International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

M.1402
(05/2012)

SERIES M: TELECOMMUNICATION MANAGEMENT, INCLUDING TMN AND NETWORK MAINTENANCE

Designations and information exchange

Formalization of data for service management

Recommendation ITU-T M.1402

ITU-T M-SERIES RECOMMENDATIONS

TELECOMMUNICATION MANAGEMENT, INCLUDING TMN AND NETWORK MAINTENANCE

Introduction and general principles of maintenance and maintenance organization	M.10–M.299
International transmission systems	M.300–M.559
International telephone circuits	M.560–M.759
Common channel signalling systems	M.760–M.799
International telegraph systems and phototelegraph transmission	M.800–M.899
International leased group and supergroup links	M.900–M.999
International leased circuits	M.1000–M.1099
Mobile telecommunication systems and services	M.1100–M.1199
International public telephone network	M.1200–M.1299
International data transmission systems	M.1300–M.1399
Designations and information exchange	**M.1400–M.1999**
International transport network	M.2000–M.2999
Telecommunications management network	M.3000–M.3599
Integrated services digital networks	M.3600–M.3999
Common channel signalling systems	M.4000–M.4999

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T M.1402

Formalization of data for service management

Summary

Recommendation ITU-T M.1402 contains definitions of data for service management. The data are defined in the way they are presented to the end users at their terminals, and conceptual or internal data structures are not defined. Therefore, this Recommendation defines an external terminology schema for the given application domain.

This Recommendation focuses on defining data for products, customers, accounts, contracts, deals, addresses, prices, various segments and relationships between all these. Some attributes that are essential for identifying objects and associations between them are defined, as well.

Order data and detailed mappings to network elements are not covered. Also, billing, marketing, sales, retailer, distributor, number management, device management, location management, logistics and finance are not covered. Finally, alarms, trouble tickets, call records and other call centre information are missing.

As the data defined in this Recommendation are designed for human usage, they are independent of the functions in which they are used. Hence, the data may be applied in any functions, e.g., in customer requests, call centres, billing, service platforms, etc.

This Recommendation uses the notation and conventions found in Recommendation ITU-T M.1401, "Formalization of interconnection designations among operators' telecommunication networks".

Appendix I explains the creation of product and customer service databases.

Appendix II explains the creation of interfaces to service platforms.

History

Edition	Recommendation	Approval	Study Group
1.0	ITU-T M.1402	2007-08-06	4
2.0	ITU-T M.1402	2012-05-14	2

Keywords

CRM, customer, data, definitions, product.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at http://www.itu.int/ITU-T/ipr/.

Table of Contents

Introduction

This Recommendation defines data as they are presented to the end users at their terminals. The data are defined in an external terminology schema that provides the glossary and grammar of all permissible data to the end user.

The external terminology schema defines elementary sentence types only, which prescribe elementary sentences like "Contract 1 contains Product instance 12345678". The schema does not prescribe compound sentences, like "Contract 1 contains Product instance 12345678, which has Installation address Storgata 6", as is required at the end-user interfaces. Note that the relative pronoun "which" is not used in external terminology schemata, but may be used in the "Contents schema" for each screen or report.

The proposed external terminology schema is intended for harmonization of data across all sub-domains within service management and all channels to this application domain.

The external terminology schema focuses on end users' need for harmonized data and not on needs for storage or communication formats.

Recommendation ITU-T M.1402

Formalization of data for service management

1   Scope

Service management is the universe of discourse (UoD) of this Recommendation. This UoD comprises products, customers, accounts, contracts, deals, addresses, prices and various segments and relationships between these.

The focus of this Recommendation is on end-user terminology as defined in an external terminology schema and which puts requirements on other schemata and implementations. As such, this Recommendation focuses on the definition of object and reference classes, and provides most identifiers and name bindings (i.e., subordinate object indicated by reversed arrowheads) for these. It also provides some attribute groups and attributes which are essential to define the data structure. Other attributes, e.g., to indicate entity history, are not covered. Also data structures for orders are not covered.

The data definitions are meant to cover the needs for customer inquiries and customer sales about the customer, their services and relations to the operator, but the definitions do not provide background information for sales people and account managers who provide offers and contracts to large customers. These may need processed information about turnover and plans for a customer/cooperation. The data structure may apply for these needs as well, but additional attributes are required. The data structure lacks the means to provide overviews for market analyses. Also, a network view for answering customer inquiries and service delivery is missing.

The data definitions cover product types as they are defined in product catalogues and cover product instances in customer databases. A subset of these product types and product instances may be implemented as telecommunication services, e.g., in a service platform. The broader context of product catalogues and customer databases is essential for definition and harmonization of service types and instances. The object class service is not defined in this Recommendation. However, the data structure defines the scope of service management.

It should be noted that product types within product catalogues are in this Recommendation defined as classes of product instances within contracts. This means that the product types may contain data that are not needed in a pure enterprise product catalogue, but these data are needed for the product catalogue to serve the role of being a schema. One contract instance may have several product catalogue instances as its schemata. One product instance in a contract is, however, only generated from one product type within a product catalogue.

2   References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

[ITU-T M.1401]   Recommendation ITU-T M.1401 (2006), *Formalization of interconnection designations among operators' telecommunication networks*.

3 Terms and Definitions

3.1 Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

3.1.1 Attribute [ITU-T M.1401]

3.1.2 Attribute class [ITU-T M.1401]

3.1.3 Attribute group [ITU-T M.1401]

3.1.4 Identifier attribute [ITU-T M.1401]

3.1.5 Object [ITU-T M.1401]

3.1.6 Object class [ITU-T M.1401]

3.1.7 Population [ITU-T M.1401]

3.1.8 Reference [ITU-T M.1401]

3.1.9 Reference class [ITU-T M.1401]

3.1.10 Schema [ITU-T M.1401]

3.1.11 Subordinate object [ITU-T M.1401]

3.1.12 Subordinate object class [ITU-T M.1401]

3.2 Terms defined in this Recommendation

None.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

CC   Country Code

CRM   Customer Relationship Management

ICC   ITU Carrier Code

KID   Customer IDentifier

LID   Local Identifier

SMS   Short Message Service

UoD   Universe of Discourse

5 Conventions

This Recommendation uses 5 mm bulleted indentations to indicate subordinate data.

