



FPA applied to UML/Use Cases

Adri Timp

Chair, NESMA Counting Practices Committee

Adri.Timp@nesma.nl

ISMA-3

September 18, 2008

FPA applied to UML/Use Cases

SIG proposal

Approach

UML and FPA

Brief explanation of the guidelines

Approach

Goal SIG (Special Interest Group)

- Application of FPA guidelines to UML/Use Cases, *no* Use Case Points
- Starting point: Acquainted with both FPA and UML

Process

- Literature study: One basic knowledge level for SIG members
- Formulate draft version
- Review by FPA and UML experts of leading companies: Getronics, Pink Roccade, Sogeti, Equens, ABN AMRO, QSM, Atos Origin, and CapGemini
- Publication

FPA and UML / Use Cases

- FPA sizes functionality based on user view of application
- So independent of:
 - Technology used for implementation
 - Analysis / Design methodologies and models
- Consequence:
 - FPA also applicable in UML designed application
 - Just different terminology → “translation” to FPA concepts
 - Many people don’t realize → use case points (and other)
 - For experienced people “open doors”

Requirements to carry out FPA

Requirement	Objective for FPA	
Model that describes Structure/Data of application (Data Model)	To identify the data functions (ILF, EIF)	Structure
Data element types and record types in Data Model	To determine complexity of the data functions	
Model that describes behaviour of application	To identify the transactional functions (EI, EO, EQ)	Behaviour
The flow of data element types and the processing logic	To determine complexity of the transactional functions	

UML and FPA

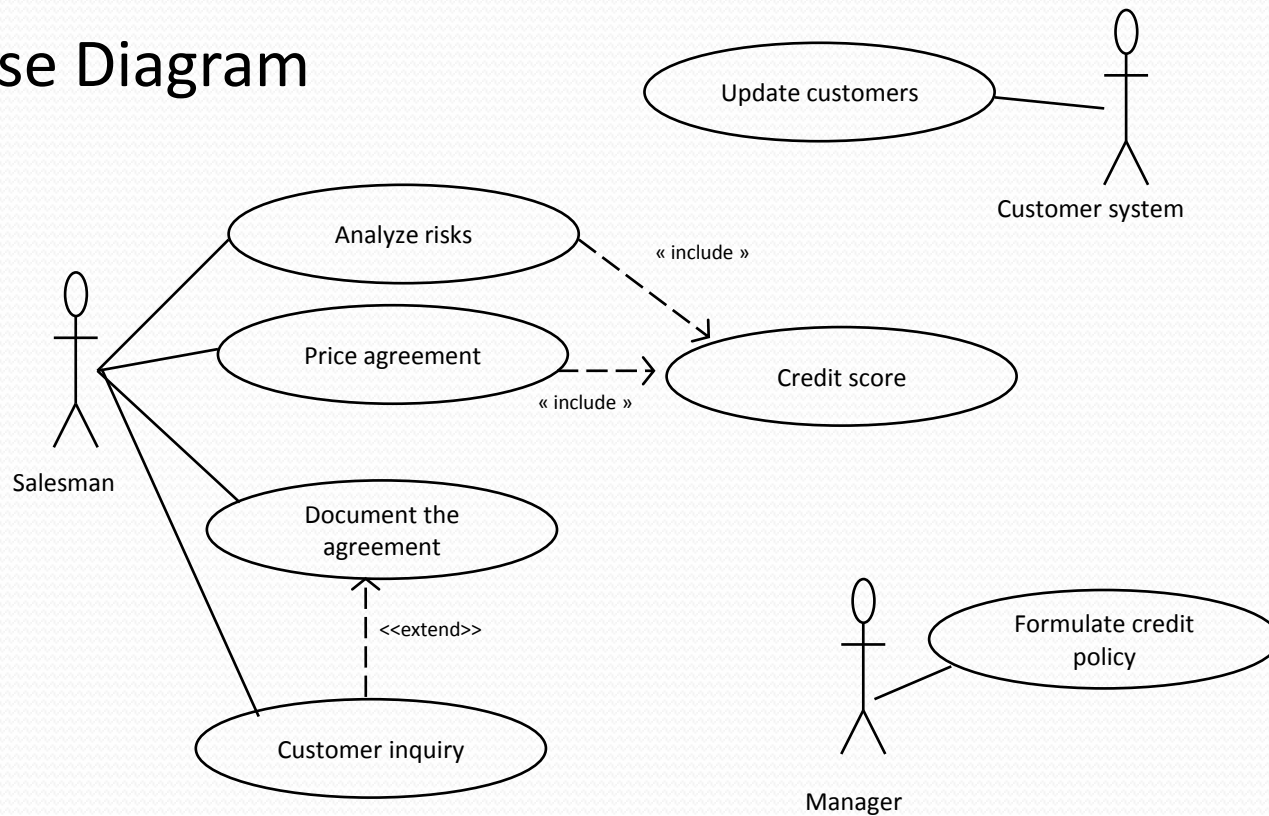
Behaviour or structure?	UML Diagram	Describes
Behaviour	Use case diagram	Actors, system and functions
	Use case description	Actions in a Use case
	Activity diagram	Actions in a scenario
	Interaction Overview diagram	Actions in a scenario
	Sequence diagram	Actions in a scenario
	Collaboration diagram	Actions in a scenario
Structure	Class diagram	Detailed information need

In practice almost always available:

- Use Case Model (both diagrams and descriptions)
- Class Model

Use Cases

Use Case Diagram



Use Case and Actor

- UML definition of “Use Case”:
“The specification of a set of actions performed by a system, which yields an observable result that is, typically, of value for one or more actors or other stakeholders of the system.”
- Use cases describe the requirements for an application from the point of view of the so-called “actor”
- Use cases represent the behaviour of the system as perceived by the actors
- UML “Actor” = FPA “User”
- UML “Use Case” = FPA “Elementary Process”?

