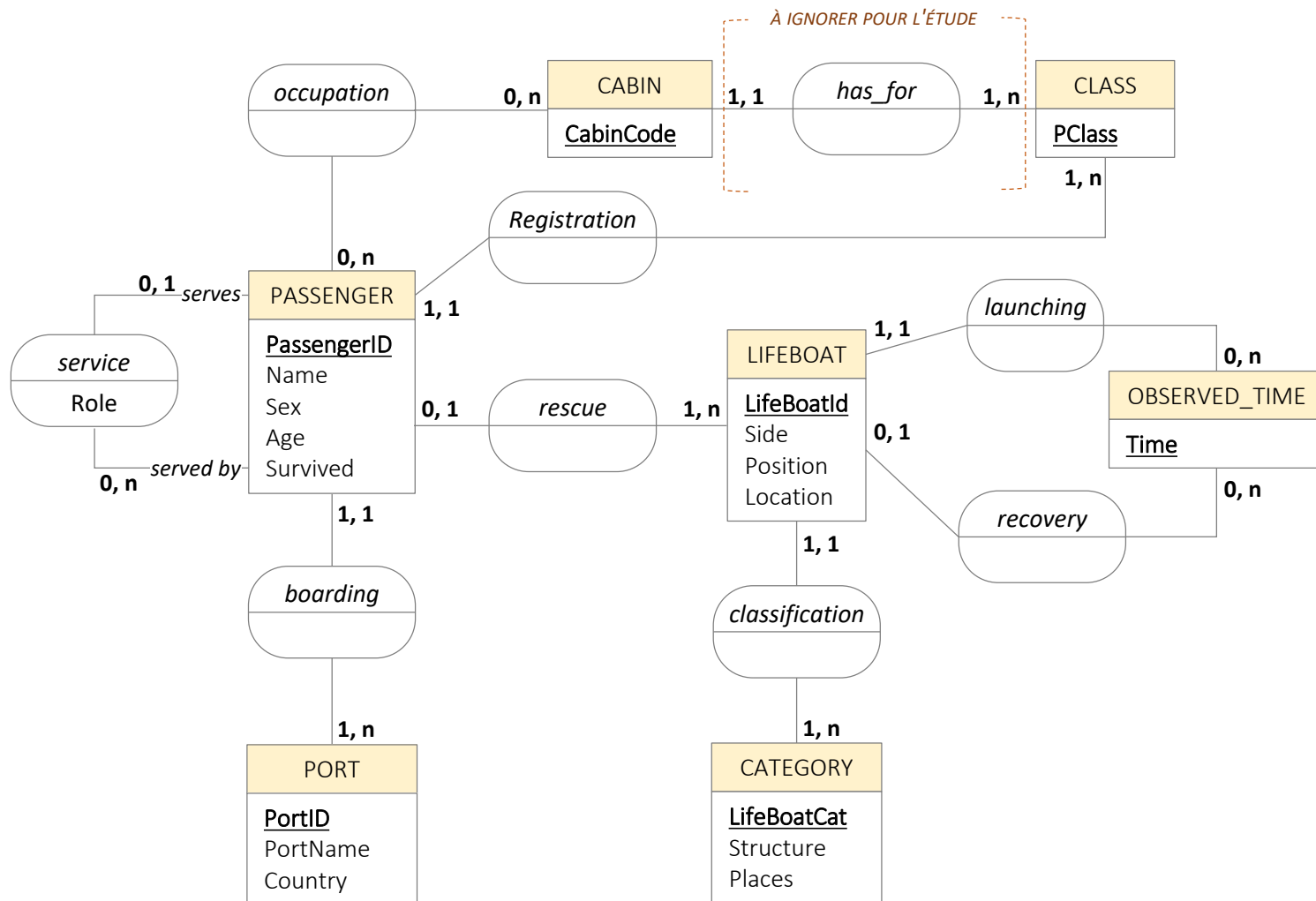


# SEA - TITANIC



## SCHÉMA RELATIONNEL (SLR) DE LA BD TITANIC

---

PORT(PortId, PortName, Country)

PASSENGER(PassengerId, Name, Sex, Age, Survived, PClass, #PortId)

OCCUPATION(#PassengerId, CabinCode)

SERVICE(#PassengerId\_Dom, #PassengerId\_Emp, Role)

CATEGORY(LifeBoatCat, Structure, Places)

LIFEBOAT(LifeBoatId, #LifeBoatCat, Side, Position, Location, Launching\_Time)

RECOVERY(#LifeBoatId, Recovery\_Time)

RESCUE(#PassengerId, #LifeBoatId)

Rappel :

