IREDES Initiative
Goethestrasse 52
D- 49549 Ladbergen
www.iredes.org

IREDES initiative
Goethestrasse 52
D - 49459 Ladbergen

IREDES
Standard for
Mine Equipment Communication

Version 1.2

Standard Architecture

D- Ladbergen, 2004-12-01

IREDES Initiative
Goethestrasse 52
D- 49549 Ladbergen
www.iredes.org

Table of Contents

IREDES Initiative
Goethestrasse 52
D- 49549 Ladbergen
www.iredes.org

1 Introduction

1.1 Purpose of the IREDES standard

IREDES is an industry standard to unify routines for the data exchange between mining equipment (machines) and office computer systems.

As the international GSM cellphone standard enables different makes of mobile phones to globally talk to different brands of base stations, it is the goal of IREDES to become the standardized data interface for mining equipment. This will enable machines talking among each other (horizontal integration) and to make corporate IT systems talk to the machines and production equipment (vertical integration). Individual development of any software interfaces to „just make the computer talk with the machine“ is not required any longer when equipment and computers „talk“ the IREDES worldwide standardized language.

The IREDES standard is set up by organizations from all over the world to meet the local requirements of the machine users. This is a precondition to make equipment usable globally and to use unified information from local mining sites for enterprise level decision finding in globally operating mining companies.

IREDES is a not for profit initiative founded by major players in the mining industry. It is jointly financed by the members on a cost sharing basis.

The standard documents are open to the public. They are accessible free of charge for any IREDES member. Non-members can purchase the documents from the IREDES office for a small contribution to the standardization work.

1.2 Benefits

The use of standards as the IREDES standard has a number of important benefits:

- Simplification of the purchasing process: The simple statement „*IREDES compliant data exchange*“ replaces extensive definitions of software and interfaces in individual tender documents.
- Substantial cost savings in development, use and maintenance of software and equipment.
- Equipment of different vendors can be used within an identical IT infrastructure using identical data exchange.
- Concentration of effort on the development of machine functions and the use of information instead of permanently reinventing the „interface wheel“.
- IREDES itself completely bases on open, established international standards, maintained by IT - provider independent organizations.
- New markets for price efficient add-on devices, software and equipment

1.3 How to use the standard

If your company is committed to the use of international standards instead of expensive individual development, here are some simple rules of how to make use of the IREDES benefits:

Mining companies:

- State „*IREDES compliant data exchange*“ in all tender documents and purchase orders for all future purchase of equipment and equipment related software to be covered by the IREDES standard. An overview of the available and coming profiles is available via the IREDES web site.
- When a new machine is put into operation, the suppliers has to submit an „IREDES schema“ on electronic media. Using this schema, standard software and database systems are able to import and export IREDES information without any software development.
- To make use of the real time machine networking features of IREDES, mine communication systems should be capable of handling the standardized TCP/IP range of networking protocols. Standard Ethernet is recommended for network connections.
- Existing internal databases and software systems can easily be made compatible with the new standard by developing easy to operate interfaces. Corresponding tools and libraries are expected to become commercially available soon.
- Existing machines can be integrated using „on top“ devices on the machines or by using separate PC’s to convert existing proprietary data formats into the new IREDES standard and vice versa.

Consequently, using IREDES means to be able to integrate machines from different vendors seamlessly into the corporate IT infrastructure.

Equipment suppliers:

- When planning a new or renovated product the automation system should be designed capable of handling the IREDES standard directly. If the machine is going to be network compliant, Ethernet and / or modem – interfaces shall be available to assure network connectivity.
- For existing products „retrofit packages“ or external (off-board) solutions may be offered to integrate such machines into IREDES compliant mining installations.
- Provided desktop software should be designed IREDES compliant to handle all data traffic to the machines via IREDES protocols exclusively.

1.4 Content Summary

This document was created in order to help understanding the IREDES Architecture. It introduces into the general concepts of IREDES compliant data exchange. It is a

guide on how IREDES data structures look like and how to add new machine information items to it.

This document explains...

...how the IREDES Standard definition is structured.

...how data is exchanged between communication partners.

...how the XML technology used for IREDES.

...how XML files (Data Sets) are structured in IREDES compliant communication.

...how vendor/mine specific structures can be added to IREDES.

...how IREDES is implemented in Embedded Systems and Central IT infrastructures

1.5 Audience

This document addresses the following readers:

- Equipment Automation Managers
- Mine Automation professionals
- IT system architects
- Decision makers and implementors of mining production related software and IT systems

Readers should be familiar with general XML concepts and object oriented software design.

It is not the intention of IREDES documentation to give an introduction into XML and XML schema related concepts nor details.

IREDES Initiative
Goethestrasse 52
D- 49549 Ladbergen
www.iredes.org

2 Document history

2000-01-04	V0.1	Initial preparation of pre-release document
2000-02-15	V0.2	First internal release for key customer presentation
2000-02-16	V0.2a	Add on of ideas for file preparation and analyses
2000-05-05	V0.3	Implementation of more flexible tables and coordinate system usage; Integration of spec. tables into main documents
2000-05-29	V0.4	File format conversion, smaller changes in text
2000-06-05	V0.4a	Review and simplification of this architecture description
2000-07-24	V0.4b	Address corrections, small redactional corrections
2000-09	V0.5	Introduction of DTD's to enable XML semantic checking Introduction of Networking Features
2000-10-08	V0.5	Document release V0.5
2002-04	V0.95	Complete renewal of entire document acc. to IREDES prototyping version
2003-02	V1.0	Document review for release
2003-04	V1.0b	Document review basing on changes made on latest meetings; Introduction of inheritance structure for equipment profiles
2004-12	V1.2	Introduced changes from field experience, General Document review Changes due to separated version handling

3 Referenced Standards

ISO 8859-1: 1987, Part 1: Latin Alphabet No 1

XML Schema Part 0: Primer: http://www.w3.org/TR/xmlschema-0/

XML Schema Part 1: Structures: http://www.w3.org/TR/xmlschema-1/

XML Schema Part 2: Datatypes: http://www.w3.org/TR/xmlschema-2/

XML Base: http://www.w3.org/TR/xmlbase/

Extensible Markup Language (XML) 1.0: http://www.w3.org/TR/2000/REC-xml-20001006

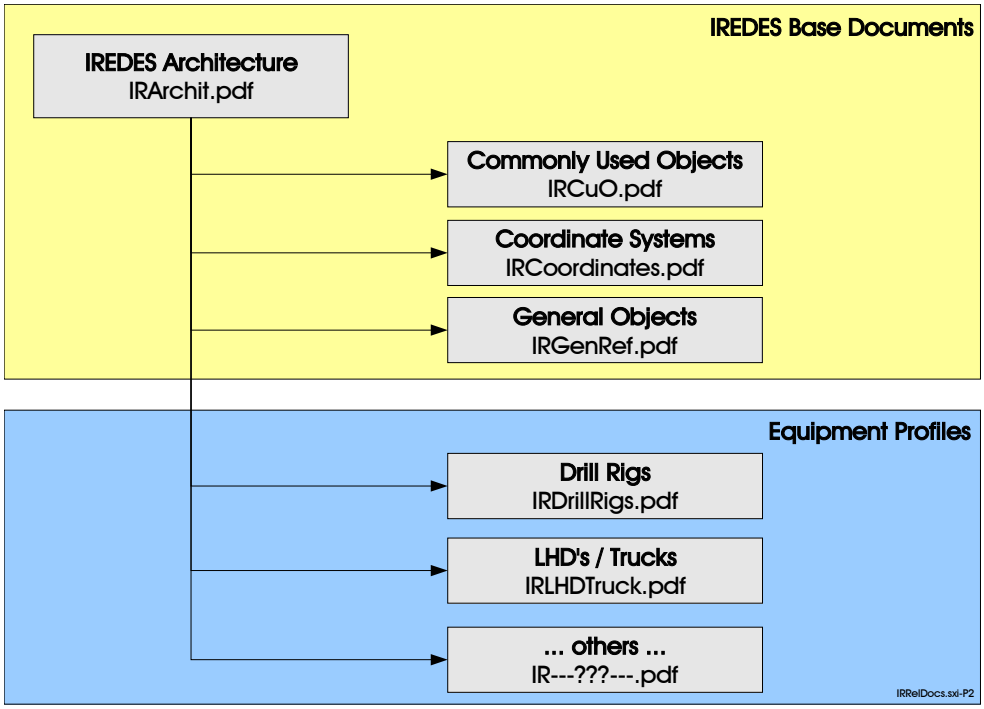
XML Namespaces: http://www.w3.org/TR/REC-xml-names

