INTERNATIONAL STANDARD

ISO/IEC 10027

First edition
1990-06-15

Information technology — Information Resource Dictionary System (IRDS) framework

Technologies de l'information — Cadre pour le gestionnaire de ressources du système d'information (IRDS)

Reference number
ISO/IEC 10027 : 1990 (E)

Table of Contents

ISO/IEC Copyright Office • Case postale 56 • CH-1211 Genève 20 • Switzerland
Printed in Switzerland

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

International Standard ISO/IEC 10027 was prepared by Technical Committee ISO/IEC JTC 1, *Information technology*.

International Standard ISO/IEC 10027 is one of a series of International Standards on Information Resource Dictionary Systems.

This page intentionally left blank

Information technology — Information Resource Dictionary System (IRDS) framework

1 Scope

This International Standard describes the framework for a number of International Standards that specify a specialised information system, called an Information Resource Dictionary System (IRDS). An IRDS is used to control and document an enterprise's information resources.

This International Standard defines the data levels relevant to an IRDS. It defines the IRDS interfaces which are prescribed by other International Standards in the IRDS family of standards. It also defines the kinds of data content that are prescribed by other International Standards in the family.

2 Normative references

The following International Standards contain provisions which, through reference in this text, constitute provisions of this International Standard. At the time of publication, the editions indicated were valid. All International Standards are subject to revision, and parties to agreements based on this International Standard are encouraged to investigate the possibility of applying the most recent editions of the International Standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards.

ISO 7498 : 1984, *Information processing systems - Open Systems Interconnection - Basic Reference Model.*

ISO/IEC 9075:1989 *Information processing systems - Database Language SQL with integrity enhancement.*

3 Definitions

Definitions in this clause are those IRDS definitions used in this Framework. These definitions are referenced in other International Standards where they are used. Further International Standards may define additional terminology.

When each term listed in this clause is introduced in a later clause of this International Standard, it is printed in bold type.

3.1 Term defined in ISO 7498 and used in this International Standard

The following term is defined and used in the OSI Reference Model. It is used in the same way in this IRDS Framework International Standard.

3.1.1 real system.

3.2 Terms originally defined in ISO 7498 and adapted for use in this International Standard

The following terms were originally defined and used in the OSI Reference Model and other OSI International Standards. Their use in this IRDS Framework is based on that in the OSI International Standards, but a revised definition is preferred.

Some terms are prefixed in OSI with "(N)-" to indicate the layer. Since the IRDS Framework does not have a formal layer concept the prefix is omitted.

3.2.1 server: A role filled by a processor when it provides services to another processor.

3.2.2 service: A capability provided by a processor to other processors.

3.3 Terms defined in this International Standard

For the purpose of this International Standard the following terms apply.

3.3.1 access control: A capability to restrict the use of services accessing data to users who have been previously authorised.

3.3.2 application level: The data level on which instances of application data are recorded.

3.3.3 application level pair: The term used to describe both the application level and its schema at the IRD level.

3.3.4 application schema: A set of definitions which control what may exist at any time in an application.

3.3.5 auditing: A process of checking that previously made changes to a collection of data have been made correctly and by an appropriately authorised user at an appropriate time.

3.3.6 client: A user requesting the services provided at an interface of a server.

3.3.7 constraint: A statement of one or more valid states of some part of a database, based on the conditions which values in the database must satisfy at any time.

3.3.8 data container: A conceptual area of storage in which data instances can be recorded.

3.3.9 data level: A stratum in a multi-level data architecture on which objects may be recorded conforming to a type definition on the next higher data level.

3.3.10 data modelling facility: A set of data structuring rules and an associated set of data manipulation rules.

3.3.11 database: A collection of interrelated data stored together with controlled redundancy according to a schema to serve one or more applications.

3.3.12 database integrity: The consistency of a collection of data in a database.

3.3.13 export: The function of extracting information from an IRDS and packaging it to an export/import file.

3.3.14 export/import file: A file created by an export function and accepted by an import function.

3.3.15 import: The function of receiving data from an export/import file into an IRDS.

3.3.16 Information Resource Dictionary (IRD): A part of a repository managed by an IRDS in which the information resources of an enterprise may be recorded.

