DELIVERABLE

Project Acronym:	**LoCloud**
Grant Agreement number:	**325099**
Project Title:	**Local content in a Europeana cloud**

D2.3: Modified MoRe Prototype

Revision: final

Authors:

Costis Dallas (DCU)
Panos Constantopoulos (DCU)
Dimitris Gavrilis (DCU)
Stavros Angelis (DCU)
Dimitra Nefeli Makri (DCU)
Eleni Afiontzi (DCU)

Project co-funded by the European Commission within the ICT Policy Support Programme		
Dissemination Level		
P	Public	x
C	Confidential, only for members of the consortium and the Commission Services	

Revision History

Revision	Date	Author	Organisation	Description
0.1	26/02/2014	Dimitris Gavrilis	DCU	Initial draft
0.2	5/03/2014	Costis Dallas, Panos Constantopoulos	DCU	Revisions
1.0	7/03/2014	Dimitris Gavrilis,	DCU	Revisions
1.1	10/03/2014	Kate Fernie	MDR	Revisions
1.2	7/03/2014	Stavros Angelis	DCU	Added updated XSLTs
1.3	10/03/2014	Dimitris Gavrilis	DCU	Added executive summary, conclusions etc.

Statement of originality:

This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both.

Contents

Executive Summary

This deliverable presents the modified cloud MoRe prototype which has been implemented according to the specifications layed out in the D2.1 deliverable. The cloud prototype has been implemented based on a pluggable storage architecture where multiple technologies can be combined in order to extend the system. The Apache Cassandra framework has been used as the prototype's storage layer with a content model inspired by the previous MoRe version (a content model that supports coxmplex digital objects with versionable datastreams). The core services implementation of the MoRe prototype are presented with the most significant to be the messaging service that acts as a glue for the rest of the core services. Finally, the scalability architecture and capabilities are presented in detail.

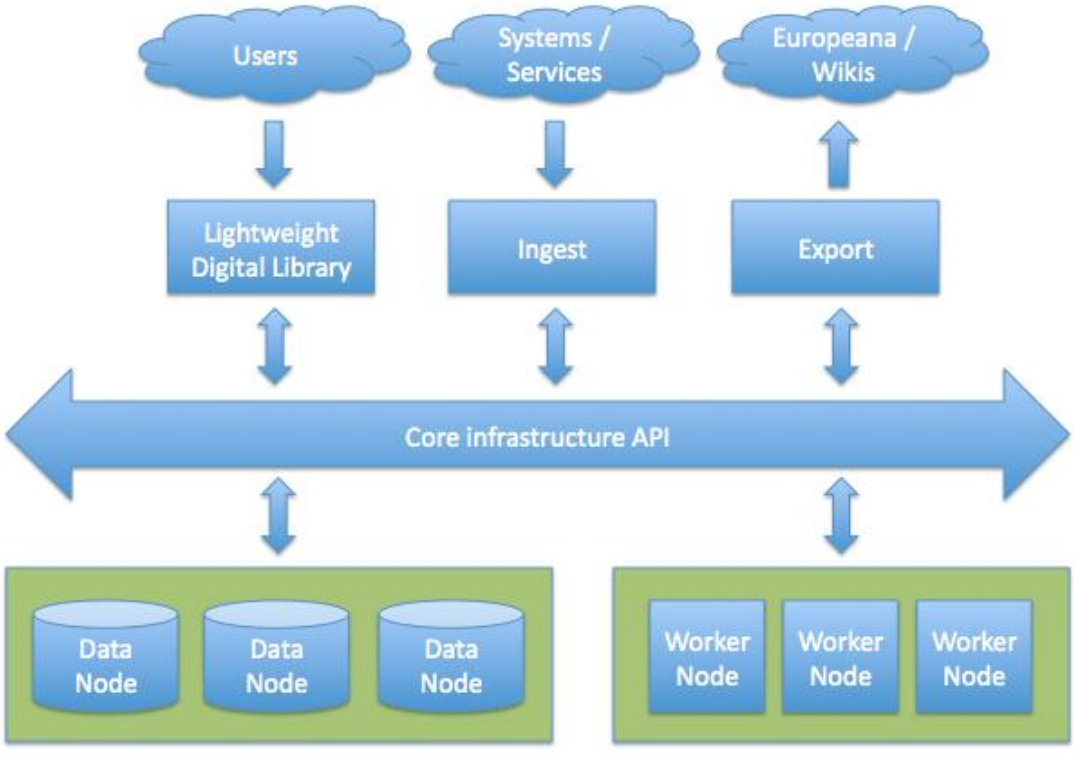
Introduction

This deliverable presents the first prototype of the MoRe metadata aggregator according to the LoCloud core infrastructure specifications as defined in D2.1. The MoRe aggregator architecture is presented in detail (focusing on its technical aspects) along with the modifications that will be made. A comparison between the traditional (monolithic) and the cloud (scalable) versions is discussed. The technologies used to construct the prototype are also presented along with a roadmap towards the final version.

The MoRe metadata aggregator in its current version uses a fedora-commons repository to store metadata records. Various enrichment and support services are built on top of MoRe and interact with the repository in order to function. This architecture, although it is robust, presents scalability issues, which fall into two categories:

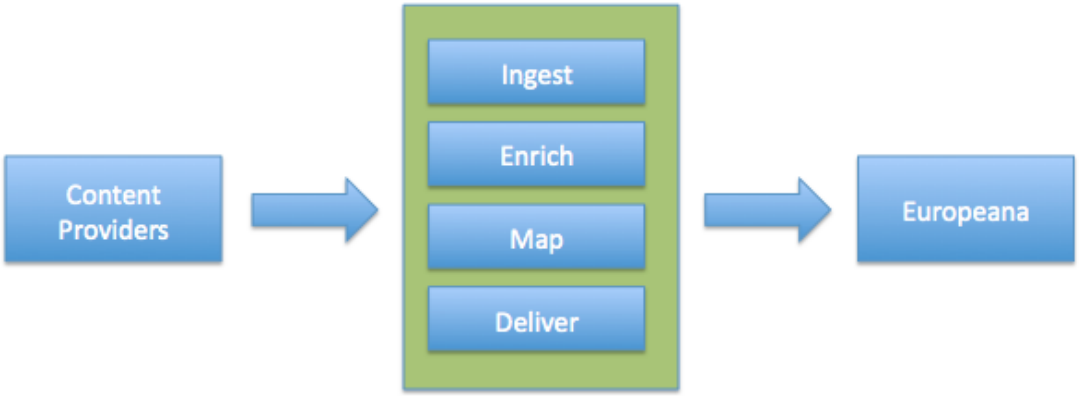
a) scalability of storage
b) scalability of services

The next version of MoRe whose first prototype is presented here, aims to overcome these scalability issues. This presents the most significant change to the aggregator. Other changes include the support of multiple intermediate schemas and a more advanced enrichment services framework.



