

Introduction paper

Reportnet architecture

An introduction

Prepared by
Sasha Milosavljevic, European Dynamics
Nedeljko Pavlovic, European Dynamics

June 2006
Version 2.0

European Environment Agency



Version management and approval

| No | Date | Changes | Author |
|-----------|-------------|--|---------------------|
| 1.0 | 06/04/2006 | Table of Contents | Sasha Milosavljevic |
| 2.0 | 19/06/2006 | First Draft | Sasha Milosavljevic |
| 2.1 | 19/09/2006 | Migrated to ODT and spell checked | Søren Roug |
| 2.2 | 15/07/2008 | Renamed references to GDEM. Brought links up-to-date | Søren Roug |

Contents

| | |
|---|-----------|
| 1. INTRODUCTION..... | 4 |
| 1.1. PURPOSE..... | 4 |
| 1.2. INTENDED AUDIENCE AND READING SUGGESTIONS..... | 4 |
| 1.3. DEFINITIONS, ACRONYMS, AND ABBREVIATIONS..... | 5 |
| 1.4. REFERENCES..... | 5 |
| 2. REPORTNET..... | 6 |
| 2.1. EIONET..... | 6 |
| 2.2. GOALS..... | 7 |
| 2.3. KEY CONCEPTS BEHIND REPORTNET..... | 8 |
| 2.4. MAIN FUNCTIONS AND COMPONENTS..... | 10 |
| 2.5. EXAMPLE OF A REPORTNET USE CASE..... | 10 |
| 3. SYSTEM ARCHITECTURE..... | 13 |
| 3.1. LOGICAL VIEW..... | 13 |
| 3.2. CONNECTOR VIEW..... | 15 |
| 4. TECHNOLOGY INFRASTRUCTURE..... | 16 |
| 4.1. APPLICATION INFRASTRUCTURE..... | 16 |
| 4.2. COMMUNICATION INFRASTRUCTURE..... | 17 |
| 4.3. CONTENT HARMONISATION INFRASTRUCTURE..... | 17 |
| 5. REPORTNET COMPONENTS..... | 19 |
| 5.1. REPORTING OBLIGATION DATABASE (ROD)..... | 19 |
| 5.2. DATA DICTIONARY (DD)..... | 22 |
| 5.3. CDR..... | 24 |
| 5.4. CONTENT REGISTRY..... | 27 |
| 5.5. CONVERSION SERVICE..... | 28 |
| 5.6. QUALITY ASSESSMENT (QA)..... | 28 |
| 5.7. WEBFORMS SERVICE..... | 29 |
| 5.8. NETWORK DIRECTORY (ND)..... | 31 |
| 5.9. DMM..... | 32 |
| 5.10. DATA SERVICE..... | 34 |
| 5.11. IMS (INDICATOR MANAGEMENT SERVICE)..... | 35 |
| 5.12. UNIFIED NOTIFICATION SYSTEM (UNS)..... | 36 |
| 6. SOFTWARE ENGINEERING METHODOLOGY..... | 39 |
| 7. CONCLUSIONS OR RECOMMENDATIONS..... | 39 |
| 7.1. LIMITATIONS OF EXISTING SOLUTION..... | 40 |
| 7.2. BENEFITS OF THE REPORTNET..... | 40 |

1. Introduction

The European Environment Agency (EEA) aims to support sustainable development and to help achieve significant and measurable improvement in Europe's environment, through the provision of timely, targeted, relevant and reliable information to policy-making agents and the public. Reportnet is an important contribution to help EEA to fulfil its objective.

Reportnet represents the Eionet's IT system for environments data exchange. Eionet (European Information and Observation Network) is EEA's mature network consisting of administrative and scientific institutions at national level in more than 30 countries. The main focus is to generate a consensus regarding classifications and standards, IT-solutions, definitions of priority data flows, where countries are continuously benchmarked, definition of common indicators linking environmental data to policy targets.

The Reportnet project could be positioned as being part of the Common Interest Projects that implement European eGovernment services in specific policy areas in the context of the IDABC programme managed by the European Commission. IDABC stands for Interoperable Delivery of European eGovernment Services to public Administrations, Business and Citizens.

1.1. Purpose

The purpose of this document is to provide a general description of the architecture and functionalities of the Reportnet IT system. The Reportnet IT system is made up of a number of independent software components optimised to support the business processes of the European Environmental Information System. These components are independent in the sense that each one performs its own function without regard to how the others are operating, except the specific are that interaction between them is necessary. This document describes the overall functionality of the Reportnet IT system by placing considerable emphasis upon the role of each component and their interactions.

1.2. Intended Audience and Reading Suggestions

This document is meant to serve as a guide to new users and developers of the Reportnet IT system. It will allow better understanding of key principles behind the Reportnet design and the way that has been implemented in the context of the Eionet

In this document, section 1 identifies the scope and the purpose of the system, and lists the definitions, acronyms, abbreviations, and reference documents.

Section 2 provides the overview of the Reportnet as a suite of IT tools optimised to support the businesses processes of the Eionet.

Section 3 provides a high-level description of the Reportnet architecture. Section 4 describes the concept of shared information infrastructure on which Reportnet is built on as well as key benefits of application development with such infrastructure.

Each individual Reportnet component is described in section 5.

1.3. Definitions, Acronyms, and Abbreviations

| Abbreviation | Description |
|--------------|--|
| EEA | European Environment agency |
| Eionet | European Environment Information and Observation Network |
| ETC | European Topic Centre |
| NRC | National Reference Centre |
| NFP | National Focal Point |
| EU | European Union |
| EUR-Lex | A service on the official website of the European Union. It provides legal texts of the union. |
| Eurostat | Statistical Office of the European Communities |
| FAO | Food and Agriculture Organisation (UN) |
| OECD | Organisation for Economic Cooperation and Development |
| SPADE | Strategic Planning of Application Development in Eionet |
| TERESA | Transparent Environmental data and information Reporting and Exchange System for Administrations |
| UI | User interface |
| XML | Extensible Markup Language |

1.4. References

| Document | Location |
|--|---|
| SPADE | https://svn.eionet.europa.eu/repositories/Reportnet/docs/SPADE.odt |
| Reportnet brochure - 2005 Web Draft | http://www.eionet.europa.eu/m/reportnet-brochure.doc |
| EEA glossary | http://glossary.eea.europa.eu/EEAGlossary/ |
| Eionet connects | http://www.eionet.europa.eu/about/eionet_web-en.pdf |
| Reportnet documents on the EEA SVN repository | https://svn.eionet.europa.eu/repositories/Reportnet/docs |

2. Reportnet

Reportnet is a system of integrated IT tools and business processes creating a shared information infrastructure optimised to support the European environmental reporting. Reportnet represents a suite of software tools that help the Eionet members to produce quality environmental data and information with minimal effort. Reporting obligations agreed between EU or international bodies and countries provide the basic framework for data/information flows inside the Eionet network. Currently inside Eionet there are dozens of different reporting obligations and corresponding data flows which makes Eionet network very complex. The Reportnet tools aim to improve the harmonisation and standardisation of these data flows, by covering all activities inside them, from definition of the data that reporters need to produce, till the generation of the final products such as EEA published reports and core set of environmental indicators. In that respect Reportnet could be considered as a valuable information infrastructure for all participants in the reporting process.

2.1. Eionet

Eionet is a partnership network of the European Environment Agency (EEA) and its members and participating countries. It consists of the EEA itself, a number of European Topic Centres (ETCs) and a network of around 900 experts from 37 countries in over 300 national environment agencies and other bodies dealing with environmental information.

European topic centres consist of organisations of thematic expertise contracted by the EEA aiming to carry out specific tasks identified in the EEA strategy (as part of a five year work program) consistent with the the annual management plans. They are designated by the EEA management board following a Europe-wide competitive selection process and provide their work as an extension of the EEA responsibilities in specific topic areas. Each ETC consists of a lead organisation and specialist partner organisations from the environmental research and information community, which combine their resources in their particular area of expertise. The ETCs, working together with member and participating countries, facilitate the provision of data and information from the countries and deliver reports and other services to the EEA and Eionet. Currently there are five ETCs covering air and climate change, water, the terrestrial environment, waste and material flows, and nature protection and biodiversity. It is expected that from 2005 The latter two will be superseded by ETCs on resource and waste management and on biological diversity .

The Eionet partnership is crucial to the EEA in supporting the collection and organisation of data and the development and dissemination of information. Information technology infrastructure (sometimes referred to as e-Eionet) supports the organisations and individuals in the network. These are the national focal points (NFPs) and the national reference centres (NRCs).

National focal points are the main contact points for the EEA in the member countries. They are in charge of cooperation with the EEA and the ETCs and organise the national coordination of activities related to the EEA strategy. Their working methods differ from

country to country. This partly reflects the diverse nature of the national environmental systems within which they are based. For example, some NFPs are located in environment agencies, others are part of the environment ministry; some are part of centralised national administrations, whereas others operate in a decentralised manner,.. NFPs maintain and develop the national network, facilitate and coordinate contacts, requests and deliveries between the national and the EU level. In addition, they act as advisers to their EEA management board members and develop contacts to other relevant networks such as Eurostat. In order to promote key EEA publications, many NFPs organise events and issue press releases.

National reference centres (NRCs) are nominated by member countries. They are nationally funded experts or groups of experts in organisations which are regular collectors or suppliers of environmental data at the national level and/or possess relevant knowledge regarding various environmental issues, monitoring or modelling. NRCs are established in specific areas of environmental activity, for example air quality, climate change, river quality, waste generation, biodiversity, energy and many more. They play a role in technical coordination of these topics and work with the EEA and the relevant ETCs. The NRC structure varies in accordance with the requirements and priorities of the EEA strategy.

For all stakeholders inside the Eionet network, the Reportnet IT system brings benefits in many fields. The software tools developed as part of the Reportnet project make aims to facilitate the work of Eionet members at any phase of the data/information flows between national institutions (NFP, NRC) and European institutions (ETC, EEA). The objective is that all data flows will fully utilise most of the Reportnet tools described in this document. Of course this objective will be reached only if exist firmly cooperation between all members of the Eionet.

2.2. Goals

Since the establishment of the Eionet network, it has been observed that the reporting burden which member countries have, towards international organisations including EEA steadily increase. This is caused by a huge number of legal and moral obligations on which countries should respond in impending deadlines. Earlier practices in environmental reporting are shown that there is no harmonisation between different data flows which exist for each reporting obligation. As a result, in many cases, countries deliver the same or almost the same type data to different organisations with different exchange formats which reside on many national repositories. Similar working practices lead to the reality that member countries do not share data, methodologies, tools and experience between each other and thereby the overall reporting system is inefficient and wasteful.

