TCH054 — SÉANCE 2 PROCESSUS DE CONCEPTION LE MODÈLE CONCEPTUEL

Anis Boubaker, Ph.D. Maître d'enseignement École de Technologie Supérieure

© Anis Boubaker (2019)

PLAN DE LA SÉANCE

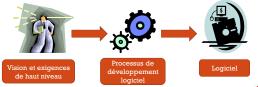
- Conception d'une base de données
- Modélisation du modèle conceptuel (modèle de classes UML)
 - Les classes et les attributs
- Les associations
- L'héritage

² ÉTS



CYCLE DE VIE D'INGÉNIERIE LOGICIELLE

 Processus qui consiste à transformer les exigences utilisateur en logiciel.



4 ÉTS

CYCLE DE VIE D'INGÉNIERIE LOGICIELLE

• Le cycle de vie se subdivise en plusieurs étapes (pas toujours linéairement)

Faisabilité

Analyse

Conception

Implantation

Tests

Déploiement et exploitation

CONCEPTION DE LA BASE DE DONNÉES

- La conception de la base de données fait partie intégrante du processus d'analyse et de conception logicielle.
- Une activité qui vise à:
 - Identifier les données à faire persister
 - Structurer les données sous forme d'entités et d'attributs
 - Identifier les relations entre ces entités.

· ÉTS

PR	OCESSUS	DE	CONC	EP	TION	D'UNE	BASE
DF.	DONNÉES	R	rt.am	ſΝ	NELL	F.	

- Plusieurs étapes visant à implanter une base de données dans un SGBD, qui sera utilisée par le logiciel;
- Plusieurs artefacts produits, processus itératif.
- · Chaque artefact découle du précédent



ı ÉTS

MODÈLE...

- Un modèle: Une représentation de la réalité faisant abstraction de tous les détails non-pertinents;
- Un modèle adopte généralement un point de vue: l'angle sous lequel on s'intéresse à l'objet à modéliser.

Exemples:

- Le plan d'architecte est un modèle qui s'intéresse aux aspects structuraux d'une bâtisse;
- Un schéma électronique représente les composantes d'un circuit électronique et leurs liaisons.
- Peut être très détaillé et concret: le code source d'un programme est un modèle!

e ÉTS

MODÉLISATION...

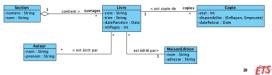
- Modélisation: Action visant à représenter la réalité à travers des modèles.
- La modélisation permet notamment:
 - De communiquer : plus facile qu'avec des mots
 - De formaliser : en utilisant des formalismes établis (ex.: UML)
 - D'abstraire: ne s'intéresser qu'aux détails pertinents de l'angle considéré
 - D'itérer : transformer un modèle abstrait à un modèle plus concret

· ÉTS

ÉTAPE 1: LE MODÈLE CONCEPTUEL DE DONNÉES (UML)



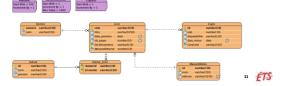
- Analyser, définir et structurer les données du problème d'affaires
- Indépendant de la base de données (et des technologies/langages)



ÉTAPE 2: LE MODÈLE RELATIONNEL (ERD)



• Compléter le modèle conceptuel avec des détails spécifiques à la persistance (clés, tables de jointure, ...)



ÉTAPE 3: LE MODÈLE PHYSIQUE (SQL)



- Constitué par les requêtes de définition de données (LDD), découle du modèle relationnel
- Détails additionnels spécifiques au SGBD choisi (dialecte SQL)

CREATE TABLE SECT	ION	
(numero	VARCHAR2(10) PRIMARY KEY,	
non	VARCHAR2(50) NOT NULL);	
CREATE TABLE MAIS		
(id	INTEGER PRIMARY KEY,	
non	VARCHAR2(50) NOT NULL,	
adresse	VARCHAR2(250));	
CREATE TABLE LIVE	E	
(cote	VARCHAR2(10) PRIMARY KEY,	
titre	VARCHAR2(50) NOT NULL,	
date parution	DATE,	
nb pages	INTEGER.	-Comp
	d INTEGER REFERENCES MAISON_EDITION(id),	. 12 E13
		. 12 <i>ETS</i>

LE MODÈLE CONCEPTUEL DE DONNÉES



OBJECTIFS

- En partant d'une description d'un problème:
- Identifier les données du problème à faire persister
- Structurer les données:
- Les regrouper en classes
- Établir les relations entre les classes



CLASSE

- Décrit les caractéristiques d'un ensemble de données de même nature
- Exemple: La classe "Livre" décrit ce qui caractérise un livre, applicable à tous les livres de la bibliothèque.
- Représentée en UML par un rectangle subdivisé en soussections – La première précise le nom de la classe.