RFC1321: http://www.faqs.org/rfcs/rfc1321.html

IREDES Initiative
Goethestrasse 52
D- 49549 Ladbergen
www.iredes.org

4 Related Documents

This document is a part of a set of documents describing the different parts of the IREDES standard:



The uppermost document is the IREDES Architecture description explaining the general setup and collaboration of the different parts of the standard. The „IREDES Architecture“ document will be the best choice to start with as it gives an overview and basic information needed to understand the structure of IREDES.

Detail information about single parts of the IREDES standard are available from separate documents. Readers of these documents should be familiar with XML, XML schemas and implementation relevant issues.

Standard definitions used in common throughout multiple equipment profiles are covered by the corresponding documentation of the „General Objects“ (Application Profiles, General Data Types etc) and „Commonly used Objects“ (CuO's).

All standard information concerning a particular equipment type (Drill Rigs, LHD's,...) is contained in a separate document related to the equipment profile.

Beside the textual descriptions, the entire standard is available as xml schemas as they contain the formal description of the standard. In case of ambigouities, the definitions in the xml schema override all definitions made in the accompanying textual documentation. Textual description available for different IREDES profiles is additional information mainly containing parts of the standards not defineable in xml schemas. Implementors should take care of these documents as they may contain important information which must be defined basing on „mutual agreement“ to make the standard work.

5 IREDES Technical Overview

5.1 Design principles

To set up the IREDES standard few design principles were set

The IREDES standard is capable of covering data exchange demands for entirely different types of machines. As many identical machine components are used for entirely different machine types (e.g. Diesel motors), the IREDES standard makes broad use of definitions already made in the standard:

1. Intense Re-Use of components defined once in the standard

A standard should be flexible to allow individual data exchange needs to be covered with a minimum of changes required. This opens up for new definitions to be introduced and tested optionally before they become part of a future standard version:

2. Provide a maximum of optional flexibility without touching the basic standardized data exchange

IREDES covers reporting information that can be easily transferred by memory media like Flash-Cards or diskettes as well as „volatile“ information like an online machine status or momentaneous power consumtion which only makes sense to be transferred online via networks. To make both ways compatible:

3. Networking data exchange has to use the same data formats as file based data exchange

One particular type of equipment does not necessarily have to „know“ about the existance of other machine types or data exchanges requirements. A smart architecture allows to extend the standard without the necessity to upgrade existing machines:

4. Easy extendability by new Equipment Profiles and Application Profiles

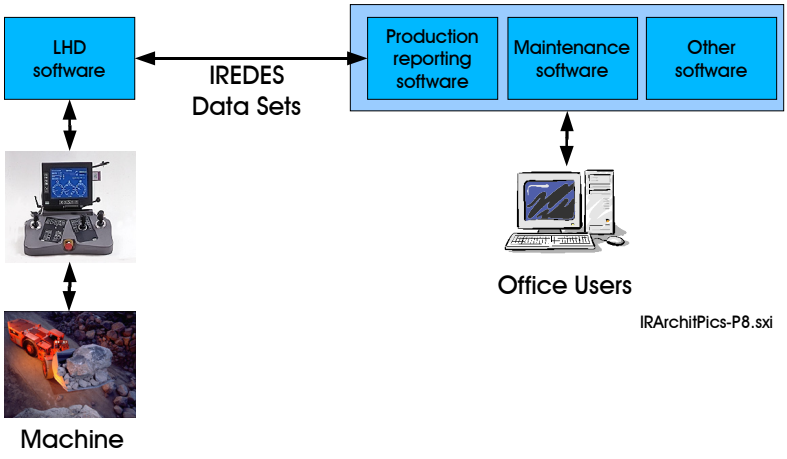
In the IT and networking industries accepted base technology is available which has to be adjusted to the particular needs of mining data exchange:

5. Use of technology broadly accepted in the IT industry but suitable also for use on resource limited Embedded Systems.

These demands lead to an completely object oriented standard structure described in this chapter.

5.2 Philosophy of IREDES data exchange

The IREDES standard unifies the data exchange of different mining equipment to enable an end - to - end application level communication between mining equipment and IT applications such as database systems, production reporting software, planning programs, maintenance systems etc.



Any information exchanged at one time is contained in a so called „IREDES Data Set“. This is the physical unit (file, network datagram) used for the exchange of IREDES information.

A machine (equipment type) is able to produce or accept different Data Sets for different application purposes e.g. to report on production performance, to report on production quality or to accept production plans.

To make the data exchange as efficient as possible reducing the number of parameter re-definitions within the standard, IREDES is set up using a component based object oriented architecture.

5.3 IREDES Components

Any IREDES Data Set consists of four main components:

- Equipment Profiles to define equipment type specific data exchange
- Application Profile information defining application purpose specific information which is used identically throughout all Equipment Profiles using it.
- Commonly used Objects to be re-used by different Equipment or Application Profiles.
- Administration Objects for controlling the data exchange

These main components are working together in an object oriented structure:

All Data Sets are defined in an Equipment profile. The Data Set is the highest level in the IREDES object structure. An Equipment Profile contains different Data Sets for

different application level purposes as e.g. for production performance reporting, for production quality reporting or for exchanging production plans.

In an simple Equipment Profile there may be entries for the follwing application level purposes:

LTPPerf Production Performance Reporting

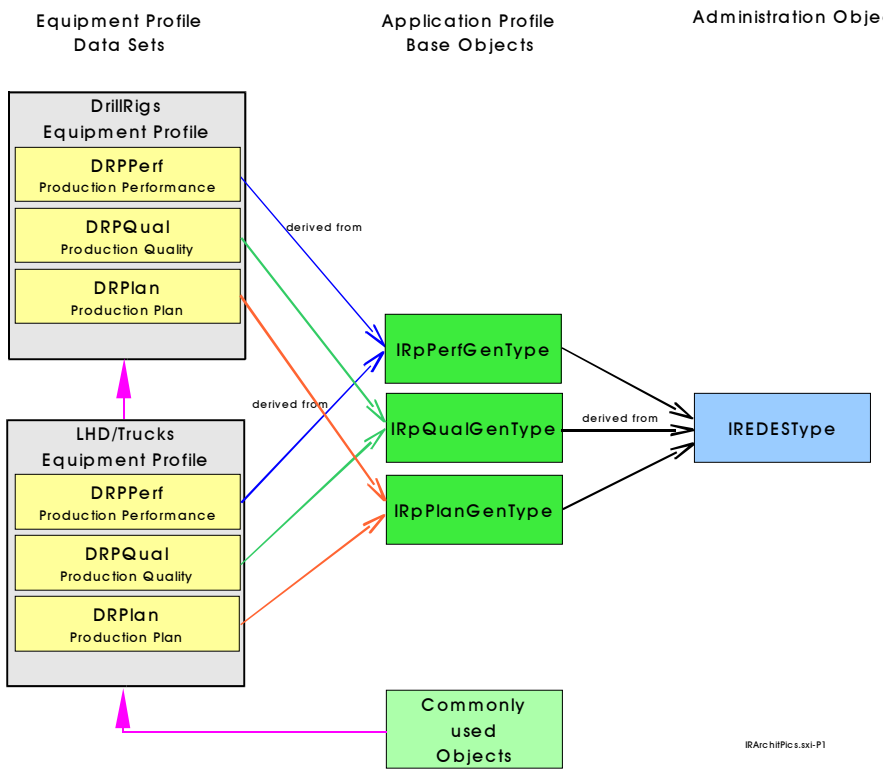
LTPQual Production Quality Reporting

LTPLan Production Plan

All these application level purposes recurr in different Equipment Profiles, so general definitions are made in a central location (in the „Application Profile“). Thereby these general definitions for e.g. Production Performance or Production Quality reporting can be used in any Equipment Profile they may be applicable to.