The main objective of the Reportnet project is to overcome those drawbacks and reduce the effort needed for producing environmental data by increasing the efficiency and productivity inside the Eionet reporting system. This may be achieved by providing a set of common tools and an infrastructure for defining, collecting, transferring, editing, analysing and streamlining environmental data. Before all, stakeholders should obtain a transparent and timely access to reliable, objective and policy relevant environmental data so that emphasis could be given also to the quality assessment of the available data. Although development of such system may sounds very complicated at the first glance, modern

information technologies and standards such as Internet, XML and RDF provide an appropriate technology foundation for easy production of the software solutions going towards the above mentioned goals.

2.3. Key concepts behind Reportnet

In order to assess whether environment policies on both national and international level are working efficiently in practice, data and information are collected by the member states, reported and analysed at the EU level. In this section the key terms and concepts that are important for understanding this reporting process are described. Various Definitions are borrowed from the EEA glossary and other EEA official documents.

Reporting obligation

Reporting obligation is a legal text defining the requirements that Member States should follow aiming to report environmental data in order to meet the international legal and moral obligations relevant to the European level. It represents the requirements to provide information agreed between countries and international bodies such as the EEA or other international conventions. Reporting obligation provides the basis for most environmental information flows.

Legal Instruments

The legal instrument is the legal basis for one or more reporting obligations. Each reporting obligation must be derived from one legal instrument. The legal instruments mostly represent a set of official documents of the EU that are usually published on the EUR-Lex but other kinds of legal instruments may be included such as various international conventions.

Data Flow

The Eionet Data Flow project deals with the analysis and description of data flows into, within and out of the EEA Reporting System, i.e. data flow modelling. It focuses on organisational aspects of data flows and how the data are processed and stored within the EEA/Eionet Reporting System. It covers data flows from countries (NFPs and NRCs), via ETCs and international organisations (Eurostat, OECD, FAO, etc.) to the EEA repositories of data used in the reporting process; the Geographic Information System and the Data Warehouse

Data set

The data sets contain aggregated environmental data at national level. They cover at least the 25 EU Member States, but in many cases a pan-European coverage of 52 countries is available. Graphs and maps can be generated from the data sets. Information about the source of each data set and its geographical and temporal coverage is attached. In the Reportnet system a dataset is a collection of tables containing the reported data. Usually datasets are provided in the form of MS Access databases or MS Excel files. They are

subject to certain data flows and obliged to be reported by Reportnet players according to current legislation.

Data warehouse

The EEA is building up a data warehouse, containing all the data that are needed for regular reporting. “Data warehouse” is a term from the computer world, and it means a database that is able to read data from other databases whichever format the source data have. The EEA data warehouse is on the one hand a repository of data that have been used for making reports, and as such, ETCs supply aggregated data to the EEA data warehouse. On the other a hand, the data warehouse is a tool to disseminate information to other users.

Environmental indicators

An indicator is an observed value representative of a phenomenon of study. In general, indicators quantify information by aggregating different and multiple data. The resulting information is therefore synthesised. In short, indicators simplify information that can help to reveal complex phenomena. Our body temperature is an example of an indicator we use regularly. It provides critical information on our physical condition. Likewise, environmental indicators provide information on phenomena that are regarded typical for and/or critical to environmental quality. Indicators always simplify a complex reality. They focus on certain aspects, which are regarded as relevant and on which data are available.

In relation to policy-making, environmental indicators are used for four major purposes:

- To supply information on environmental problems, in order to enable policy-makers to evaluate their seriousness;
- To support policy development and priority setting, by identifying key factors that cause pressure on the environment;
- To monitor the effects and effectiveness of policy responses, and
- To raise public awareness on environmental issues. Providing information on driving forces, impacts and policy responses is a common strategy to strengthen public support for policy measures.

Core set of indicators

The EEA in collaboration with various countries has developed a core set of indicators covering:

- Environmental issues: water stress, air pollution, climate change, biodiversity and nature protection, soil and land issues, waste and material flows
- Economic sectors and environment: transport, energy, agriculture, tourism, fisheries

It is expected that the core set of indicators will be gradually implemented over the time period of 2004-2008, and will provide the main focus for information flows in Reportnet between countries and the EEA..

2.4. Main functions and components

For easier understanding of the main functions that are performed inside the Reportnet system, groups of functions are divided by software components which implement them. These software components and data repositories are available in various types of services via the web. The system can carry out its main tasks without the presence of some components and this is because people responsible for specific data/information may design their data/information flows by utilising only a sub-set of the existing components. However in the near future, the intention is that data/information flows should try to utilise all or the most of the components inside Reportnet, by following the best practices in the data flow management. The main software component inside the Reportnet system are:

- Reporting Obligations Database (ROD) - Enables storing, searching, browsing and updating reporting obligations;
- Central Data Repository (CDR) - Uploads, downloads, versioning, approvals and sealing of official datasets, packaging for delivery, and tagging with meta-information;
- Data Dictionary (DD) - Defines datasets and stores definitions of its data elements, their attributes, their data types, allowable values, and relationships between other elements;
- Data exchange modules (DEMs) - Improve automation of reporting and contribute to the development of a data exchange format in European level in order to ease exchange, enhance transparency and access to environmental information;
- Network Directory (ND) - Allow user authentication, security services, getting contact information, and routing of workflow processes; .
- Content Registry (CR) – Enables the searching by meta-data information fields in reporting products in the Eionet;
- Unified Notification System (UNS) - Enables Eionet members to receive notifications about specific events;
- Data service - Provides a central place for storing and merging reported deliveries which belongs to the same reporting obligations into one dataset;
- Data Merging Module (DMM) - Enables easy merging of new reported data with existing one inside data service;
- Indicator Management Service Provide centralised and formatted storage for contents of the environmental indicators.

2.5. Example of a Reportnet Use Case

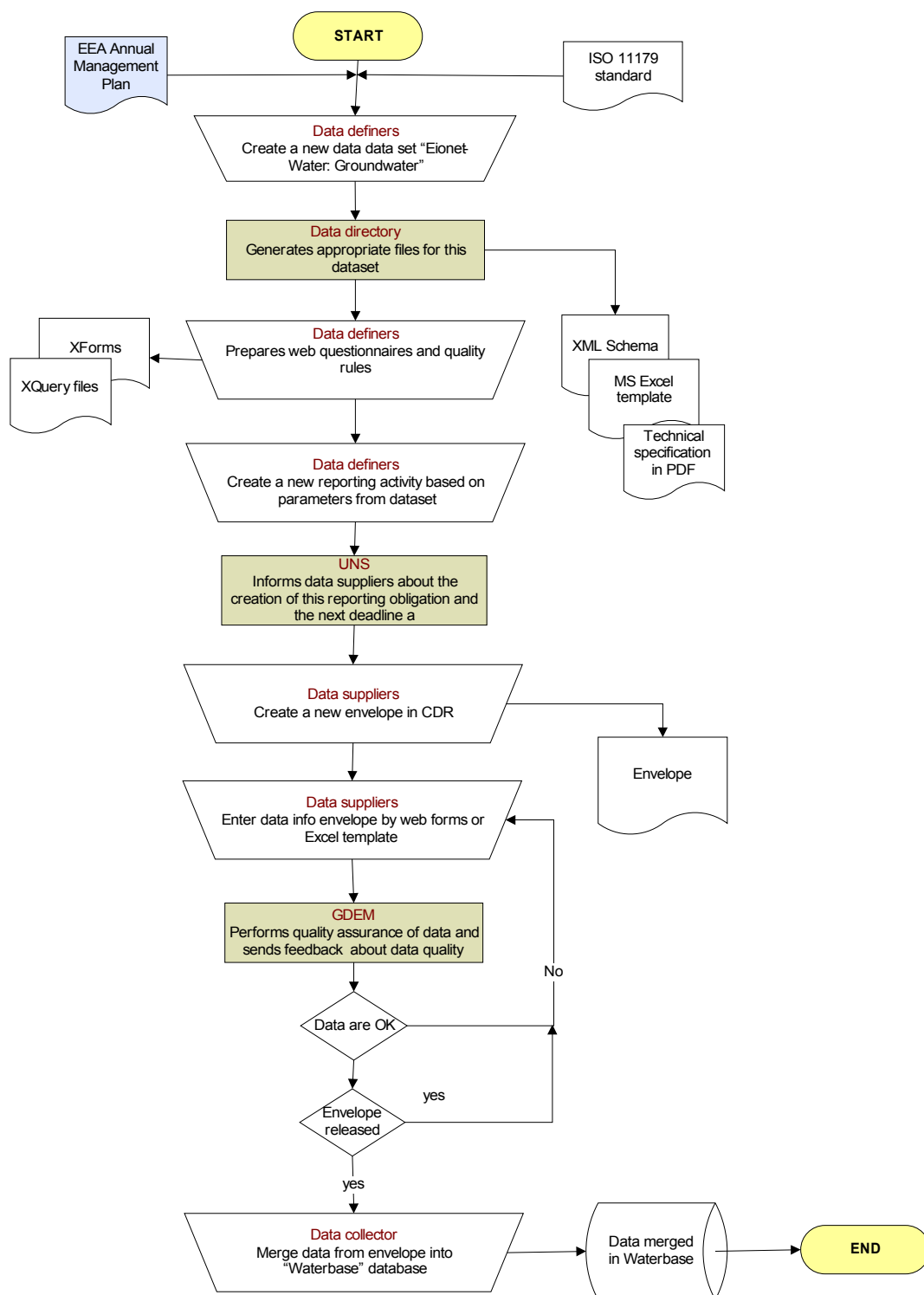
In order to clarify how the different Reportnet components work together it is best to describe one use case of the Reportnet that demonstrates the use of the components in various stages inside one data/information flow. For this purpose a specific data flow from Eionet priority data flows named “EWN-3: GroundWater quality” has been selected. The described workflow is simplified picture of the real activities that need to be performed in

the real data flow management. Many data/information flows currently exist inside Eionet, *but only few of them fully utilise Reportnet tools*. It should be clear that the Reportnet may not be totally static and rigid in the sense of the exact type of components that could be utilised in one data/information flow. Therefore the steps described here stands for this data flow, but for others may be different.

