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

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.

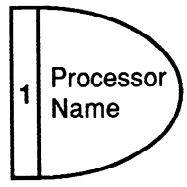

Figure 1 - Notation for a processor and its interface

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.

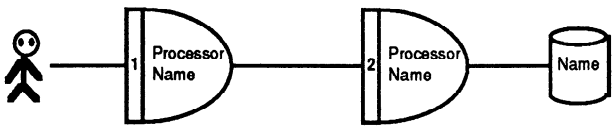

Figure 4 - Conventions to show Client - Server associations

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.

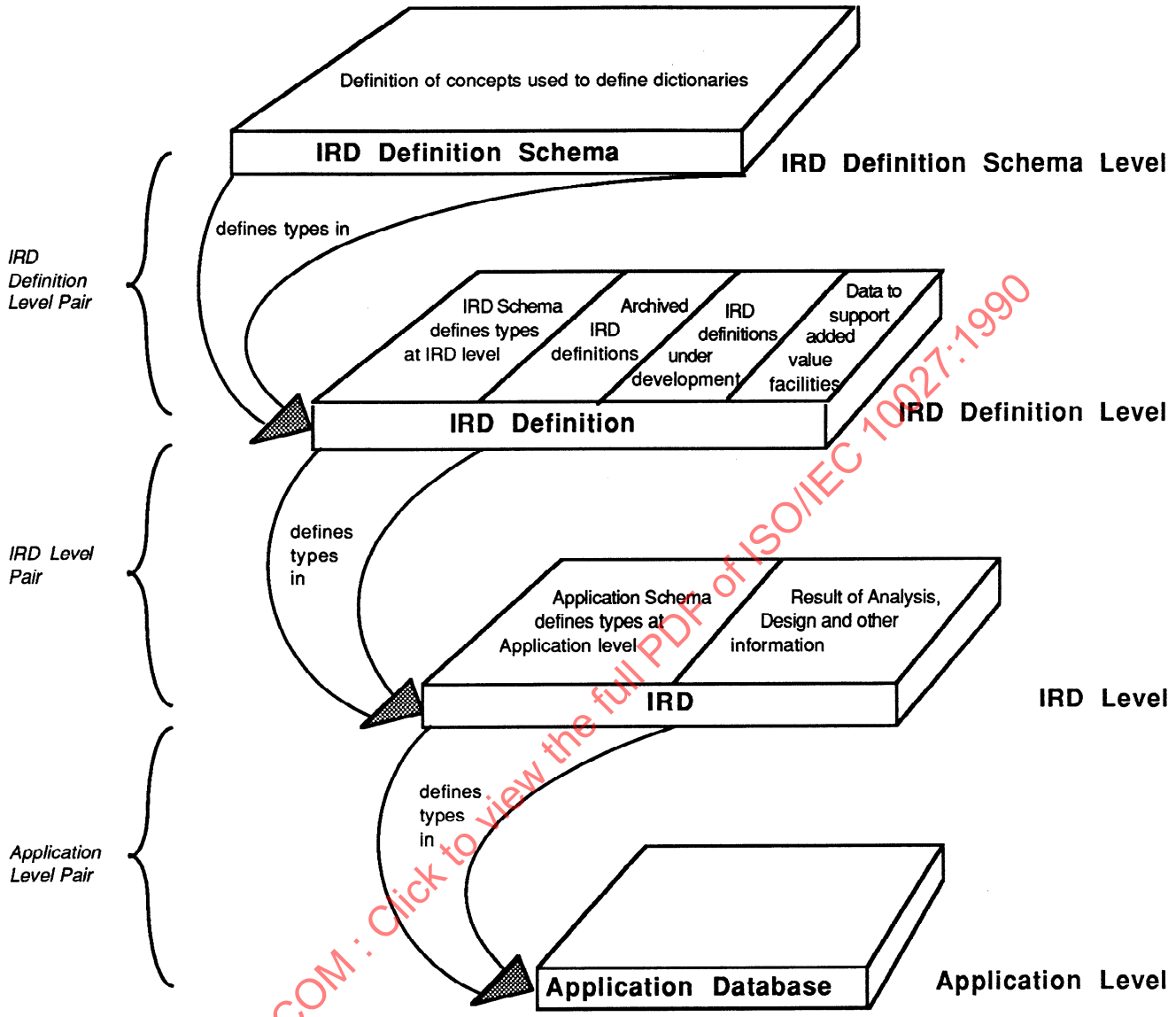

Figure 5 - Levels and the IRDS

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.