A Data Set following a specific Application Profile simply inherits all general definitions from this Application Profile. An DRPPerf object by example which reports on the Production Performance of a drill rigs inherits general reporting parameters from the IRpPerfGenType. The LHD/Trucks production performance reporting Data Set also inherits identical general objects from the same IRpPerfGenType.

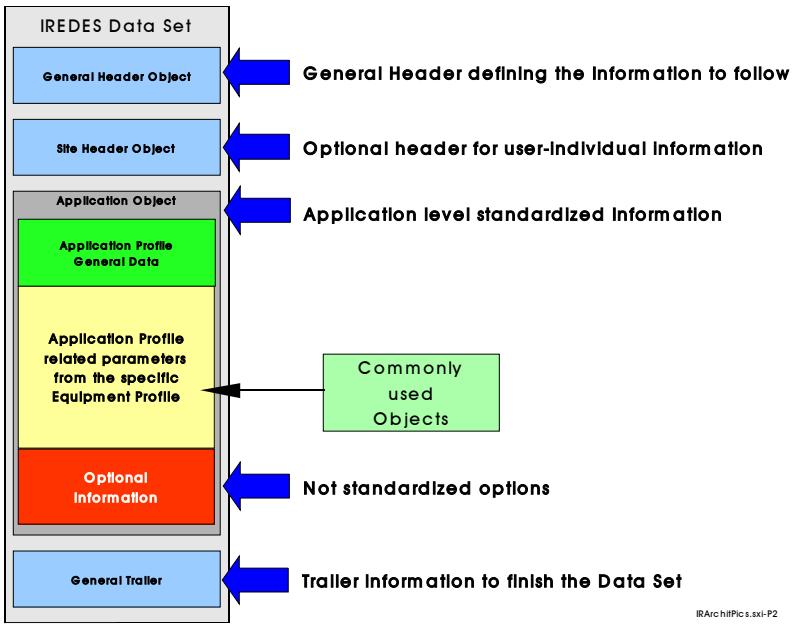
The Equipment Profile itself then extends these Application Profiles by adding new, individual parameters that are exclusively used with the particular equipment type.



Picture 5-1: IREDES components

All Application Profile data types are derived from the IREDESType object supplying the Data Set with administration information required to organize the data exchange.

Additional objects to be used in a Data Set definition are Commonly used Objects (CuO's). In these CuO's information reusable by many Equipment Profiles is defined. This information may be used by any Data Set in any place. Examples for Commonly used Objects are coordinate system definitions or definitions for commonly used equipment components like a diesel motor.



Picture 5-2: IREDES Data Set

In practice a valid IREDES data set consists of:

- A „General Header" (GenHead) object to contain general information on the data set setup (see 5.4)
- An optional Site Header object to contain information the user of a machine may define individually (e.g. inventory ID's, machine nickname, special working site info etc). (See 5.5)
- An Application Object which contains all application information, the „data payload" of an IREDES data set. The Application Object is a combination of the generic Application Profile object (see 5.6) and the equipment type specific Equipment Profile object (see 5.8). Both of these may contain any universally used „Commonly used Objects (see 5.10)
- The Trailer object to contain data integrity information such as checksums (see 5.4).

In the following sections, the components are described in more detail:

IREDES Initiative
Goethestrasse 52
D- 49549 Ladbergen
www.iredes.org

5.4 Administrative information

Each IREDES data set contains some administrative information of general importance for the data exchange itself and the communication partners involved.:

GenHead - General Header

The General Header (Fig. 5-1) contains general information such as the time of data set creation or the IREDES version used. The GenHead object is integral part of any file based IREDES Data Set.

GenTrailer - General Trailer

The Data Set trailer object is used to secure data integrity. The object simply contains a checksum.

The IREDES definitions for Administrative information are stored in the xml schema file *IRtypes.xsd*.

5.5 Optional Site Information

For machine users it is often very important to integrate their individual machine identifications, working locations and other individual information into a data exchange. This enables formal identification of equipment and data exchange routines according to mine individual formal identification principles.

To enable such procedures without having to change standardized definitions, the optional „SiteHeader“ object was integrated into the standard.

This object may be freely defined by a machine user. The machine manufacturer does not have to make any adjustments to it's own software to meet these individual demands.

Once a machine is receiving such an optional Site Header information, it simply stores it. With each IREDES data set the machine creates, this Site Header Object is mirrored back to the user's computer system enabling this system to make use of the previously sent identification information.

The IREDES definitions for the optional Site Header are stored in the xml schema file *IRtypes.xsd.*

5.6 Application Profiles and reports

There is information that all IREDES compliant equipment may use in common regardless of the particular equipment type. Such information like „overall machine runtime during reporting period“ etc. is covered by the IREDES „Application Profiles“.

Application Profiles are set up for different purpose which multiple equipment types are able to support as e.g.:

PQual - Production quality information

PPlan - Production planning information

PPerf - Production performance information

Further Application Profiles - e.g. for online machine status, maintenance and machine monitoring will be added later.

The application profiles differ in the purpose of the reported information and in the timing of data generation or activation:

A **Production Performance** Data Set contains information on the working performance of the reporting machine during the reporting period. The reporting period can be set by the machine user (e.g. by operator / administrator input or parameter setup. Production Performance Data Sets are generated by the machine in fixed time intervals regardless of the working sequences.

A **Production Plan** (PPlan) Data Set typically is generated by the planning department containing information on which work has to be done by a machine and how this work has to be performed. One example is a drill pattern for a drill rig telling the machine where to place the holes and how the holes have to be drilled. A machine should be able to store multiple Production Planning data sets to enable the operator to manually select a specific set for use.

A **Production Quality** (PQual) report contains information on how the machine has carried out it's work in relation to the corresponding Production Plan. For the drill rig, this data set contains all information on how the drill rig has carried out the drilling of the planned drill pattern in reality. Therefor, a Production Quality data set is created always directly after a production plan has been completed.

For each IREDES Application Profile general data objects are defined for IREDES standard wide use. These profiles are stored in the schema file *IRappBaseClasses.xsd*.

An Equipment profile using the information of a particular application profile simply is derived from the Application Profile. Thereby it automatically uses all parameters defined in the Application Profile. Additionally, individual parameters are added for the Equipment Profile's internal use.