1. In first step data definers define dataset in data dictionary for the new data flow which is described in a legislative instrument “EEA Annual Management Plan”. This will be done by defining data tables with attributes that belongs to the created dataset, following [ISO 11179 standard](#) for describing data elements. The data dictionary component will automatically generate an XML Schema and an Excel template for this data set thereby providing ability for automatic data quality assessment. Detailed guidelines that would explain how templates should be populated is also provided.
2. Data definers prepare the web questioners form to make easy transfer of the national data to the Central Data Repository (CDR). They also set up set the relevant quality rules regarding this dataset to be used for verification of that the data sent by member countries are correct, complete and precise.
3. EEA exports the published obligation on the ROD website with specified parameters such as Legislative instruments on which this obligation is based on, countries that should respond to this obligation, organisation for which the data set should be delivered and the time frequency of reporting.
4. The Unified Notification System (UNS) informs data suppliers about the creation of this reporting obligation and the next deadline for report delivery.
5. Data suppliers create an envelope in the central repository (CDR) in order to fill out the metadata about this envelope such as reporting obligation name, envelope name and period of time which is covered in this envelope.
6. Data suppliers choose the method of delivering national data to central repository. It may use web questionnaires created in step 2 or download an empty template from data dictionary in form of excel file that is created in first step. The user will fill out the national data in chosen form and save (web questionnaires) or upload data (excel template) inside the envelope on central data repository (CDR). The Web questionnaire service which belongs to generic data exchange module (DEM) converts filled forms in xml files and store them into the envelope on CDR.
7. After reported data are stored in the envelope on the central repository (CDR) the system performs automatic quality assurance and provides the feedback to data supplier which contains the result of quality control of data. Additional feedback may also be provided at a later date to national experts by ETC.
8. After the delivery has passed the quality control and it is released for public the data flow collectors merge reported data for each county by using DMM toll, into database, in this case “WaterBase “ which is maintained y the European Topic Centre on Water and replicated in the EEA data warehouse.
9. The dataset merged in the previous step is used for creation of the environmental indicator “Nutrients in freshwater” that belongs to the EEA Core set of indicators.

10. Finally, when a user looks at an indicator fact sheet, he/she will be able to follow the data sources all the way back to a country's deliveries. From this collected data set the deliveries (reports) may be produced to international organisations in their specific-formats .

A Graphical presentation of this use case is shown in the following figure:

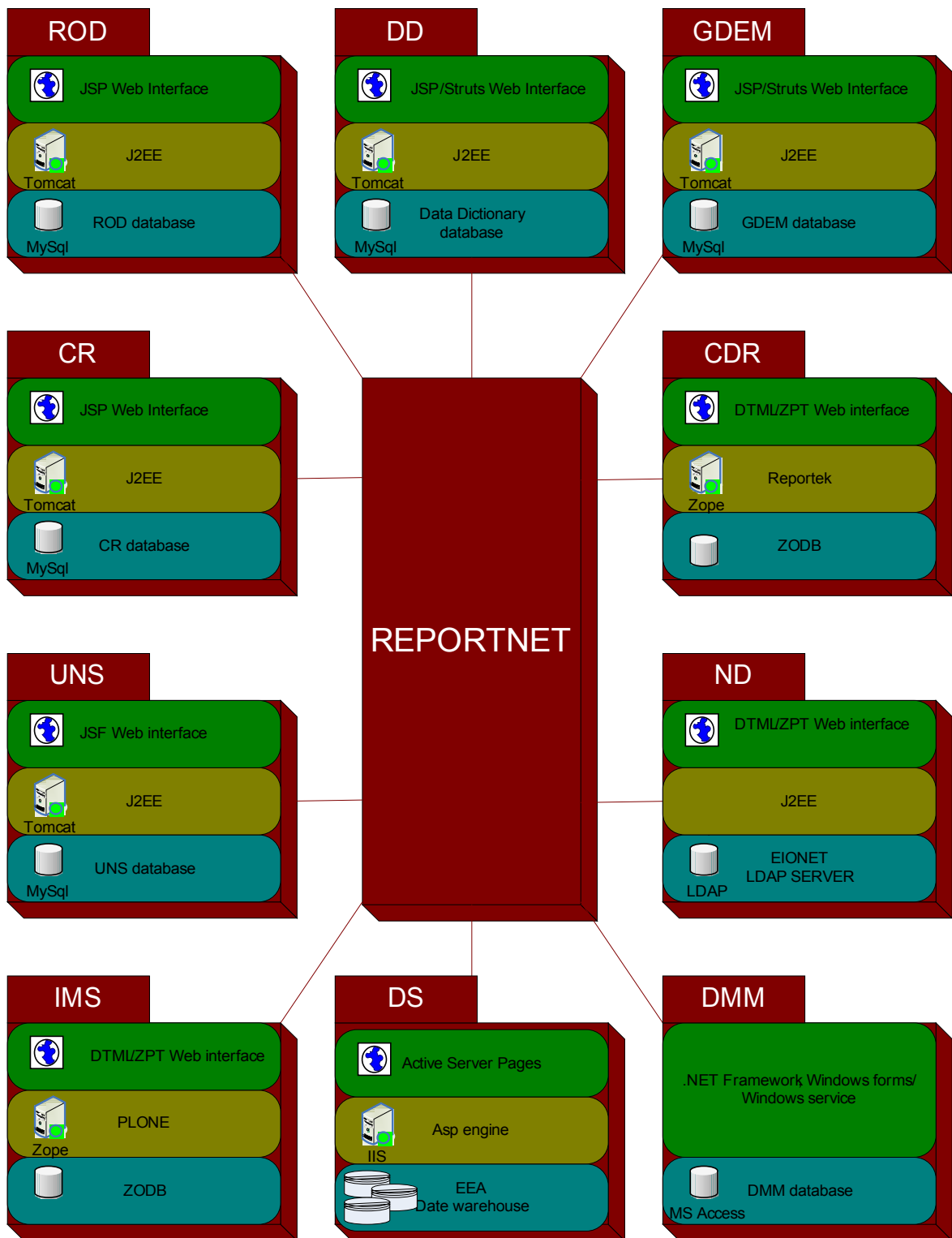


3. System Architecture

Software architecture defines the Reportnet IT system in terms of components and interactions between them. The system architecture is used in order to understand what the systems does, how it works and the role of each part of the system. This section gives a general overview of the software architecture of the current system, and how it can be represented conceptually. It presents the main interrelation between the components, where components implement a certain functionality, which is exposed to the rest of the architecture through one or more interfaces. In order to capture the overall picture of the Reportnet we may observe the system from different views: such as logical , connector, data or physical view. We consider a view as being a representation of all or part of the architecture, from the perspective of one or more *concerns* which are held by one or more of its stakeholders.

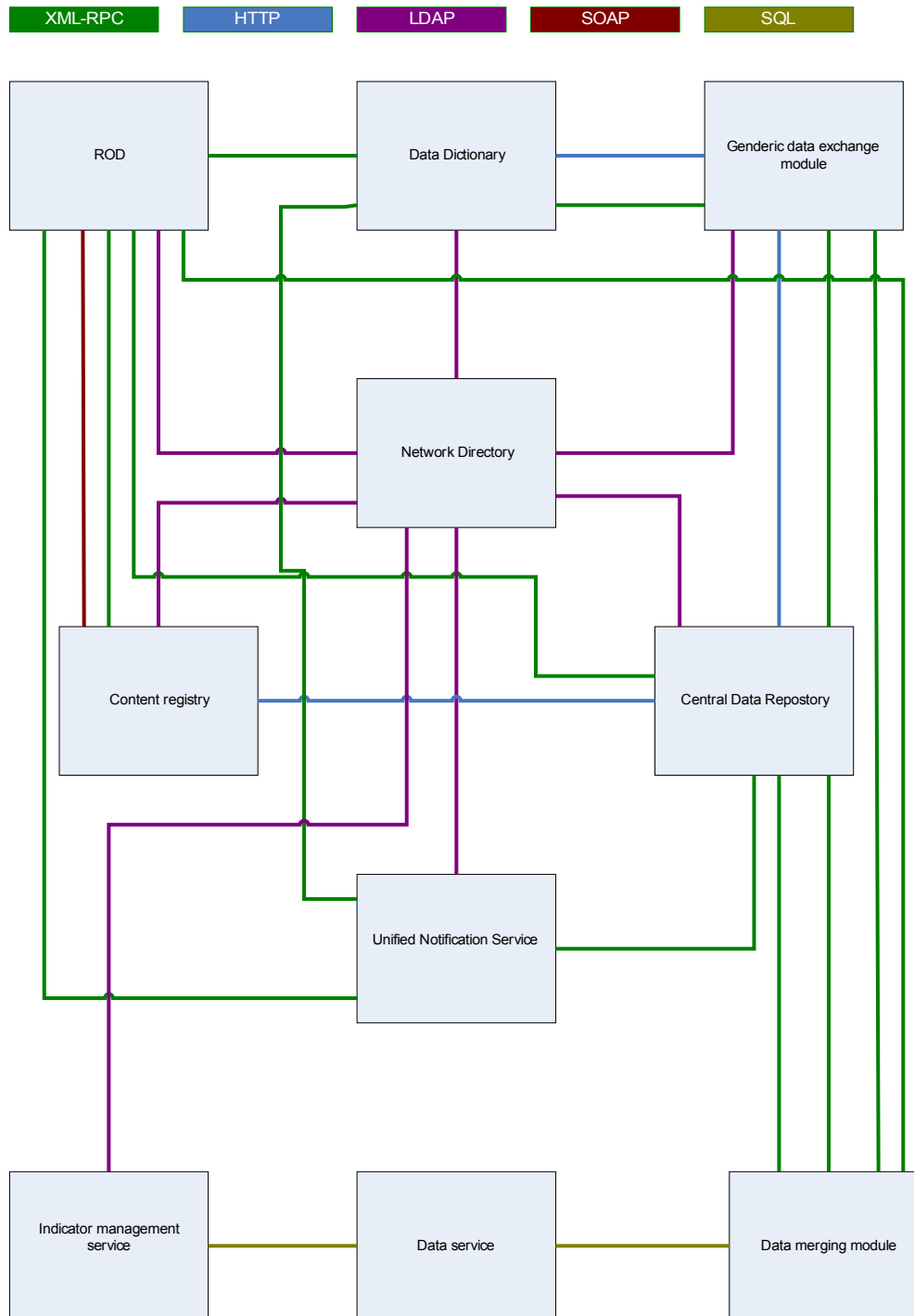
3.1. Logical view

The Reportnet system logical architecture is depicted in the following figure. It consists of a set of independent but interoperable Reportnet components built on different technologies and accessible via Web:



3.2. Connector view

Connector view depicts the main components of Reportnet and their interactions in the sense of data flows with specifications of concrete protocols that are used in these interactions.



4. Technology infrastructure

‘If I have seen farther than other men, it is because I have stood on the shoulders of giants’ – *Isaac Newton*

The Reportnet architecture places into relationship various components and technologies that comprise a framework or an information infrastructure on which businesses processes inside Eionet or any other reporting network may be successfully implemented. We may define information infrastructure as the framework of data standards, interoperability mechanisms, and other permanent structures that enables information exchange and sharing within and between information systems.

Reportnet software development has been based on two principles: Open Source and Modularity. Open source development enables code reuse, protection of investment, peer review and real competition for the tendering of development by making available all information and tools inside Reportnet. A large portion of the development effort is spent on writing general-purpose components based on data and technology standards that can be re-used throughout the main application, rather than on domain-specific functionality of the application’s features. The system can be easily modified or extended. Changes to an individual module can be made without affecting functionality of other modules.

4.1. Application infrastructure

Application infrastructure in the Reportnet is mostly based on Open Source products. The rapid evolution of the open source community produces better and more reliable software solutions than the traditional proprietary software. Examples of the open source products successfully utilised in the Reportnet are Apache Tomcat, Zope and MySQL.