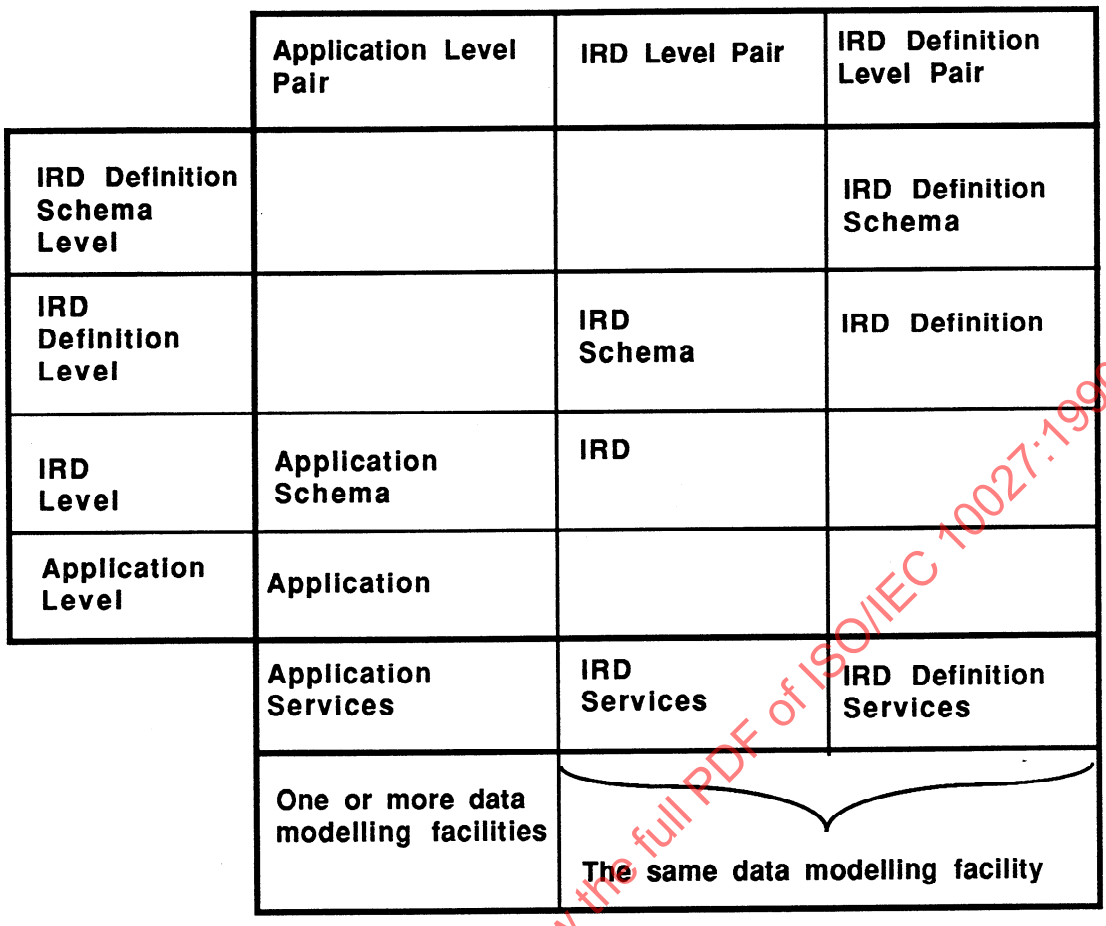

Figure 6 - Use of data modelling facilities and services

The data on the Application Level is processed using programs built by or generated by an information systems builder. General-purpose programs suitably adapted by the information systems builder may be used. These programs may be written in an International Standard programming language, or built in some other way, perhaps using a higher level application generation facility. In either case the programs or higher level language facilities will always include references to types of data defined in an IRD.

6.5 Level Pairs

An IRDS provides services at two levels: the IRD Definition Level and the IRD Level. In each case the services may be thought of as operating on a **level pair.** A level pair consists of two adjacent data levels. The upper level of the two will always contain the "type" of information relevant to the "instances" on the lower level of the two. That part of the data at the upper level of the two which defines types for the level below, is termed the schema for the lower level.

The successful operation of an IRDS requires the data on the lower level of any level pair to be consistent with a specified version of its schema at the higher level of the level pair.

If a schema is changed, for which data exists at the lower level, consistency must be maintained between that data and the schema.

It is important to note that the upper level of a level pair may also contain information which is not type information associated with lower level instances.