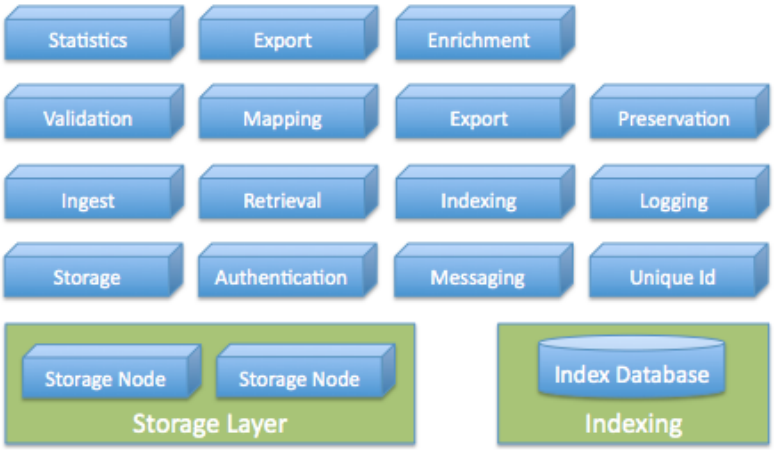
The cloud based overall architecture can be seen in the above diagram where in the lower part, the data (or storage) nodes and the worker nodes can be seen. The storage nodes in LoCloud are responsible for storing metadata and the worker nodes for providing a scalable and fault-tolerant set of aggregator sercvices. These services are responsible for the content aggregation tasks whch can be seen in the picture blow and follow four main steps:

a) content is first ingested into the infrastructure,
b) metadata records are enriched through one ore more of the enrichment micro-services,
c) metadata are mapped from one of the intermediate schemas supported to (EDM)
d) metadata are exported to Europeana or other interested parties



1. MoRe cloud prototype architecture

From a services point of view, the MoRe cloud metadata aggregator comprises the following services:



These services are used in order to complete all aggregation tasks and can also be used to communicate with complementary LoCloud services such as MINT, the Lightweight Digital Library, and additional enrichment micro-services (such as the Historic place names micro-service).

The following are key characteristics of the MoRe cloud prototype:

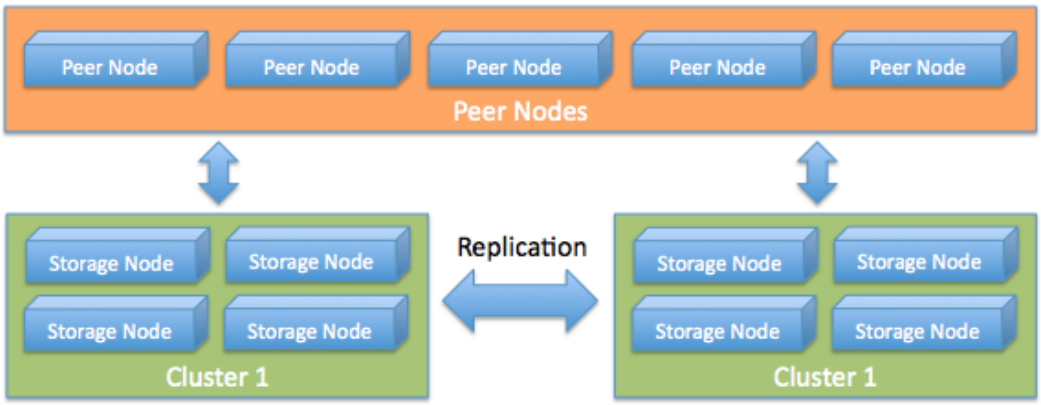
- fault-tolerance,
- high-availability,
- elasticity, and
- scalability.

These are provided by:

a) the storage layer, and
b) the decentralized and clustered design of the various services.

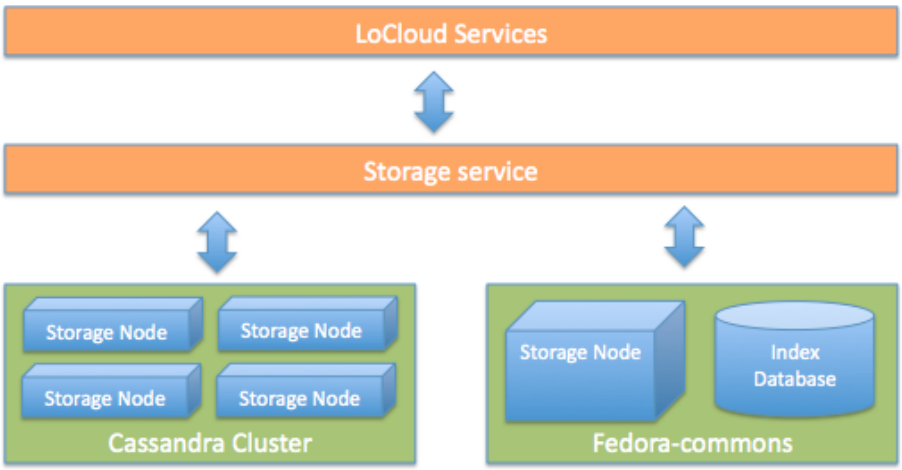
2.1 Storage layer

The storage layer provides a scalable, elastic and high-availability persistent storage for storing metadata. The main technology for providing this persistent storage used to build the prototype was Apache Cassandra. Apache Cassandra uses noSQL database in a P2P environment to provide all of the above features. Multiple clusters of nodes can be defined with various replication scenarios and each cluster can be scaled with more nodes. Futhermore, as the figure below depicts, a number of peer nodes are used to query the cluster thus eliminating any single point of failure and bottleneck issues.



2.1.1 Pluggable storage drivers

In order to ensure support for multiple storage technologies, a storage service was built on top of the cloud prototype in order to allow for integration of other technologies such as Fedora-commons, MySQL clusters, Mongo DB, etc. This storage service uses the bridge design pattern to register multiple drivers of storage technologies. In this deliverable the Cassandra driver is presented.



This architecture allows existing and future technologies to be easily integrated into LoCloud. In order to achive this, the current implementation makes the following assumptions:

a) The content model is complex. This requires that every digital object is comprised of parts (called datastreams).
b) The versionable datastreams and replication is handled by each storage technology internally (e.g. it is transparent to the rest of the services).

2.1.2 Content model

The content model used is described in detail in Annex I and it is based on the proven model used by previous versions of MoRe. The main advantages of this model are that:

- it makes very few assumptions on the specifics of the content,
- it defines complex objects, and
- it supports datastream versioning.

2.2 Digital preservation

One significant characteristic of the MoRe metadata aggregator is that it supports preservation management of digital resources. This goes beyond backups which are covered by the storage infrastructure but include a mechanism for:

- storing the entire set of changes of an object's datastream, and
- keeping an audit log in PREMIS with all the modification events performed on the object.

Both features have been implemented and are transparent to the rest of the services. The new versions and the PREMIS audit log are generated automatically on write events.

2.3 Services layer

The services layer provides a framework for building services and integrating them into the LoCloud infrastructure. Each service must be able to have access to the storage layer and be able to scale up thus not creating bottlenecks or a single point of failure.