3.3.17 Information Resource Dictionary System (IRDS): A software product which maintains IRDs and IRD definitions.

3.3.18 information resource management: The task of maintaining and controlling information processing systems.

3.3.19 interface: A defined set of services made available by a processor.

3.3.20 IRD definition: A set of objects which collectively defines the data which may be held in an IRD.

3.3.21 IRD definition level: The data level at which potential IRDS content is defined.

3.3.22 IRD definition level pair: The term used to describe both the IRD definition level and its schema at the IRD Definition Schema level.

3.3.23 IRD definition schema: A set of definitions which control what may exist at any time in an IRD definition.

3.3.24 IRD definition schema level: A data level on which the types of object that may be recorded in an IRD definition are prescribed.

3.3.25 IRD level: The data level at which the information resources of an enterprise are defined.

3.3.26 IRD level pair: The term used to describe the IRD level and its schema at the IRD definition level.

3.3.27 IRD schema: A set of definitions which control what may exist at any time in an IRD.

3.3.28 level pair: Two adjacent data levels, the upper level of which will always contain the "type" information relevant to the "instances" on the lower level.

3.3.29 life cycle: A conceptual framework used to trace the evolution of objects over time.

3.3.30 life cycle phase: A sub-division of a life cycle.

3.3.31 partition: A logical sub-set of the objects in either an IRD or an IRD Definition.

3.3.32 processor: An abstract conceptualisation of an executable piece of code.

3.3.33 status of dictionary content: A status of a collection of data in a dictionary indicating whether the data may be freely modified, not modified, or is regarded as archived.

3.3.34 storage medium: A device on which data of any kind may be recorded.

3.3.35 user: A person or an application program which requests services for data management.

3.3.36 value: An abstraction with a single characteristic which can be compared with other values, and may be represented by an encoding of the value.

4 Conventions

4.1 Processors

This International Standard contains diagrams illustrating the processors and interfaces prescribed by the Standard.

A **processor** is an abstract conceptualisation of an executable piece of code.

Each processor is represented by a named icon as illustrated in figure 1.

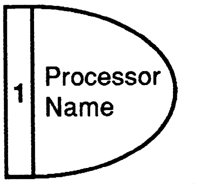

Figure 1 - Notation for a processor and its interface

The name in the icon is the name of the processor.

4.2 Interfaces

An **interface** is a defined set of services made available by a processor.

Each processor is defined as having one interface at which services are made available. The interface is represented by the rectangular part of the icon shown in figure 1. The number identifies the interface which is named and described in accompanying text.

4.3 Person

A person is represented by the icon shown in figure 2.

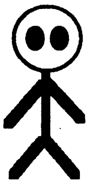

Figure 2 - Icon for a person

4.4 Storage medium

A **storage medium** is a device on which data of any kind may be recorded such as a hard disc, a floppy disc or tape. A storage medium is represented by the icon shown in figure 3.

Data will be held on some storage medium.

The name within the icon identifies the content of the storage medium.

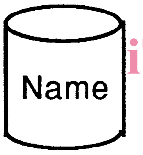

Figure 3 - Icon for a storage medium

4.5 Client - Server association

Users of **services** provided at an interface are termed **clients.**

The processor that provides the services at an interface is termed the **server.** A processor may fill the role of client and the role of server. A processor can be the server to one interface. A processor may be a client of many interfaces.

A client may be a person or a processor depending upon the nature of the interface.

Use of an interface by a client is shown by a line connecting them. Use of a storage medium by a processor is shown by a line between them as illustrated in figure 4.

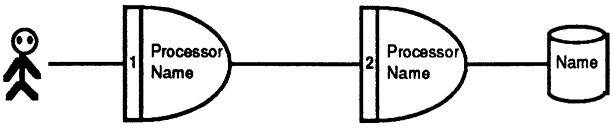

Figure 4 - Conventions to show Client - Server associations

4.6 Diagramming conventions

All figures using the conventions described above are drawn so that the client is shown to the left of the server. This convention is illustrated in figure 4.

5 The Structure of the IRDS Standards

5.1 Purpose of the IRDS Standards

The purpose of the family of International Standards for Information Resource Dictionary Systems is to provide a common basis for the development of Information Resource Dictionaries. This IRDS Framework International Standard defines the context within which the other parts of the IRDS family of International Standards are prescribed.