The three level pairs are identified as follows (abbreviations in parentheses):

a) IRD Definition Level Pair (IRDD_LP)

b) IRD Level Pair (IRD_LP)

c) Application Level Pair (AP_LP)

Only the first two level pairs are the subject of the IRDS family of International Standards. Each service relates to a level pair and never to a single level. Figure 5 shows the levels and level pairs and how they inter-relate. Figure 6 illustrates the terminology associated with the data levels and level pairs. Figure 7 gives examples of objects at the four levels as they are interpreted in the context of the relevant level pairs.

Each level pair is now described in turn.

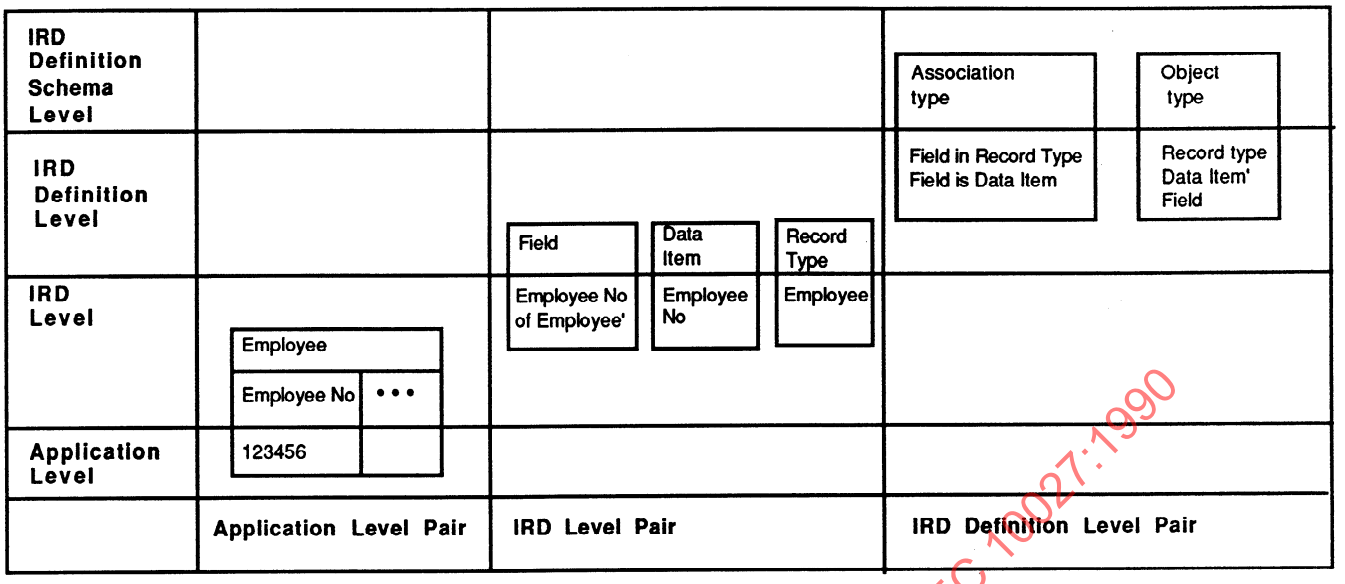

Figure 7 - Examples of data types and data at data levels

6.5.1 IRD Definition Level Pair

The **IRD definition level pair** consists of the IRD Definition Schema Level and the IRD Definition Level.

The IRDS family of International Standards define IRD Definition Services which operate on the IRD Definition. These services operate by reference to the IRD Definition Schema on the IRD Definition Schema Level.

A dictionary administrator will use IRD Definition Services to define the types of data that subsequently may be used at the IRD Level by an application designer.

6.5.2 IRD Level Pair

The **IRD level pair** consists of the IRD Definition Level and the IRD Level.

The IRDS family of International Standards defines IRD Services which operate on an IRD. The IRD Services operate by reference to the applicable IRD Schema on the IRD Definition Level.

When the designer of an application system is evolving and documenting the design of a database or of a set of application programs, he/she will use the IRD Services provided on this level pair.

6.5.3 Application Level Pair

The Application Level and IRD Level together represent the **application level pair**.

No functionality is prescribed for the application level pair in the IRDS family of International Standards. Furthermore, the services prescribed in the family of IRDS International Standards do not impinge on this level pair in any way.

It is important to note that International Standards for Database Languages and for Programming Languages do provide functionality which is normally associated with the application level pair.

Although the IRDS family of International Standards does not provide explicit services for the application level pair, there is a clear possibility that other International Standards defined for other services associated with the Application Level Pair (for example query languages) might require access to data associated with the IRD level, such as a record description or a screen format description.