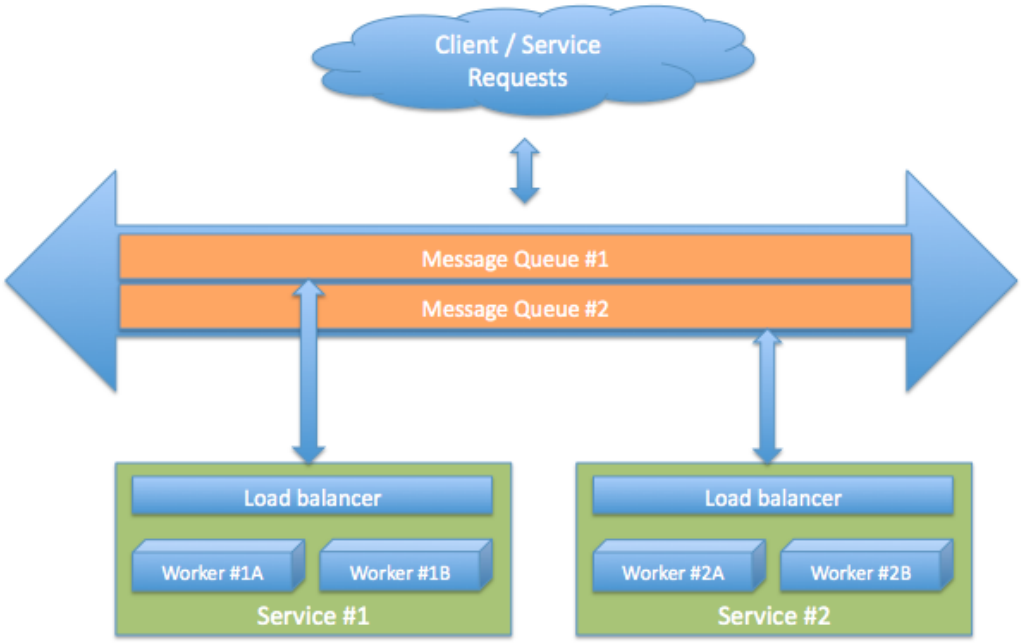
2.3.1 Scalability

The cloud prototype of MoRe includes a scalable services layer which allows more worker nodes for each service to be added. The load balancing is implemented with two main technologies:

- Using a simple round robin balancer to evenly distribute requests among a cluster group of worker nodes. For this case, the apache mod_proxy is used (see Annex II).
- Using a message queue that supports load balancing using durable queues. For this case, the RabbitMQ is used (see Annex III).

The figure below shows the general architecture of the various core services within LoCloud. The inter-service communication is implemented using RabbitMQs queue capabilities. When a new service request is generated (e.g. a transformation request after an ingest), a message is published to the respective queue and the consumer service (in this case, the mapping service) receives the message, performs the task and upon completion, notifies the rest of the services by publishing another message.

In a scalable environment, where multiple consumer services exist (e.g. multiple transformation services on multitple nodes) messages are consumed by the next available worker node. It is possible to add another layer of load balancing behind clusters of workers (for cases of highly processs intensive tasks) using mod_proxy.



2.3.2 Inter-service communication

The inter-service communication or, more specifically in our case, Message Queuing is a crucial component of the infrastructure. The messaging service is not just a mechanism for producing messages to the end users/services but it is used to execute workflows, provide load balancing etc. It essentially acts as glue in a de-coupled service environment. The Messaging technology provides the following advantages:

- Performance: as it allows to perform tasks asynchronously
- Decoupling: by de-coupling services, complexity is reduced dramatically
- Scalability: it can be used to act as a load balancer (when using durable queues)

The messaging technology that has been used for the construction of the prototype is based on JMS (Java Messaging Service) specification. There are several implementations of JMS:

- ActiveMQ
- Apache Qpid
- RabbitMQ
- Apollo
- HornetMQ
- ZeroMQ

RabbitMQ was chosen for the implemenetation of the prototype because it is by far the most robust when the number of messages scale up. The decision is supported by an extensive benchmark which is available in Annex IV.

The Message Queue (MQ) supports a number of routing schemes:

- multicast
- broadcast
- unicast

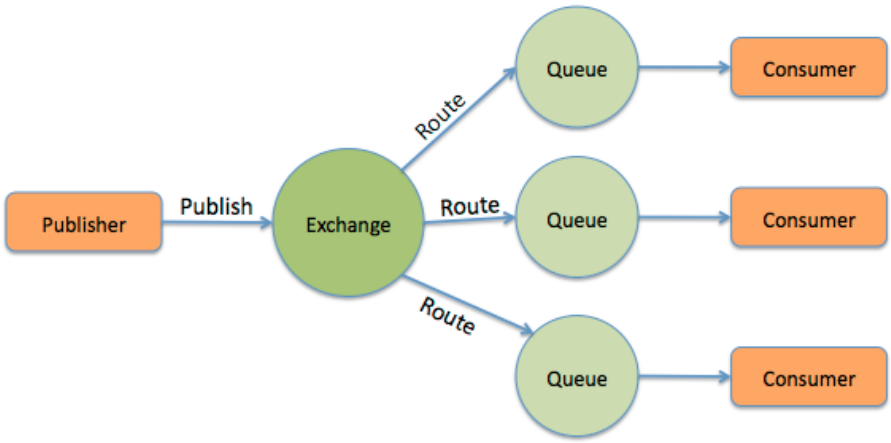
Whereas the message store can be:

a) persistent
b) transient

The core services of LoCloud employ all of the schemes and stores mentioned above in order to provide the full functionality to all the services described.

JMS also defines two main protocols: AMPQ and STOMP. Both protocols are robust but in the current implementation the AMPQ protocol was chosen because it provides better and wider cross-platform interoperability.

AMPQ uses the following message flow:



Purpose	Implementation
Task execution in a load balanced environment	Information is published on a persistent queue using a multicast scheme.
Notifications	Information is published on a transient queue using a broadcast scheme.
Direct service communication	Information is published on a persistent queue using a unicast scheme.

2.4 Metadata specifications

In contrast with the previous version, the new, cloud-based implementation of MoRe can handle multiple metadata schemas and make them available to the various services. These schemas are refered as Intermediate Schemas, and for the LoCloud project they have been defined as:

- CARARE
- EAD
- ESE / Dublin Core
- LIDO

Thus, the notion of the schema identifier is embedded in the content model of the storage infrastructrure (see schema_id in the datastreams Column Familiy in Cassandra configuration). Each service that needs to perform a task associated with the context of a record, needs to be context aware. This means that it must be independent in terms of how to access/modify the content of each record. The infrastructure is and should be schema agnostic. Most services are designed to be able to handle the variety of the metadata schemas.

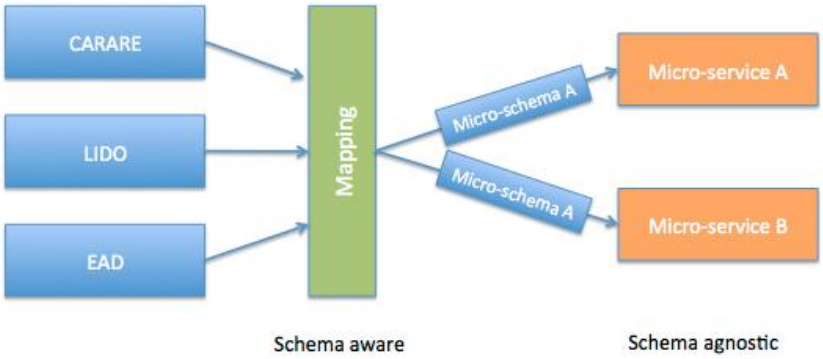
However, this is not the case for the enrichement service, as this needs to inter-operate with other micro-services. This requirement introduces some complexity, which is addressed with the introduction of micro-schemas.

2.4.1 Handling complexity: Micro-schemas

A micro-schema is defined as a small in size metadata schema, which focuses on a specific thematic area and is used to transfer specific information to micro-services. The micro-schema should be able to provide bi-directional mappings from-to its Native schemas. In the LoCloud case, the generic enrichment service, instead of serving the various intermediate schemas to each micro-service, it serves only one micro-schema to each micro-service depending on the domain it operates.