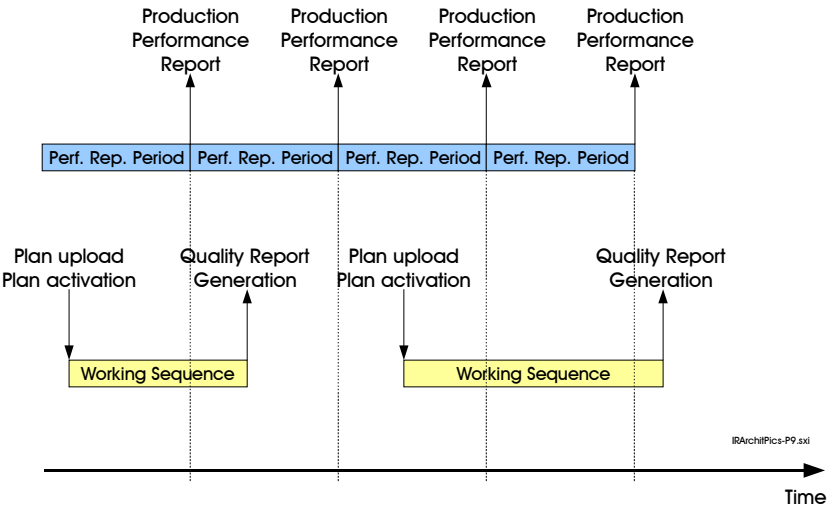
5.7 Timing of report generation

The Application Profile also determines the timing of the data exchange, because Application Profiles are set up to meet different reporting demands of the machine users:

PPlan: Defines the work to be carried out in a specific working sequence. It has to be available on the machine before the working sequence starts. PPlan Data Sets may be physically transferred at any time. Before starting the working sequence, the related plan is activated automatically or by a user.

PPerf: Generated in fixed time intervals (shifts, days,...) to report on production performance („WHAT was done during reporting period?“)

PQual: Generated each time a working sequence defined in a PPlan is completed to report on the qualtiy of work („HOW was the work carried out?“)



Picture 5-3: Exchange of PPerf, PPlan and PQual data sets

The picture shows how different profiles are exchanged over time.

Before each sequence (yellow) starts, a Production Plan has to be sent to the machine and activated.

During work, a number of Production Performance Data Sets (blue) are created by the machine any time the reporting period (shift, day,...) is completed. Therefor, during different working cycles different numbers of Production Performance reports may be created. If the working cycle is shorter than one Production Performance reporting period, this period may even cover multiple working cycles.

Once a working sequence is finished, the related Production Quality Data Set is created reporting on the results of the work carried out. The next working sequence is ready to start as soon as the next production plan is activated.

5.8 Equipment Profiles

The Equipment Profiles contains all data specific to a special machine type. So each machine type supported by the IREDES standard has it's own Equipment Profile definition. This is necessary due to significant differences between the different types of mining machines. Only machines with similar data content can be defined in the same Equipment Profile. Profiles currently available are:

1. DrillRig (released by IREDES V1.0)
2. Profiler

3. LHD/Trucks

Further Equipment profiles (e.g. for Chargers or rail bound transport) are setup as soon as there is sufficient interest in the IREDES workgroups.

When defining a new Equipment Profile it is decided, which general Application Profiles shall be applicable to the new machine type.

In these profile definitions all equipment type specific standardized information is covered.

This structure of separation between Application Profiles and Equipment Profiles has been chosen to make the standard usable for entirely different equipment types in uniform application environments. All general information defined in the Application Profiles used for completely different machine types is directly comparable with each other.

Another consequence of this setup is that new Equipment Profiles as well as Application Profiles may be added to the standard without having to modify existing components. New equipment types may be integrated into the standard without the requirement of updating other equipment already running the IREDES standard. Equipment types covered by IREDES „do not need to know“ about the other profile's existence making the standard very flexible in use.

5.9 Collaboration of Application Profiles and Equipment Profiles

The IREDES standard is set up in a hierarchical, object oriented structure. A consequence of this design is that the origin of parameters and the place of their definition may not be clearly visible by an user of a particular IREDES Data Set.

The reason for this is the use of object oriented „inheritance“ technologies, deriving specialized structures from commonly used base components:

Each Equipment Profile Data set (as e.g. DRPPerf, DRPQual,...) is derived by extension from an Application Profile data type. Thereby all elements defined in the base type (Application Profile) automatically become part of the equipment specific definitions.

Additionally any Application or Equipment Profile can make use of the Commonly used objects.

A key issue when setting up IREDES profiles is the decision in which component a definition has to be made, whether it has to be inside an equipment profile, within the corresponding application profile or as an commonly used object which makes the definition accessible to all profiles.

For this decision, the following guidelines shall be followed carefully:

1. Objects to be **used by multiple Application Profiles** shall be defined as „Com-

monly Used Object“ (CuO) as defined in 5.10.

2. Objects to be **used by multiple Equipment Profiles**, but limited to one particular „Application Profile“ shall be defined together with the corresponding Application Profile. This is by example applicable to maintenance relevant information such as „Time to next service“ which may be used for any equipment type, but it’s use makes only sense when used together with the Maintenance Application Profile.
3. Objects that are **entirely individual for specific machine types** shall be defined inside the corresponding Equipment Profile. As most people’s work today is mainly related to one single machine type, it is natural to define everything as individual for this particular type of equipment. However, it should be carefully evaluated whether the information (- or parts of it -) may be used by other equipment types, too. In this case all commonly used definitions are made in the corresponding Application Profile or as a Commonly Used Object.

In the documentation of each Application Profile as well as for any Equipment Profile it shall be clearly documented which external parts of the IREDES standard and especially which CuO’s the profile is using.

5.10 Commonly Used Objects

The IREDES namespace contains all basic types needed for the definition of several extensions made in all modules. These basic types can be found in The Commonly Used Object module although there is no special namespace for them. They are expected to be „atomic“ and unique.

5.11 Adding Customized Structures

There are mainly two hookup points in the standard to integrate non standardized, optional information:

- The SiteHead object
- Inside any Equipment Profile

The SiteHead object is mainly used for identification purposes by the equipment users („mine“). The machine does not perform any processing with this information.

The optional elements inside an Equipment Profile are used for individual reporting parameters to be set up by a machine.

In both cases, separate xml schemas may be used to specify an xml structure „below“ the hookup point in the IREDES standardized profiles. The hookup points are defined in the xml schema as „IRoptionType“.

6 Data Set Exchange

This section explains how a central computer system and a machine perform the data exchange. This procedure is completely independent from any carrier media. All file based IREDES data sets may be transferred via removable media (Floppy, Flash-Cards,...) or in identical form via networks.

Networking will become a part of the standard and will be covered by a separate Application profile containing the definitions for the network data exchange to provide some kind of administrative frame the header and trailer of the IREDES XML files.

6.1 Memory media

The IREDES standard does not make any restrictions on the kind of memory media to be used for off-line data exchange. However, memory capacity has to be sufficient to support plain, uncompressed and unencrypted storage of the IREDES data sets used by the machine.

An IREDES compliant machine shall not require any supplyer specific hardware at the machine user for accessing and use of the IREDES compliant information.

6.2 Networking

The definitions for networking are currently set up by the IT/Networking group. The following guidelines can be made already now:

- For transfer of IREDES information via networks, the TCP/IP protocol will be used.
- There will not be any restrictions made regarding the underlying networking hardware. However, for hardware layer compatibility a machine should support an IEEE 802.3 10 Base T / 100 Base T Ethernet connection and a serial port for PPP traffic and modem connection. Using these ports, nearly any communication demands can be met.
- The application level framework for data exchange and handling of administrative issues (Request information, subscribe information, error handling,...) is currently under discussion. The XML based Simple Object Access Protocol (SOAP) is one potential candidate.

IREDES Initiative
Goethestrasse 52
D- 49549 Ladbergen
www.iredes.org

7 Technology

This chapter briefly introduces into the technology used for IREDES to represent the standard in a format broadly used throughout the IT industry.

7.1 Why XML?

XML can be read and understood by humans and easily be processed within an IT application. In addition, it is very flexible and can be used to transport all kind of information. That is why XML becomes more and more important for data interchange between different machines and organizations. More information about the advantages of XML can be obtained at http://www.w3.org/XML/1999/XML-in-10-points.