6.6 Role of data modelling facilities

One data modelling facility is used at the IRD Definition Schema Level to define the data at the IRD Definition Level. The same data modelling facility is also used at the IRD Definition Level to define the data at the IRD Level. One or more data modelling facilities may then be used at the IRD Level to define data at the application level. This allows the IRDS to support the definition of application databases using any data modelling facility.

For example, two data modelling facilities may be defined in the IRD Definition, and each would be included as part of an IRD schema. For the same given area of interest, two application schemas may then be developed, one corresponding to each data modelling facility. One of these **application schemas** may be such that no application database container would be instantiated, whereas a container would be instantiated for the other. The former data modelling facility would be one for which no data manipulation services have been defined, whereas the latter might be that provided in an International Standard database language.

This Framework does not prescribe a limit on the number of different data modelling facilities which may be defined in an IRD Definition.

6.7 Extensibility

The types of data which can be maintained at the IRD Level are themselves defined at the IRD Definition Level in the IRD Schema. It is possible to add new types of data at the IRD level and also to modify existing types of data.

An extension may be specified in the form of an International Standard, which may belong to the IRDS family of International Standards or to some other family of standards. Alternatively, an extension may be specified either by a National Standard, an IRDS implementor, a vendor of IRDS extensions or by a user.

7 IRDS facilities

7.1 Classification of IRDS facilities

A definition of an IRDS is given in 5.1 above. Since an IRD is a database covering data in a particular data subject area, that of information resource management, it is convenient to classify the facilities of an IRDS accordingly. Those facilities which an IRDS shall provide because of its role as a database management system are described in 7.2; those which arise from the nature of the data subject area are covered in 7.3.

Throughout this clause the emphasis is on the facilities as perceived by the user. Where possible this Framework specifies only facilities as perceived by users of an IRDS, and does not concern itself with the mechanisms used to provide those facilities. Subsequent International Standards in the IRDS family will provide such detailed specifications.

7.2 General database management facilities

Other International Standards, particularly those concerned with database languages, specify general-purpose database management facilities. While an IRDS is not a general-purpose database management system in this sense, it does have to provide to its users many of the same features, as detailed below. It may well do this using a general-purpose database management system, but is not required to do so.

In addition to the facilities mentioned explicitly below, an IRDS shall provide services to insert, update and delete data under its control, and to select and retrieve data.

7.2.1 Enforcement of constraints

An IRDS shall provide means of specifying **constraints** on the values associated with objects, and on the associations between objects. These constraints shall include, but are not limited to, validation of individual values, and cardinality constraints on named types of association between objects. When a constraint is specified, it shall be possible to state under what circumstances it shall be enforced.

An IRDS shall ensure that all data within an IRD or an IRD Definition is consistent with such constraints at all times when the data is generally available to suitably-authorised users, and when indicated by the definition of the constraint. It shall be permissible for the enforcement of constraints to be switched off under certain circumstances, such as during any period when the relevant data is accessible only to a single user.

7.2.2 Access Control

An IRDS shall provide means of limiting access to data in an IRD or an IRD Definition, or operations on that data, to suitably authorised users. Such authorisation may be in terms of types of data, development status of the data concerned or other relevant criteria. It is therefore to be understood that all facilities specified in the remainder of clause 7 are available only to suitably-authorised users.

An IRDS may also provide means to limit access to individual objects within an IRD or an IRD Definition, or even to individual attributes of those objects.

The process of authentication of the identity of a user is outside the scope of the IRDS Framework.

7.2.3 Audit trail

An IRDS shall provide means of optionally **auditing** changes to an IRD or an IRD Definition. The application of auditing facilities shall be optional at least at the installation level.

7.2.4 Limits and defaults

An IRDS shall allow the specification of limit and default values for the values of specified attributes, and shall then apply these when objects are created or modified.

7.2.5 Database integrity

An IRDS shall ensure that the integrity of the data in an IRD or an IRD definition is preserved in both a single-user and a multi-user environment (if such an environment is supported). The mechanism for achieving this is implementor-defined.

Such support shall include the ability to recover from system or program failure, by automatic rollback, or to process a user initiated rollback.

7.2.6 Query and reporting facilities

In addition to any specific implementor-defined reports, an IRDS shall provide general-purpose query and reporting capabilities on an IRD or an IRD Definition, either directly or by providing access for general-purpose query and reporting facilities.

7.2.7 Remote data access

There is no assumption in this Framework that the user of IRDS services, at any of the IRDS interfaces defined in clause 8, shall be required to be located at the same real system as the IRDS.