An **Information Resource Dictionary** is a shareable repository for a definition of the information resources relevant to all or part of an enterprise. This may include information about any or all of the following:

a) data needed by the enterprise;

b) the computerized and possibly non-computerized processes which are available for presenting and maintaining such data;

c) the available physical hardware environment on which such data can be represented;

d) the organization of human and physical resources which can make use of the information;

e) the human resources responsible for generating that information.

An **Information Resource Dictionary System** (IRDS) is a system which provides facilities for creating, maintaining and accessing an Information Resource Dictionary (IRD) and its IRD definition.

It must be emphasized that this family of International Standards does not provide a standard definition for all of the above kinds of information. It does provide a framework for defining such information and in which the information can be represented and managed. The definition of those kinds of information to be represented in an Information Resource Dictionary, that are suitable for standardisation, are a subject for other International Standards in this family.

The content of an Information Resource Dictionary can be compared with the content of a typical application database. An application database contains data of relevance to the day to day operation of an enterprise. Such data may refer to things such as employees, suppliers, customers and purchase orders.

The data in an Information Resource Dictionary is in many ways similar to that in an application database, but it is on a higher level. Such data may refer to things such as data item types, data files, computer programs and sub-systems.

5.2 The IRDS family of Standards

This International Standard for an IRDS Framework specifies the overall architecture in which each member of the IRDS family of International Standards should be positioned. The Framework identifies, in general terms, the kinds of data together with the major processors and their associated interfaces and the broad nature of the services provided at each interface.

Some, but not all, of the interfaces identified in this International Standard are candidates for standardisation as members of the IRDS family of International Standards.

More than one standard may be developed for a given interface. Two International Standards for the same interface may differ according to any mix of the following aspects:

a) Programming language dependence;

b) Interface style;

c) Data modelling facility used

d) Data interchange format.

Each aspect will be considered in turn.

5.2.1 Programming language dependence

Numerous programming language dependencies are possible and there exist many International Standards for programming languages.

5.2.2 Interface styles

5.2.2.1 Interface styles for processors

Possible alternative interface styles which may be used by processors are the following:

a) Programmatic - procedure call;

b) Syntax (execution time interpretation);

c) Service conventions (as used with OSI services).

A procedure call interface defines a sequenced set of parameters and the associated binding rules for the CALL statement of an International Standard programming language. A procedure call may also be implicit which means that a statement is used which is translated into a CALL statement prior to compilation.

A syntax for execution time interpretation is similar to that provided for a human user with the difference that the linguistic forms are interpreted at execution time by a processor specific to an International Standard programming language.

A service convention is a standard set of programming language independent conventions for specifying parameter lists and service primitives for use in an open system environment.

5.2.2.2 Interface styles for persons

Possible alternative styles for interfaces to persons are the following:

a) Panels (abstract screen formats);

b) Concrete syntax;

c) Graphics.

A panel style of interface defines a grouping of services which may possibly have been defined using some other style.

A concrete syntax (such as a command language) is the traditional way of defining interfaces for persons. It is important to note that using a syntactic language to define an International Standard does not imply that the same concrete syntax must be used by a person using the interface.

A graphic interface style may be used in conjunction with either a panel interface or with an abstract syntax.

5.2.2.3 Abstract syntax

An interface style which can be used to define both interfaces to processors and interfaces to persons is that of an abstract syntax. An abstract syntax is the specification of a service by using notation rules which are independent of the encoding techniques used to represent them.

An abstract syntax emphasises the semantics of the interface and a person may initiate a standard service in a number of different ways such as selecting from a menu, touching a screen or using a special keyboard. Similarly, a processor interface can also be implemented using any one of several interface styles as required.

An International Standard using an abstract syntax defines a set of services without prescribing any linguistic form to be used by a person when each service is initiated or invoked. Such an International Standard also includes the semantics of the services.

5.2.3 Data modelling facility

A **data modelling facility** is a set of rules for defining the structure of data (including constraints) and the semantics of the associated data manipulation services.

Each International Standard for either an interface to a person or an interface serving another processor is dependent on one or more data modelling facilities.