Nom de la classe	Livre
Attributs 🔫	-cote: String -titre: String -dateParution: Date -nbPages: int -isbn: String

CLASSE — CONVENTIONS DE NOMMAGE	
• Le nom d'un classe doit être représentatif.	
• Le nom doit commencer par une majuscule. Exemples: Etudiant Livre	
Si le nom est composé, on utilise la notation CamelCase: Exemples: MaisonEdition	
16 ÉTS	
ATTRIBUTS	
• Un attribut est une donnée qui caractérise les éléments d'une classe.	
Dans la classe livre, chaque livre est caractérisé par les attributs : cote, titre, dateParution et nbPages. Livre -cote : String -titre : String -titre : String	
Chaque occurrence (instance) de la classe aura des valeurs d'attributs qui lui sont - Chaque occurrence (instance) de la classe aura des valeurs d'attributs qui lui sont - String	
propres (par forcément uniques!)	
п 🚜	
ATTRIBUTS - CONVENTION DE NOMMAGE	
• Le nom de chaque attribut doit être représentatif de la donnée.	
• Même syntaxe que les variables (en C ou en Java)	
• Le nom doit être unique <u>au sein d'une classe</u>	
 Le nom commence par une minuscule. On utilise la convention CamelCase. Exemples: adresseLivraison dateNaissance 	
• Le nom de l'attribut est suivi de son type (int, String, double, etc.)	
Exemples: nbPages: int dateNaissance: Date 18 ETS	

INSTANCE							
 On appelle ins d'une classe. 	stance d'une	classe , un éléi	ment spécifique				
Par exemple, pour la classe "Livre", le livre ayant les valeurs d'attributs suivantes est une instance de la classe: Cote: "AB-100a" Titre: "Le Seigneur des anneaux" dateParution: 1984-01-01 nbPages: 140 isbn: "978-2266232999"							
				19 ÉTS			
INSTANCE			Etudiant -codePerm: String -nom: String -prenom: String -dateNaissance: I				
Exemple : Insta	ances de la cla	asse Etudiant .					
CodePerm	Nom	Prénom	DateNaiss				
SNOJ1982102101	Snow	Jon	1982-10-21				
STAA1987071509	Stark	Arya	1987-01-10				
TARD1979071502	Targaryen	Daenerys	1979-07-15				
GRET1981091501	Greyjoy	Theon	1981-09-15				
LANT1970061203	Lannister	Tyrion	1970-06-12				
				20 ÉTS			
IDENTIFIC!	IDENTIFICATION DES CLASSES/ATTRIBUTS						
La modélisation conceptuelle commence par identifier les informations à traiter à partir de la description du problème : → Identification des classes et de leurs attributs							

21 ÉTS

 Pour identifier les classes/attributs à partir de la description du problème, on a recours à plusieurs approches, notamment:
 Analyser les formes nominales de la descriptions du problème

• Identifier les entités tangibles (ex. Livre) du domaine d'application, les rôles (ex.: Adhérent), les événements (ex.: Emprunt), les interactions (ex. Rendez-vous), les localisations (ex.: Aile, Département).

IDENTIFICATION DES CLASSES/ATTRIBUTS		
IDENTIFICATION DES CEMBERS/ ATTAIDOTS		
• Écarter les concepts identifiés qui sont:		
En dehors du périmètre du logiciel		
Réfèrent au système au complet		
Dupliquent d'autres concepts		
Sont trop vagues (beaucoup trop d'instances) ou trop spécifiques (trop peu d'instances)		
22	ÉTS	
EXERCICE		
EVEUCICE		
 Une compagnie de transport dispose d'un parc automobile composé de camions de diverses capacités (en tonne) identifiés par leur plaque. 		
 La société offre ses services à des clients qui lui requièrent le transport de marchandises d'une adresse de départ à une adresse d'arrivée. 		
 La compagnie conserve les nom, prénom et numéros de téléphone de ses clients. 		
 Le transport est facturé au client selon la distance parcourue, le poids transporté et le volume. 		
→ Identifiez les classes du problème ainsi que leurs		
attributs	ÉTS	
	Lis	
RELATIONS ENTRE LES CLASSES		
 La seconde étape de la modélisation conceptuelle Deux types de relations: 		
• Les associations		
• Les relations de généralisation / spécialisation (héritage)		
(Note: d'autres types de relations existent mais ne seront pas abordées dans le cadre de ce cours).		
24	ÉTS	

LES ASSOCIATIONS	
 Une association entre deux classes définit une relation entre les instances de chacune des classes. 	
• Chaque livre, est associé à l'auteur (ou les auteurs) qui l'a écrit.	
Auteur -nom: String -prenom: S	
25 ÉTS	
ASSOCIATION - SYNTAXE	
ClasseA cardinalité de A pour B nom associaton (verbe) role de B pour A role de A pour B Cardinalité de B pour A	
• Exemple :	
Auteur -nom : String -prenom : String -trenom -t	
≈ £TS	
NOM DE L'ASSOCIATION	
ClasseA cardinalité de A pour B nom associaton (verbe) role de B pour A ClasseB role de A pour B Cardinalité de B pour A	
• Nom de l'association: Explique pourquoi un lien entre A et B existe.	
 • Utilise une forme verbale • En cas d'ambiguïté, on utilise > pour préciser le sens de 	
Auteur from : String sentom :	
• Un Auteur est associé aux livres qu'il a é crit .	