7.3 Facilities specific to information resource management

A large part of the task of information resource management is concerned with controlling the development,

introduction and use of information processing systems. The following IRDS facilities are those which assist in this task, and which go beyond the scope of a general-purpose database management system.

7.3.1 Naming

An IRDS shall provide facilities for any object within an IRD to be identified externally by a unique name, if the definition of the object type so specifies or the user creating the object so chooses. If an IRDS implementor provides a unique identifier for an object where the user did not provide one, no user shall need to be aware of any such system-generated identifier.

The name spaces for each IRD and for each IRD definition shall be distinct.

The ability shall be provided to impose constraints on the naming at each level, either globally or by object type.

7.3.2 Status of dictionary content

Data that is not stable is termed "uncontrolled". Data that is stable is termed "controlled". Data that is no longer used is termed "archived". An IRDS shall enable the user to distinguish between these states.

7.3.3 Information system life cycle management

To assist in controlling the development, introduction and use of information processing systems, an IRDS shall support the concept of a **life cycle** split into a number of phases.

The concept of **life cycle phases** shall apply at both IRD and IRD Definition levels.

7.3.4 Version control

In the development, introduction and use of information processing systems there is commonly a need for two or more versions of an object or collection of objects to be of interest concurrently. An IRDS shall therefore provide means for users to create and maintain versions of objects or user-defined groups of objects, and to indicate when necessary which version is of interest.

7.3.5 Partitioning

For convenience of information resource management and control, both an IRD and an IRD Definition may be divided into partitions. A **partition** is a logical subdivision of an IRD or an IRD Definition. Any object shall be in only one such partition; however, other objects representing different versions of the same information resource may exist in other partitions.

7.3.6 Subsetting

To control user access and provide each user with a convenient context, it shall be possible to define subsets of an IRD or an IRD Definition. Each subset shall provide update access to no more than one partition, and optionally read-only access to other partitions. Such a subsetting mechanism may also be used to constrain access to only a specified version or versions.

7.3.7 Copy creation

To enable users to create similar objects and groups of objects without unnecessary duplication of input, an IRDS shall provide facilities for a user to make a copy of an object or group of objects. These facilities may or may not extend to the subsequent management of identical copies.

7.3.8 Impact analysis

An IRDS shall provide means for answering queries such as those concerned with impact analysis. In particular, the concepts of impact dependency and query path shall be supported at both IRD and IRD Definition levels.

An impact dependency of an object A on an object B in an IRD or an IRD Definition indicates that if B is modified, A is considered to have been modified.

A query path is a predefined means of accessing a collection of related objects, potentially of more than one object type.

8 IRDS Interfaces

8.1 Positioning of interfaces

Figure 8 shows diagrammatically the major interfaces covered in this Framework. Each processor in the figure may act as a server to one or more clients, shown to the left of the processor.

The figure is divided into two parts by a dashed line. To the left of the dividing line are shown the processors which collectively comprise the IRDS Framework.

To the right of the dividing line are shown two processors and storage media. These two processors and their interfaces are non-normative for the IRDS Framework International Standard, but are included here to indicate that use may be made of some kind of Database Services interface to access the data managed by the IRDS.

Each storage medium represents one of the data levels discussed in clause 6.