Les entités CLASS et OBSERVED\_TIME du SEA n'ont pas été conservées après dérivation, car réduites à un seul attribut (leur clé primaire)

### TYPE ET CONTRAINTES DES ATTRIBUTS (HORS CLÉ PRIMAIRE OU CLÉ ÉTRANGÈRE)

*Note : les attributs qui portent le même nom dans plusieurs relations ont le même type*

PORT		
PortId	char(1)	C, Q ou S
PortName	varchar	doit être renseigné
Country	varchar	doit être renseigné
PASSENGER		
PassengerId	int	
Name	varchar	doit être renseigné
Sex	varchar	doit être renseigné
Age	int	
Survived	int	valeurs 0 ou 1
PClass	int	doit être renseigné, compris entre 1 et 3
OCCUPATION		
CabinCode	varchar	
SERVICE		
PassengerId_Dom	int	
PassengerId_Emp	int	doit être renseigné
Role	varchar	doit être renseigné
CATEGORY		
LifeBoatCat	varchar	valeurs <i>standard</i> , <i>secours</i> ou <i>radeau</i>
Structure	varchar	doit être renseigné, valeurs <i>bois</i> ou <i>bois et toile</i>
Places	int	doit être renseigné
LIFEBOAT		
LifeBoatId	varchar	
Side	varchar	doit être renseigné, valeurs <i>babord</i> ou <i>tribord</i>
Position	varchar	doit être renseigné, valeurs <i>avant</i> ou <i>arriere</i>
Location	varchar	doit être renseigné, valeur par défaut <i>pont</i>
Launching_Time	Time	doit être renseigné
RECOVERY		
Recovery_Time	Time	doit être renseigné
RESCUE		
PassengerId	int	
LifeBoatId	varchar	doit être renseigné

## S1.04 – Naufrage du Titanic – ETAPE 2

### OBJECTIFS :

- ✓ Création sous PostgreSQL de votre propre base de données relative au naufrage du Titanic: **implémentation du SLR**
- ✓ Écriture de **tests de conformité** avec les contraintes devant être respectées
- ✓ **Peuplement** de votre base à partir de fichiers csv fournis et premières recherches d'informations

### PRELIMINAIRES

DANS LE PARCOURS DE CETTE ETAPE, NOUS VOUS FOURNISSONS

- LE SEA MODELISANT LA BD A IMPLEMENTER : **SEA\_BD\_TITANIC**
- LE SCHEMA RELATIONNEL (SLR) DERIVE DU SEA : **SLR\_BD\_TITANIC**
- UN FICHIER EXEMPLE, POUR LA JUSTIFICATION DES RELATIONS DU SCHEMA RELATIONNEL : **EX\_JUSTIFICATION\_SLR**

DANS VOTRE REPERTOIRE S1.04 :

- Créez un répertoire **Etape\_2** et placez-vous dans ce répertoire
- Téléchargez les fichiers SEA\_BD\_TITANIC, SLR\_BD\_TITANIC, EX\_JUSTIFICATION\_SLR et peuplement.sql disponibles sur Chamilo (SAE1.04 > Etape 2) et placez ces 4 fichiers dans votre répertoire personnel S1.04/Etape\_2.

### A. JUSTIFICATION DU SCHEMA RELATIONNEL DE LA BD TITANIC ET IMPLEMENTATION SOUS POSTGRESQL

#### A1. Justification du schéma relationnel de la BD

##### RAPPELS

- Votre étude est centrée sur le naufrage du Titanic et s'intéresse exclusivement aux passagers de 1<sup>ère</sup>, 2<sup>ème</sup> et 3<sup>ème</sup> classe (elle pourra être étendue plus tard)
  - La reconstitution des familles des passagers n'est pas dans la portée de l'étude, de même que l'influence de la localisation de leurs cabines sur la survie des passagers
  - Les données à prendre en compte étaient toutes présentes (*mais non organisées de façon exploitable*) dans le fichier **Data\_Titanic\_S1\_04.ods** que vous avez récupéré lors de la 1<sup>ère</sup> étape
- Relisez les règles de passage d'un SEA au schéma relationnel d'une BD à implémenter (cf. cours 3)
  - Étudiez attentivement le SEA qui vous est proposé (**SEA\_BD\_TITANIC**) et vérifiez que le schéma relationnel qui vous est proposé (**SLR\_BD\_TITANIC**) est cohérent avec ce SEA
  - **Dans le répertoire Etape\_2 du chef de projet :**
    - ✓ Créez un document LibreOffice\_Writer et enregistrez-le sous le nom **etape2 suivi du login du chef de projet** (exemple : **etape2\_barbapapa**)
    - ✓ Dans ce document, vous justifierez pour chaque relation du SLR\_BD\_TITANIC :
      - l'origine de chacun de ses attributs
      - le choix de sa clé primaire
      - la raison pour laquelle un de ses attributs, ou un sous-ensemble de ses attributs, est clé étrangère et à quelle clé primaire d'une autre relation cet attribut, ou sous-ensemble d'attributs, fait référence
- Vous suivrez pour cela l'exemple donné dans le fichier **EX\_JUSTIFICATION\_SLR**