In the IREDES project it is very important, that XML can contain data defined as objects. For this purpose, XML Schema was developed which supports the inclusion of other XML Schema definitions into a definition file and the extension of predefined objects.

7.2 A short introduction into XML

A data object in XML is represented by a tag. The basic tag is the name of the object within brackets:

```
<name>
```

These tags are called „elements“ in XML. To show that this object is complete, a slash „/“ has to be inserted before the closing bracket:

```
<name/>
```

All attributes of this object are added as a list of identifier value pairs after the objects name:

```
<name attr1="value1" attr2="vauel2" attr3="value3" .../>
```

The XML specification does not restrict the length of this attribute list. The content, if any present, of an object has to be placed between two Tags. One marks the beginning of the content and the other marks the end:

```
<name attr1="value1" attr2="vauel2" attr3="value3" ...>
  This is the content.
</name>
```

It is a big advantage of XML that objects can be placed in other objects to perform a logical data structure. Here is an example for a car:

```
<car color="red" manufacturer="does_not_matter">
  <carwheel position="left_front" status="ok"/>
  <carwheel position="left_back" status="ok"/>
  <carwheel position="right_front" status="ok"/>
```

```
  <carwheel position="right_back" status="ok"/>
  <engine type="diesel" horsepower="124.7"/>
  <navigation system="human" reliability="20%"/>
</car>
```

There is a difference between an object like a wheel and an attribute like color that should be noticed. For this example the writer of the XML definition has decided that it is not necessary to define a color as an object. He probably does not expect extensions to be made for a color. But an engine and a wheel could be extended as there are quite a lot of different types of wheels and engines with different properties. So we want to have more information about the wheel itself but no more information about the color.

7.3 A short introduction into XML Schemas

XML Schemas are used to define the allowed content of XML files. XML Schema files are XML files itself which can easily be processed by any parser enabled to read XML. XML Schema allows to include other XML Schema documents which makes modularisation possible. XML Schemas also offers basic object orientation like the extension of already defined objects. All Tags that can be used in XML Schema definitions are documented in the XML Schema specification.

As a short example it might be interesting to see how a developer might define a wheel like that one mentioned above in XML Schema:

```
<xs:element name="carwheel">
  <xs:annotation>
       <xs:documentation>This is the wheel of a car.</xs:documenta
            tion>
  </xs:annotation>
  <xs:complexType>
       <xs:attribute name="position" type="xs:string"/>
       <xs:attribute name="status" type="xs:string"/>
  </xs:complexType>
</xs:element>
```

There is a Tag that is named with „xs:element“. This is the element-Tag as defined in the XML Schema specification and is used to define an element, we might think of as an object. The text before the colon specifies the namespace the Tag is defined in. Here „xs“ stands for XML Schema and we expect this namespace to be defined in the head of the file. All Tags with names starting with „xs:“ are Schema specific keywords and defined in the XML Schema namespace. The documentation for this object is surrounded by a documentation and an annotation Tag. The complexType Tag now contains the main information about the attributes of the carwheel. Here the attribute Tags tell us that there are two attributes both with string type values. One is named position and the other one is named status.

There are much more Schema-Tags that can be used to define XML that can not all be explained here. It also is possible to offer several objects at one place or define a range of numbers an object must occur in the file.

IREDES Initiative
Goethestrasse 52
D- 49549 Ladbergen
www.iredes.org

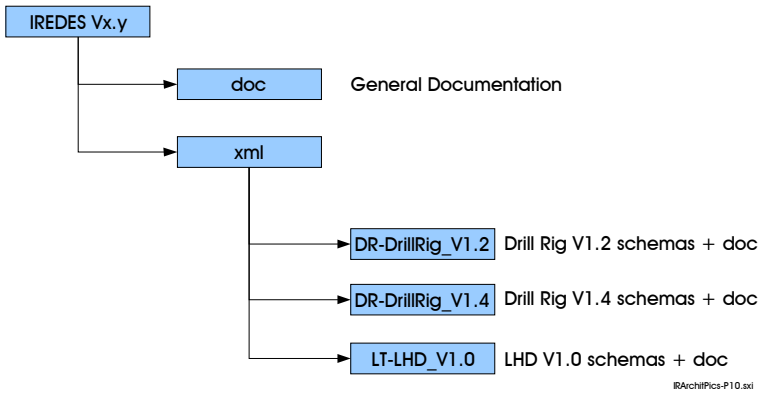
7.4 Namespaces

Namespaces mean that an identifier has to be unique within a namespace. Namespaces are used for big definitions and Programs to avoid cryptic or multiply defined names for objects. That is why IREDES uses a namespace for every machine definition. Dealing with IREDES definitions in Schemas, the usage of a special namespace for the IREDES structures is recommanded in order to avoid problems. The default IREDES namespace identifier is „IR“. It is used as a prefix for all names used in this namespace. See also section 8.3.

8 IREDES representation in XML

8.1 Files

IREDES Data Sets are exchanged in XML files. These data sets have to be set up according to the rules defined by the corresponding XML Schemas. The IREDES schema files are organized as follows:



For each IREDES Base version one top level directory is available containing sub directories „xml“ and „doc“:

xml — This directory contains the xml schemas and profile definitions.

doc — This directory contains the documentation accompanying the related IREDES base version.

The *xml* directory contains all general schema files used in the related IREDES base version.

Additionally, the xml directory contains one single subdirectory for each Equipment Profile: This subdirectory contains all xml Schema files exclusively used by the related Equipment Profile. Usually there will be one xml Schema file named as the Equipment profile itself. This file contains all top level elements - one for each Data Set defined in the profile. Equipment profile related documentation is also stored in the Equipment Profile directory.

8.2 Data Set representation in XML

Any Equipment Profile consists of independent entries for each Application Profile supported by the equipment type. For each Application Profile supported by the

Equipment Profile one top level (root) element exists. All top level elements of the Drill Rigs Profile are shown in the following picture (taken from the XMLSpy tool):

DrillRig.xsd

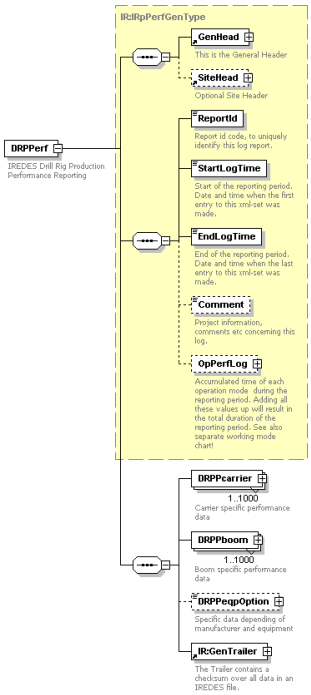
import	loc: ./DRbaseClasses.xsd	ns:http://www.iredes.org/xml/DrillRig
import	loc: ../IRappBaseClasses.xsd	ns:http://www.iredes.org/xml
import	loc: ../IRtypes.xsd	ns:http://www.iredes.org/xml
annotation	IREDES Drill rig XML schema version 1.0 (RFC)	
annotation	Schema created by IREDES special intrestgroup for Drill rigs, see www.iredes.org.	
annotation	This version is for review and comments ONLY, not for design!	
element	**DRPPlan**	ann:IREDES Drill Rig Production Plan
element	**DRPPerf**	ann:IREDES Drill Rig Production Performance Reporting
element	**DRMaint**	ann:IREDES Drill Rig Maintenance Reporting
element	**DRPQual**	ann:IREDES Drill Rig Production QualityPerformance Reporting
element	**DRMWD**	ann:IREDES Drill Rig Measurement-While-Drilling Reporting
element	**DRTunnelLine**	ann:IREDES Tunnel Line definitions
element	**DRLaserLine**	ann:IREDES Laser Line definitions
element	**HoleDeviation**	ann:IREDES Hole deviation reporting

Picture 8-1: Top level elements of the Drill Rigs Equipment Profile

Each of the elements stated in this picture represents a separate IREDES Data Set that can be exchanged independently.