4.1.1. Apache Tomcat

Tomcat is a web container developed by the Apache Software Foundation. Tomcat implements the servlet and the Java Server Pages (JSP) specifications from Sun Microsystems, providing an environment for Java code to run in cooperation with a web server. It adds tools for configuration and management but can also be configured by editing configuration files that are normally XML-formatted. Because Tomcat includes its own HTTP server internally, it is also considered as a standalone web server.

4.1.2. Zope

Zope is an open source application server for building content management systems, intranets, portals, and custom applications. The Zope community consists of hundreds of companies and thousands of developers all over the world, working on building the platform and Zope applications. Zope is written in Python, a highly-productive, object-oriented scripting language.

4.2. Communication infrastructure

Application integration between Reportnet components is supported by the set of standardised communication protocols that glue components together so they work as if they were one. The following communication protocols are used for transferring data between the various Reportnet components.

4.2.1. Lightweight Directory Access Protocol (LDAP)

LDAP is a [networking protocol](#) for querying and modifying [directory services](#) running over [TCP/IP](#). An LDAP directory usually follows the [X.500](#) model: It is a tree of entries, each of which consists of a set of named attributes with values. While some services use a more complicated "forest" model, the vast majority use a simple starting point for their database organisation.

4.2.2. XML-RPC

XML-RPC is a remote procedure call protocol which uses XML to encode its calls and HTTP as a transport mechanism. It is a very simple protocol, defining only a handful of data types and commands, and the entire description can be printed on two pages of paper. This is in stark contrast to most RPC systems, where the standards documents often run into the thousands of pages and require considerable software support in order to be used.

4.2.3. SOAP

SOAP is a protocol for exchanging XML-based messages over a computer network, normally using HTTP. There are several different types of messaging patterns in SOAP, but by far the most common is the Remote Procedure Call (RPC) pattern, in which one network node (the client) sends a request message to another node (the server), and the server immediately sends a response message to the client. Indeed, SOAP is the successor of XML-RPC.

Interoperability is the key to mitigating the problem of heterogeneity. To make systems interoperable, middleware is usually required. Middleware is software that wraps the architectural details of a system and offers services to other systems on heterogeneous platforms across a network XML as data interchange standard is the key building block for interoperability architecture inside Reportnet and in the same time the fundamental unit of transfer and storage data within the system.

SOAP and XML-RPC have become industry-wide accepted standards for XML-based messaging and both protocols applications can adopt any interfaces at any time, including the necessary data types and function (method) declarations. This approach allows Reportnet components to be written in different languages, and operate on different platforms.

4.3. Content harmonisation infrastructure

Data harmonisation means that definitions of data entities, their attributes, and values are agreed upon in committees and stored in an electronic data registry. From there, the

definitions are reused by all components that use that data. Data harmonisation also ensures that data can be retrieved, filtered and synthesised by end users and remote applications.

Key technologies that enable data harmonisation in Reportnet are: XML, XML Schema, RDF and the Dublin Core metadata element set.

4.3.1. The Extensible Markup Language (XML)

XML is recommended by the W3C consortium to be used as a general-purpose markup language in order to create special-purpose markup languages. XML is capable of describing many different kinds of data. In other words: XML is a way of describing data and an XML file can contain the data too, as in a database. It is a simplified subset of the Standard Generalized Markup Language (SGML). Its primary purpose is to facilitate the sharing of data across different systems, particularly systems connected via the Internet.

4.3.2. XML Schema

XML Schema is published as a W3C Recommendation in May 2001, is one of several XML schema languages. It was the first separate schema language for XML to achieve Recommendation status by the W3C. Like all XML schema languages, XML Schema can be used to express a schema: a set of rules to which an XML document must conform in order to be considered 'valid' according to that schema. However, unlike most other schema languages, XML Schema was also designed with the intent of validation resulting in a collection of information adhering to specific data types, which can be useful in the development of XML document processing software, but which has also provoked criticism.

4.3.3. The Resource Description Framework (RDF)

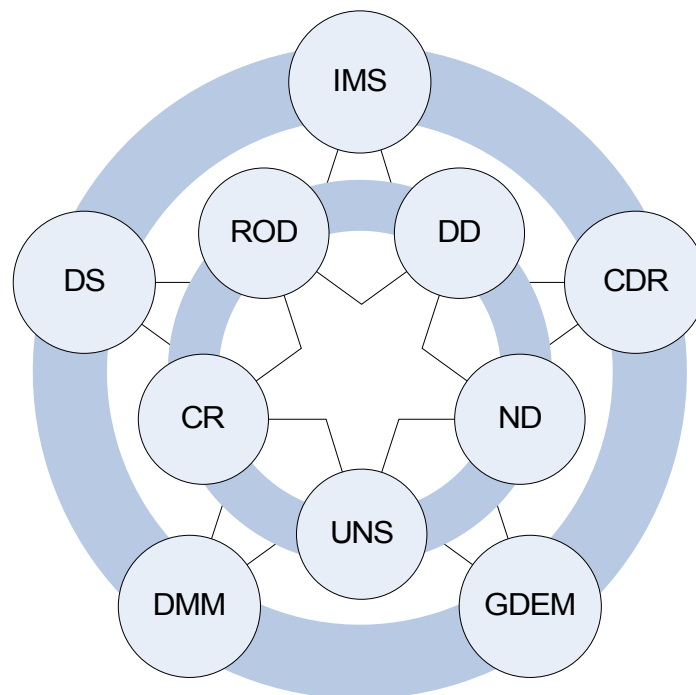
RDF is a language for representing information about resources in the World Wide Web. It is particularly intended for representing metadata about Web resources, such as the title, author, and modification date of a Web page, copyright and licensing information about a Web document, or the availability schedule for some shared resource.

4.3.4. Dublin Core

The Dublin Core metadata element set is a standard for cross-domain information resource description. In other words, it provides a simple and standardised set of conventions for describing resources online in ways that make them easier to find. Dublin Core is widely used to describe digital materials such as video, sound, image, text and composite media like web pages. Implementations of Dublin Core are typically XML and Resource Description Framework based.

5. Reportnet components

This section describes what the components in Reportnet system look like, what they do and how they are implemented. This includes a fairly detailed description of component main functionalities, users and technology. The position of the component within the overall system and its interactions with other components is also highlighted and when necessary appropriate links to other artefacts which deal with particular component are provided.



5.1. Reporting Obligation database (ROD)

5.1.1. Definition

ROD is both the EEA's reporting obligations database and the web module that enables browsing, searching and updating content in database. It contains records describing environmental reporting obligations that EEA member countries have towards the DG environment, European marine conventions, Eurostat, OECD, UN, UNECE, as well as the EEA itself. Excluded from ROD are non-environmental reporting, requirements for proposals or the seeking of permits, requirements for specific actions to be taken (as opposed to reporting on actions taken), and administrative processes such as notifications of contact points, instances of issuance/withdrawal of permits (rather than reports that summarise permits issued), notifications of legal transposition and implementation measures. Environmental experts from EEA and ETC use ROD to find duplication of data sets in various obligations.

5.1.2. Functions

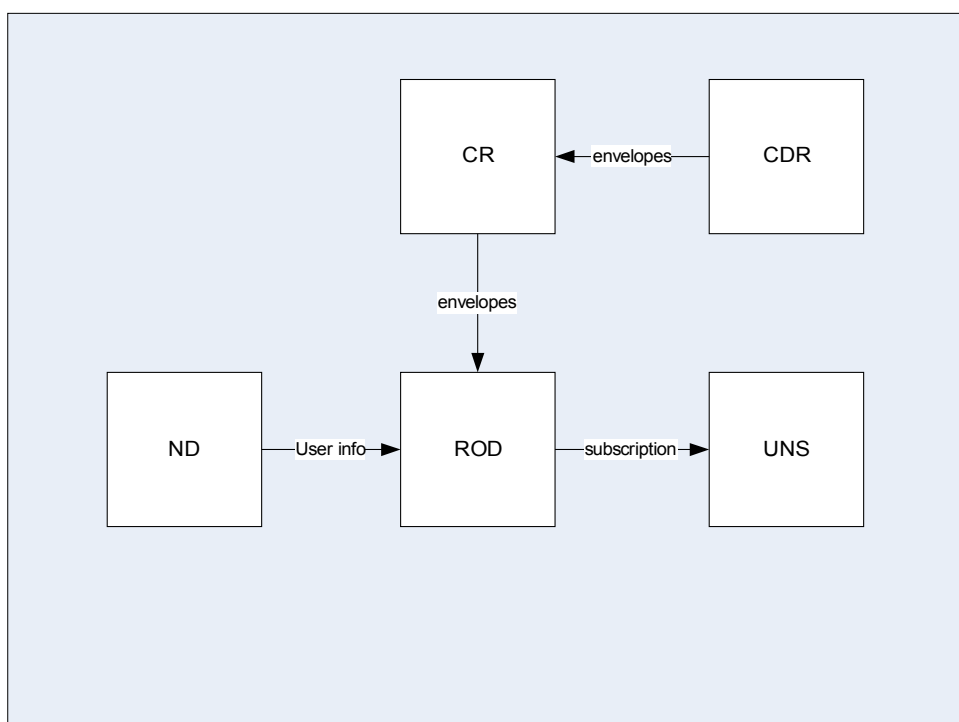
- Creating, updating, searching, browsing and sorting reporting obligations.
- It provides a classification of legal instruments based on Eur-lex.
- It sends notifications to local coordinators in NFPs regarding approaching deadlines.
- Provide framework for analysing overlapping obligations.
- Provide detailed reports about obligations in “EA Core set of indicators”, “Eionet priority data flows” and overlapping obligations.
- Enables classifying obligations according to the DSPIR framework. DSPIR stands for Driving forces, Pressures, States, Impacts and Responses. Particularly useful for policy-makers, DSPIR builds on the existing OECD model and offers a basis for analysing the inter-related factors that impact on the environment.

5.1.3. Users

- The EEA thematic experts who are involved in environmental reporting enabling them to add content and update the database .
- People involved in environmental policy development who are gathering information about an environmental issue (what information is reported where?); They also try to reduce overlapping by searching for duplication of datasets between obligations.
- Country representatives who are involved in environmental reporting (when is reporting due and what do I need to do?);
- People involved in environmental reporting from the perspective of the international organisations receiving and making *use of the data*.

5.1.4. Collaboration

ROD refers to the data definition of a reporting obligation in the Data Dictionary. It uses ACL mechanism and LDAP server from Network Directory (ND) for user authorisation and authentication. Status of deliveries is retrieved from the Content Registry. Provides an RDF channel that contains data about changes in reporting obligations to the Content Registry for harvesting. There is a separate web page where users may subscribe to UNS where statements in the UNS filter are derived from sets of obligation’s metadata.