For example, the Historic Place Names and Spatial enrichment micro-services only need access to the Spatial micro-schema. Similarly, the thesauri enrichement micro-service only needs access to the subject micro-schema.

This approach allows for easier development and support of micro-services and more importantly, it allows the infrastructure to expand with more Intermediate shcemas in the future without affecting the internals of the corresponding micro-services.



In the LoCloud MoRe prototype the micro schemas for:

a) thematic information
b) spatial information
c) temporal information

are defined for the CARARE 1.x schema. These microschemas are implemented using the mapping service and a series of XSLT files, which are available in Annex V.

2.5 Registries

In order for the MoRe metadata aggregator to work, support information is required. This information mainly includes objects, content providers, services, and schema related information. All this support related information is implemented within the various LoCloud registries:

2.5.1 Schema registry

The schema registry includes the following information:

- schema XSDs used for validation
- schema XSLT mappings used for transformation
- micro-schema XSLT mappings

2.5.2 Services registry

The services registry includes the following information:

- service description
- service URI

2.5.3 Content providers registry

The content providers registry includes various information about the content providers.

2.5.4 Object registry

The object registry is a list of all objects and their URIs. This is used by the unique identifier service.

2.6 Disaster planning & recovery

An important parameter introduces in the cloud infrastructure is a robust disaster recovery plan. The current version of MoRe is based on the Mopseus (http://www.dcu.gr/) digital library system which includes a proven disaster recovery plan. This important characteristic has been introduced in the prototype implementation using two methodologies:

a) by introducing a highly redundant architecture of Apache Cassandra (the prototype has been setup with a replication factor of 1 but this can easily be increased).
b) by maintaing a PREMIS audit log for all changes within a digital object. This is represented within the auditlog column familiy (see the Cassandra configuration).

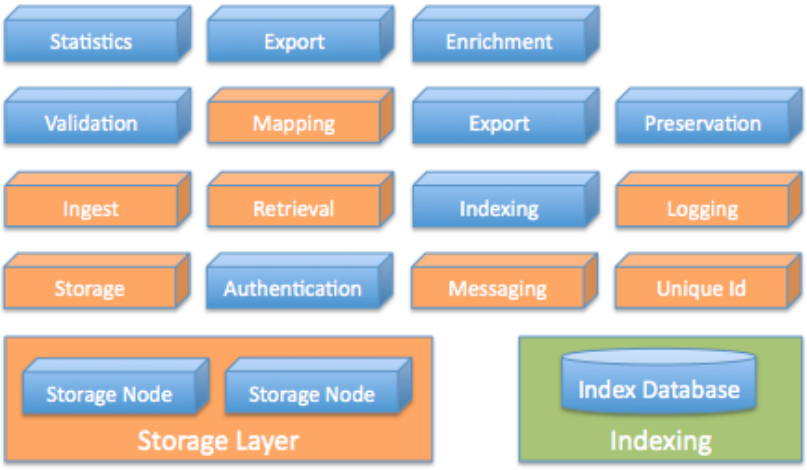
In case of a disaster where only the lower level storage survives, it is possible to:

- re-build the object index (and thus re-enable the unique identifier service)
- re-build the schemas index
- re-index all objects
- re-build the content provider index

Some information such as the name of the content provider can be encoded within the various PREMIS records (hence they are retrievable).

2. Prototype demonstrator services

The MoRe cloud prototype introduces a number of newly implemented services that can be seen in orange in the following figure:



3.1 Unique identifier service

The unique identifier service allows assigning unique identifiers for every object. This is a REST service accessible by:

Relative URI:	/uuid/next
Parameters	
ns	[optional] : the namespace of the cloud storage (used when having multiple cloud storage plugged in).
Returns	
The URI of the newly created object	

3.2 Storage service

The storage service allows communicating with the storage layer (currently only the Cassandra driver has been implemented). The main services that accomplish this are:

Create a new digital object:

Relative URI:	/store/create
Parameters	
object_id	The id of the object
label	A label for the object to be created
provider_id	The provider

Adding a new datastream:

Relative URI:	/store/add

Parameters	
datastream_id	The id of the datastream (e.g. CARARE)
object_id	The id of the object
label	A label for the object to be created
provider_id	The provider
content	The contents of the datastrem in XML
schema_id	The id of the schema (e.g. CARARE)
logmessage	A log message describing the transaction (e.g. ingest src add new ds)

3.3 Messaging service

The messaging service has been built using RabbitMQ. It has been described extensively in previous sections.

3.4 Ingest service

The ingest service ingests a digital object to MoRe. It performs the following tasks:

1) Validation of the datastream
    a. Validates the datastream according to its schema
    b. Validates the provider id information
2) Checks for existence of an object_id.
3) If an object id is not provided, it assigns a new one and creates a new object.
4) It adds the new datastream. If the datastream already exists,a new version of this datastream is created.

Relative URI:	/store/ingest
Parameters	
datastream_id	The id of the datastream (e.g. CARARE)
object_id	The id of the object
label	A label for the object to be created
provider_id	The provider
content	The contents of the datastrem in XML
schema_id	The id of the schema (e.g. CARARE)
logmessage	A log message describing the transaction (e.g. ingest src add new ds)

3.5 Retrieval service

The retrieval service currently retrieves information about a digital object or a datastream.

Retrieve information about an object:

Relative URI:	/store/get
Parameters	
object_id	The id of the object

Returns
The object information along with its list of datastreams in XML

Retrieve the contents of a datastream:

Relative URI:	/store/get
Parameters	
object_id	The id of the object
datastream_id	The id of the datastream
Returns	
The contents of the datastream in XML	

3.6 Mapping service

Tha mapping service executes mappings between schemas

Retrieve information about an object:

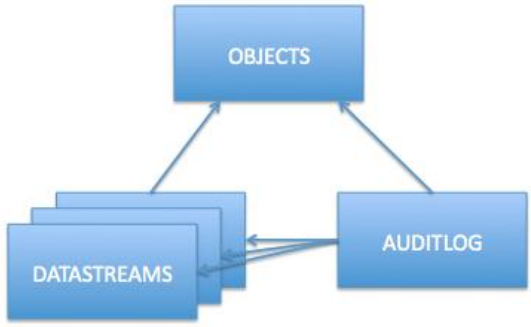
Relative URI:	/map/transform
Parameters	
object_id	The id of the object
datastream_id	The datastream id
source_schema_id	The source schema id
target_schema_id	The target schema id
store_target	A flag (0/1) indicating whether to store the result as a new datastream
Returns	
The transformed datastream in XML	

3.7 Logging service

The logging service has been implemented using the well-known Java Log4j logging mechanism. All services include Log4j and log information both locally and to a centralized service by including two Appenders.

Annex I – Cassandra content model

The Cassandra object model that was used follows the content model below.



This model represents a digital object with the following parameters:

- id (a URI)
- label
- provider id
- creation date

Each digital object comprises of one or more datastreams with the following parameters:

- id (a URI)
- schema id (one of the intermediate schemas)
- label
- content (XML content)
- creation date
- version id
- a logmessage

Each datastream can have multiple versions (distinguished using the schema id and version).