In the Drill Rigs example above the DRPPlan object represents the Production Plan Application Profile extended by the Drill Rigs special entries. The DRPQual object represents the Production Quality Application Profile extended by the drill rig entries and so on. There may also be top level elements not extending an Application Profile. For the drill rig the „Hole Deviation“ object is such an element (see below).

Any of those objects has a similar construction which is explained using the DRPPerf Production Performance reporting for the Drill Rig:



The DRPPerf element as any other top level IREDES Data Set element contains two groups of elements:

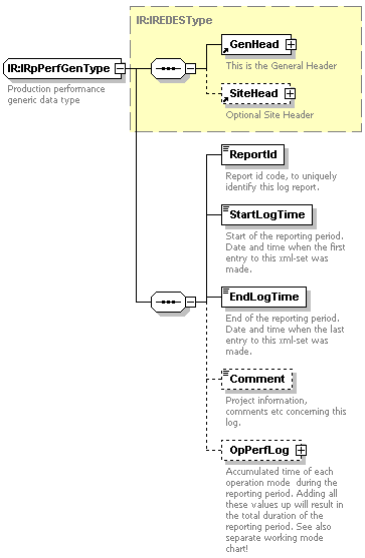
The elements of the first group represented in the orange area are inherited from it's base data type which is „*IR:IRpPerfGenType*“. This general (IREDES wide) data type defines the Application Profile elements to be used for any such entry (in this case Production Performance reporting) in any Equipment Profile of the IREDES standard.

These derived elements are used by DRPPerf for drill rig production performance reporting in the same way as in LTPPerf used for LHD/Trucks production performance reporting.

The second element group is a sequence of elements individually defined for this particular Data Set's tasks. These elements „extend“ the *IR:IRpPerfGenType*.

The last element in this list is always the global element IR:GenTrailer (defined in the schema IRTypes.xsd). This element is used for securing data consistency of the exchanged Data Set using a checksum algorithm.

IREDES Initiative
Goethestrasse 52
D- 49549 Ladbergen
www.iredes.org



The IRpPerfGenType object providing common Application Profile objects has to cover all generic information applicable to all IREDES Data Sets. Therefor, this object is derived from another data type, the generic IREDESType, shown by the orange area in the graphic. This generic type covers the IREDES Data Set header objects: The GenHead (generic header) and the SiteHead (site header) objects.

The IRpPerfGenType object then extends the IREDESType base type by the application profile specific objects as e.g. Report ID, StartLogTime etc.

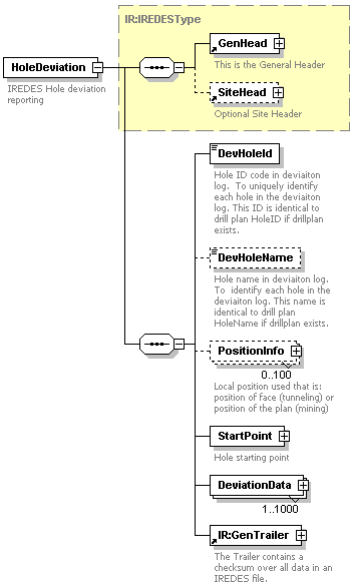
In similar form, all Application Profile objects are handled. All of them are derived from the IREDESType generic object.

If no Application Profile is applicable to an Equipment Profile specific Data Set, the top level object of this special Data Set in any case has to be directly derived from the generic data type IREDESType.

This procedure assures that all IREDES Data Sets regardless of which Application Profile or Equipment Profile they belong to are derived from the IREDES Type data type providing the administrative information to any IREDES Data Set exchanged.

By deriving a data set from an Application Profile generic data type (as IRPpPerfGenType in the example above) it is assured that an application specific data set from a Drill Rig behaves exactly as the same application profile data set originating from another machine type (as e.g. LHD). Thereby the generic information becomes comparable between different machine types!

There is one element in the Drill Rigs profile which is not covered by any Application Profile, the „HoleDeviation“ object. Such top level objects are individually designed for an equipment profile only if there is no suitable Application Profile available for use with the particular element and if the purpose is that special that it may never be used by other machine types.



To make these individual top level elements compliant with the IREDES general Data Set structure, they have to be derived from the IREDESType data type as defined in the IRTypes.xsd schema. The elements covered by IREDESType then become the first entries of the Data Set, representing the general IREDES Data Set headers.

As the headers covered by the IREDESType elements, the individual Data Set also has to supply the GenTrailer parameter, which simply is a reference to the GenTrailer parameter as defined in the IRTypes.xsd schema.

The GenTrailer object always has to be stated as the last parameter in the Data Set's parameter sequence.

Any parameters between those two sections can be freely defined in the Equipment Profile's schema specification.

8.3 Use of XML namespaces

It can be expected that the IREDES standard soon will cover a whole range of different equipment types. Data Sets prepared by different machine types are processed by one single database system. Therefor it is important to prevent from use of parameters with identical names in different application context which makes interpretation ambigious.

Therefor, IREDES makes internal use of the XML namespace capabilities. Different namespaces are available for:

	Target Namespace	Prefix
IREDES General defininitions and data Types	http://www.iredes.org/xml	IR:
Drill Rigs Equipment Profile	http://www.iredes.org/xml/DrillRig	DR:
Other Equipment Profiles	http://www.iredes.org/<EqupProfile>I	<tbd>

One top level IREDES namespace contains the xml schemas to be used by any Application Profile or Equipment Profile. These are as for IREDES V1.2:

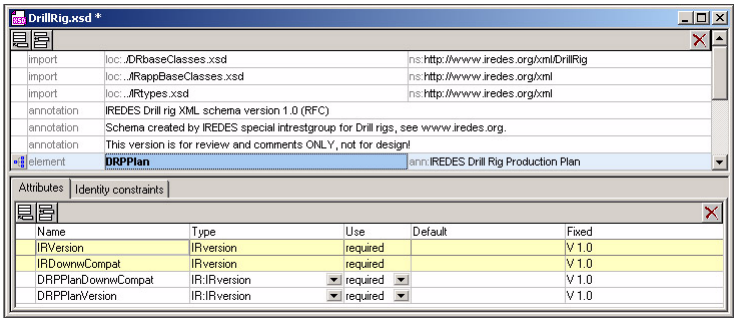
- General IREDES data type definitions in IRTypes.xsd
- General Application Profile definitions in IRappBaseClasses.xsd
- Coordinate system definitions in IRcoordNav.xsd

Further schemas are expected to be set up as the standard develops. Please check in the IREDES schema top level directory!

IREDES uses the xml qualified namespace format, so elements referenced from namespaces other than the schema's own target namespace have to be accessed by using their namespace's prefix as defined in the schema headers.

8.4 IREDES Version handling

As also the IREDES standard will be developing over time, a version handling is available for each top level IREDES element. Thereby any IREDES Data Set exchanged automatically carries the version number of the IREDES xml schema the Data Set was set up with.



Picture 8-2: IREDES version handling: Example

Each top level („root") element of any IREDES Data Set carries four version attributes to provide information about the IREDES standard version:

IRVersion: IREDES Base version needed to process this Equipment Profile schema

IRDownwCompat: Earliest version the IREDES Base system version stated in IRVersion is downward compatible to. Since this version, only extensions have been made but no changes affecting compatibility issues (data type changes etc).

*Profile*Version: Version number of this Equipment Profile

*Profile*DownwCo: Earliest version the Profile version stated in *Profile*Version is downward compatible to. Since this version, only extensions have been made but no changes affecting compatibility issues (data type changes etc).