5.1.5. Technology

| Purpose | Used software |
|---------------------------|--|
| UI client | HTTP browser, HTML-capable (Internet Explorer 5.0, Netscape 6.0, or later) |
| UI Server | HTTP server with JSDK 2.2 compatible servlet engine (Apache Tomcat http://httpd.apache.org/ ; etc.) |
| UI Client-Server protocol | HTTP on TCP/IP |
| Common Gateway Interface | JSDK 2.2 (Java Servlet Development Kit) compatible Java servlets and Java Server Pages (JSP), accessed through the HTTP server's servlet engine. |
| Application Software | Java™ 2 platform http://www.javasoft.com/j2se/1.3/ . |
| Database | MySQL 3.23.43 http://www.mysql.com |
| Database driver | JDBC http://mymysql.sourceforge.net |
| Operating system | Linux |

5.1.6. Artefacts

| Artefact | Location |
|----------------------|---|
| ROD Conceptual Model | https://svn.eionet.europa.eu/repositories/Reportnet/webrod/trunk/docs/ROD_Elements.doc |
| ROD Data Model | https://svn.eionet.europa.eu/repositories/Reportnet/webrod/trunk/docs/ROD_DataModel.doc |
| ROD Detailed Design | https://svn.eionet.europa.eu/repositories/Reportnet/webrod/trunk/d |

| Artefact | Location |
|-------------|---|
| Document | ocs/ROD_DetailedDesign.odt |
| Source code | https://svn.eionet.europa.eu/repositories/Reportnet/webrod/ |

5.2. Data dictionary (DD)

5.2.1. Definition

The Data Dictionary is a central registry for technical specifications of data that should be delivered for reporting obligations. Detailed definition of the data is a key element for achieving an harmonised content in a reporting system such as Reportnet, when definitions may be reused for many reporting obligations and where data suppliers know exactly what data they need to report. It also represents the main mechanism that enables validation and quality assurance of the reported data.

Data Dictionary holds definitions of datasets, tables and data elements. Each of these three levels is defined by a set of attributes, the core set of which corresponds to an ISO 11179 standard for describing data elements. The whole attribute set is flexible and attributes can be added or removed from/to the system.

Data dictionary is the central registry for the XML schema representation of the datasets/dataflows. All quality assurance/control and transformation services inside Reportnet are underpinned with XML schemas from Data dictionary.

5.2.2. Functions

- Provide countries with detailed specifications of what to produce and report, on a website as well as in PDF documents for download;
- Provide data reporters with Excel templates to facilitate reporting;
- Provide definitions of parameters as input for technical quality control and automated validation of the reported data;
- Provide to the people using the reported data by various countries with a single point of reference for obtaining information about the specifications;
- Generate PDF guidelines for data reporters;
- Enables importing new dataset specifications to DD using the MS Access import tool.

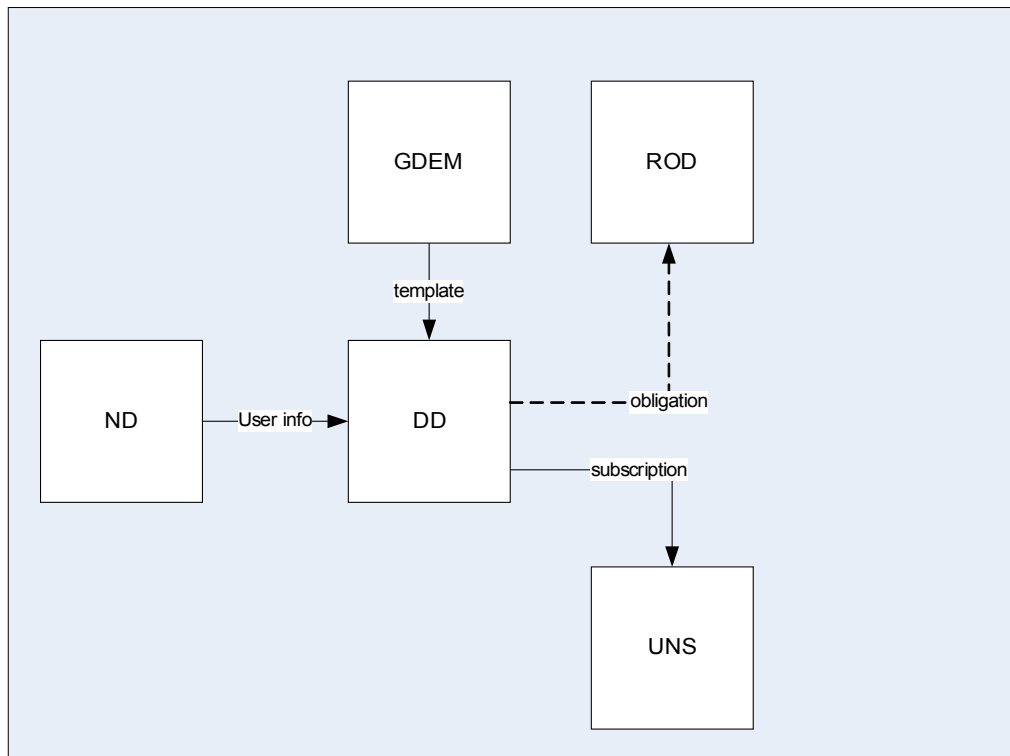
5.2.3. Users

- Data Reporters download from the Data Dictionary web site Excel templates and guidelines in PDF format for particular data flow.
- Data Definers create and update data definitions for reporting obligations according to rules specified in the ISO 11179 standard.

- Administrators maintain the functions in DD that the other two user groups are using.

5.2.4. Collaboration

Data Dictionary uses the conversion services to generate template files (MS Excel, MS Access) from data definition file – XML Schema. It refers to ROD obligations that use particular definition from DD. There is a separate web page where users may subscribe to UNS where statements in the UNS filter are derived from set of elements in data definition. It uses ACL mechanism and LDAP server from Network Directory for user authorisation and authentication.



5.2.5. Technology

| Purpose | Used software or technology |
|---------------------------|--|
| UI client | HTTP browser, HTML-capable (Internet Explorer 5.0 and above, Netscape 6.0 and above). |
| UI Server | HTTP server with JSDK 2.2 compatible servlet engine (Apache 2.0.40, http://httpd.apache.org/ ; Tomcat 4.1.30, http://jakarta.apache.org/tomcat/) |
| UI Client-Server protocol | HTTP on TCP/IP |
| Common Gateway Interface | JSDK 2.2 (Java Servlet Development Kit) compatible Java servlets and Java Servlet Pages (JSP), accessed through the HTTP server's servlet engine. |
| Application Software | Java™ 2 platform http://www.javasoft.com/j2se/1.3/ . |
| Database | MySQL 4.0.1-alpha |

| Purpose | Used software or technology |
|--|--|
| | http://www.mysql.com |
| Database driver | JDBC driver (MySQL Connector/J) http://www.mysql.com/products/connector/j/ |
| Operation system | Linux RedHat 7.1 |
| XML parsing | Xerces Java Parser http://xml.apache.org/xerces2-j/index.html |
| PDF creation on the fly | iText, an open source Java library for generating PDF files on the fly. http://www.lowagie.com/iText/ |
| MS Excel creation on the fly | Apache POI-HSSF, an open source Java library for generating MS Excel files on the fly. http://jakarta.apache.org/poi/hssf/index.html |
| Protocols for accessing API | XML-RPC, a popular protocol that uses XML over HTTP to implement remote procedure calls. http://www.xmlrpc.com/ http://ws.apache.org/xmlrpc/ |
| User management | LDAP (Lightweight Directory Access Protocol) http://www.openldap.org/ |
| Image handling | PJA (Pure Java AWT) Toolkit, an open source Java library for handling images on machines where the native graphics resources is missing. |
| Helper tools for creating data definitions | MS Access 2000. http://office.microsoft.com |

5.2.6. Artefacts

| Artefact | Location |
|-----------------------------|---|
| DD Detailed Design Document | https://svn.eionet.europa.eu/repositories/Reportnet/datadict/trunk/doc/DD%20Detailed%20Design.odt |
| DD user guide | https://svn.eionet.europa.eu/repositories/Reportnet/datadict/trunk/doc/DD%20User%20Guide.odt |
| Source code | https://svn.eionet.europa.eu/repositories/Reportnet/datadict/ |

5.3. CDR

5.3.1. Definition

CDR is content management system for storing national deliveries prepared for reporting obligations issued by ROD. Each county inside CDR has its own folder that may contain collections as organisational units, envelopes or links to other national repositories. The envelope is the unit processed by the data handlers when merging national data into the European data set and contains the files for one delivery in response to a reporting obligation and metadata such as title, description, coverage and year. If the data definition exists in the Data Directory for the dataset to be delivered, data suppliers may choose two different ways to upload their data. They may upload Excel files previously downloaded from DD as templates which will be automatically converted to XML files or input data directly in web forms provided by the CDR. In both cases CDR will store data as XML

files and will perform automatic quality assurance on deliveries so users will be immediately notified about the quality of their data. Besides the automatic quality assurance, storing well defined delivery data as XML files provides the opportunity to end users to display and download delivery content in their favourite formats.

In the heart of CDR is the workflow engine used by data flow managers to construct all the necessary steps that should be performed for a particular envelope to be released.

5.3.2. Functions

- Organises, uploads, downloads, versioning, and tagging of deliveries
- Performs quality assurance and validation of deliveries if they are in XML format
- Enables easy retrieval by end users in their appropriate formats
- Give opportunity to data suppliers to fix issues therefore ensuring that stakeholders will always have access to accurate and reliable information.

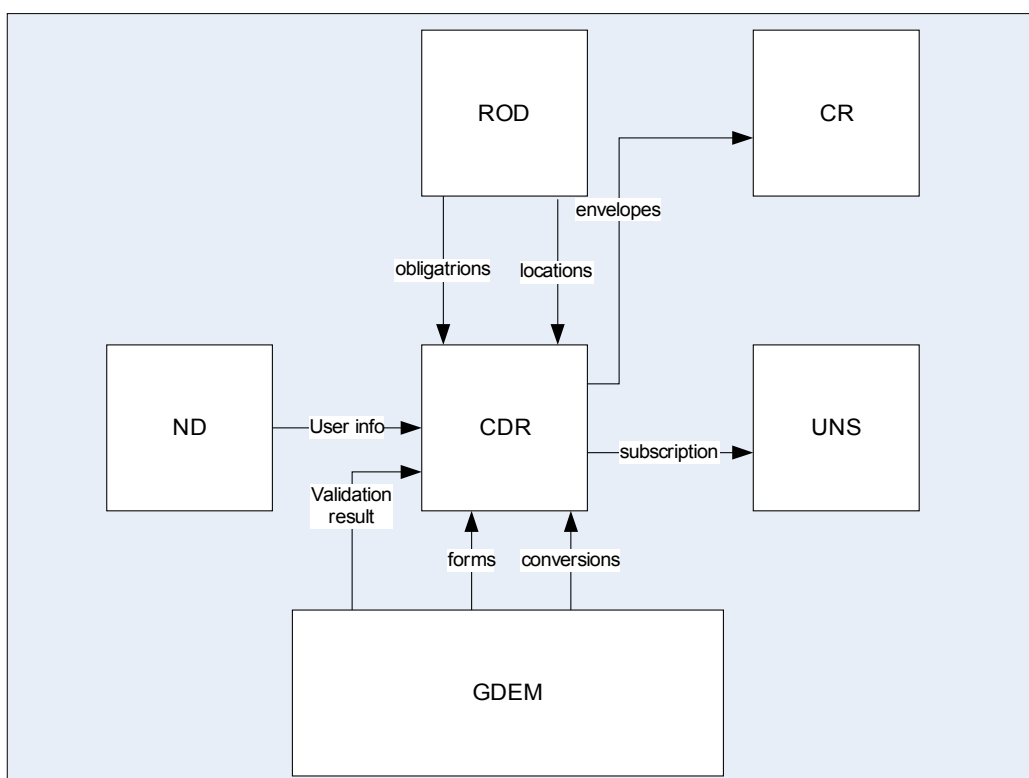
5.3.3. Users

The visitor, who can browse released reports and download the files.