Textual definitions and explanations are provided in paragraphs that are indented 5 more mm to the right.

Object classes are underlined; attribute group and attribute classes are not.

References to other object classes are written in blue fonts, in italics and underlined.

The figures use rectangles to indicate object classes.

Lines with reversed arrowheads indicate subordinate object class.

Two-way arrows indicate references between object classes.

A dashed one-way arrow supported with an S at the arrowhead indicates a schema reference from a population. Each element subordinate to a schema may be instantiated to several elements subordinate to the population. And, the schema reference may itself be instantiated, like any other reference.

A dashed one-way arrow supported with a colon (:) at the arrowhead indicates a data type. This is analogous to super-class references in object-oriented languages, where properties of the subordinate class are inherited from the data type.

A reference one level up in the data tree is indicated by a (').

A recursive use of any number of dashes is indicated by (&).

An existential condition is stated by (<>). A data node on the left-hand side of the condition can only exist if the node referenced on the right-hand side exists.

A reference to a data node without referencing its label is stated by a parenthesis ( ), e.g., (Operator) refers to a superior node of Operator, i.e., the Corporation in the graph. However, by not referring to the Corporation, we can reference any node corresponding to the Corporation, i.e., one of its Populations.

Red (combined dashed and dotted) lines indicate derived references.
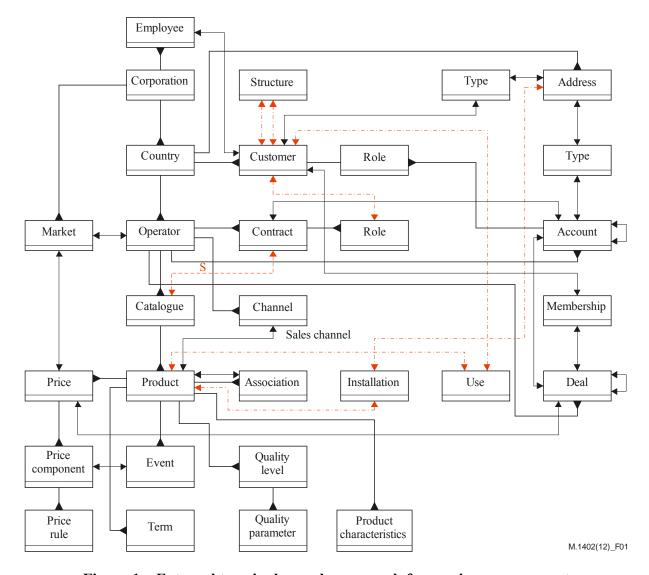
6 External terminology schema

Figure 1 – External terminology schema graph for service management

Corporation

This Recommendation defines a schema of a system of a Corporation. The schema is called Corporation.

The schema contains data classes of the system. Each class acts as an original/prototype that can be copied into (several) instances subordinate to each of the populations of the schema. The S reference from a population to another data item makes the other item act as a schema relative to the population item.

Overlapping subsets of a schema may be implemented as separate systems.

▪ Country

A Country may be an independent state or a state within a union.

Operators are identified with a unique International Carrier Code within a Country. This means that a corporation must be defined with (minimum) one Operator within each Country.

▪ ▪ CC

Country Code is a three-character value according to ISO 3166-1.

- ▪ Name

   The Country Name identifies a Country, dependency, or other area of particular geopolitical interest. Countries are assigned a globally unique Name in ISO 3166-1.

- ▪ Address

   An Address identifies a geographical place. The identifier is independent from application domain, vendor or market segment of its use.

   The Address notion does not cover bank accounts and the like. However, the post box notion is covered by the identifiers. Also, postal addresses given by place and postal code/post area are covered by the identifiers.

   In this Recommendation, Address is treated as one object class only; this requires further study, e.g., separate object classes for streets, local areas, etc.

- ▪ *Installation*

   An Address may have one or more subordinate Installations, which each defines a role of an Installation in this Address.

- ▪ *Type*

   An Address may have one or more subordinate Types, which each indicates a role of a Type of an Account.

- ▪ *Type*

   An Address may have one or more subordinate Types, which indicates a role of a Type of a Customer.

- ▪ Customer

   Customer indicates a legal person whom the Operator relates to. This legal person may have one or more roles to Accounts, Contracts, Product instances, Deals or Employees in or from the Operator. The Role notions allow the Operator to register the Customer relative to these legal notions and not to register the Customer's internal organization structure, which may change without the Operator's knowledge.

   A Customer may be a private person or a legal organization. The Customer's internal organization may indicate family, cohabitant and friendship relations, or indicate organization of geographically distributed companies and collaborative relations. The Operator may not want or be able to record this internal organization of the Customer.

- ▪ KID

   Each Customer of the corporation's Operators shall be assigned a customer identifier (KID) that is unique within the corporation in a Country and is not specific for each business unit/Operator. It shall be possible to undertake credit control of the Customer based on the customer identifier (KID).

- ▪ Name

   Each Customer shall have a Name within a Country. Several Customers may have identical Names. Customers may be distinguished by Address or other information.

- ▪ Business role

   This attribute may have the following value set:

   A – Employee in the corporation

   F – Retailer of the corporation

   D – Distributor of the corporation

   L – Operator in the corporation

   K – Competitor of the corporation

Sources for this information are not identified and responsibility for updating is not assigned in this Recommendation.

▪ ▪ ▪ Customer status

Customer status is provided by the business register within the Country.

K   – valid customer

UK   – (terminated) bankrupt

UF   – (terminated) wrong registration

US   – (terminated) merged

UO   – (terminated) closed

UT   – (terminated) legal action

For persons, the following states are used:

K   – customer

KU   – customer emigrated

KF   – customer disappeared

▪ ▪ ▪ Importance

This attribute indicates that the Customer is a Very Important Person in his relationships to the corporation.

V   – Very Important Person

This Recommendation does not prescribe how this attribute will be maintained, but the attribute may become updated if the Customer gets a Role as Very Important Person to an Account or Contract, and be deleted if the Role as Very Important Person is deleted.

▪ ▪ ▪ *Employee*

A Customer may have one or more subordinate Employees that each is a role of an Employee. This way, it is possible to state that the "customer" is employed by some Operator of some Product instances. However, only employment in the corporation is registered and not by each Operator.