The general attributes for the IREDES version information (IRVersion and IRDownwCompat) are taken from the IREDESType datatype. They are indirectly inherited from the IREDESType data type via the Application Profile's base type.

The individual version attributes for use with the Equipment Profile are added when the Equipment Profile is set up.

IREDES Initiative
Goethestrasse 52
D- 49549 Ladbergen
www.iredes.org

The version parameter in a usable Data Set always informs about the IREDES schema version that was used to set up the particular data set.

The values of all version attributes are defined as fixed values in the corresponding schemas. Values other than those defined in the schemas are not allowed.

The version information allows to automatically process IREDES Data Sets in an environment of different IREDES versions used e.g. by equipment of different age.

If an IREDES Data Set has to be validated against a corresponding schema the interpreting computer is able to automatically assign the right schema version to be used according to the version tags.

Is the computer not able to find a suitable schema, it also may use available schemas down to the version stated in the tag „..DownwCompat“. These attributes are used if new tags are added in a newer version and other changes have been made that did not affect the interpretability of the „older“ parameters.

Consequently, when the meaning and the definition of parameters is changed resulting in the loss of the compatibility, the DownwCompat attributes are set to a higher version. In the version with such changes, the DownwCompat attribute will be identical to the version attribute.

All version numbers have to follow the character sequence (patterns) as defined in the IRVersion data type.

9 IREDES and non standardized information

9.1 XML schema elements for IREDES extensions

Any standard should have the possibility of adding individual entries introduced e.g. by technical progress, new demands or user specific requirements.

However, this should not affect the standard's general usability. Data Sets containing non - standardized, optional information should be compatible for use with any standard compliant equipment or computer system, even if it is not able to make use of the additional non - IREDES information provided additionally.

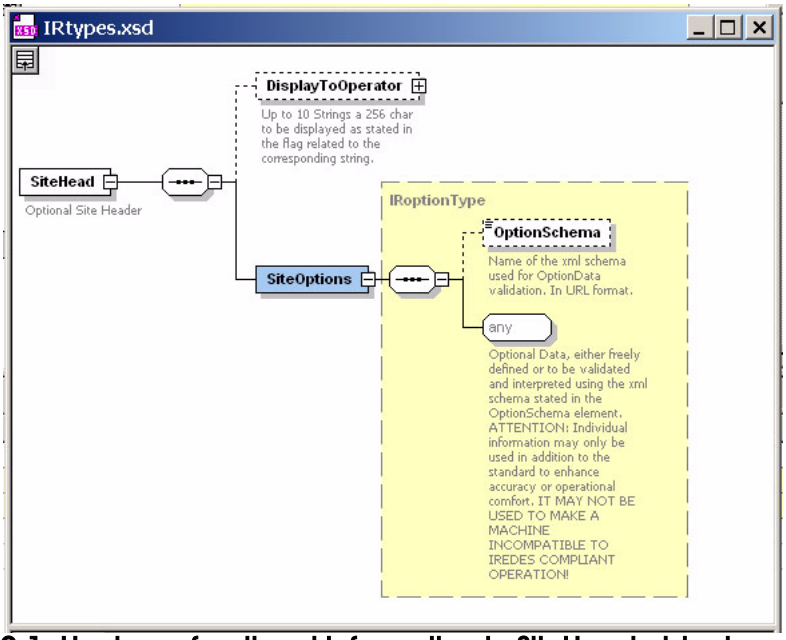
Consequently, a few rules apply to the use of individual options and extensions:

1. The basic IREDES Schemas may not need to be altered for adding options
2. Adding optional information is only allowed at special points in the IREDES Data Sets. These entry points are clearly marked in the IREDES xml schemas.
3. Multiply defined structures should be avoided. The use of clearly defined prefixes in the parameter names is highly recommended!

XML schemas allow the integration of any information not covered by schema definitions at any point in the schema using the „*xsd:any*“ element.

When validating a schema, any information below a „*xsd:any*“ element is ignored. A Data Set will be regarded „valid“ even if the information in the structure below the „*xsd:any*“ element cannot be interpreted.

This feature is used in the IREDES standard to define „hookup points“ for individual, optional information.



Picture 9-1: Hookup of optional information in SiteHead object

In the IREDES data type IRoptionType the standard provides a structure for xml schema based handling of optional information (see picture):

The first tag „OptionSchema“ provides an optional possibility of stating an xml schema to be used for interpretation of the optional elements, if they also are set up using xml schema based principles. The schema name has to be stated in xml URL format allowing the interpreting computer system to gather this file „on the fly“ from any accessible location in a network or even from the Internet.

The second element „*any*“ provides the hookup point for nearly any kind of information as explained above.

ATTENTION: Individual information may only be used in addition to the standard to enhance accuracy or operational comfort. IT MAY NOT BE USED TO MAKE A MACHINE INCOMPATIBLE TO IREDES COMPLIANT OPERATION!

9.2 Recommended extension handling

The handling of IREDES extensions not using xml schema procedures has to be carried out entirely individual.

However it is very recommended to set up XML schemas for the extensions in the same way as the IREDES schemas are set up. In this case, IREDES and xml based procedures for information parsing and validation can be used without additional effort.

A recommended handling is a multi pass procedure:

1. Validation and interpretation of the IREDES standardized elements, ignoring all extensions.
2. Separate validation and interpretation of any optional extension in the IREDES Data set: Loading the corresponding xml schema, validation against the schema and call to the interpretation routines.

In this procedure, just the interpretation routines remain individual. For schema validation, standard xml procedures are used.

IREDES Initiative
Goethestrasse 52
D- 49549 Ladbergen
www.iredes.org

10 IREDES administative Elements

This section describes details for the use of the general IREDES administrative elements and data types. These are defined in the schema IRTypes.xsd.

10.1 General Header - GenHeader

The General Header is used in any IREDES Data Set to define general administrative information on the Data Set to follow:

FileCreateDate: Date and time stamp when a Data Set is initialized.

10.2 Site Header - SiteHeader

The Site Header is used optionally to specify information to be used individually for different mining sites. Major parts of this header can be freely defined.

The site header is prepared by a mine. Once sent to a machine, the machine has to mirror the site header back to the mine until a new site header is received. Once an IREDES Data Set without any optional site header is received, the machine stops mirroring site headers back to the mine.

The equipment does not any changes with the Site Header information. The information contained in the Site Header is just stored and mirrored back as-is.

10.3 General Trailer - GenTrailer

FileCloseDate: Date and time stamp when the Data Set was finalized (closed). This value has to be written immediately before the checksum is calculated!

The General Trailer additionally contains a checksum for assuring the data consistency of the IREDES Data Set and to prevent from unauthorized manual Data Set modifications.

The checksum is calculated as follows:

1. The xml Data Set is created using the value '0' (zero) as a checksum entry.
2. Then checksum is calculated over the ENTIRE xml Data Set, including the checksum itself (which at this point is set to 0 (zero)). (Algorithm see below).
3. The calculated checksum is written into the checksum parameter value of the xml Data Set, now replacing the preliminary '0' value.
4. The Data Set is stored or sent.

The interpretation has to be carried out in exactly reverse order:

1. Read the checksum from the Data Set
2. Set the checksum value in the Data Set to 0 (zero).

IREDES Initiative
Goethestrasse 52
D- 49549 Ladbergen
www.iredes.org

3. Calculate the checksum
4. Compare it with the checksum originally stored in the Data Set.

The checksum algorithm is a CRC32 routine as mathematically descibed in ISO 3309.

IREDES Initiative
Goethestrasse 52
D- 49549 Ladbergen
www.iredes.org

11 Abstracting Machine Data

Abstracting a machines data means to take a closer look on data in- and output. To describe a machine in XML, the vendor needs to take care about all information relevant to the outside world and the machine itself. This can be both static and dynamic information. The number of wheels can be expected to be a static value but the time it is working for a session should be a dynamic value. The vendor will probably want to find out what data the machine needs from outside and what data the outside world needs or wants to have from the machine.