- A collection administrator, who maintain the hierarchical structure, but who doesn't upload any reports.
- A release-coordinator, who releases an envelope to the public. The public must be confident that they are not downloading an incomplete report, so a release/-revocation is logged in the activity log. Once a report is released, it is no longer possible to upload files. If a mistake is found then the release-status can be revoked.
- A reporter, who creates the actual reports. A reporter can request a report to be released when he is finished.
- And then there still is the Zope Manager, who can fix issues if and when the security paradigm doesn't cut it.

5.3.4. Collaboration

CDR depends on two lists that must be retrieved from a different software component: A list of obligations and a list of localities. Both lists are retrieved from ROD via the XML-RPC protocol. Content Registry harvests from CDR a list of envelopes. That list is required by ROD to display the reporting progress of an obligation. There is a separate web page where users may subscribe to UNS where statements in the UNS filter are derived from a set of envelope's metadata. CDR is heavily integrated with other components and uses the conversion, quality assurance and web forms services via a set of XML-RPC calls.



5.3.5. Technology

| Purpose | Used software or technology |
|-----------------------------|--|
| UI client | HTTP browser, HTML-capable (Internet Explorer 5.0 and above, Netscape 6.0 and above). |
| UI Server | ZServer/1.1 |
| UI Client-Server protocol | HTTP on TCP/IP |
| Application Software | Reportek - Zope product for management of collections, envelopes and documents on CDR |
| Database | ZODB Zope Object Database |
| Application server | Zope (Zope 2.9.1-, python 2.4.2) |
| Operation system | Linux RedHat 7.1 |
| Protocols for accessing API | XML-RPC, a popular protocol that uses XML over HTTP to implement remote procedure calls. http://www.xmlrpc.com/ http://ws.apache.org/xmlrpc/ |
| User management | LDAP (Lightweight Directory Access Protocol) http://www.openldap.org/ |

5.3.6. Artefacts

| Artefact | Location |
|---------------------|---|
| README for Reportek | http://www.eionet.europa.eu/software/reportek/readmereportek1 |
| Source code | https://svn.eionet.europa.eu/repositories/Zope/trunk/Reportek/ |

5.4. Content Registry

5.4.1. Definition

Content Registry (CR) is a web application where users can search for the content of services in Eionet, based on their metadata. Not all of the Eionet services are included, only those that have been specified by the CR web site administrators.. CR uses the DublinCore Metadata Element Set to describe the content, but it also keeps track of the Eionet specific metadata elements.

There are a number of owners of metadata (services) inside the Eionet and as a result the Content Registry may be configured to collect metadata from any service. The process of collecting metadata from services is called harvesting, where harvesting scope represents a set of data services that will be harvested. Each service is identified by its URL.

5.4.2. Functions

- Harvest xml files in RDF format from repositories such as CDR;
- Store the entire RDF file in the database for future reuse ;
- Enable basic and advanced search by metadata elements via the web interface

5.4.3. Users

- People that supervise the data reporting activities. It allows them to search delivered data sets by date, obligation, and responsible person.
- Administrators configure harvesting scope, that is set of services that will be harvested.

5.4.4. Collaboration

All components in the Reportnet may use Content Registry service to get metadata about a particular topic. Content Registry harvest repository of obligations from ROD and repository of reporting deliveries from CDR.

5.4.5. Technology

| Purpose | Used software or technology |
|---------------------------|--|
| UI client | HTTP browser, HTML-capable (Internet Explorer 5.0 and above, Netscape 6.0 and above). |
| UI Server | HTTP server with JSDK 2.2 compatible servlet engine (Apache 2.0.40, http://httpd.apache.org/ ; Tomcat 4.1.30, http://jakarta.apache.org/tomcat/) |
| UI Client-Server protocol | HTTP on TCP/IP |
| Common Gateway Interface | JSDK 2.2 (Java Servlet Development Kit) compatible Java servlets and Java Servlet Pages (JSP), accessed through the HTTP server's servlet engine. |
| Application Software | Java™ 2 platform http://www.javasoft.com/j2se/1.3/ . |

| | |
|-----------------------------|--|
| Database | MySQL 4.0.1-alpha http://www.mysql.com |
| Database driver | JDBC driver (MySQL Connector/J) http://www.mysql.com/products/connector/j/ |
| Operation system | Linux RedHat 7.1 |
| XML parsing | Xerces Java Parser http://xml.apache.org/xerces2-j/index.html |
| RDF parsing | ARP: Another RDF Parser. This is an open source Java library for Jena RDF/XML parsing. http://www.hpl.hp.com/personal/jjc/arp/ |
| Protocols for accessing API | XML-RPC, a popular protocol that uses XML over HTTP to implement remote procedure calls. http://www.xmlrpc.com/ http://ws.apache.org/xmlrpc/ |
| User management | LDAP (Lightweight Directory Access Protocol) http://www.openldap.org/ |

5.4.6. Artefacts

| Artefact | Location |
|-----------------------------|---|
| CR Detailed Design Document | http://eea.eionet.europa.eu/Public/irc/eionet-circle/eionet-telematics/library?l=/reportnet/cr_ddd_v02_doc/EN_0.2_&a=d |
| Source code | https://svn.eionet.europa.eu/repositories/Reportnet/contreg/ |

5.5. Conversion service

When data is stored as XML it is not readily viewable by anyone. The Conversion Service is responsible for converting XML files into human friendly formats like HTML, MS Excel, PDF or even into SQL scripts. Conversion Service is converting also MS Excel files into an XML format. But the prerequisite of that function is that the MS Excel file has been generated from the Data Dictionary and the user strictly followed the DD specifications and has not deleted any required attributes in the file.

5.6. Quality Assessment (QA)

Quality Assessment (QA) service is checking the quality of the delivered XML files that may be produced either by filling out fields in web forms or convert an Excel template generated in DD. The quality check scripts are based on the defined quality rules. QA service uses currently XML Schema validation and XQuery scripts for making quality assurance. While the XML Schema validation is just for simple syntax and allowed values check, the XQueries can do more complex validation checks. For example it could check that the XML file contains information about all the stations in a given country, and that no station occurs twice. The validation service is integrated into the Quality Assessment service, but it can be used also separately.

5.6.1. Users

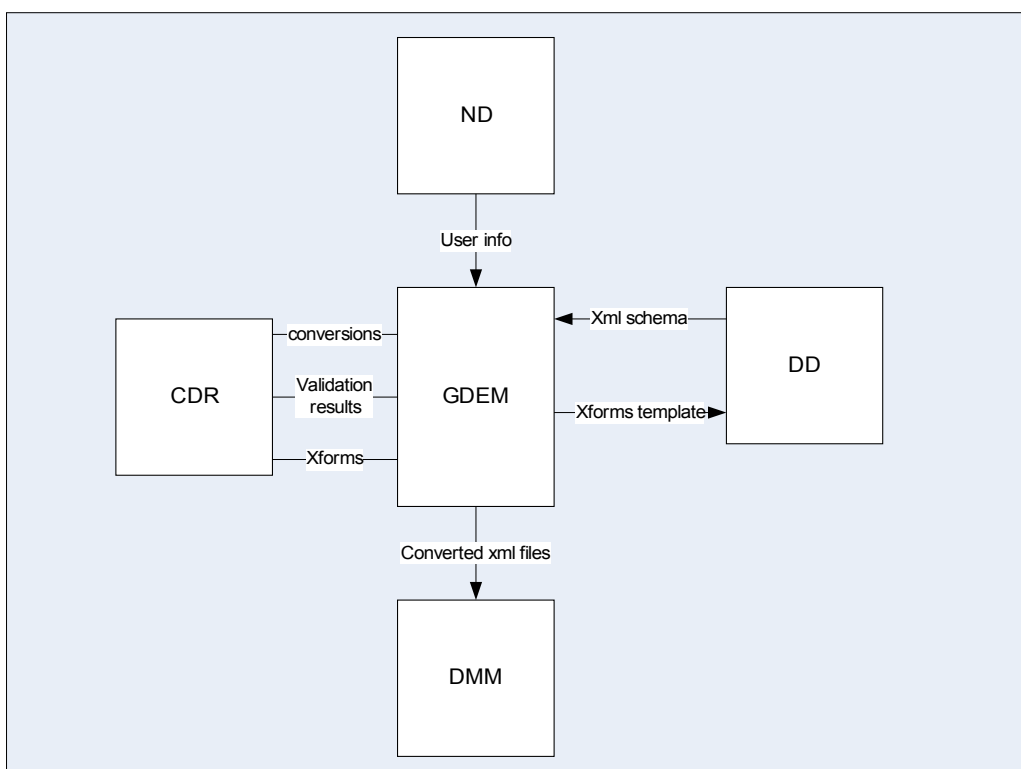
Data definers define quality analysis rules for automatic quality review of each requested data set.

5.7. WebForms Service

WebForms Service is an option for delivering data into the central data repository (CDR). CDR is using WebForms service for allowing the user to upload deliveries in XML format. The logic how a data supplier reaches to web forms is handled already in CDR. Following the user's input, WebForms service produces automatically valid XML files as deliveries into his/her relevant CDR envelope. Users can pre-fill the web forms with data coming from last year's deliveries and make the updates as needed. They can also start to fill out an empty web form.

5.7.1. Collaboration

The WebForms service will use the XML-Schema's that are generated by the Data Dictionary in order to generate the necessary web questionnaires. Web questionnaire also retrieves the element definitions normative description from the DD through the XML-RPC protocol. This way the input received can be validated.