Normally, there is a one-to-one correspondence between Customer and Employee, but in some cases there may be several Employee roles per Customer. This relationship shall not be used to register that an Employee has responsibility for a set of Customers. This information is registered by letting the employee Customer have Roles to several Accounts or Contracts.

▪ ▪ ▪ *Membership*

A Customer may have several subordinate Memberships, which each indicates a role of a Membership, which has a subordinate Deal, which is a role of a Deal.

▪ ▪ ▪ *Role*

A Customer may have several subordinate Roles, which each indicates a role of a Contract's subordinate Role.

▪ ▪ ▪ *Role*

A Customer may have several Roles, which each indicates a role of an Account's subordinate Role.

▪ ▪ ▪ *Subordinate structure*

A Customer may have several Subordinate structures, which each indicates a role of a Structure object, used to define the Customer's internal organization structure.

For a corporate Customer, the Subordinate structure's Subordinate customer indicates an organization unit within the corporation.

For a private person Customer, the Subordinate structure's Subordinate customer indicates a family member, etc.

Sometimes the Subordinate structure is not explicitly registered, but is derived from geographic addresses and relationships to Accounts, Contracts or Product instances, e.g.:

    − When looking upon a Customer, list all subordinate Accounts and Contracts which have usage address at a given address.

    − The terminal user assumes that the Customer with the Role Type Owner has a subordinate unit at the given address.

▪ ▪ ▪ *Superior structure*

A Customer may have a subordinate Superior structure, which indicates a role of a Structure indicating the Customer's organizational association. The Customer's subordinate Superior structure's Superior customer indicates this relationship.

▪ ▪ ▪ *Type*

A Customer may have several subordinate Types, which each indicates a role to a Type (of Address), which has a subordinate Address, which is a role of an Address.

▪ ▪ ▪ *Use*

A Customer may have several subordinate Uses, which each indicates a role of a Use, which has a subordinate Product, which indicates a role of a Product instance.

▪ ▪ Operator

An Operator – i.e., an instance of the data class Operator – defines a business unit for sales and delivery of telecommunication services.
An Operator must always be defined locally to a Country.
The definition of Operators may reflect the organization structure of the corporation.
For more information on Operator, see [ITU-T M.1401].

▪ ▪ ▪ ICC

An Operator is assigned an ITU Carrier Code by the Administration/Regulator in each Country.
For more information on Operator, see [ITU-T M.1401].

▪ ▪ ▪ LID

Every Operator in the corporation may be assigned a unique Local identifier (LID) within the corporation.

▪ ▪ ▪ Billing system

This attribute provides the identification of default Billing system for all Accounts under this Operator.

▪ ▪ ▪ Rating system

This attribute provides the identification of default Rating system for all Contracts under this Operator.

▪ ▪ ▪ Account

An Account provides information on agreed and delivered payments from a Customer.

▪ ▪ ▪ ▪ Account number

The Account number is a unique identifier of an Account within an Operator.

- Payment form

  E – prepaid

  O – postpaid

- Role

  An Account may have several subordinate Roles, which each indicates a relationship to a Customer.

- Category

  The Category indicates which role a Customer plays to an Account. The Category is an attribute group that consists of attributes.

  If the value of the Type attribute is updated by the terminal user, then the value of the Update attribute is set to manual (M). If the role is created by the system itself, then the value of the Update attribute is set to derived (B). Only roles that are updated manually need to be stored.

  - Type

    A Role's subordinate Type may have one of the following values:

    E – owner

    B – payer

    F – billing address

    Y – authorized

    K – contact

    V – very important person

    U – user

    A – provider agent

    R – responsible employee

    An Account may only have one Role of Type owner (E), payer (B), user (U) or provider agent (A), but may have several Roles of Type contact (K) and very important person (V).

    If some Roles are not explicitly provided for an Account, and the Account has one Contract only, then the Roles of the Account are copied from the Roles of the Contract.

    If owner (E) is provided, and payer (B) is missing, then the payer is equal to the owner.

    If owner (E) is provided, and billing address (F) is missing, then the billing address is equal to the owner.

    If owner (E) is provided, and user (U) is missing, the user is equal to the owner.

    If payer (B) is provided, and owner (E) is missing, the owner is equal to payer.

    If payer (B) is provided, and user (U) is missing, the user is equal to payer.

    If user (U) is provided, and owner (E) is missing, the owner is equal to user.

    If user (U) is provided, and payer (B) is missing, the payer is equal to user.

    If some Roles are not provided for a related Contract, then the Roles of the Contract are equal to those of the Account.

Derived Types may be indicated by a separate colour at the user interface, or they may be hidden.

▪ ▪ ▪ ▪ ▪ ▪ Update

The Type of a Role may be updated

M   –   manually

B   –   automatic

▪ ▪ ▪ ▪ ▪ *Customer*

A Role must have a subordinate Customer that is a role of a Customer.

▪ ▪ ▪ ▪ *Address type*

An Account may have a subordinate Address type, which indicates a role to a Type of an Address.

▪ ▪ ▪ ▪ *Contract*

An Account may have a subordinate Contract, which is a role of a Contract, which is billed by this Account.

Note that the Account may be owned by an employee and have a reference to the Contract of this employee. The company in which the person is employed may have an Account that refers to the same Contract. The company Account may have references to a Contract of each employee.

▪ ▪ ▪ ▪ *Deal*

An Account may have a subordinate Deal, which is a role of a Deal, which is billed by this Account.

This reference is typically established when the Deal is settled between an organization Customer and the Operator. The Deal refers to Price of a Product in a product Catalogue of an Operator. When a member of the organization uses the Deal, the Price is instantiated to a Product instance under a Contract that has a reference to one or more Accounts. One of the Accounts may be the Account that has the reference to the Deal.

▪ ▪ ▪ ▪ *Subordinate account*

An Account may have several Subordinate accounts, which each indicates a role of another Account. The Subordinate account provides information on parts of the Superior account. The account structure indicates how the Customer wants his payment information to be structured and split. Figures from the current Account may then be calculated to all Superior accounts. Therefore, the account structure may have several top nodes, but one Account cannot be contained (recursively) via different paths within the same node. An attribute indicates if the Account shall result in a bill or not. One Customer may receive several bills on the same Product instance and the same period, e.g., partly prepaid and postpaid.