#### A2. Implémentation sous PostgreSQL

##### 1. Expression des contraintes :

Complétez votre document **etape2\_Login** avec, pour chaque relation de la BD, l'écriture SQL des contraintes qui doivent être vérifiées (contraintes *référentielles*, *d'attribut*, ou *de relation* – cf. **SLR\_BD\_TITANIC**)

##### 2. Fichiers de création et suppression des relations de la BD – **Répertoire Etape\_2 du chef de projet**

Créez et complétez 2 fichiers :

- Création des relations de la BD : fichier **create\_login.sql** (exemple : **etape2\_barbapapa.sql**)
- Suppression des relations de la BD : fichier **drop\_login.sql**

**IMPORTANT** : Les instructions de création de chaque relation de votre BD **titanic** doivent impérativement respecter l'ordre des attributs, tel que défini dans le SLR

### 3. Implémentation du schéma de la BD – base du chef de projet

- ✓ Connectez-vous à la base PostgreSQL du chef de projet
- ✓ Nettoyez le contenu de cette base (il ne doit subsister aucune relation)
- ✓ Exécutez le fichier **create\_login.sql** (\i create\_login.sql)  
En cas d'erreur, détruisez les relations créées (\i drop\_login.sql)

## B. TESTS DE CONFORMITE DES RELATIONS DE LA BD

### B1. Création d'un fichier de test des contraintes exprimées en A2

- Dans le répertoire **Etape\_2** du chef de projet, créez un nouveau fichier : **testconfo\_login.sql**  
Dans ce fichier :
    - pour chaque contrainte exprimée en A2, donnez un exemple de requête d'insertion qui ne fonctionne pas car la contrainte en question n'est pas respectée. Incluez aussi en commentaire dans le testconfo\_login.sql, le résultat de l'exécution de la requête et votre explication en français sur la raison pour laquelle elle ne fonctionne pas. A titre d'exemple, voir TP3 partie 1 test.sql
- INDICATION : les noms des passagers utilisés dans certains tests devront être simples (ex : 'Passager 1')

## C. PEUPLEMENT DES RELATIONS DE LA BD ET PREMIERES REQUETES

### C1. Peuplement de votre BD **titanic**

- Dans la **base du chef de projet** :
  1. Détruisez, puis recréez le schéma de votre BD
  2. Exécutez le fichier **peuplement.sql** que vous avez copié au tout début de cette étape (cf. page 1)  
**NOTE** : aucune erreur ne doit se produire !
- Affichez le nombre de lignes de chaque relation et vérifiez que vous obtenez bien les valeurs suivantes :

PORT	PASSENGER	OCCUPATION	SERVICE	CATEGORY	LIFEBOAT	RECOVERY	RESCUE
3	1309	349	40	3	20	17	490

## D. PREMIERES REQUETES : RECHERCHE DE DONNEES PARMI CELLES DU FICHIER **Data\_Titanic\_S1\_04.ods**

- Dans le répertoire **Etape\_2** du chef de projet, créez un fichier **testinfos\_login.sql**
- Écrivez dans ce fichier des requêtes permettant de vérifier :
  - ✓ les informations relatives au passager n°916 : son nom, son âge, sa classe, sa ou ses cabines, le nom du port où il a embarqué, numéro et catégorie de l'embarcation de sauvetage qui l'a éventuellement secouru
  - ✓ si le nom et le rôle des domestiques du passager n°1264 est exact
  - ✓ si la liste des passagers ayant été secourus par le canot n°7 est exacte

### RENDUS CONCERNANT CETTE ETAPE – 21 DECEMBRE 2023

- \* votre document **etape2\_login** au format pdf, portant en bas de la page de garde, le nom et le prénom des membres de votre groupe projet.
- \* les fichiers **create\_login.sql**, **drop\_login.sql**, **testconfo\_login.sql** et **testinfos\_login.sql**

Ces documents sont à glisser et déposer sur Chamilo dans le répertoire concernant **votre groupe TP** et **l'étape 2** du projet, situé dans le **Travaux** du parcours R1.05