Use Case=Elementary Process?

- Yes and No
- Use Cases are well suited for counting function points (user view; observable result/self contained)
- But
 - Level of detail (“granularity”) of uses cases widely vary
 - UML does not give a definite answer for the level of detail for a use case
- So
 - Additional analysis is always a must
 - Automated “counting” not possible

Analysis of Use Case A Must!

- Decomposition needed

For example:

- 1 Use Case “Maintain Customer”

Analysis might show there are 4 EP’s:

Customer Inquiry & Add, Change, Delete Customer

- Composition Needed

For example:

- 5 Use Cases “Validate Customer”, “Validate Article”, “Add Orderline”, “Calculate Shipping Date”, “Confirm Order”
Analysis might show that this is 1 EP: “Place an Order”

Use Case Scenarios

- Use Case Description may contain several “Scenarios”
- A Scenario is a “series of steps that describe the interaction between an actor and an application”
- All scenarios for a use case have in common that they support the same basic objective for a user
 - “Happy Flow”: everything goes well
 - Alternate Flows: describe what can go wrong and how a user can achieve his goal in a different way
 - handling an error situation: typically no extra EP
 - additional steps for actor: could be extra EP if self contained

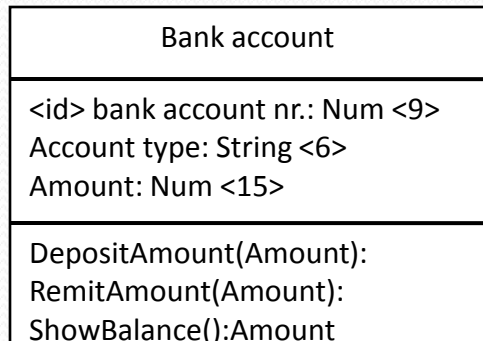
Include Relationship

- Include Relationship: When a specific type of behaviour occurs in several Uses Cases, but is described once as separate Use Case
- Include Use Case can be:
 - Not a separate EP if it is not a self contained process, for example a validation that is documented separately
 - An EP if it does meet the criteria for an EP for example a drop down list in which the contents of an ILF is presented to make a selection
- Magic Word again: Analysis!

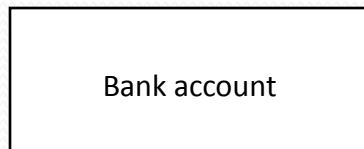
Summary Use Cases

- Analyse Granularity
- Analyse Scenarios
- Analyse “Include” Use Cases
- Analyse “Storyboards”

Class Model



Or



Class/Object relationships

- Association
- Aggregation
- Composition
- Generalization


Objects and Classes

- How to identify Data Functions based on the OO Class Model
- Class: the definition of the attributes and the operations (“methods”) of similar Objects
- Object: concrete or abstract “ things” from the real world
- Sounds like Entity (Occurrence) and Entity Type
- Approach: Analyse the Class Model using the FPA rules and guidelines for identifying Logical Files in an Entity (Relationship) Model


Guidelines




Exclude the Code Data



Exclude Association Classes without attributes:
("Key-Key entities"): do not occur

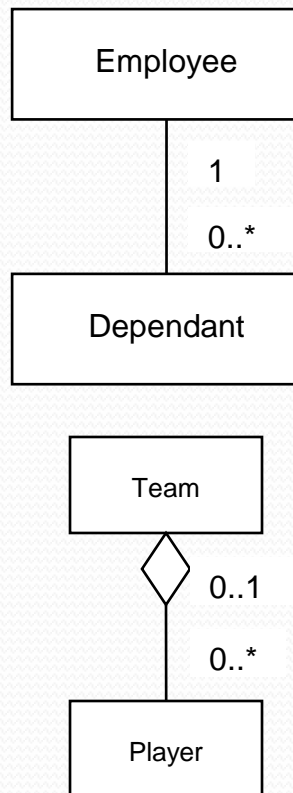


Include Association Classes with attributes
("Key-Key with attributes")



Analyse type and optionality of the relationship
(dependent/independent) and group

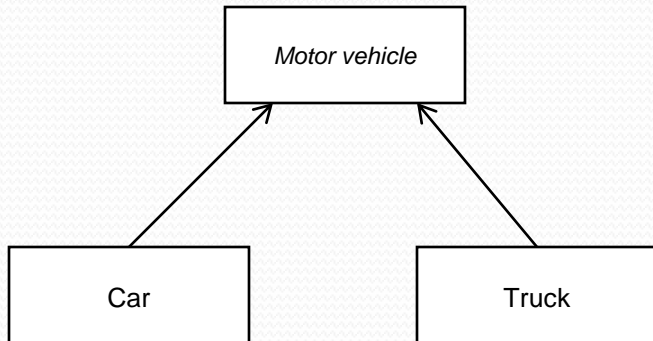
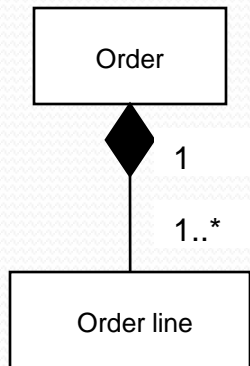
Guidelines



Association and Aggregation:

- Mutually a 1 → 1 Logical File
- Mutually a 0 → 2 Logical Files
- A 0 on one side? → “delete rule”
(class dependence/independence)

Guidelines



Composition

- Together always 1 logical file

Generalization

- Separate “discrete” items? Then 2 Logical files

FPA applied to UML/Use Cases Guide

Guide “FPA applied to UML/Use Cases”

www.nesma.nl/english

(Free download)

office@nesma.nl

Adri.Timp@nesma.nl