CARDINALITÉ

ClasseA	cardinalité de A pour B	nom associaton (verbe)	role de B pour A	ClasseB
		nom associatori (verbe)		
	role de A pour B		Cardinalité de B pour A	

- Cardinalités: À combien d'instances de B est-ce qu'on associe une instance donnée de A?
- Définie sous la forme: min..max
- Le min ou le max peuvent être omis, selon les cas.
- Exemples:
 - 1..10 : de 1 à 10
 - 3..* : un minimum de 3 et un maximum indéfini
 - *: équivalent à 0..*
 - 1: équivalent à 1..1 (exactement 1)



CARDINALITÉ - EXEMPLE



- Un auteur donné est associé à 0 ou plusieurs (*) Livres
- Un Livre donné peut-être associé à au moins un Auteur, et peut avoir plus d'un auteur (1..*)



RÔLE



- Rôle: Que représente, pour l'instance de A, l'instance de B qui lui est associée?
- Exemple:



- Un auteur est associé à des livres qui représentent ses ouvrages
- Un livre est associé à un ou plusieurs auteurs, qui sont ses auteurs,

TYPES D'ASSOCIATIONS - BINAIRE

- Nous considérons les associations binaire et unaires (les ternaires existent)
- Exemple d'association binaire (cas typique):



31 ÉTS

TYPES D'ASSOCIATIONS - UNAIRE

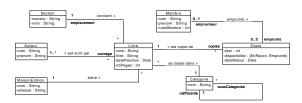
• Exemple d'association unaire (ou réflexive)



- Une personne donnée est associée à au plus 2 personnes, qui représentent ses parents
- Une personne donnée est associée à zéro ou plusieurs personnes qui représentent ses enfants.
- Dans une relation unaire, il est primordial de spécifier les rôles de l'association.

32 ÉTS

EXEMPLE - GESTION DE LIBRAIRIE



TW:	ri T	'n	7/	٠,	r
F.X	ı.ı	(l.	и		ŀ

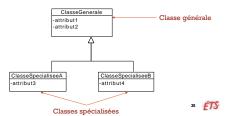


- Reprendre l'exercice précédent et le compléter en y ajoutant:
 - · Les associations entre les différentes classes
 - Les cardinalités et les rôles pour chaque association

34 ÉTS

GÉNÉRALISATION

 La relation de généralisation décrit la relation entre une classe générale est un/des classe(s) qui représentent des cas particuliers – les sous classes.



GÉNÉRALISATION - EXEMPLE

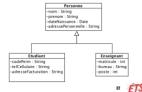
 Un Animal peut-être soit un Oiseau, un Mammifère ou un Reptile



- Toute instance d'Oiseau est également une instance d'Animal
- Toute instance d'une classe spécialisée est aussi une instance de la classe générale.
- La relation de généralisation s'appelle aussi relation de spécialisation...

GÉNÉRALISATION - HÉRITAGE

- La relation de généralisation s'appelle aussi relation
- Héritage: Les classes spécialisées « héritent » des attributs de la classe générale.
- Les classes Etudiant et Enseignant héritent des attributs de Personne.



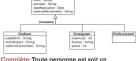
TYPES DE GÉNÉRALISATION

- Une généralisation peut-être:
 - Complète ou Incomplète: Elle est complète si toute instance est forcément une instance d'une des classes spécialisées.



Incomplète: Certains animaux ne sont pas des oiseaux, des mammifère ou des reptiles (ex.: ver de terre)

> Ils sont des instances de Animal uniquement.



Complète: Toute personne est soit un étudiant, un enseignant ou un

professionnel.

→ Il ne peut y avoir d'instances de
Personne uniquement (classe abstraite)

38

FTS

TYPES DE GÉNÉRALISATION

- Une généralisation peut-être:
 - Disjointe ou Chevauchante: Elle est disjointe si une instance ne peut faire partie de plus d'une classe spécialisée.



Disjointe: Un animal ne peut être à la fois (ex.) un Oiseau et un Mammifère.



Note: Ceci n'est pas possibles dans les langages OO (on utilise la délégation) 39

EXERCICE	
Trouvez quatre exemples de généralisations	
Madálian an gáglastina an másifiant laura tuna	
 Modéliser ces généralisations en spécifiant leurs types (complète/incomplète, disjointe/chevauchante). 	
• <i>ÉTS</i>	
PROCESSUS DE CONCEPTION D'UNE BASE	
DE DONNÉES RELATIONNELLE	
Modèle de conceptuel de données (ou relationnel) (SQL)	
a ÉTS	
MEANS.	