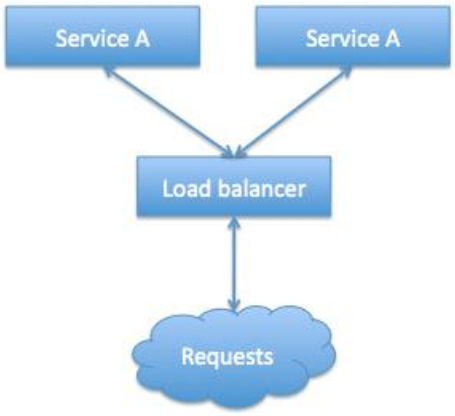
The Auditlog is a PREMIS log describing the history of the object.

The CQL3 commands below describe the details of how to construct the above model in a Cassandra node:

```
CREATE KEYSPACE LOCLOUD WITH REPLICATION = { 'class' : 'SimpleStrategy', 'replication_factor': 1 };

USE LOCLOUD;

CREATE TABLE objects ( object_id varchar, label varchar, provider_id varchar, creation_date timestamp, PRIMARY KEY(object_id, provider_id) );

CREATE TABLE datastreams ( datastream_id varchar, object_id varchar, provider_id varchar, schema_id varchar, label varchar, content TEXT, creation_date timestamp, version timeuuid, logmessage VARCHAR, PRIMARY KEY((object_id, provider_id, datastream_id), version) );

CREATE INDEX object_id ON datastreams(object_id);
CREATE INDEX provider_id ON datastreams(provider_id);
CREATE INDEX datastream_id ON datastreams(datastream_id);

CREATE TABLE auditlog ( auditlog_id timeuuid, object_id varchar, datastream_id varchar, label varchar, log text, creation_date timeuuid, PRIMARY KEY(auditlog_id, object_id, datastream_id) );
```

Annex II – Services load balancing

This section shows the configuration for load balancing web services using the apache mod_proxy. The configuration below (which is part of apache httpd.conf) enables to distribute requests to two internal worker nodes using round robin algorithm.



The mod_proxy load balancer forwards each request to one of the balancer members (e.g. Service A). The algorithm that is used is round robin.

In a case where mod_proxy is used, the balancer is registered as the service (instead of the individual Service A nodes) thus simplifying the setup.

```
<VirtualHost *:80>
    ProxyRequests off

    ServerName n1.dcu.gr

    <Proxy balancer://mycluster>
        # Worker node 1
        BalancerMember http://192.168.10.101:80
        # Worker node 2
        BalancerMember http://192.168.10.102:80

        Order Deny,Allow
        Deny from none
        Allow from all

        # Round robin load balancing.
        ProxySet lbmethod=byrequests

    </Proxy>

    ProxyPass / balancer://mycluster/

</VirtualHost>
```

Annex III – Message Queueing

This RabbitMQ implementation has been setup from its generic linux binary packages. It is launched from the command line in detached mode and the **rabbitmq_management** plugin has been activated. The RabbitMQ server is accessible through its Web based management GUI: http://core.cloud.dcu.gr:15672/#/

RabbitMQ configuration

The configuration includes the usage of queues for inter-service communication. Each service publishes and consumes messages to/from one or many queues. Currently the following queues are used:

- INGEST
- TRANSFORMATION
- LOG

For the current status of the prototype two main configurations are used:

A) Persistent queues for assigning tasks to worker nodes.

Parameter	Value
durability	Durable
auto-delete	No
x-message-ttl	0ms
x-expires	1024000ms
x-max-length	1000000

B) Non-persistent queues for publishing notification messages

Parameter	Value
durability	Transient
auto-delete	No
x-message-ttl	10240ms
x-expires	1024000ms
x-max-length	1000000

Annex IV – JMS implementations benchmark

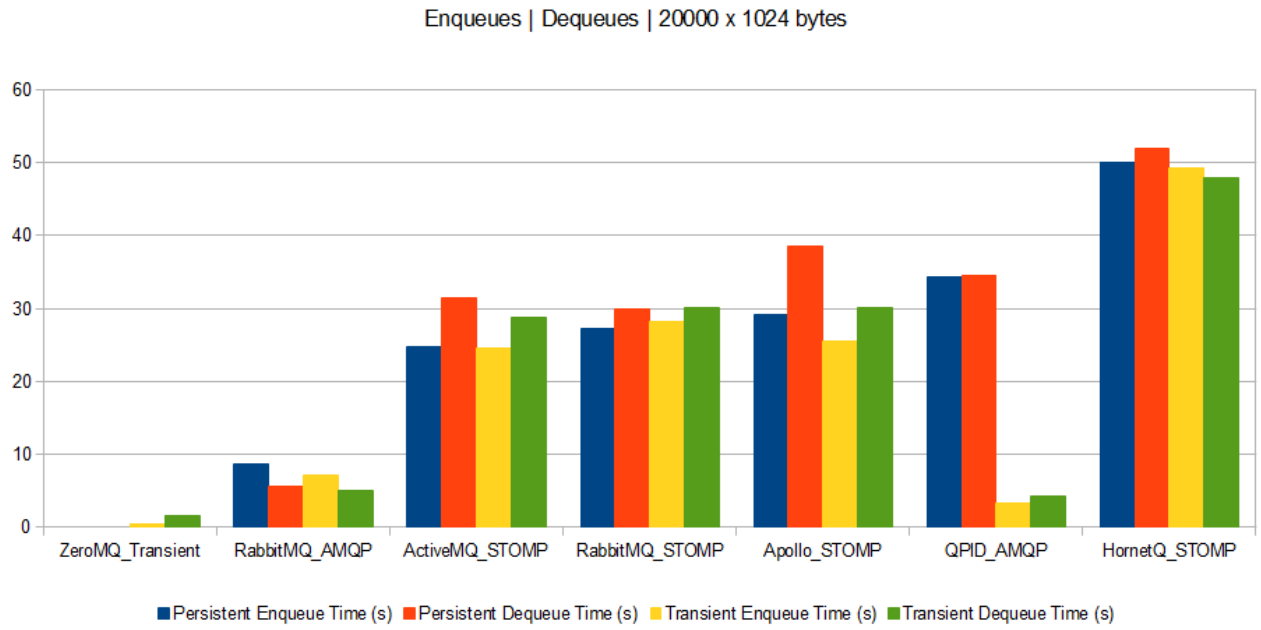
The following benchmark is available from the following URL:

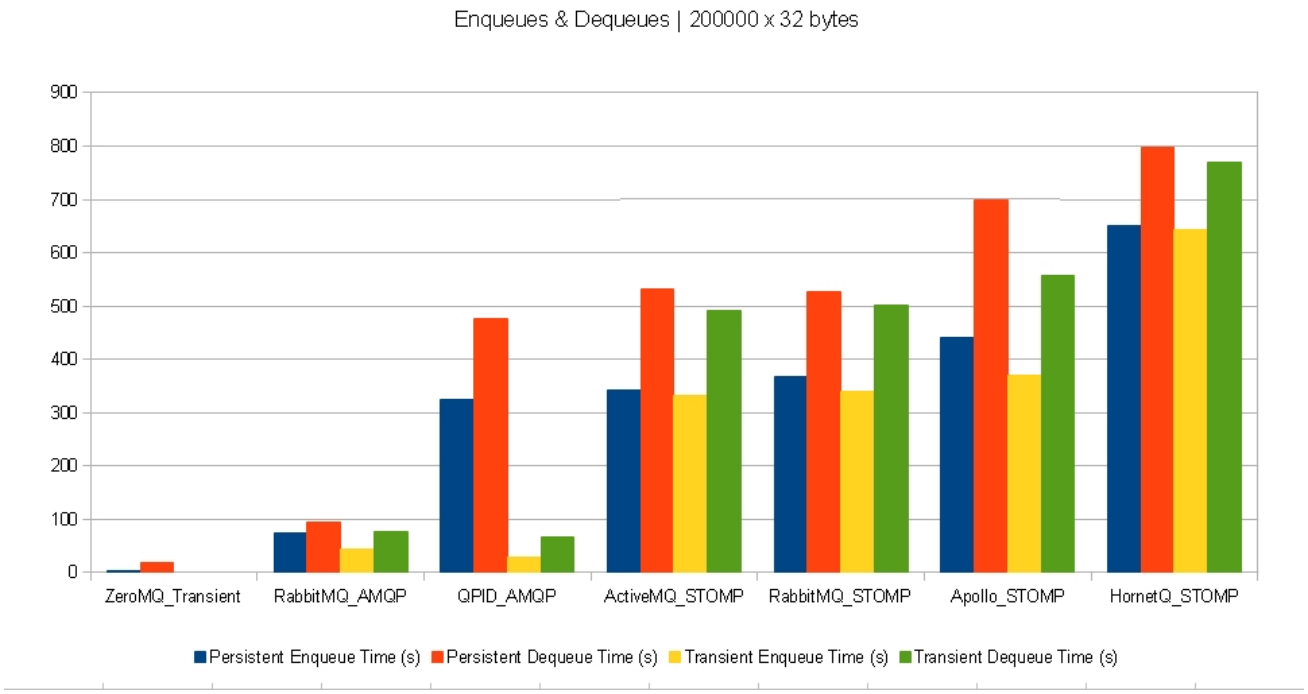
http://rivierarb.fr/presentations/messaging-systems/#1.

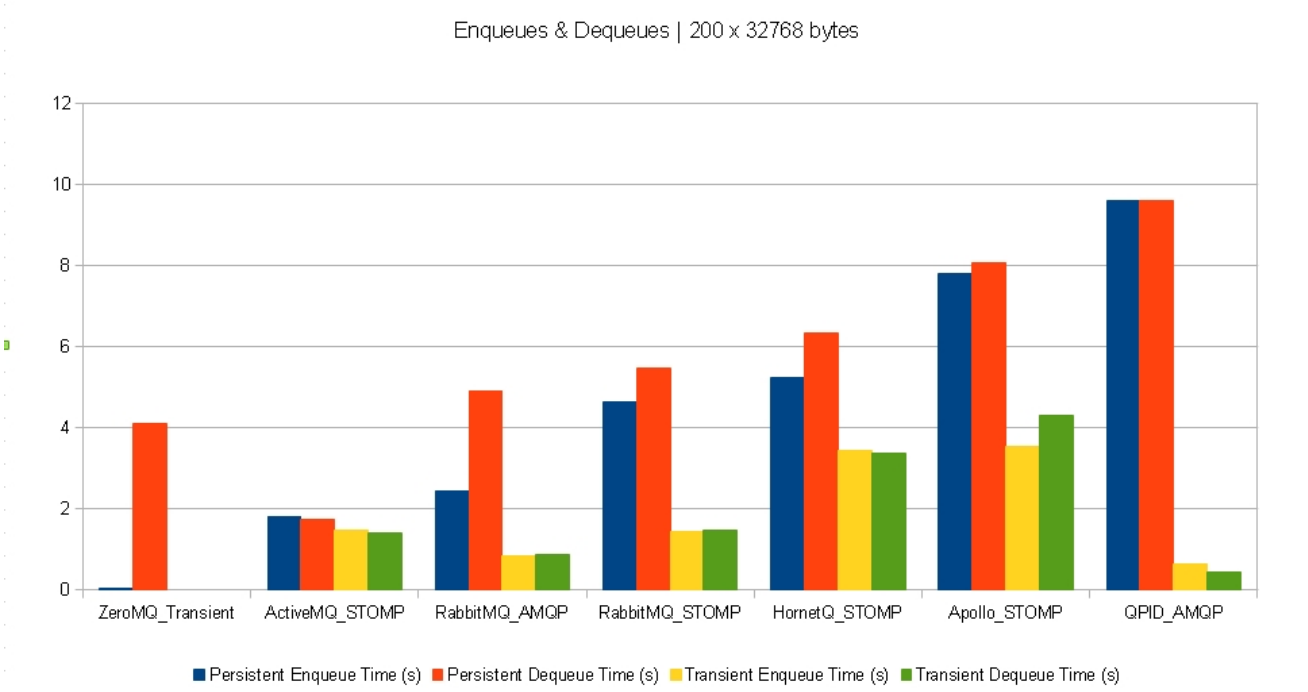
This benchmark probably the most widely accepted and cited. It has been carried out by Adina MIHAILESCU and presents a comparison of different JMS implementations:

- ZeroMQ
- RabbitMQ
- ActiveMQ
- Apollo
- Qpid
- HornetMQ

For the AMPQ and STOMP protocols and for small (32 bytes), medium (1024 bytes) and large (32 Kbytes) message sizes. The results that are shown in the following diagrams show the response times for each case (lower values indicate better performance). It is clear that RabbitMQ outperform all other implementations with the exception of ZeroMQ for small and medium sized messages.




Enqueues & Dequeues | 200000 x 32 bytes
900
800
700
600
500
400
300
200
100
0
ZeroMQ_Transient
RabbitMQ_AMQP
QPID_AMQP
ActiveMQ_STOMP
RabbitMQ_STOMP
Apollo_STOMP
HornetQ_STOMP
Persistent Enqueue Time (s)
Persistent Dequeue Time (s)
Transient Enqueue Time (s)
Transient Dequeue Time (s)


Enqueues & Dequeues | 200 x 32768 bytes
12
10
8
6
4
2
0
ZeroMQ_Transient
ActiveMQ_STOMP
RabbitMQ_AMQP
RabbitMQ_STOMP
HornetQ_STOMP
Apollo_STOMP
QPID_AMQP
Persistent Enqueue Time (s)
Persistent Dequeue Time (s)
Transient Enqueue Time (s)
Transient Dequeue Time (s)

Annex V – Micro-schemas definition for CARARE

The section presents a first attempt at creating micro-schemas for the CARARE 1.x schema for the following three domains:

- subjects
- spatial
- temporal

In the three tables below the respective XSLTs are presented.