5.7.2. Technology

| Purpose | Used software or technology |
|-----------|---|
| UI client | HTTP browser, HTML-capable (Internet Explorer 5.0 and |

| Purpose | Used software or technology |
|-------------------------------|--|
| | above, Netscape 6.0 and above). |
| UI Server | HTTP server with JSDK 2.2 compatible servlet engine (Apache 2.0.40, http://httpd.apache.org/ ; Tomcat 4.1.30, http://jakarta.apache.org/tomcat/) |
| UI Client-Server protocol | HTTP on TCP/IP |
| Common Gateway Interface | JSDK 2.2 (Java Servlet Development Kit) compatible Java servlets and Java Servlet Pages (JSP), accessed through the HTTP server's servlet engine. |
| Application Software | Java™ 2 platform http://www.javasoft.com/j2se/1.3/). |
| Database | MySQL 4.0.1-alpha http://www.mysql.com |
| Database driver | JDBC driver (MySQL Connector/J) http://www.mysql.com/products/connector/j/ |
| Operation system | Linux RedHat 7.1 |
| XML parsing | Xerces Java Parser http://xml.apache.org/xerces2-j/index.html |
| XSL Transformations | Xalan-Java 2.5 - transforming XML documents into HTML, text, or other XML document types. http://xml.apache.org/xalan-j/index.html |
| MS Excel creation and reading | Apache POI-HSSF, an open source Java library for generating and reading MS Excel files. http://jakarta.apache.org/poi/hssf/index.html |
| XQueries processor | Saxon XQuery engine for running XQuery scripts http://saxon.sourceforge.net/ |
| XForms server-side processor | Chiba - an Open Source Java Implementation of the W3C XForms standard that delivers most XForms functionality to today's browsers. http://chiba.sourceforge.net/ |
| PDF creation | Apache FOP (Formatting Objects Processor) - print formatter driven by XSL formatting objects (XSL-FO) http://xml.apache.org/fop/ |
| Protocols for accessing API | XML-RPC, a popular protocol that uses XML over HTTP to implement remote procedure calls. http://www.xmlrpc.com/ http://ws.apache.org/xmlrpc/ |
| User management | LDAP (Lightweight Directory Access Protocol) http://www.openldap.org/ |

5.7.3. Artefacts

| Artefact | Location |
|-------------------------------|---|
| Source code WebQ module | https://svn.eionet.europa.eu/repositories/Reportnet/webq/ |
| Source code Converters module | https://svn.eionet.europa.eu/repositories/Reportnet/xmlconv/ |

5.8. Network Directory (ND)

5.8.1. Definition

Network Directory represents a web interface to the LDAP based registry concerning all persons, organisations, and their roles inside Eionet and other familiar networks such as EPER.

It should be considered as the core service for user authentication and authorisation (together with the developed ACL library) for all Reportnet components.

5.8.2. Functions

- Enables easy searching and browsing information about users, roles and groups by different criteria.
- Provides complete role and organisation information, which can be used by other application.
- Allow user authentication.
- Enables user authorisation provided through the appropriate ACL mechanism.
- Provides contact information for all members of the Eionet.
- Enable the creation of mailing list inside common mail clients.

5.8.3. Users

National coordinators maintain the content for members at national level.

5.8.4. Collaboration

Network Directory is called from other components in Reportnet that need to authenticate the users and software components that need information about users and roles stored in Eionet LDAP directory.

5.8.5. Technology

| Purpose | Used software or technology |
|---------------------------|---|
| UI client | HTTP browser, HTML-capable (Internet Explorer 5.0 and above, Netscape 6.0 and above). |
| UI Server | ZServer 1.1.b1 based on Medusa |
| UI Client-Server protocol | HTTP on TCP/IP |
| Application Software | Python |
| Database | ZODB Zope Object Database |
| Application server | Zope/(Zope 2.6.4 (binary release, python 2.1, linux2-x86), python 2.1.3, linux2) |
| Operation system | Linux RedHat 7.1 |

| Purpose | Used software or technology |
|-----------------------------|--|
| Protocols for accessing API | XML-RPC, a popular protocol that uses XML over HTTP to implement remote procedure calls. http://www.xmlrpc.com/ http://ws.apache.org/xmlrpc/ |
| User management | LDAP (Lightweight Directory Access Protocol) http://www.openldap.org/ |

5.9. DMM

5.9.1. Definition

The Data Merging Module is a client desktop application that merges an entire obligation and imports it into the single European database. DMM represents the final component that provides quality assurance and control in a data/information flow between national and European institutions. The main objectives of DMM are to create datasets on European level with reliable data and to make the merging process much easier and efficient for environmental experts. DMM utilise menu other components to support data merging, witch is a proof of good design of overall Reportnet architecture.

5.9.2. Functions

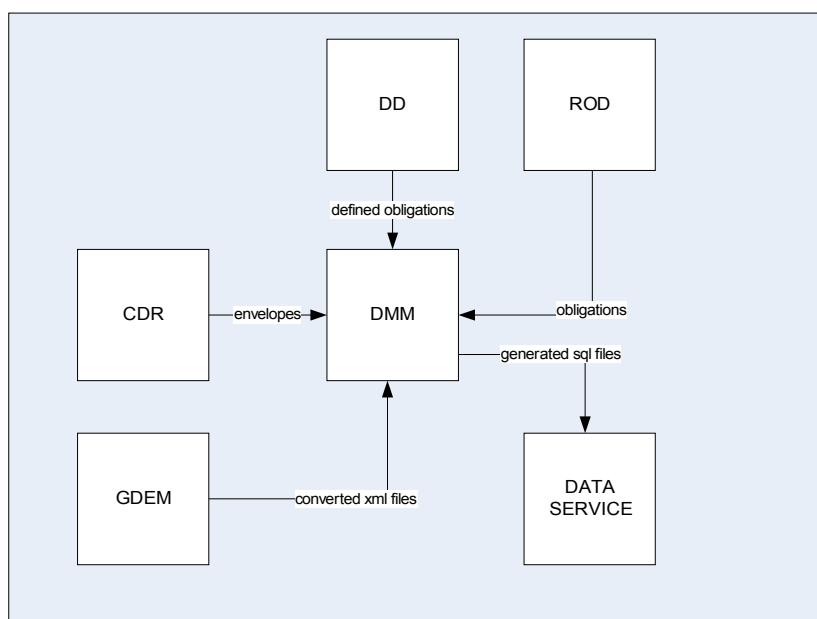
- Enables easy retrieving of obligations in a local database or in a file with XML or HTML format;
- Compares a new extraction with previously extracted data on the basis of uploaded time;
- Enables automatic and scheduled realisation of merges

5.9.3. Users

- Data collectors dealing with the merging of European datasets

5.9.4. Collaboration

DMM workflow begins when it retrieves obligations which have data definition entry in the Data Directory. After that DMM retrieves all such obligations from ROD and released envelopes for those obligations from CDR, it convert envelopes using GDEM conversion service to SQL files, that will be merged in reference databases hosted in ETCs.



5.9.5. Technology

| Purpose | Used software or technology |
|----------------------|--------------------------------------|
| Application Software | C# on .NET 1.1 |
| Database | MS Access |
| Graphical library | Windows Forms |
| Operation system | Windows NT, Windows 2000, Windows XP |

5.9.6. Artefacts

| Artefact | Location |
|---------------------------|--|
| FSD - Data Merging Module | http://eea.eionet.europa.eu/Members/irc/eionet-circle/eionet-telematics/library? l=/reportnet_project/specialist_files/wp05_merging_module/dmm_ guide_v10doc/ EN 1.0 &a=d |
| TSD - Data Merging Module | http://eea.eionet.europa.eu/Members/irc/eionet-circle/eionet-telematics/library? l=/reportnet_project/specialist_files/wp05_merging_module/mergi ng_module_v10doc/ EN 1.2 &a=d |
| DMM - User Guide | http://eea.eionet.europa.eu/Members/irc/eionet-circle/eionet-telematics/library? l=/reportnet_project/specialist_files/wp05_merging_module/dmm_ guide_v10doc/ EN 1.0 &a=d |
| Source code | https://svn.eionet.europa.eu/repositories/Reportnet/datamerge/ |

5.10.Data Service

5.10.1. Definition

The data service provides almost all data sets and applications which have been used in EEA's periodical environmental reports. All data is reported to the Data Warehouse inside data service, be they at ETC or NFP. The EEA Data Warehouse will be integrated with the ETC reference databases so that data can automatically be replicated to it from the topic databases.

The data service contains 3 major elements that each of them has it's specific purposes. – Data sets, Applications and Maps/Graphs. The elements are needed to allow relationships between reference centre content and the used datasets. A published indicator fact sheet or a map in a report always comes from one or more datasets. The map in the report is a timestamp where the dataset is a continues growing object. These three elements allow us to link those together.

5.10.2. Functions

- Provides data discovery and visualisations services
- Enables downloading of datasets inside data warehouse
- Provides custom applications for browsing datasets , e.g. AirView

5.10.3. Users

The data collectors merge national data into European data, supported by the definitions stored in Reportnet's data dictionary.

5.10.4. Collaboration

This component is processed from the documents contained in Central Data repository.

In some cases DEMs can directly upload to the DW. The Data Service are accessed by Indicator Management tools for the purpose of producing assessments and scenarios.

5.10.5. Technology

| Purpose | Used software or technology |
|---------------------------|---|
| UI client | HTTP browser, HTML-capable (Internet Explorer 5.0 and above, Netscape 6.0 and above). |
| UI Server | Microsoft-IIS/6.0 |
| UI Client-Server protocol | HTTP on TCP/IP |
| Application Software | Active Server Pages ++ Visual basic (Active X components) |
| Database | MS SqlServer 2000 |
| Application server | ASP script engine |
| Operation system | Windows Server 2003 |

5.11. IMS (Indicator Management Service)

5.11.1. Definition

For the core set of indicators, EEA is developing a web-based indicator management service that allows thematic and sectoral experts working with EEA to draft specifications for the indicators required by the EEA work programme. IMS is based on the indicator profile that was developed from the EEA indicator facts-sheet template to support the preparation of the core set of indicators. The main objective of the IMS is to close the gap between indicator preparation and publishing.

The publishing facility in IMS is used to make approved specifications available to the public via the EEA web site. The IMS also supports the drafting and web-publishing of the regular indicator assessments based on the data sources and methodology in the specification.

Eionet partners have the facility to review and comment the final drafts of indicator specifications and assessments through IMS. The IMS will also provide the possibility to maintain information on the quality of an indicator according to specific rules defined for the relevant indicator set.

5.11.2. Functions

- Provides a transparent audit trail enabling efficient information flows and tracking of the quality of data from the original source in countries to final presentation at the European level;
- Enables drafting, reviewing (including consultation with countries), quality assurance and publishing (including dissemination calendar) of all indicators in the core set;
- Enables downloading of indicator data sets for own use and calculation;
- Provides discussion forums on indicator outcomes and to support regular review processes;
- Provides glossary for management of the core set indicators;
- Provides access to procedural guidelines and templates.

5.11.3. Users

- Assessment producers that implements EEA's core set of environmental indicators.
- Environmental experts may see indicator specification and put comments on draft specification of indicators.
- Policy makers or the general public, can view the environmental indicators.