Examples of kinds of data modelling facilities include those:

a) Based on an International Standard Database Language (such as NDL or SQL);

b) Based on a non-standard database language;

c) Specific to an International Standard programming language (such as COBOL or PL/I);

d) Specific to a non-language International Standard (such as OSI Directory Services);

e) Which are non-standard data modelling facilities (such as entity relationship modelling).

Each data modelling facility is an intrinsically independent means of representing data and possibly the services which may be specified for such data.

5.2.4 Data Interchange formats

If an International Standard in the IRDS family is concerned with the transfer of data from one real system to another, then a data interchange format must be adopted or defined in it. The transfer may be either by means of communications facilities or by physical transportation of data from one location to another using a transportable storage medium.

5.3 Support for different fields of application

This IRDS Framework is intended to supplement standardisation work in such general areas as data interchange formats and to facilitate the development of consistent International Standards in many specific fields of application.

The use of an International Standard in the IRDS family is useful not only within each field of application, but also as a bridge between them. This is enabled by allowing the rules according to which data is represented at one real system themselves to be defined, changed and extended according to specific requirements.

5.4 Means of support using International Standards

The support through International Standards can be provided in different ways, depending on the technologies involved and on the field of application. Examples are the following:

a) Standardised services at an interface;

b) Data content using generalised services;

c) Data interchange formats.

An International Standard for services at an interface defines in general terms the contents of some part of an Information Resource Dictionary and an IRD Definition, and the services by which those contents may be accessed and manipulated.

A data content standard defines in precise terms the content of some part of an information resource dictionary according to some prescribed data modelling facility. The services which may be performed on that data (including the semantics of such services) may or may not be implicit in the general data manipulation services associated with that data modelling facility.

An International Standard for a data interchange format is one designed to facilitate the inter-operability of several real systems by standardising the formats of the various kinds of message sent from one real system to another. A data interchange format may be specific to a field of application.

The facilities to be described in clause 7 of this International Standard relate to the first two of these three categories. The facilities to be described in clause 8.5 relate to the first and last of these categories.

6 IRDS Data Content

6.1 Data Levels

The cornerstone of the IRDS Framework is the concept of four data levels and the associated three "level pairs". The purpose of these four data levels is to make it possible to extend the types of data that can be held in the IRD.

An understanding of the levels and level pairs is critical to an understanding of how an IRD relates to its environment and of how an IRDS provides services to its users.

6.2 Concept of types and instances

The idea of types and instances (often referred to as occurrences) is well established in many programming languages and in database management systems, although these two have a different approach to the separation of program from data.

A "type" of data, such as an EMPLOYEE, is defined, either in a program or, in the case of a database management system, in a separate language used for defining data. This definition of a type of data essentially creates an open-ended data container. Sometimes it is identified as a record type, sometimes as a table, and sometimes in other ways.

Application programs, which may be separate from the above data definition or may contain it, refer to EMPLOYEE in their executable code. When such programs are executing, they will subsequently cause data about specific EMPLOYEES (each of which is sometimes called an instance or an occurrence) to be stored in a file or in a database.

When data about a specific EMPLOYEE needs to be accessed, it is necessary to refer to the type of data in a program and subsequently to pick out the specific instance of that type.

The IRDS concept of data levels is an extension of this basic type and instance concept which one can regard as having two levels and one level pair. These two levels are in fact the bottom two of the` four identified in this IRDS International Standard.

The concept of an application program, which in its source form references a defined type of data, such as EMPLOYEE, and when executing accesses an instance of that type, provides the basis for understanding how an IRDS service relates to a level pair.

6.3 Data Containers

The concept of a data container must be distinguished from that of a type or schema. A **data container** is a conceptual area of storage in which data instances can be recorded. At any point in time, a data container may contain data instances or it may be empty.

Data types are described in some kind of data definition language. The creation of a container in which instances can be recorded may be a separate event from the definition of the associated schema in which the data types are defined. Instances of a type can only be recorded after a container has been created.

The set of rules which governs how the data instances in a data container must conform to the types of data with which they are associated is collectively called a data modelling facility. One or more data modelling facilities may be associated with each level pair.

The semantics of each service provided for a level pair are necessarily specific to one data modelling facility.

6.4 Identification of Data Levels

The IRDS functionality is associated with three data levels, but it is useful for expository reasons to cover four data levels.