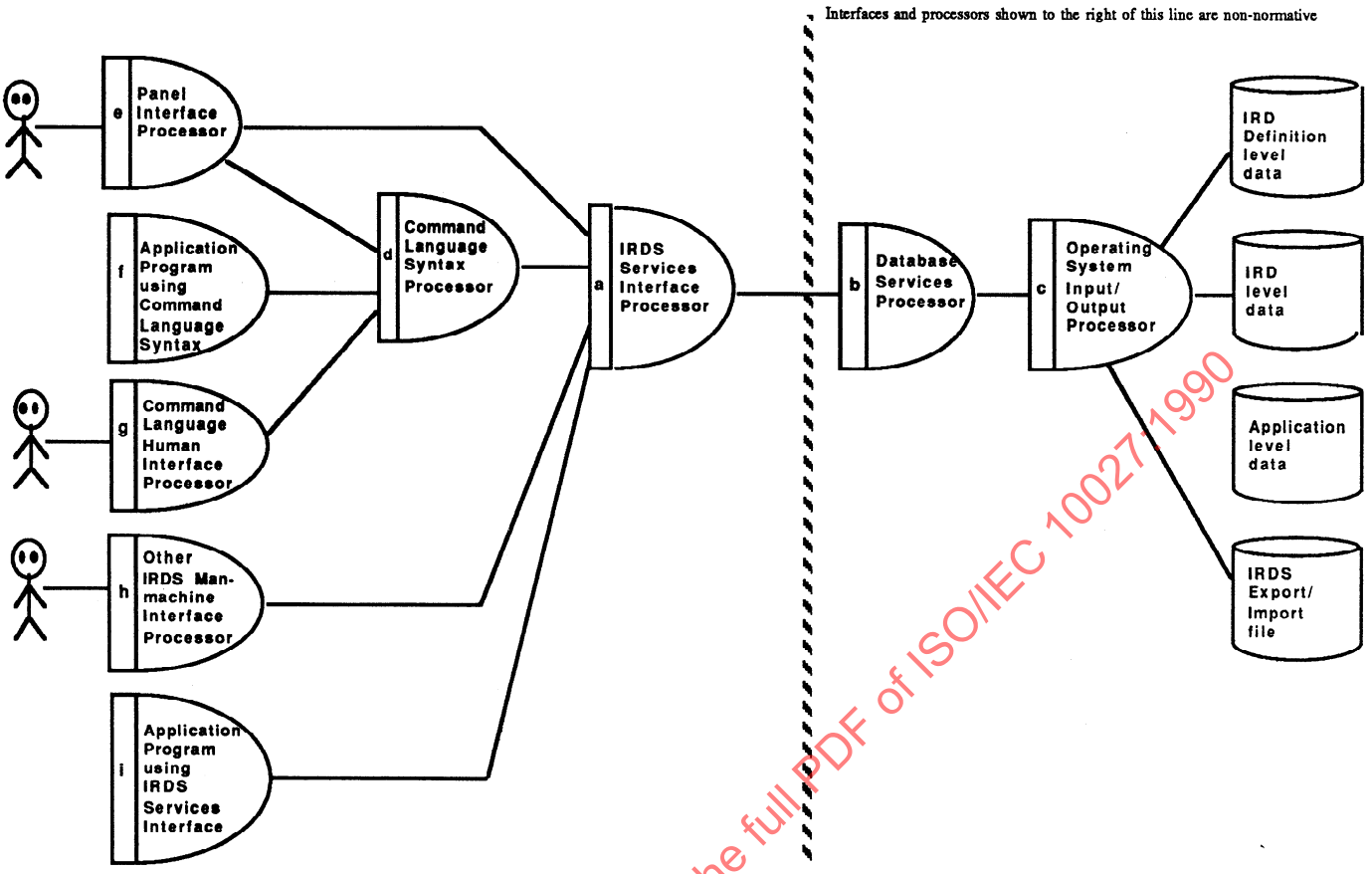

Figure 8 - Major IRDS Processor Interfaces

8.2 IRDS Interfaces Overview

Figure 8 positions several of the interfaces relevant to the family of International Standards for an IRDS.

These interfaces are

a) the IRDS Services Interface;

b) a database services interface;

c) an operating system Input/Output processor interface;

d) the Command Language Application Program Interface, supported by the Command Language Syntax Processor;

e) the IRDS Panel Interface;

f) a class of interfaces, provided by application programs using Command Language syntax through the Command Language Application Program Interface;

g) the IRDS Command Language Human Interface;

h) a class of interfaces, provided by other IRDS man-machine interface processors;

i) a class of interfaces, provided by application programs using the IRDS Services Interface.

The interfaces lettered a), d), e) and g) are described in more detail in 8.3. Certain of these interfaces are prescribed in complete detail in other International Standards in the IRDS family of International Standards. The distinction between interfaces f), h) and i), is for informative purposes only.

An application program or a vendor provided product is not precluded by this International Standard from accessing more than one IRDS interface concurrently.

8.3 Description of Interfaces

8.3.1 IRDS Services Interface

The IRDS Services Interface is a processor to processor interface.