Subject:

```
<?xml version="1.0" encoding="utf-8"?>

<xsl:stylesheet version="2.0"
	xmlns:carare="http://www.carare.eu/carareSchema"
	xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
	xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <xsl:strip-space elements="*" />
<xsl:output method="xml" version="1.0" encoding="utf-8" indent="yes"/>

<xsl:variable name="lower">abcdefghijklmnopqrstuvwxyz</xsl:variable>
<xsl:variable name="upper">ABCDEFGHIJKLMNOPQRSTUVWXYZ</xsl:variable>

<xsl:template match="carare:carareWrap">
<carare:carareWrap>
	<carare:carare>

		<!-- HERITAGE ASSET -->
		<xsl:for-each select="carare:carare/carare:heritageAssetIdentification">
			<xsl:element name="carare:heritageAssetIdentification">
				<xsl:element name="carare:references">
					<xsl:for-each select="carare:references/carare:subject">

						<xsl:element name="carare:subject">
							<xsl:if test="@lang > '0'">
								<xsl:attribute name="xml:lang">
									<xsl:value-of
select="translate(@lang,$upper,$lower)" />
								</xsl:attribute>
							</xsl:if>
							<xsl:value-of select="text()" />
						</xsl:element>
					</xsl:for-each>
				</xsl:element>
			</xsl:element>
		</xsl:for-each>

		<!-- DIGITAL RESOURCE -->
		<xsl:for-each select="carare:carare/carare:digitalResource">
			<xsl:element name="carare:digitalResource">
				<xsl:for-each select="carare:subject">
					<xsl:element name="carare:subject">
						<xsl:if test="@lang !=''">
							<xsl:attribute name="xml:lang">
								<xsl:value-of
select="translate(@lang,$upper,$lower)" />
							</xsl:attribute>
						</xsl:if>
```

```
                                        <xsl:value-of select="text()" />
                                    </xsl:element>
                                </xsl:for-each>
                            </xsl:element>
                        </xsl:for-each>

                        <!-- COLLECTION -->
                        <xsl:for-each select="carare:carareWrap/carare:carare/carare:digitalResource">
                            <xsl:element name="carare:digitalResource">
                                <xsl:for-each select="carare:subject">
                                    <xsl:element name="carare:subject">
                                        <xsl:if test="@lang !=''">
                                            <xsl:attribute name="xml:lang">
                                                <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                            </xsl:attribute>
                                        </xsl:if>
                                        <xsl:value-of select="text()" />
                                    </xsl:element>
                                </xsl:for-each>
                            </xsl:element>
                        </xsl:for-each>

            </carare:carare>
</carare:carareWrap>
</xsl:template>

</xsl:stylesheet>
```