5.11.4. Collaboration

- For each data set, there may be an identified linkage to a reporting obligation (in ROD).
- For each indicator in the EEA Core Set, the IMS lists the Data Sets from the data service, that are used in the calculation of the indicator.

5.11.5. Technology

| Purpose | Used software or technology |
|---------------------------|---|
| UI client | HTTP browser, HTML-capable (Internet Explorer 5.0 and above, Netscape 6.0 and above). |
| UI Server | Medusa HTTP server integrated into Zope |
| UI Client-Server protocol | HTTP on TCP/IP |
| Application Software | Plone Content management system for Zope |
| Database | ZODB Zope Object Database |
| Application server | Zope |
| Operation system | Linux RedHat 7.1 |
| User management | LDAP (Lightweight Directory Access Protocol) http://www.openldap.org/ |

5.12. Unified Notification System (UNS)

5.12.1. Definition

The Unified Notification System allows Eionet authenticated users to receive notifications with events from any of the RSS/RDF channels available inside Eionet. A set of the user roles assigned to a channel would determine on which channels users may subscribe. The most often types of events are related to metadata in a data/information flow such as obligation deadlines or changed data definition in the Data Directory. Notifications may be delivered to the users in several different ways, email message, jabber message, RSS feed or users may see its notifications on a web dashboard panel that user may customise. Advanced option that UNS offers to the users is the creation of filters. Filter is a set of statements (conditions), in shape of key-value pair, that excludes all notifications that doesn't satisfy certain conditions.

UNS may work in two modes: the PULL mode and the PUSH mode. In PULL mode UNS will harvest information from a channel's URL where in PUSH mode external applications will push events to the UNS.

5.12.2. Functions

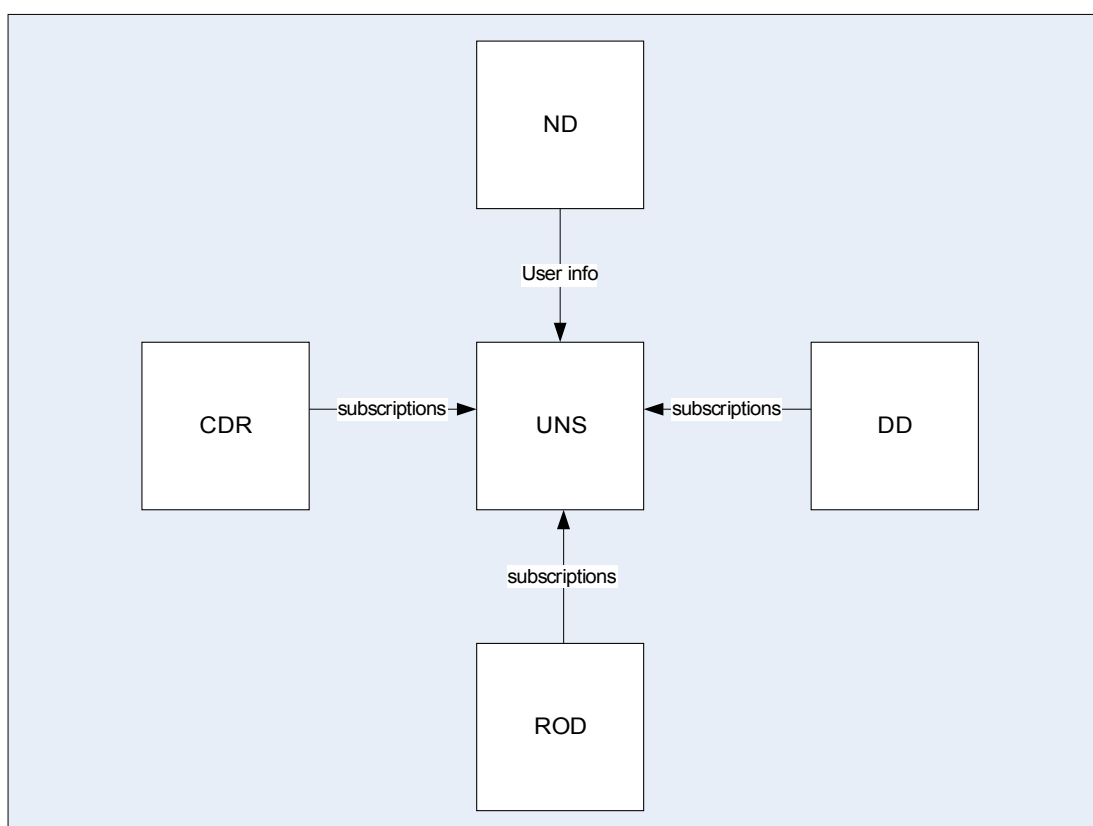
- Harvests events from available channels and store its content in the database;
- Generates reports about notification based on different criteria;
- Receives events from external applications and store its content in the database;
- Sends notifications to subscribers;

5.12.3. Users

- All Eionet authorised users may subscribe to channels;
- UNS Administrator, creates updates and delete channels, defines parameters of channels like assigned roles, delivery types , notification templates etc.;
- RPC users create and update PUCH channels and send their events to UNS.

5.12.4. Collaboration

UNS authorise users and get their roles and personal data via ACL mechanism and Network Directory. There is a separate web page on CDR, DD and ROD from which users can make subscriptions to channels in UNS along with filters.



5.12.5. Technology

| Purpose | Used software or technology |
|---------------------------|---|
| UI client | HTTP browser, HTML-capable (Internet Explorer 5.0 and above, Netscape 6.0 and above). |
| UI Server | Tomcat 5.0.28, http://jakarta.apache.org/tomcat/ |
| UI Client-Server protocol | HTTP on TCP/IP |
| Application Software | Java™ 2 platform http://www.javasoft.com/j2se/1.3/ . |
| Database | MySQL 4.0.1-alpha http://www.mysql.com |
| Database driver | JDBC driver (MySQL Connector/J) |

| Purpose | Used software or technology |
|-----------------------------|--|
| | http://www.mysql.com/products/connector/j/ |
| Operation system | Linux RedHat 7.1 |
| XML parsing | Xerces Java Parser http://xml.apache.org/xerces2-j/index.html |
| XSL Transformations | Xalan-Java 2.5 - transforming XML documents into HTML, text, or other XML document types. http://xml.apache.org/xalan-j/index.html |
| Protocols for accessing API | XML-RPC, a popular protocol that uses XML over HTTP to implement remote procedure calls. http://www.xmlrpc.com/ http://ws.apache.org/xmlrpc/ |
| User management | LDAP (Lightweight Directory Access Protocol) http://www.openldap.org/ |

5.12.6. Artefacts

| Artefact | Location |
|---------------------------|---|
| Functional Specifications | https://svn.eionet.europa.eu/repositories/Reportnet/UNS/trunk/doc/FSD%20-%20Unified%20Notification%20Service%20-%20V2_0.doc |
| Technical Specifications | https://svn.eionet.europa.eu/repositories/Reportnet/UNS/trunk/doc/UNS%20Detailed%20Design.odt |
| UNS User Guide | https://svn.eionet.europa.eu/repositories/Reportnet/UNS/trunk/doc/UNS%20User%20Guide.odt |
| Source code | https://svn.eionet.europa.eu/repositories/Reportnet/UNS/trunk/ |

6. Software engineering methodology

Official software development methodology for the Reportnet project is a modified spiral model, originally defined by Barry Boehm, called “The Win-Win spiral model”, which extends the spiral software development model by adding “Theory W” activities to the front of each cycle. Detailed description of this model may be found in Boehm work called “Anchoring the Software Process”. The intention of spiral models is not to define in detail the entire system at first but using an evolutionary approach with active participations of stakeholders during all phases of the software development. This methodology represents the best approach when user requirements are not fully defined ahead of the implementation and where it is expected that the nature of requirements may be changed during the development phase of the project or the general evolution of the business case itself that has been used to produce the initial set of requirements,. Prototypes have a key role in software development when stakeholders agreed that implemented software match their needs. Modular architecture of the Reportnet assures a relatively fast development of prototypes.

Steps in the spiral model include:

- The new system requirements are defined. This usually involves interviewing a number of users representing all the end users of the system.

- A preliminary system design is created.
- An initial prototype of the new system is constructed from the preliminary design. This is usually a scaled-down system, and represents an approximation of the final product's characteristics.
- After evaluating the strengths, weaknesses and risks of the first prototype, a second prototype is developed and tested.
- If the risk is considered to be too high, it may be possible to decide the termination of the project or parts of the project at this point. Risk factors to be considered include development cost overruns, operating-cost miscalculation, etc.
- Subsequent prototypes are developed until the customer is satisfied that the latest prototype represents the desired product.
- The system is constructed, based on the final prototype.
- The final system is evaluated and tested. Routine maintenance is carried out continually to prevent large-scale failures and to minimise downtime

7. Conclusions or Recommendations

We describe in previous sections key concepts behind Reportnet with the objective of allowing an easier understanding of its overall functionality. Reportnet represents a strong information framework for reporting system inside Eionet. By following key principles for comprehensive reporting network such as streamline data/information and data harmonisation it makes all business processes inside the community much easier and efficient. In this section we describe in short well known drawback of old EIONTET reporting system which are caused born of Reportnet as well as key benefits of implementing new solutions.

7.1. Limitations of existing solution

Eionet is a young network connecting people and organisations across Europe that share information and data, and collaborate to in order to put in practice environmental policies defined by environment decision-making institutions and organisations... The number of member countries involved, organisations, reporting obligations etc., continually increase, therefore various constraints and limitations are starting to appear within the Eionet network:

- Reporting burden increase all time without synchronisation and harmonisation between different data flows.
- No central place inside community that will answer on basic questions related to reporting that is why, when, what, who, where and how stakeholders must deal with obligations.
- Different methodology and technology used in collection of data and reporting data from national to European level.
- Stakeholder don't always know of each others data collection activities.

- Much of material, technical and human resources needed to serve one data flow.
- Poor response rates to many reporting obligations.

7.2. Benefits of the Reportnet

Reportnet contributes to all functionalities inside Eionet reporting system by:

- Reducing manual work in all data/information phases
- Eliminating overlaps in data/information flows between different reporting obligations
- Harmonising data between different data flows with central data registry
- Providing transparency and availability information by content registry and web site publishing
- Notifying users about important events in data/information flows
- Reducing effort for manual quality assurance/control and validation of data
- Facilitating import data in European Data Service
- Making efficient management of data/information flows
- Providing more policy target data
- Providing advance data handling function and conversion
- Enabling data reuse by conversation service which convert reported data in stakeholder appropriate data.
- Providing metadata framework for transparent handling of data flows.
- Harmonising tools for reporting data in different topics
- May be reusable in other reporting networks
- Easy to learn through standard webbrowser interface
- Enhancing Corporate memory
- Enabling strong support for standardisation as key process for interoperability