The four data levels described are as follows:

a) IRD Definition Schema Level;

b) IRD Definition Level;

c) IRD Level;

d) Application Level.

These levels, while being inter-related, exist to serve different purposes, as will be described later. The levels are illustrated in Figure 5.

6.4.1 IRD Definition Schema Level

The purpose of the **IRD Definition Schema level** is to prescribe the types of object about which data may be recorded on the IRD Definition Level.

The definition of the types of data that can be stored on the IRD Definition level is called the **IRD Definition Schema.**

6.4.2 IRD Definition Level

The purpose of the **IRD Definition Level** is to contain IRD definitions.

The types of data whose instances are recorded in **IRD Definitions** are defined on the IRD Definition Schema Level.

There may be any number of IRD Definitions existing, all described by one IRD Definition Schema.

A part of an IRD Definition, referred to as an **IRD Schema**, prescribes the types of object about which data may be stored in one or more IRDs.

An IRD Definition may contain one or more IRD Schemas. Some of the content of the IRD Definition may be under development and intended to replace the content of an IRD Schema or to add to the definitions in an IRD Schema. Other definitions may have previously been replaced and are categorised as "archived". This is illustrated in figure 5.

The content of an IRD Schema may be defined in three ways:

a) by International Standards;

b) by a supplier of IRD definitions;

c) by a user.

At any point in time an IRD Schema is a subset of an IRD Definition, consisting of a part of the IRD definition that the dictionary administrator has chosen to make active.

An IRD Definition includes data to support added value facilities such as those described in 7.3.

6.4.3 IRD Level

The purpose of the **IRD Level** is to contain IRDs.

There may be any number of IRDs existing, all described by one IRD Schema. There may also be other IRDs described by other IRD Schemas.

Some, but not all, of the content of an IRD defines types at the application level.

For example, an IRD would contain the information that EMPLOYEE and PURCHASE ORDER are two record types. An IRD might also contain information about which programs use these record types.

Figure 5 shows that some of the information in an IRD will define the currently active application schemas. Other information in an IRD will include non-active application schemas and other analysis, design and control information about the applications.

The purpose of the data in an IRD is to enable an IRDS to support the design, construction and operation of computerised information systems, and any other functions for which the IRDS is seen as an appropriate tool.

Most data instances in an IRD are placed there by a business analyst, an information systems designer or an information systems builder. This may be done explicitly and consciously, as part of the activity involved with planning, analysing and designing information systems to support the running of the enterprise. In some cases, the recording of data in an IRD may be done automatically by a software product (for example, a language compiler) which supports a particular development activity. Some of the data in an IRD makes it possible for the dictionary administrator to carry out the activities for which he is responsible.

The types of data in an IRD are completely defined by the data held in the applicable IRD Schema. Some of these types will be defined in one or more International Standards in the IRDS family or possibly in other standards. Furthermore, the types of data in an IRD may be extended by an implementor of a product, an installer of a product or by a dictionary administrator.