▪ ▪ ▪ ▪ *Superior account*

An Account may have a Superior account, which indicates a role of another Account. In this case, the current Account provides information on parts of the Superior account. An Account may have several Superior accounts. Figures from the current Account may then be calculated to all Superior accounts. Therefore, the account structure may have several top nodes, but one Account cannot be contained (recursively) via different paths within the same node. An attribute indicates if the Account shall result in a bill or not. One Customer may receive several bills on the same Product instance and the same period, e.g., partly prepaid and postpaid.

▪ ▪ ▪ Catalogue

A Catalogue contains an overview of all Products, i.e., Product types, available from the superior Operator of the Catalogue. Due to needs to manage several versions of offers, an Operator may have several Catalogues. Also, separate Catalogues may be defined for technical products and Product types which are inherited, or by other means are used to construct products for sale. A Product from one Catalogue may be inherited or composed of Products from other Catalogues.

▪ ▪ ▪ ▪ Identifier

A Catalogue has a unique identifier within the superior Operator. If there are several versions for a product Catalogue, version information should be contained in the identifier.

▪ ▪ ▪ ▪ Valid Period

The period during which the catalogue is applicable.

▪ ▪ ▪ ▪ Product

Product type

A Product type (under a Catalogue) is a type of service with a predefined functionality.

Note that a Product type is an instance of Product class (under Catalogue). Product types are normally for sale, and these only are visible in the customer channels. Separate Product types may be defined as technical products, which are used to implement products for sale. Also, particular Product types may be defined for inheritance of definitions. See the Association Type I.

The Product type may be defined as a market product or a technical product. This allows one Operator A to deliver a Product type to another Operator B. The delivered Product type may be a technical product from A that is used to deliver a bundled market product from B, and the market product only is visible to the Customers of B. This way, products may be bundled in several steps.

Due to the bundling mechanisms indicated above, there is no distinction between Product, service and resource. Products may have service attributes, and Products may have attributes that refer to resources.

Note that a service may be defined to be a Product that is delivered from a service platform to a user. However, this notion is outside the scope of this Recommendation. See Appendix I.

As only Products may have Prices, if the parameters of the resources imply different pricing, a separate Product must be defined for each of these parameter values. This means that if the service platform has a parameter for bandwidth and the different bandwidths have different Prices, then a separate Product may be defined for each bandwidth.

The Product type may be an individual Product type or be bundled as a predefined set of other Product types, see Association Type B.

A Product type may have a set of permissible additional Product types, see Association Type M, and the Product type may exclude use of other Product types, see Association Type I.

The topmost node in the Product type hierarchy, by bundling or inheritance, i.e., Association Type B or I, shall itself be an ordinary Product type (for sale), and the hierarchy shall not be used for categorization of Products, shall not indicate organization, deals or other information.

A Product type may inherit properties or subordinate structures from several other Product types.

Each Product type will have a Type (identifier) that is unique within the superior Catalogue.

In addition, a Product type may have a Resource type identifier, provided by the Operator, and which tells that an instance of this Resource type may be replaced by another instance of the same type.

Product instance

A Product instance is an instance of exactly one Product type, and there may be several instances of the same type. This is accomplished by the S(chema) reference from Contract to Catalogue. This specifies that:

– All data classes subordinate to Catalogue are considered to be subordinate to Contract, as well;

– The S(chema) reference itself may be instantiated to refer from a specific Contract instance to one or more specific Catalogue instances;

– The data instances in the tree subordinate to a Contract are instances of the data, i.e., Product types, under the referenced Catalogues.

A Product instance may be contained in a Product instance bundle, may have additional Product instances and exclude use of others according to the prescriptions in its Product types.

Note that a Product instance may only be contained in one superior product bundle, while a Product type may be contained in several superior product bundle types.

The Product instance has an Identifier. This Identifier is made up of a Catalogue, a Type and a Number. The Catalogue attribute indicates from which Catalogue the Product type is taken to generate this Product instance. Also, the Catalogue makes the identifier unique within an Operator. The Number may be a telephone number, circuit number or a similar resource identifier. In addition, the Product instance may have a Resource type attribute.

Note that generalization of Number to an alphanumeric field is for further study.

- Identifier

    A Product has a unique identifier within the superior Catalogue or Contract.

    The Type is unique within its Catalogue, while the combination of Catalogue, Type and Number is unique within the Contract.

- Catalogue

    The Catalogue attribute identifies the Catalogue instance from which the Product instance is generated. The Catalogue attribute is strictly needed only for Product instances.

- Type

    Each Product type will have a unique Type within its superior Catalogue.

    Each Product instance under a Contract instance will be assigned the Type of the Product type from which it is generated. The Type must be filled in both for Product types and Product instances.

- Number

    Product instances within a Contract instance will have a Number which is unique together with the Catalogue and Type within the Product's superior Contract.

However, for some Types, e.g., TLF, the Number may be globally unique, such that one may look up a product instance directly, based on the Number only.

The Number attribute is strictly needed only for product instances.

▪ ▪ ▪ ▪ ▪ Name

A word, term, or phrase by which the Product is known and distinguished from other Products. The Name attribute is applicable for both Product types and Product instances.

▪ ▪ ▪ ▪ ▪ Description

An explanation of what the Product is. The Description attribute is applicable for both Product types and Product instances.

▪ ▪ ▪ ▪ ▪ Technology type

The technology type of the Product, such as fixed phone, wireless, data communication, etc. This attribute is only applicable for Product types.

▪ ▪ ▪ ▪ ▪ Customer base type

The industry or customer base the Product is designed for, such as government, education, finance, family, etc. This attribute is only applicable for Product types.

▪ ▪ ▪ ▪ ▪ Valid period

The period during which the Product is applicable. This attribute is applicable for both Product types and Product instances.

▪ ▪ ▪ ▪ ▪ Status

The condition in which the Product exists. This attribute is applicable for both Product types and Product instances. However, Product type and Product instance may have different status values. For example, Product type may take such values as planned, obsolete, active, while Product instance may take such values as planned, designed, activated, disconnected.