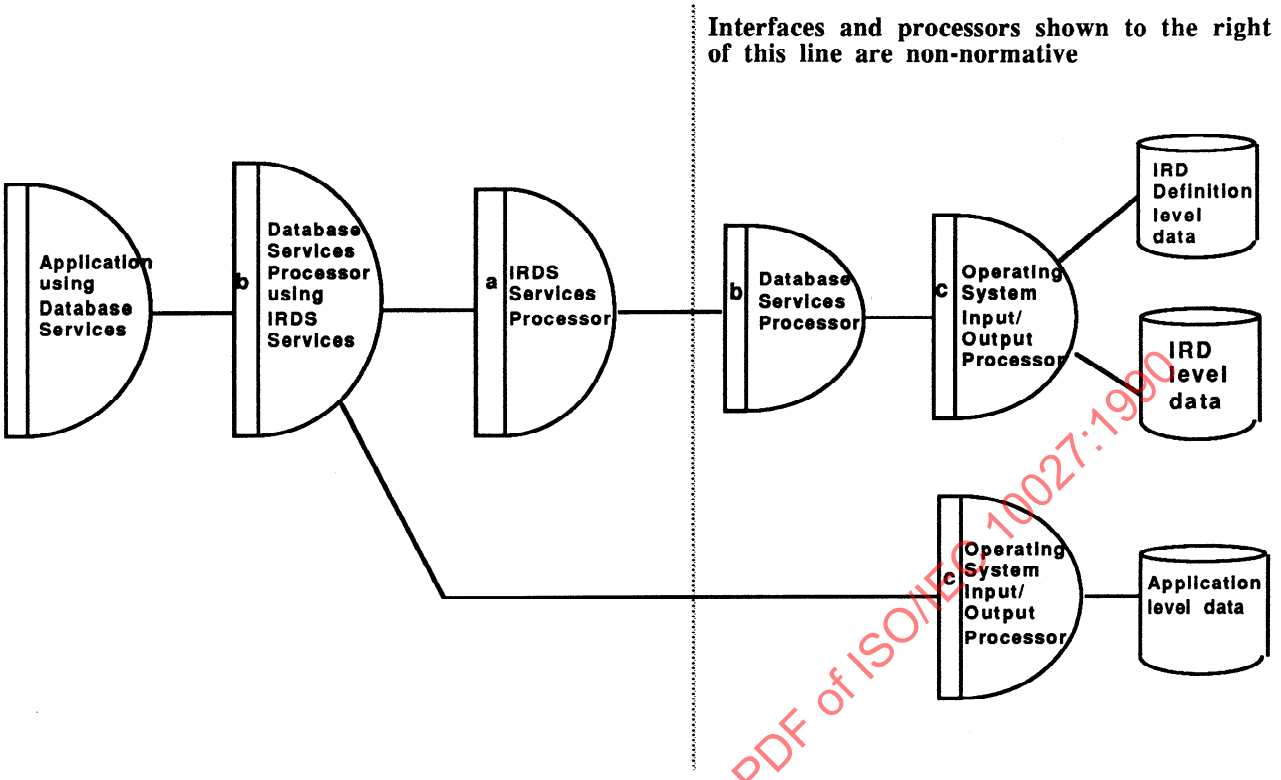

Figure 9 - A Database Services processor using the IRDS Services Interface

The IRDS Services Interface provides all the services needed by any processor wishing to access and manipulate IRDS data. This interface could be used by processors such as Panel Interface Processors, Command Language Syntax Processors, other IRDS Man-machine Interface Processors and Application Programs wishing to access IRD Definition Level and IRD Level data.

This International Standard does not preclude an implementation of an IRDS offering concurrent access through a single IRDS Services Interface to more than one IRD.

The IRDS Services Interface is provided as the primary means of access to data on the IRD Definition Level and on the IRD Level. All access to IRD Definition Level and IRD Level data shall be through the IRDS Services Interface.

Each service available at the Services Interface is associated with either the IRD Definition Level Pair or with the IRD Level Pair.

Many other processors may make use of the services at the Services Interface. Several such processors are shown in figures 8 and 10. Some of these may relate to programming language standards or to Database Language standards. In addition, International Standards developed to support the fields of application referred to in 5.2 may identify or require processors which may make use of the services provided at this interface.

The IRDS Services Processor may make use of the services provided at the Database Services Interface to access the data on appropriate data levels, as illustrated in figure 8. Alternatively, the IRDS Services Processor may access this data directly or make use of operating system services.

This Framework does not prescribe which of these three alternatives should be used.

8.3.2 IRDS Panel Interface

A Panel Interface is a style of interface for human use. It consists of a set of panels or screen formats, each providing access to a prescribed set of sevices defined in terms understandable to human users who may be non-technical. Each panel may reference data on either the IRD Definition Level or on the IRD Level.

The data modelling facility used to define the data seen by a user of a Panel Interface may differ from that used at the Services Interface. It is also possible for a different data modelling facility to be used at each of the two data levels for which services are provided. There may thus be more than one Panel Interface for the same Services Interface, depending on the data modelling facility which is felt to be appropriate.

Each mapping between data modelling facility used at the Services Interface on a data level and a data modelling facility used at a Panel Interface on the same data level

shall be performed by the relevant Panel Interface Processor.