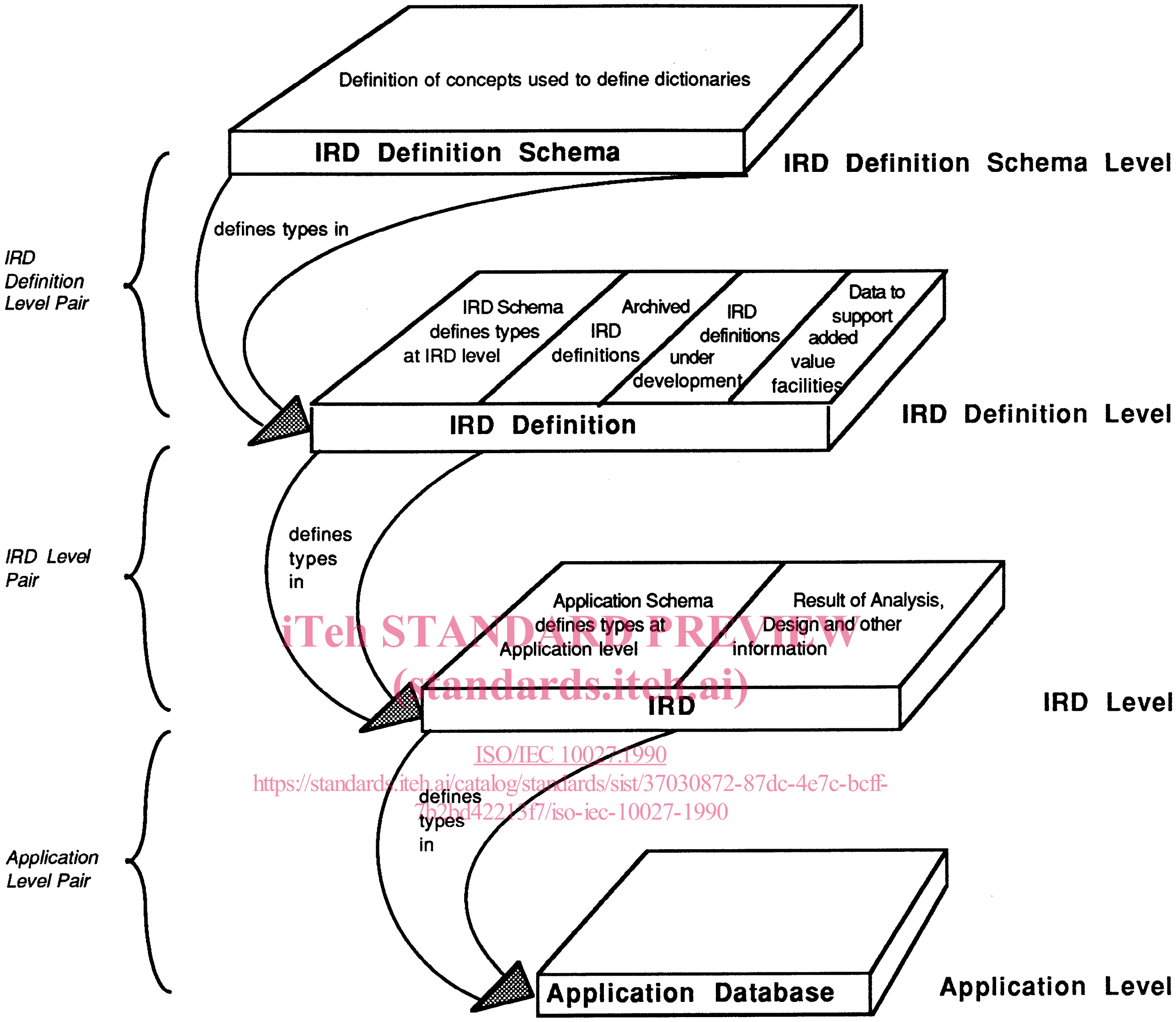
Definition of concepts used to define dictionaries
IRD Definition Schema
IRD Definition Schema Level
defines types in
IRD Definition Level Pair
IRD Schema defines types at IRD level
Archived IRD definitions
IRD definitions under development
Data to support added value facilities
IRD Definition
IRD Definition Level
defines types in
IRD Level Pair
Application Schema defines types at Application level
Result of Analysis, Design and other information
IRD
IRD Level
defines types in
Application Level Pair
Application Database
Application Level

Figure 5 - Levels and the IRDS

An IRD Definition specifies further types of data in an IRD, necessary for day to day administration by a dictionary administrator.

Some of the data instances in an IRD may be defined in an International Standard. Other content of an IRD may be added.

There is no restriction on the types of data which may be held on the IRD level, provided that the instances of this data correspond to types in the applicable IRD Schema.

6.4.4 Application Level

The **Application Level** is the level on which instances of business data are recorded.

The data associated with the application level is for the user of the information system. Such data helps to run the business activities of an enterprise in some way. Thus, the data about specific instances of EMPLOYEE will be recorded on the Application Level.

For example, the data pertinent to John Smith who happens to be an employee would be on the Application Level, as would data about PURCHASE ORDER 738942, as issued on 17 May 1989.

Data about the type, namely about the general concept of EMPLOYEE will be recorded in an IRD. Thus, the types of data are recorded in an IRD and the corresponding instances are on the Application Level. Application source programs refer to the data types which are specified at the IRD Level. When such programs are executed, then data instances on the Application Level are retrieved and possibly also updated.