▪ ▪ ▪ ▪ ▪ Product characteristic

Product characteristic specifies characteristic quality or distinctive features of a Product and the values assigned to the characteristics. The characteristic can either take on a discrete value, such as colour, or take on a range of values (for example, sensitivity of 100-240 mV). Product characteristic is applicable for both Product types and Product instances.

▪ ▪ ▪ ▪ ▪ ▪ Identifier
The local unique identifier for the Product characteristic within the Product.

▪ ▪ ▪ ▪ ▪ ▪ Name
A word, term, or phrase by which the characteristic is known and distinguished from other characteristics.

▪ ▪ ▪ ▪ ▪ ▪ Description
A narrative that explains the characteristic.

▪ ▪ ▪ ▪ ▪ ▪ Value unit
This indicates the unit that is used by this characteristic. When no unit is used for this characteristic (such as a ratio), this field is filled with "--", indicating "not applicable". This attribute is optional.

- Configurable

  A boolean value determines if a characteristic value can be assigned for Product types by the provider or configured for Product instances by the customers.

- Value

  Values that the characteristic can take on. The value may be a discrete value or a value range. This attribute is optional.

- Valid period

  The period of time for which a characteristic and its value are applicable.

- *Sales channel*

  A Product's subordinate Sales channel refers to a Channel subordinate to an Operator.

  A Product may have a variety of Sales channels, such as online purchase, SMS purchase, hotline purchase, business hall purchase, and so on. Alternative Sales channels may exist for a Product type, while a Product instance is sold through a particular Sales channel.

- Term

  Term specifies a condition under which the Product is made available to customers. The Term includes financial terms, such as acceptable methods of payment, shipment terms and service terms. Term is only applicable for Product types.

- Identifier

  The local unique identifier of the Term within the Product.

- Type

  The type of the Term, such as financial terms, shipment terms, service terms, and so on.

- Term information

  The detailed information of the specific Term.

- Quality level

  The information about different quality level options to be chosen or agreed for the Product type, or a quality level that is chosen by a customer for a particular Product instance.

- Service class

  This attribute is used to identify the Quality level of the Product, such as "platinum", "gold", "silver" and so on. The Service class is unique within the Product.

- Description

  Explanation of the specific Service class.

- Quality parameter

  Quality parameter specifies the detailed information of each quality parameter of the Product.

- Identifier

  The unique identifier assigned for the parameter for reference and identification purpose. The Identifier is unique within the Quality level.

- Name

  This attribute is used to specify the full name or the abbreviation of the specified Quality parameter.

- Value unit

  This indicates the unit that is used by this Quality parameter. When no unit is used for this Quality parameter (such as a ratio), this field is filled with "--", indicating "not applicable".

- Value

  Values that the Quality parameter can take, indicating the possible value range within which the Quality parameter should be to comply to the service level. The value may be a discrete value or a value range.

- Description

  This attribute explains the meaning of the Quality parameter.

- Parameter area

  This attribute indicates which area this Quality parameter is dealing with. For example, the areas could be Network Performance, Traffic or Service.

- Association

  A Product's subordinate Association indicates a role to one or more other Products relative to the first Product.

- Type

  A Product's Association to one or more other Products may be of the following Types:

  B –   Bundle, where the other Products are components

  M –   Main product, where the other Products are additional products

  C –   Constrained product, where the other Products are excluded products

  I –   Product instance, where the other Product is the Product type of the current Product

  N –   Next service; if the user is not present at the current service, try the next

- *Contained product*

  An Association's subordinate Contained product is a Product that is referenced in the Association.

- *Installation*

  A Product's subordinate Installation indicates a role of an Installation.

- *Superior association*

  A Product's subordinate Superior association is an Association that references the Product.

  The recursive references on the Product types under a Catalogue put constraints on the Product instances under Contracts.

A Product instance must contain exactly the Association Type Bundle, which are prescribed by the Product type.

A Product instance may only have a subset of the Association Type Main Product that are given for the Product type.

A Product instance may not contain or be part of a Product that is associated by the Association Type Constrained product of the Product type.

A Product instance can only have the Association Type Next to Products of Product types having this association from the Product type of the Product instance.

A Product instance obeys all rules that are prescribed for its Product type, as well as the type's recursively superior types.

A Product type or Product instance may have an Association Type I to another Product type. When this Association is between two Product types, it is used to inherit properties. When it is between an instance and a type, it is used for instantiation. Inheritance may be seen as a constrained form of instantiation. Note that the Association from instance to type may or may not be explicitly stored. Also, the Type of the instance Identifier will tell about this Association.

Note that specialized associations may be derived from the various Association Types, e.g., a S(chema) reference from the Association Type I. Such derived associations are not covered by this Recommendation.

▪ ▪ ▪ ▪ ▪ Event

A special set of circumstances under which a service management policy would be triggered.

Normally, only service Products may have Events.

▪ ▪ ▪ ▪ ▪ ▪ Identifier

A globally unique identifier of the Event. The identifier may consist of a prefix indicating the environment, e.g. Service platform, of the identifier, followed by the unique identifier within that environment.

▪ ▪ ▪ ▪ ▪ ▪ ▪ Environment

A globally unique identification of the hardware or software environment in which Events are identified.

▪ ▪ ▪ ▪ ▪ ▪ ▪ Event-identifier

A unique identifier of the Event within its Environment.

▪ ▪ ▪ ▪ ▪ ▪ *Price component*

An Event's subordinate Price component refers to a Price component subordinate to a Price of the same or a recursively superior Product.

▪ ▪ ▪ ▪ ▪ Price

A Product type may have several Prices. A Product instance has only one Price.

The Price of a Product instance may depend on Deal and Market. The same Price may apply for several Deals or Markets, or may be different for alternative Deals and Markets.

▪ ▪ ▪ ▪ ▪ ▪ Price component

A Price is made up of a set of Price components. Normally, Price components are added up to make up the Price. However, the Price type may indicate subtraction, multiplication or other operators.

- Price type

  A category that describes the price, such as recurring, discount, allowance, penalty, and so forth.

- Unit

  The unit used for this Price component. The unit may be given in a certain currency, minutes, SMSes or other.

- Amount

  The amount in which this Price is defined.

- Description

  A narrative that explains in detail the semantics of this Price component.

- Valid period

  The period for which the pricing is valid.

