## **Example:**

# **Operative and Tactical Marketing Architecture**

#### Ideal data model

- Definition of Sorts
- <u>Definition of connections</u>

#### **Definition of Sorts**

Name of Sort: business action

Type: Kernel

An object from the category (#business action) is every action the aim of which is to transfer rights to business into a contract, or to identify, negotiate and create a new rights to business.

Business action examples:

- tender participation
- phone call approach to a partner or customer
- meeting
- conference presentation
- particular distributor deal on a new solution distribution
- business oriented paper on network security monitoring

Examples of what is not considered to be a business action in this context:

- market probe
- new customer categorization creation
- design of a new solution for given needs fulfilling

Name of Sort: business action categorization

Type: Kernel

An object of the category (#business action categorization) is each aiming segmentation of a set of business actions expressing in a given context a necessary classification.

See connection to Diam4.Context.

Name of Sort: business action category

**Type:** Kernel

An object from the category (#business action category) is each purposely selected container (class) joining a subset of business actions by chosen criterion which is given by appropriate categorization.

See Diam4.Model

Name of Sort: complex solutions producer

Type: Kernel

**Super-Type: subject** 

An object from the category (#complex solutions producer) is each subject which composes several products and solutions into a complex product. This product the complex solutions producers provides/supplies to its customers. Particular solutions or products embedde in this complex product are invisible by their brands to end-customers.

Name of Sort: customer

**Type:** Kernel

An object from the category (#customer) is every end customer who is or can be a consumer or user of our services or products. It is an agent acting on the market, which is autonomous and co-operative in a sense of holonic approach, to which attention can be reasonably payed.

See Diam4.Agent

Name of Sort: customer categorization

Type: Kernel

An object of the category (#customer categorization) is each aiming segmentation of a set of customers expressing in a given context a necessary classification.

See connection to Diam4.Context.

Name of Sort: customer category

**Type:** Kernel

An object from the category (#customer category) is each purposely selected container (class) joining a subset of customers by chosen criterion which is given by appropriate categorization.

See Diam4.Model

Name of Sort: customer pain/need

**Type:** Kernel

An object from the category (#customer pain/need) is every formulation of a required state-of-things more or less definite. When expressed positively (as desired) it is a need, when expressed negatively (as undesirable) it is a pain. Often it is expressed using requirements or absence of requirements. Requirement is every singular documented need of what a particular product or service should be or should do. It is a statement that identifies a necessary attribute, capability, characteristic, or quality of a system in order for it to have value and utility to a user.

See Diam4.Goal and Diam4.Requirement.

Name of Sort: dealer

**Type:** Kernel

**Super-Type:** subject

An object from the category (#dealer) is every agent who sells or mediates to a customer our products or services.

Name of Sort: distributor

**Type:** Kernel

**Super-Type:** subject

An object from the category (#distributor) is every agent who creates a network of dealers to whom he/she/it sells our solutions.

Name of Sort: functionality

**Type:** Kernel

An object of the category (#functionality) is each particular way of requirement(s) satisfaction.

See Diam4.Requirement

Name of Sort: negotiated deal

**Type:** Associative

An object from the category (#negotiated deal) is each representation of a relationship between (#solution), (#customer), (#customer pain/need) and (#subject) with the meaning: results (text description) achieved by given (#subject) while using given (#solution) for curing given (#customer pain/need) by a sale to given (#customer) / 0,1:0,M

Name of Sort: revenue model

**Type:** Kernel

An object from the category (#revenue model) is each model which specifies components of the income-flow and their mutual composition and a way of this income-flow realization in a time period.

See Diam4.Model

Name of Sort: revenue model application

**Type:** Associative

An object from the category (#revenue model application) is each representation of a relationship between (#solution), (#customer category) and (#revenue model) with the meaning:

particular use (text description) of given (#revenue model) when it is applied for given (#customer category) within a sale of given (#solution) / 0,1:0,M

Name of Sort: rights to business

**Type:** Associative

An object from the category (#rights to business) is each representation of a relationship between (#solution), (#customer category), (#customer pain/need) and (#subject) with the meaning:

conditions (text description) undertaking given (#subject) while he/she/it offers given (#solution) for curing given (#customer pain/need) to given (#customer category) / 0,1:0,M

Name of Sort: solution

**Type:** Kernel

An object from the category (#solution) is a complex answer to a need/pain expressed in terms of accomplishment of appropriate requirements arising from a registered need/pain. According to Wikipedia, solution is a product, service, or combination of both which is said to solve a business or consumer's problem.

See Diam4.Service.

Name of Sort: solution categorization

**Type:** Kernel

An object of the category (#solution categorization) is each aiming segmentation of a set of solutions expressing in a given context a necessary classification.

See connection to Diam4. Context.

Name of Sort: solution category

**Type:** Kernel

An object from the category (#solution category) is each purposely selected container (class)

joining a subset of solutions by chosen criterion which is given by appropriate categorization.

See Diam4.Model

Name of Sort: subject

**Type:** Kernel

An object from the category (#subject) is every agent acting on the market, which is autonomous and co-operative in a sense of holonic approach and who is or can be our partner in business or a competitor. Subjects are currently distinguished into dealers, distributors, system integrators and complex solutions producers. Distinguishing between partners and competitors is orthogonal to the above sorting and is recorded by special subject categories within appropriate subject categorization.

See Diam4.Agent

Name of Sort: subject categorization

**Type:** Kernel

An object of the category (#subject categorization) is each aiming segmentation of a set of subjects expressing in a given context a necessary classification.