8.3.3 IRDS Command Language Interface

A Command Language Interface is a style of interface oriented towards use by human users. It defines a means of invoking IRDS Services using statements expressed according to a concrete syntax, consisting of a set of syntactic rules which must be adhered to by a user preparing the statements. Each statement defines a service in terms understandable to people. Each statement may reference data on either the IRD Definition Level or on the IRD Level.

The data modelling facility used to define the data seen by a user of a Command Language may differ from that used at the Services Interface. It is also possible for a different data modelling facility to be used on each of the two data levels for which services are provided. There may thus be more than one Command Language for the same Services Interface, depending on the data modelling facility which is felt to be appropriate.

Each mapping between a data modelling facility used at the Services Interface on a data level and a data modelling facility used at a Command Language on the same data level shall be performed by the relevant Command Language Processor.

8.3.4 IRDS Command Language Application Program Interface

The Command Language statements may be called from or embedded in an International Standard programming language. This possibility is depicted in Figure 8 as interface d) which is a separate interface from the human Command Language interface.

8.3.5 Other interfaces

The other interfaces shown in figure 8 may or may not be candidates for standardisation.

8.4 Use of IRDS Services by Database services

Database services processors may themselves call IRDS Services as illustrated in figure 9. This allows a database management system to store information about the structure of databases in the IRD. Thus the integration of database management systems and the IRDS is enabled.

8.5 Export and Import of IRDS data

A copy of some or all of the data managed by an IRDS may be transmitted from one **real system** (in the OSI sense) to another. This may be done in two ways.

The first method is by making a copy, called an **Export/Import file**, on a storage medium and then either sending the copy by a telecommunications link or by physically transferring the medium from one real system to the other. In this case the IRDS places no constraints on how the Export/Import file is transported.

The process of making the Export/Import file on the storage medium is called "**export**" and the process of transferring the data from the Export/Import file to the second IRDS at the second real system is called "**import**".

The second method is for two IRDS's at two different real systems to communicate directly using OSI protocols to transfer information between them. This International Standard does not prescribe interfaces for this type of communication.

In both cases, the process of "import" involves checking the consistency of the imported data.

The data which is exported or imported may be on any of the bottom three data levels. Only data on the IRD Definition Level and on the IRD Level is covered by this Framework International Standard.

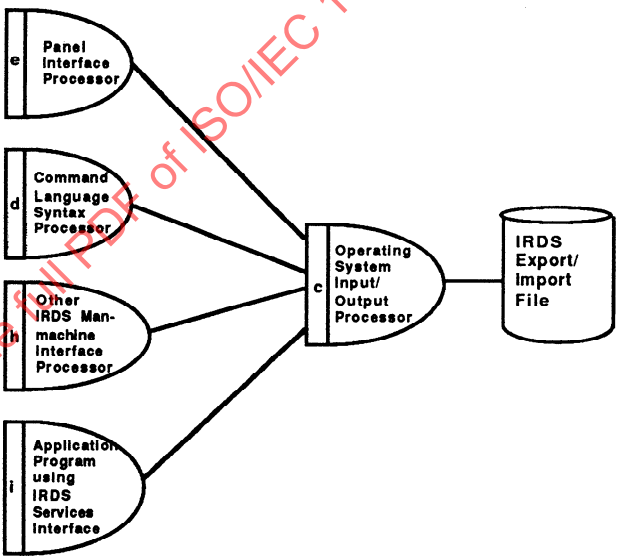

Figure 10 - IRDS Export/Import File

The Export/Import File provides a common data format for exchange of IRDS data between different IRD's.

The IRDS Export/Import file format is prescribed by a separate IRDS International Standard.

Files in this format can be created and accessed by IRDS processors via operating system Input/Output Services. This is shown in figure 10.

8.6 Remote IRDS Access;

It is necessary to be able to access the IRD Definition data and IRD data located at one real system from another real system.

This requirement differs from that of Export/Import in that the amounts of data accessed are small and OSI protocols would be used in all cases.

8.7 Services provided at Interfaces

Several services are available at each interface making up the IRDS Framework.

Each service relates to either the IRD Level Pair or to the IRD Definition Level Pair. In some cases, similar services may be provided for both of these two level pairs, but for definitional convenience they must be defined as separate services.

Many of the services provided at the interfaces described in the IRDS Framework are similar to those provided at a typical Database Services Interface. Other IRDS services are not typical of database services.

This IRDS Framework does not identify each individual service as this is the role of International Standards based on this Framework.

The facilities described in clause 7 are provided through the IRDS Services.

9 Conformance

Conformance requirements are not stated in this International Standard. Rather, they are stated in each of the other Standards in the IRDS family of International Standards.