- *Event*

  A Price component's subordinate Event refers to an Event subordinate to a service Product.

- Price rule

  A Price component is governed by Price rules. All Price rules consist of three types of components, which define the semantics of the rule.

- Event

  A Price rule may refer to the Price component's subordinate Event which may trigger the evaluation of the Price rule.

- Price condition

  Part of a Price rule representing constraints that define the assessment of the Price rule.

- Price action

  The action to be taken when the Price condition is satisfied.

- *Deal*

  A Price's subordinate Deal indicates a role of a Deal subordinate to the Price.

- *Market*

  A Price's subordinate Market indicates a role of a Market. The Market indicates the market segment for which this Price applies.

- *Use*

  A Product instance may have a Use, which is a role of a Use. Use refers to a Customer and his relationship to the Product instance.

- *Market*

  An Operator may have several subordinate Markets, which each indicates a role of a Market of this Operator.

- Channel

  A medium or organization through which Products may be distributed or sold.

- Identifier

  A unique identifier of the Channel within the Operator.

- ▪ ▪ ▪ ▪ Name

  The name of the Channel.

- ▪ ▪ ▪ ▪ Channel information

  The detailed information of the Channel.

- ▪ ▪ ▪ ▪ *Product*

  A Channel's subordinate Products refer to the Products that may use this sales or distribution Channel

- ▪ ▪ ▪ Contract

  A Contract is a contract with a Customer on delivery of services, which are called Product instances in this data structure.

  Permissible Product types for sale can be found in two ways:

  - By the schema reference (S) to Catalogues, which are explicitly referred to from the Contract;
  - If no explicit schema reference is provided, the permissible Product types are found under the latest Catalogue under the Contract's superior Operator.

  Note that older Product types under the current Contract may come from older Catalogues under the current Operator.

  A Customer may have one or more Contracts. Sometimes it may be convenient to have one Contract per Account.

- ▪ ▪ ▪ ▪ Contract number

  A Contract's subordinate Contract number is a unique identifier of a Contract within the Operator.

  A Contract number may by default be identical to a phone number used to identify a Product instance subordinate to the Contract. If a Contract number is assigned this way, one must avoid that this number comes in conflict with numbers assigned differently.

  The customer may not always be informed about the Contract number, and the Operator may not expect that the Customer is able to inform about his Contract number. Rather, the Customer may inform about a Number of a Product instance, e.g., a phone number.

- ▪ ▪ ▪ ▪ Amount

  A Contract's subordinate Amount indicates the available amount for usage under this Contract.

- ▪ ▪ ▪ ▪ ▪ Prepaid amount

  An Amount's subordinate Prepaid amount indicates the set amount for usage from the Prepaid date under this Contract.

  The Prepaid amount is the Prepaid amount minus the Spent amount before the Prepaid date plus the additional amount paid at this date.

- ▪ ▪ ▪ ▪ ▪ Prepaid date

  The Prepaid date gives the last date when the Prepaid amount under this Contract was updated.

- ▪ ▪ ▪ ▪ ▪ Spent amount

  An Amount's subordinate Spent amount indicates the used amount under this Contract since the Prepaid date. The available amount for usage is Prepaid amount ÷ Spent amount.

NOTE – More attributes, e.g., for Available amount, Last usage update, etc., could be defined, if needed.

- Internet address

  A Contract's subordinate Internet address indicates the Customer's contact address. A Customer may have more Internet addresses, and each Internet address is recorded as a Product instance.

- *Account*

  A Contract may have one or more subordinate Accounts, which each is a role of an Account. Normally a Contract has one Account only, but use of multiple Accounts allows for split bills. Rules given for each Account specifies which costs go where, e.g., before 17.00 hours to the company customer and after 17.00 hours to the private customer. Also, the Account hierarchy may help to accomplish this split.

- Role

  A Contract's subordinate Role indicates the Customer's relationship to this Contract.

- Category

  The Category indicates which role a Customer plays relative to a Contract. The Category is an attribute group that consists of attributes.

  If the value of the Type attribute is updated by the terminal user, then the value of the Update attribute is set to manual (M). If the role is created by the system itself, then the value of the Update attribute is set to derived (B). Only roles that are updated manually need to be stored.

- Type

  A Role's subordinate Type may have one of the following values:

  E – owner

  B – payer

  Y – authorized

  K – contact

  V – very important person

  U – user

  A – provider agent

  R – responsible employee

  A Contract may only have one Role of Type owner (E), payer (B), user (U) or provider agent (A), but may have several Roles of Type contact (K) and very important person (V).

  If some Roles are not explicitly provided for a Contract, and the Contract has one Account only, then the Roles of the Contract are copied from the Roles of the Account.

  If the Role of Type user (U) is not indicated for the Contract, but is the same for several subordinate Product instances, then this role is copied to the Contract.

  If owner (E) is provided, and payer (B) is missing, then the payer is equal to the owner.

If owner (E) is provided, and user (U) is missing, the user is equal to the owner.

If payer (B) is provided, and owner (E) is missing, the owner is equal to payer.

If payer (B) is provided, and user (U) is missing, the user is equal to payer.

If user (U) is provided, and owner (E) is missing, the owner is equal to user.

If user (U) is provided, and payer (B) is missing, the payer is equal to user.

If Use is not provided for some subordinate Product instances, then these are copied from the user (U) Type of the Contract.

If some Roles are not provided for a related Account, and this Account has only this Contract, then the Roles of the Account are equal to those of the Contract.

Derived Types may, e.g., be indicated by a separate colour at the user interface, or they may be hidden.

▪ ▪ ▪ ▪ ▪ ▪ Update

The Type of a Role may be updated

M – manually

B – automatic

▪ ▪ ▪ ▪ ▪ *Customer*

A Role must have a subordinate Customer that is a role of a Customer.

▪ ▪ ▪ ▪ *S <>'& (Corporation (Country (Operator (Catalogue*

A Contract's superior Operator's first subordinate Catalogue is a S(chema) for the Product instances subordinate to the Contract.

Note that the reference may go all the way up to the object Corporation via wild card (&) down to some other Operator than the directly superior Operator. If this option is used, the value of the Catalogue attribute of the Product instance must be made unique across the Operators.