See connection to Diam4.Context.

Name of Sort: subject category

**Type:** Kernel

An object from the category (#subject category) is each purposely selected container (class) joining a subset of subjects by chosen criterion which is given by appropriate categorization.

See Diam4.Model

Name of Sort: system integrator

**Type:** Kernel

**Super-Type:** subject

An object from the category (#system integrator) is each subject which integrates several products and solutions into a complex aggregate. This aggregate the system integrator provides/supplies to its customers. Particular solutions or products integrated in this aggregate are visible by their brands to end-customers.

#### **Definition of Connections**

Name of Connection: 1:worrying

Cardinality: customer category -> customer pain/need: 0,M, reversely: 0,M

Connection between entities <u>customer category</u> and <u>customer pain/need</u> with the following meaning

(#customer pain/need)-s worrying given (#customer category) / 0,M:0,M

Name of Connection: 2: curing

Cardinality: customer pain/need -> solution: 0,M, reversely: 0,M

Connection between entities <u>customer pain/need</u> and <u>solution</u> with the following meaning (#solution)-s curing given (#customer pain/need) / 0,M:0,M

Name of Connection: 3: subject is actor

Cardinality: business action -> subject: 0,M, reversely: 0,M

Connection between entities <u>business action</u> and <u>subject</u> with the following meaning (#subject)-s which are actors of given (#business action) / 0,M:0,M

Name of Connection: 4: customer category addressed

Cardinality: customer category -> business action: 0,M, reversely: 0,M

Connection between entities <u>customer category</u> and <u>business action</u> with the following meaning

(#customer category)-s addressed by given (#business action) / 0,M:0,M

Name of Connection: 5: addressed

**Cardinality:** business action -> customer: 0,M, reversely: 0,M

Connection between entities <u>business action</u> and <u>customer</u> with the following meaning (#customer)-s addressed by given (#business action) / 0,M:0,M

Name of Connection: 6: topic

Cardinality: customer pain/need -> business action: 0,M, reversely: 0,M

Connection between entities <u>customer pain/need</u> and <u>business action</u> with the following meaning

(#customer pain/need)-s which are topics of given (#business action) / 0,M:0,M

Name of Connection: 7: offered by

Cardinality: business action -> solution: 0,M, reversely: 0,M

Connection between entities <u>business action</u> and <u>solution</u> with the following meaning (#solution)-s offered or presented by given (#business action) / 0,M:0,M

Name of Connection: 8:solution has

Cardinality: functionality -> solution: 0,M, reversely: 0,M

Connection between entities <u>functionality</u> and <u>solution</u> with the following meaning (#functionality)-s which has given (#solution) / 0,M:0,M

Name of Connection: belongs to\_1

**Cardinality:** customer -> customer category: 0,M, reversely: 0,M

Connection between entities <u>customer</u> and <u>customer category</u> with the following meaning (#customer category)-s to which belongs given (#customer) / 0,M:0,M

Name of Connection: belongs to 2

Cardinality: subject -> subject category: 0,M, reversely: 0,M

Connection between entities <u>subject</u> and <u>subject category</u> with the following meaning (#subject)-s which belong to given (#subject category) / 0,M:0,M

Name of Connection: belongs to 3

Cardinality: solution category -> solution: 0,M, reversely: 0,M

Connection between entities <u>solution category</u> and <u>solution</u> with the following meaning (#solution)-s which belong to given (#solution category) / 0,M:0,M

Name of Connection: belongs to\_4

Cardinality: business action category -> business action: 0,M, reversely: 0,M

Connection between entities <u>business action category</u> and <u>business action</u> with the following meaning

(#business action)-s which belong to given (#business action category) / 0,M:0,M

### Name of Connection: of\_1

Cardinality: customer category -> customer categorization: 1,1, reversely: 0,M

Connection between entities <u>customer category</u> and <u>customer categorization</u> with the following meaning

(#customer categorization) in a frame of which given (#customer category) is defined / 1,1:0,M

#### Name of Connection: of 2

Cardinality: subject category -> subject categorization: 1,1, reversely: 0,M

Connection between entities <u>subject category</u> and <u>subject categorization</u> with the following meaning

(#subject categorization) in a frame of which given (#subject category) is defined / 1,1:0,M

#### Name of Connection: of 3

Cardinality: business action categorization -> business action category: 0,M, reversely: 1,1

Connection between entities <u>business action categorization</u> and <u>business action category</u> with the following meaning

(#business action categorization) in a frame of which given (#business action category) is defined / 1,1:0,M

#### Name of Connection: of 4

Cardinality: solution categorization -> solution category: 0,M, reversely: 1,1

Connection between entities <u>solution categorization</u> and <u>solution category</u> with the following meaning

(#solution categorization) in a frame of which given (#solution category) is defined / 1,1:0,M

Name of Connection: subject communicates competency to solve

Cardinality: customer pain/need -> subject: 0,M, reversely: 0,M

Connection between entities <u>customer pain/need</u> and <u>subject</u> with the following meaning

(#customer pain/need)-s which are comunicated by given (#subject) as these he/she/it is able to solve  $/\ 0,M:0,M$ 

Name of Connection: worrying

Cardinality: customer -> customer pain/need: 0,M, reversely: 0,M

Connection between entities <u>customer</u> and <u>customer pain/need</u> with the following meaning

(#customer pain/need)-s worrying given (#customer) / 0,M:0,M