11.1 Modularization

Modularization is very important for the creation of data structures. A lot of machines can be expected to have an engine, for example. That is why an engine's abstraction should be available as a separated data structure in order to be used by several machines. This leads to less effort needed for every machine definition. Engine and other structures are simply included if already defined. So a deeper look at the IREDES XML files should be taken to find out which modules are defined. All modules that fit the application should be used in order to keep the data definitions clear and small.

11.2 Object Orientation

Object orientation in general means that a module's properties can be changed by derivation. Such a module is called an object. In XML Schema data structures can be extended. A good example for object oriented definitions can be different diesel or electric engines derived from an object called „base-engine". All data structures needed by both types of engines should be defined in „base-engine" and all specific information is added by extending the basic engine definition. This also leads to less effort for new machines and a lean and clear definition of a machine's data.

11.3 Base Types

Basic IREDES data types are defined in the IRtypes.xsd file and can be used in all Schemas dealing with IREDES.

IREDES Initiative
Goethestrasse 52
D- 49549 Ladbergen
www.iredes.org

12 IREDES Application integration

For implementing the IREDES standard in applications two different environments have to be distinguished determining the possibilities for implementing the standard and it's components:

12.1 Machine implementation

An implementation on a computer system used right on a machine is characterized by:

1. Limited computing resources are available on the (embedded) system on a machine
2. Only those parts of the standard dealing with the particular machine type shall be implemented; A dynamic „schema" handling is not required.
3. The implementation is more or less „hardwired" to the machine's control software. Therefor, a dynamic implementation allowing different machine types and versions to be interpreted is not required.

Carefully observe that IREDES do not necessarily require XML file validation against the corresponding XML schema on the machine!

These characteristics make a „static" implementation possible:

Preparing the reporting data sets is possible right from a single software library containing the source code for one particular IREDES equipment profile and profile version.

Interpreting incoming data sets (e.g. plan files) is also possible by using a static implementation realized within one single software library.

It can be expected that IREDES source libraries for static implementations will become commercially available soon.

12.2 Central Server implementation

An implementation on a server system connected to a fleet of (different) machines is characterized by:

1. Commercial computing resources
2. Full dynamic schema validation is required
3. Identical equipment profiles have to be processed in different versions (machine ages) simultaneously
4. Loading and unloading equipment profiles has to be possible dynamically during runtime

Observe that IREDES requires validation of the XML data files against the corresponding XML schemas on the Central Server side of the data exchange!

IREDES Initiative
Goethestrasse 52
D- 49549 Ladbergen
www.iredes.org

These characteristics make a „static“ implementation possible, preparing the reporting data sets right from a single software library containing the source code for one particular IREDES equipment profile and profile version.

12.3 Application Software implementation

For an application software (e.g. dealing with Drill Plan handling) it may be chosen between a static or dynamic implementation. IREDES sets the following demands on compliant software applications:

1. The application shall be able to deal with different IREDES versions of any supported Equipment Profile simultaneously!
2. Application software NOT running directly on a machine is regarded as a Server side device. Therefor, the software shall perform XML file validation upon before interpretation and after creation of any IREDES data set.
3. Application software not necessarily has to be able to interpret optional supplier or project individual extensions of the standard. But any software shall be able to tolerate the existence of such extensions. If the application is not able to interpret those extensions, they should simply be ignored.

12.4 Implementation Guidelines

To assure smooth integration of the standard into heterogenous system and machine environments, a few guidelines for implementation shall be followed:

- Be as tolerant as possible in standard interpretation when accepting foreign data sets
- Be very restrictive in standard interpretation while creating data sets for distribution to foreign computer systems
- Each software system running on office computer systems shall perform data set validation against the corresponding XML schemas before accepting data sets and after creation of data sets.

13 Nomenclature

To make navigation in the IREDES standard easier, a nomenclature is used in the XML schemas. Using this nomenclature, the origin of all major objects can be easily identified by their name.

The first prefix (two capital letters) also determines the XML-namespace the corresponding object is residing in.

These prefixes are used as part of the name of the corresponding XML tag names. Thereby they safely prevent from unintended name collisions if different profiles are developed in different work groups.

13.1 Object Prefixes

IR	IREDES general objects, IREDES standard wide data types and other tool objects available throughout the entire standard. Also applicable to IREDES wide available Commonly used Objects.
CH	Objects specifically defined for the Chargers Equipment Profile.
DR	Objects specifically defined for the Drill Rigs Equipment Profile.
LT	Objects specifically defined for the LHD/Trucks/Loaders Equipment Profile

13.2 Additional Application Profile Prefix

A second additional prefix (also two capital letters) s introduced for each Application Profile. This prefix follows the Equipment Profile prefix and is as follows:

PL	Determines the Production Plan Application Profiles.
PP	Determines the Production Performance Application Profile
PQ	Determines the Production Quality reporting Application Profile
CM	Determines the Condition Monitoring Application Profile

13.3 Example

The following examples show the setup of XML name tags following this nomenclature:

DRPPerf	Uppermost object of the Production Performance object for Drill Rigs
LHPQual	Uppermost object of the Production Quality object for LHD's/Loaders/ Trucks

IREDES Initiative
Goethestrasse 52
D- 49549 Ladbergen
www.iredes.org

Attachment 1: IREDES Glossary

Application Object	Data structure within a Data Set containing all information exchanged for a particular application level information for the purpose of the Data Set. This might e.g. be Planning data, Production Performance data etc. The "Application Object" may consist of multiple sub-objects containing detail information. In the Data set, all information sub-objects are located in a sequential order. The type of Application Object carried by a Data Set is defined in the Data Set's General Header.
Application Profile	Application data profile definition for the specific Data Set's purpose. This might be a Planning Data Set, a Production Performance log or a Production Quality log. An Application profile consists of: 1. A Data Block and the corresponding parameter definitions for general information applicable to any equipment regardless of it's type when using the corresponding Application Profile 2. Equipment type specific information e.g. for Drill Rigs, LHD's etc, for the Application Profile's purpose as defined in the corresponding Equipment Profiles.
ApplicationObject	see Application Object
CUO	see Commonly Used Object
Commonly Used Object	In many cases, a set of belonging parameters is used multiple times in different profiles. To prevent from multiply defining nearly identical parameter sets, „Commonly Used Objects" are used e.g. for a hole definition, definition of the tree dimensional coordinates of one single navigation point etc.
Data Set	Complete Information structure for one single purpose including all Header and Trailer Data Blocks. The purpose might e.g. be planning Data, Production Performance data or Production Quality data. The Data Set is created by an IREDES conformant application.
Data File	ASCII file containing information of one single Data Set
Equipment Profile	Standard Definition to cover all standardized application parameters for one individual type of equipment (e.g. Drill RIg, LHD,...) regardless of it's manufacture. A complete Equipment Profile covers one set of equipment type individual parameter definitions for each relevant Application Profile.
General Header	General Data Set Header specifying general information on file generation, software and standard versions as well as on the type of Application Object transferred by this Data Set. The

IREDES Initiative
Goethestrasse 52
D- 49549 Ladbergen
www.iredes.org

	General Header is always located on the very top of the Data Set.
GenHead	see General Header
Site Header	Optional Site Header specifying customer specific information not to be modified by the machine. For customer use only. The Site Header is mirrored back to the customer with each response telegram until the machine receives a new Site Header or a telegram without any Site Header. The Site Header is not object for standardization.
SiteHead	Keyword used to specify the Site Header. See Site Header
Trailer	Administrative Data Block containing information required for consistency check of an exchanged Data Set.