If no explicit S(chema) reference is provided, the latest Catalogue of the directly superior Operator is used.

Note that the S(chema) reference may itself be instantiated. Then the reference goes via the Population of the Schema.

The Product instances subordinate to the Contract are instances of the Product types subordinate to the S(chema) referenced Catalogue.

The recursive references on the Product types put restrictions on the Product instances subordinate to the Contract:

– A Product instance must contain exactly the Associations with the Type Bundle that are prescribed for its Product type;

– A Product instance may only contain a subset of the Associations with the Type Main product that are allowed for its Product types;

– A Product instance must not contain Associations with the Type Constrained that are prescribed for its Product types;

– A Product instance must obey all constraints that are defined for its Product types as well as its recursively superior Product types, indicated by the Association Type I.

▪ ▪ ▪ <u>Deal</u>

A Deal defines a relationship from an Operator via a Membership to a Customer on one or more product instance prices of given Product types. A Deal is valid for a limited time period, with a defined start and stop.

A Deal may be a campaign for a product type or a result of a campaign for a product instance.

Note that a Product instance may be related via Price to several Deals, and Deals may become terminated, while a Product instance is contained in a Contract that cannot be removed without removing its Product instances, as well.

▪ ▪ ▪ ▪ *Account*

A Deal may have a subordinate Account, which is a role of an Account, which is the billing Account for this Deal.

▪ ▪ ▪ ▪ *Membership*

A Deal may have subordinate Memberships, which each is a role of a Membership.

▪ ▪ ▪ ▪ *Price*

A Deal may have several subordinate Prices, which each is a role of a Price. The Prices may be both types (under Product in a Catalogue) and instances (under Product in a Contract).

▪ <u>Employee</u>

Employee is an employee of an Operator of Product instances to Customers.

▪ ▪ Employee number

Employee number is a unique identifier of an Employee of an Operator of Product instances. Some employees may have several Employee numbers. In this case, several Employees are related to the same Customer, who plays the role of being the person that is employed.

Note that Employee number is unique across all Operators of a corporation.

▪ ▪ *Customer*

An Employee may have a subordinate Customer, who indicates a role of the employee as a Customer.

Note that the employee's role to other Customers is registered from his Customer role to the other Customers.

▪ <u>Installation</u>

Installation indicates a reference to the installation address of a Product instance.

▪ ▪ Change

Change indicates state and time interval for the installation.

▪ ▪ ▪ State

State indicates

H – historic

I – installed

▪ ▪ ▪ Time interval

Time interval indicates the time period for the establishment or termination of an installation.

- ▪ *Installation address*

    An Installation must have a subordinate Installation address that indicates the Address where the Product instance is installed.

- ▪ *Product*

    An Installation must have a subordinate Product, which indicates the Product instance that is installed at the Address.

- Market

    A Market defines a market segment, which the Operator chooses to handle in a harmonized manner.

    A Market may be defined by customer groupings, channel groupings or by other means.

- ▪ *Operator*

    A Market may have a subordinate Operator, which is a role of an Operator for whom the given Prices apply.

- ▪ *Price*

    A Market may have subordinate Prices, which each indicates a role of a Price of a Product (type or instance).

- Membership

    A Membership indicates an association between a Customer and a Deal of an Operator.

- ▪ Category

    A Category indicates a role that a Customer plays in a Deal. Category is an attribute group, which contains two attributes.

    If Type is updated manually by the terminal user, the Update attribute value is set to manual (M). If the Type is updated by the system itself, the Update attribute value is set to derived (B). Only manual updates need to be stored.

- ▪ ▪ Type

    The Type of a Membership may have one of the following values:

    K – contact

    M – member

    E – owner

    Note that often the members will only be registered by a reference to their phone numbers, i.e., Number within a Product instance. In this case, the memberships will be derived automatically from Product instances via Use to Customer.

- ▪ ▪ Update

    Types may be updated

    M – manually

    B – derived

- ▪ *Customer*

    A Membership may have a subordinate Customer that is a role of a Customer of the corporation.

- ▪ *Deal*

    A Membership may have a subordinate Deal, which indicates a role of a Deal subordinate to Operator.

- Structure

   A Structure indicates the organization of the Customer.

   The Operator will normally not register this Structure, but sometimes the implied organization of a corporation into its constituent business units may be derived from the geographical locations, or of families from their relationships to common account numbers and addresses.

   The Operator will normally only register the Customer's associations to Accounts, Contracts and Products, which indicates agreed relationships between the Customer and Operator.

- ▪ Type

   Type indicates which role a Subordinate customer plays to a Superior customer.

   - V – indicates subordinate business unit
   - M – indicates Member of a family
   - U – indicates subordinate Unspecified membership
   - H – indicates historical entity of the current entity

- ▪ *Subordinate customer*

   A Structure's Subordinate customer indicates a role of a Customer that is superior to this Structure.

- ▪ *Superior customer*

   A Structure's Superior customer indicates a role of a Customer that is subordinate to this Structure.

- Type

   A Type indicates the type of Address of the Customer.

- ▪ Type

   Type indicates the role of the referenced Address relative to the Customer.

   - S – location
   - P – postal address

- ▪ *Address*

   A Type must have a subordinate Address, which indicates a role of an Address.

- ▪ *Customer*

   A Type must have a subordinate Customer that indicates a role of a Customer.

- Type

   A Type indicates the type of Address of the Account.

- ▪ Type

   Type indicates the role of the referenced Address relative to the Account.

   - A – address of responsible person
   - B – billing address

   A company customer may have several Accounts, which each may have different responsible persons.

- ▪ *Account*

   A Type must have a subordinate Account, which indicates a role of an Account.

- ▪ *Address*

   A Type must have a subordinate Address, which indicates a role of an Address.

- Use

  Use indicates a role between a Product instance and a Customer. The role is defined in the attribute group Category. Use is not relevant for Product types within a Catalogue.

- ▪ Category

  Category indicates the role that a Customer has to be a Product instance. Category is an attribute group which consists of two attributes.

  If the attribute Type is updated by the terminal user, the Update attribute is assigned the value Manual (M). If the role is established by the system itself, the Update attribute is assigned the value automatic (B). Only roles that are updated manually need to be stored.

- ▪ ▪ Date

  Date for establishment of a certain Type of Use.

