est nommé : ATL_IND.
3. Rédiger l'instruction de création de la table BATEAU qui sera partitionnée par liste sur le type de bateau (F pour le fret ou V pour les voyageurs)
4. Proposez une optimisation pour le stockage du nom de liaison :
Exemple :
Liaison numéro 15 :
Nom de la liaison : **Quiberon - Le Palais** (nom du port de départ concaténé avec le nom du port d'arrivée avec un tiret en séparateur).

3. Requête SQL à faire sur votre modèle optimisé

Travail à faire

- La rédaction des requêtes doit être conforme aux règles de présentation mises en œuvre en TP.
- L'écriture des jointures doit être effectuée en **formulation ANSI avec la syntaxe : Inner join et Outer join avec la clause ON.**
- L'utilisation d'**alias de table** est **obligatoire.**

1. Écrire la requête SQL permettant de calculer le nombre de traversée pour toutes les liaisons sur les 3 dernières années.

Afficher le numéro et le nom de liaison (port de départ – port d'arrivée) et le nombre de traversées.

Le résultat sera trié par numéro de liaison.

Note : la colonne nom de liaison est disponible dans le modèle optimisé dans la table Liaison.

Exemple de résultat attendu

Numéro de liaison	Nom de liaison	2016	2015	2014
11	Le Palais - Vannes	120	115	118
15	Quiberon - Le Palais	125	117	110
16	Quiberon - Sauzon	130	119	102
17	Sauzon - Quiberon	135	121	94
19	Vannes - Le Palais	140	123	86
21	Lorient - Port-Tudy	145	125	78
22	Port-Tudy - Lorient	150	127	70
24	Le Palais - Quiberon	155	129	62
25	Quiberon - Port St Gildas	160	131	54
30	Port St Gildas - Quiberon	165	133	46

4. Optimisation des requêtes SQL

Requête 1 :

Liste des liaisons pour le secteur de Belle-Ile-en-Mer

```
select
  LIA.*
from
  Liaison LIA
where
  LIA.nom_secteur='Belle-Ile-en-Mer'
;
```

Plan d'exécution associé :

Plan hash value: 2983519079

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		1	67	2 (0)	00:00:01
* 1	TABLE ACCESS FULL	LIAISON	1	67	2 (0)	00:00:01

Predicate Information (identified by operation id):

1 - filter("LIA"."NOM_SECTEUR"='Belle-Ile-en-Mer')

Travail à faire

1. A partir du plan ci-dessus, analyser la méthode d'accès pour la table liaison.

Ce plan est-il optimal ?

Proposer une solution pour optimiser cette requête.

Requête 2 :

```
select
  LIA.*
,TRA.NOMBAT
,TRA.DATDEPTRAV
,TRA.HEUDEPTRAV
from
  Liaison LIA inner join Traversee TRA
    on LIA.NUMLIAISON=TRA.NUMLIAISON
where
  TRA.DATDEPTRAV Between '1/6/2017' and '30/6/2017'
and
  TRA.NOMBAT='Luce Isle'
and
  LIA.nom_secteur='Belle-Ile-en-Mer' ;
```

Les colonnes de clé étrangère sont indexées :

nom_secteur dans liaison : I_FK_LIAISON_SECTEUR
num_liaison dans traversée : I_FK_TRAVERSEE_LIAISON
nombat dans traversée : I_FK_TRAVERSEE_BATEAU

Plan d'exécution associé :

Plan hash value: 1655938889

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		1	125	1 (0)	00:00:01
1	NESTED LOOPS					
2	NESTED LOOPS		1	125	1 (0)	00:00:01
3	TABLE ACCESS BY INDEX ROWID	LIAISON	1	67	1 (0)	00:00:01
* 4	INDEX RANGE SCAN	I_FK_LIAISON_SECTEUR	1		1 (0)	00:00:01
* 5	INDEX RANGE SCAN	I_FK_TRAVERSEE_LIAISON	1		0 (0)	00:00:01
* 6	TABLE ACCESS BY INDEX ROWID	TRAVERSEE	1	58	0 (0)	00:00:01

Predicate Information (identified by operation id):

```
4 - access("LIA"."NOM_SECTEUR"='Belle-Ile-en-Mer')
5 - access("LIA"."NUMLIAISON"="TRA"."NUMLIAISON")
6 - filter("TRA"."NOMBAT"='Luce Isle' AND "TRA"."DATDEPTRAV">=TO_DATE(' 2017-06-01
    00:00:00', 'syyyy-mm-dd hh24:mi:ss') AND "TRA"."DATDEPTRAV"<=TO_DATE(' 2017-06-30 00:00:00',
    'syyyy-mm-dd hh24:mi:ss'))
```

Travail à faire

2. À partir du plan ci-dessus, analyser les méthodes d'accès pour la table liaison et traversée ainsi que la jointure associée

Ce plan est-il optimal ?

Proposer une solution pour optimiser cette requête.

5. Programmation PL/SQL

Travail à faire

1. Rédiger la fonction PL/SQL GET_NOMLIAISON permettant d'afficher le nom complet de la liaison (nom du port de départ concaténé avec le nom du port d'arrivée avec un tiret en séparateur) à partir du numéro de la liaison.
2. Rédiger la requête permettant d'afficher le numéro de liaison et le nom de la liaison à partir de la table des liaisons.
3. Rédiger le **corps du package de gestion de la table PA_BATEAU** avec les procédures ou fonctions :
 - Add : pour créer un nouveau bateau,
 - GetByPK : pour afficher un bateau à partir de son nom (PK)
 - GetALLByType : pour afficher tous les bateaux d'un type (F ou V)
 - Upd : pour mettre à jour toutes les informations du bateau
 - Remove : pour supprimer un bateau