- ▪ ▪ Type

  The subordinate Type of a Use may have one of the following values:

  K – contact person

  U – user

  A Product may have one user (U) only, but may have several contacts (K).

  If a particular Role is not provided for the superior Contract, and this role is given for the current Product instance in the Contract, then the Role of the Contract is derived automatically from the role of the Product instance.

  If Use is not given for a Product instance, this is derived automatically from the Role of the Contract.

  From the above, if a role is given for a particular Product instance only, then this role is inherited to all Product instances within the Contract. However, the inherited roles may be overridden by manual insertion of particular roles for a Product instance, Contract or Account.

  Derived Types may be indicated by use of a separate colour in the end-user dialogue, or they may be hidden.

- ▪ ▪ Update

  The Type of a Use may be updated

  M – manual(ly)

  B – automatic

- ▪ *Customer*

  A Use must have a subordinate Customer, who is a role of a Customer.

- ▪ *Product*

  A Use must have a subordinate Product, which is a role of a Product instance.

Appendix I

Product and customer service databases

(This appendix does not form an integral part of this Recommendation.)

Products in a Catalogue may be instantiated into Products in a Contract. This is illustrated in Figure I.1, which depicts how classes within the Catalogue class apply within any Contract. The Contract plays the role of being a P(opulation) of the Catalogue, which acts as a S(chema) of the Contract.
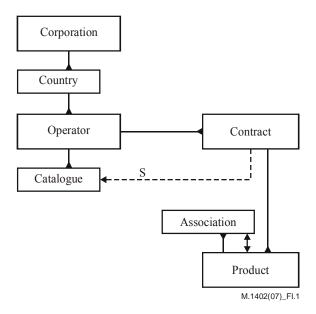

M.1402(07)_FI.1

Figure I.1 – Example inheritance by instantiation from the product Catalogue

NOTE 1 – The Schema reference from Contract to Catalogue in Figure I.1 allows for an n:1 mapping from Product under Contract to Product under Catalogue. Only a 1:1 mapping is depicted. The purpose is to show that the same classes apply under Contract as of under Catalogue. This is right, but in practice, the instantiation is more complicated: Contract is instantiated, Catalogue is instantiated, and Schema references are stated between the Contract instances and Catalogue instances. Product instances under Contract instances are instantiated from Product type instances under Catalogue instances.

NOTE 2 – The term inheritance is used in many object oriented notations, but not in the notation used in the ITU-T M.1400 series, as the ITU-T M.1400 series prescribe syntax, not concepts. The purpose of Figure I.1 and the accompanying text is to show that something similar to inheritance, but more complex, can be stated. Most object-oriented notations apply only between classes. In the current approach, the same notation applies between instances.

The Product class and its subordinate classes, such as Price, may be instantiated into a product database. An example instantiation is shown in Figure I.2.
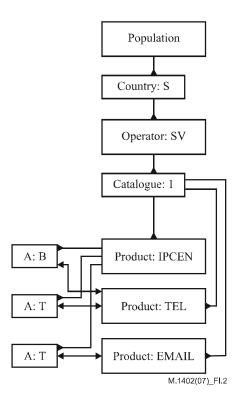
M.1402(07)_FI.2

Figure I.2 – Example instances in a product database

Note that the labels used in Figure I.2 are incomplete, due to lack of space inside the boxes. For example A: B means Association Type B. Product: IPCEN means a Product of Type IPCEN. Population is the name of the product database.

The instances in a product database may act as classes of Product instances in a customer service database. This is illustrated in Figure I.3.
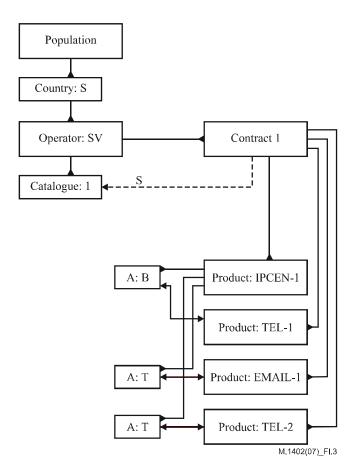
Figure I.3 – Example instances in a customer service database

Note that there is a homomorphic mapping from instances to their classes, i.e., there is a many-to-one mapping from instances to classes, and for each box, line or label among the instances there is an identical box, line or label among their classes. The Catalogue attribute of the Product instances is not indicated.

Appendix II

Interface to a service platform

(This appendix does not form an integral part of this Recommendation.)

Some of the Product instances in the customer service database may be implemented as services within a service platform. Hence, information of the subset of Product types and Product instances need to be communicated over the interface to the service platform. Figure II.1 depicts the basic data structure needed for communicating instance data to and from the service platform.
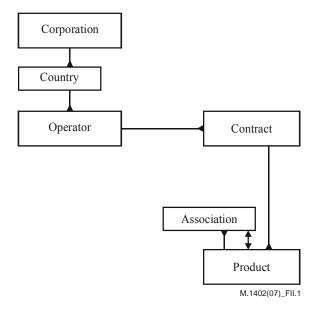

M.1402(07)_FII.1

Figure II.1 – Example data structure for interface specification

The interface specification towards the service platform is designed to be simple.

The interface is anticipated to be implemented as a replication of a subset, i.e., horizontal partition, of the customer service database.

Alternatively, the interface may be designed as an exchange of service orders; this alternative is not shown in this Recommendation.

For pricing information, both Price and Deal may be added to the shown data structure.

Note that what appears as a networked data structure (in the Application layer of the Data Architecture) may be turned into a tree (defined in a Distribution Schema) when being communicated over the interface to the service platform.

Note that the data structure provides multi-operator support on one and the same service platform.

Bibliography

[b-TMF GB922A3]   TM Forum GB922 (2010), *Shared Information/Data (SID) Model. Addendum 3 – Product Business Entity Definitions, Release 9.0.*

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	General tariff principles
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Construction, installation and protection of cables and other elements of outside plant
Series M	**Telecommunication management, including TMN and network maintenance**
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Terminals and subjective and objective assessment methods
Series Q	Switching and signalling
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks, open system communications and security
Series Y	Global information infrastructure, Internet protocol aspects and next-generation networks
Series Z	Languages and general software aspects for telecommunication systems