Spatial:

```
<?xml version="1.0" encoding="utf-8"?>

<xsl:stylesheet version="2.0"
            xmlns:carare="http://www.carare.eu/carareSchema"
            xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
   <xsl:strip-space elements="*" />
<xsl:output method="xml" version="1.0" encoding="utf-8" indent="yes"/>

<xsl:variable name="lower">abcdefghijklmnopqrstuvwxyz</xsl:variable>
<xsl:variable name="upper">ABCDEFGHIJKLMNOPQRSTUVWXYZ</xsl:variable>

<xsl:template match="carare:carareWrap">
<carare:carareWrap>
            <carare:carare>

                        <!-- COLLECTION INFORMATION -->
                        <xsl:for-each select="carare:carare/carare:collectionInformation">
                            <xsl:element name="carare:collectionInformation">
                                <xsl:for-each select="carare:coverage">
                                    <xsl:element name="carare:coverage">
                                        <xsl:for-each select="carare:spatial">
                                            <xsl:element name="carare:spatial">
                                                <xsl:for-each select="carare:locationSet">
                                                    <xsl:element
name="carare:locationSet">
                                                        <xsl:for-each
```

```
select="carare:namedLocation">
                                                                                <xsl:element
name="carare:namedLocation">
                                                                                    <xsl:if
test="@lang !=''">
        <xsl:attribute name="xml:lang">
        <xsl:value-of select="translate(@lang,$upper,$lower)" />
        </xsl:attribute>
                                                                                    </xsl:if>
                                                                                    <xsl:if
test="@preferred !=''">
        <xsl:attribute name="xml:preferred">
        <xsl:value-of select="text()" />
        </xsl:attribute>
                                                                                    </xsl:if>
                                                                                    <xsl:if
test="@namespace !=''">
        <xsl:attribute name="xml:namespace">
        <xsl:value-of select="text()" />
        </xsl:attribute>
                                                                                    </xsl:if>

                                                                                    <xsl:value-of
select="text()" />

                                                                                </xsl:element>
                                                                            </xsl:for-each>


                                                                            <xsl:for-each
select="carare:address">
                                                                                <xsl:element
name="carare:address">
                                                                                    <xsl:for-each
select="carare:buildingName">
        <xsl:element name="carare:buildingName">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
        <xsl:if test="@authority !=''">
```

```
                <xsl:attribute name="xml:authority">
                        <xsl:value-of select="text()" />
                </xsl:attribute>
        </xsl:if>
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                </xsl:for-
each>
                                                                                <xsl:for-each
select="carare:numberInRoad">
        <xsl:element name="carare:numberInRoad">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
        <xsl:if test="@authority !=''">
                <xsl:attribute name="xml:authority">
                        <xsl:value-of select="text()" />
                </xsl:attribute>
        </xsl:if>
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                </xsl:for-
each>
                                                                                <xsl:for-each
select="carare:roadName">
        <xsl:element name="carare:roadName">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
```

```
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                </xsl:for-
each>
                                                                                <xsl:for-each
select="carare:townOrCity">
        <xsl:element name="carare:townOrCity">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
        <xsl:if test="@authority !=''">
                <xsl:attribute name="xml:authority">
                        <xsl:value-of select="text()" />
                </xsl:attribute>
        </xsl:if>
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                </xsl:for-
each>
                                                                                <xsl:for-each
select="carare:postcodeOrZipcode">
        <xsl:element name="carare:postcodeOrZipcode">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                </xsl:for-
each>
                                                                                <xsl:for-each
select="carare:locality">
```

```
		<xsl:element name="carare:locality">
		<xsl:if test="@lang !=''">
			<xsl:attribute name="xml:lang">
				<xsl:value-of select="translate(@lang,$upper,$lower)" />
			</xsl:attribute>
		</xsl:if>
		<xsl:if test="@authority !=''">
			<xsl:attribute name="xml:authority">
				<xsl:value-of select="text()" />
			</xsl:attribute>
		</xsl:if>
		<xsl:value-of select="text()" />
		</xsl:element>
	</xsl:for-each>
	<xsl:for-each select="carare:adminArea">
		<xsl:element name="carare:adminArea">
		<xsl:if test="@lang !=''">
			<xsl:attribute name="xml:lang">
				<xsl:value-of select="translate(@lang,$upper,$lower)" />
			</xsl:attribute>
		</xsl:if>
		<xsl:if test="@authority !=''">
			<xsl:attribute name="xml:authority">
				<xsl:value-of select="text()" />
			</xsl:attribute>
		</xsl:if>
		<xsl:value-of select="text()" />
		</xsl:element>
	</xsl:for-each>
```

```
                                                                                <xsl:for-each
select="carare:country">
        <xsl:element name="carare:country">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
        <xsl:if test="@authority !=''">
                <xsl:attribute name="xml:authority">
                        <xsl:value-of select="text()" />
                </xsl:attribute>
        </xsl:if>
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                        </xsl:for-
each>
                                                                                </xsl:element>
                                                                        </xsl:for-each>

                                                                        <xsl:for-each
select="carare:geopoliticalArea">
                                                                                <xsl:element
name="carare:geopoliticalArea">
                                                                                <xsl:for-each
select="carare:geopoliticalAreaName">
        <xsl:element name="carare:geopoliticalAreaName">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                        </xsl:for-
each>
                                                                                <xsl:for-each
```

```
select="carare:geopoliticalAreaType">

        <xsl:element name="carare:geopoliticalAreaType">

        <xsl:if test="@lang !=''">

                <xsl:attribute name="xml:lang">

                        <xsl:value-of select="translate(@lang,$upper,$lower)" />

                </xsl:attribute>

        </xsl:if>

        <xsl:if test="@namespace !="">

                <xsl:attribute name="xml:namespace">

                        <xsl:value-of select="text()" />

                </xsl:attribute>

        </xsl:if>

        <xsl:value-of select="text()" />

        </xsl:element>
                                                                        </xsl:for-
each>
                                                                </xsl:element>
                                                        </xsl:for-each>


                                                        <xsl:for-each
select="carare:cadastralReference">
                                                                <xsl:element
name="carare:cadastralReference">
                                                                        <xsl:if
test="@ref !=''">

        <xsl:attribute name="xml:ref">

        <xsl:value-of select="text()" />

        </xsl:attribute>
                                                                        </xsl:if>
                                                                        <xsl:value-of
select="text()" />
                                                                </xsl:element>
                                                        </xsl:for-each>


                                                        <xsl:for-each
select="carare:historicalName">
                                                                <xsl:element
name="carare:historicalName">
                                                                        <xsl:if
test="@lang !=''">

        <xsl:attribute name="xml:lang">
```

```
        <xsl:value-of select="translate(@lang,$upper,$lower)" />
        </xsl:attribute>
                                                                                    </xsl:if>
                                                                                    <xsl:if
test="@preferred !=''">
        <xsl:attribute name="xml:preferred">
        <xsl:value-of select="text()" />
        </xsl:attribute>
                                                                                    </xsl:if>
                                                                                    <xsl:value-of
select="text()" />
                                                                                        </xsl:element>
                                                                                    </xsl:for-each>

                                                                                </xsl:element>

                                                                            </xsl:for-each>

                                                                            <xsl:for-each
select="carare:spatialReferenceSystem">
                                                                                <xsl:element
name="carare:spatialReferenceSystem">
                                                                                    <xsl:if test="@lang !=''">
                                                                                        <xsl:attribute
name="xml:lang">
                                                                                            <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                                                        </xsl:attribute>
                                                                                    </xsl:if>
                                                                                    <xsl:value-of  select="text()"  />

                                                                                </xsl:element>
                                                                            </xsl:for-each>

                                                                            <xsl:for-each
select="carare:cartographicReference">
                                                                                <xsl:element
name="carare:cartographicReference">
                                                                                    <xsl:for-each
select="carare:spatialFeatureType">
                                                                                        <xsl:element
name="carare:spatialFeatureType">
                                                                                            <xsl:if
test="@lang !=''">
        <xsl:attribute name="xml:lang">
        <xsl:value-of select="translate(@lang,$upper,$lower)" />
        </xsl:attribute>
                                                                                            </xsl:if>
                                                                                            <xsl:value-of
select="text()" />
```

```
                                                                    </xsl:element>
                                                                </xsl:for-each>
                                                                <xsl:for-each
select="carare:coordinates">
                                                                    <xsl:element
name="carare:coordinates">
                                                                        <xsl:element
name="carare:x">
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                        <xsl:element
name="carare:y">
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                        <xsl:element
name="carare:z">
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                    </xsl:element>
                                                                </xsl:for-each>
                                                            </xsl:element>
                                                        </xsl:for-each>


                                                        <xsl:for-each select="carare:geometry">
                                                            <xsl:element name="carare:geometry">
                                                                <xsl:for-each
select="carare:boundingBox">
                                                                    <xsl:element
name="carare:boundingBox">
                                                                        <xsl:element
name="carare:minX">
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                        <xsl:element
name="carare:maxX">
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                        <xsl:element
name="carare:minY">
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                        <xsl:element
name="carare:maxX">
            <xsl:value-of select="text()" />
```

```
        </xsl:element>
                                                                                    </xsl:element>
                                                                                </xsl:for-each>
                                                                                <xsl:for-each
select="carare:quickpoint">
                                                                                    <xsl:element
name="carare:quickpoint">
                                                                                        <xsl:element
name="carare:x">
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                        <xsl:element
name="carare:y">
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                    </xsl:element>
                                                                                </xsl:for-each>
                                                                                <xsl:for-each
select="carare:entity">
                                                                                    <xsl:element
name="carare:entity">
                                                                                        <xsl:if
test="@lang !=''">
        <xsl:attribute name="xml:lang">
        <xsl:value-of select="translate(@lang,$upper,$lower)" />
        </xsl:attribute>
                                                                                        </xsl:if>

                                                                                        <xsl:value-of
select="text()" />
                                                                                    </xsl:element>
                                                                                </xsl:for-each>
                                                                                <xsl:for-each
select="carare:storedPrecision">
                                                                                    <xsl:element
name="carare:storedPrecision">
                                                                                        <xsl:if
test="@lang !=''">
        <xsl:attribute name="xml:lang">
        <xsl:value-of select="translate(@lang,$upper,$lower)" />
        </xsl:attribute>
                                                                                        </xsl:if>

                                                                                        <xsl:value-of
select="text()" />
                                                                                    </xsl:element>
                                                                                </xsl:for-each>
```

```
                                                                    <xsl:for-each
select="carare:height">
                                                                        <xsl:element
name="carare:height">
                                                                            <xsl:if
test="@lang !=''">
            <xsl:attribute name="xml:lang">
            <xsl:value-of select="translate(@lang,$upper,$lower)" />
            </xsl:attribute>
                                                                            </xsl:if>

                                                                            <xsl:value-of
select="text()" />
                                                                        </xsl:element>
                                                                    </xsl:for-each>

                                                                    <xsl:for-each
select="carare:area">
                                                                        <xsl:element
name="carare:area">
                                                                            <xsl:if
test="@lang !=''">
            <xsl:attribute name="xml:lang">
            <xsl:value-of select="translate(@lang,$upper,$lower)" />
            </xsl:attribute>
                                                                            </xsl:if>

                                                                            <xsl:value-of
select="text()" />
                                                                        </xsl:element>
                                                                    </xsl:for-each>

                                                                </xsl:element>
                                                            </xsl:for-each>
                                                            <xsl:for-each select="carare:representations">
                                                                <xsl:element
name="carare:representations">
                                                                    <xsl:if test="@lang !=''">
                                                                        <xsl:attribute
name="xml:lang">
                                                                            <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                                        </xsl:attribute>
                                                                    </xsl:if>

                                                                    <xsl:value-of select="text()" />
                                                                </xsl:element>
                                                            </xsl:for-each>
                                                        </xsl:element>
                                                    </xsl:for-each>
                                                </xsl:element>
                                            </xsl:for-each>
```

```
                        </xsl:element>
                    </xsl:for-each>


                    <!-- HERITAGE ASSET -->
                    <xsl:for-each select="carare:carare/carare:heritageAssetIdentification">
                        <xsl:element name="carare:heritageAssetIdentification">
                            <xsl:for-each select="carare:spatial">
                                <xsl:element name="carare:spatial">
                                    <xsl:for-each select="carare:locationSet">
                                        <xsl:element
name="carare:locationSet">
                                            <xsl:for-each
select="carare:namedLocation">
                                                <xsl:element
name="carare:namedLocation">
                                                    <xsl:if
test="@lang !=''">
        <xsl:attribute name="xml:lang">
        <xsl:value-of select="translate(@lang,$upper,$lower)" />
        </xsl:attribute>
                                                    </xsl:if>
                                                    <xsl:if
test="@preferred !=''">
        <xsl:attribute name="xml:preferred">
        <xsl:value-of select="text()" />
        </xsl:attribute>
                                                    </xsl:if>
                                                    <xsl:if
test="@namespace !=''">
        <xsl:attribute name="xml:namespace">
        <xsl:value-of select="text()" />
        </xsl:attribute>
                                                    </xsl:if>

                                                    <xsl:value-of
select="text()" />

                                                </xsl:element>
                                            </xsl:for-each>


                                            <xsl:for-each
select="carare:address">
                                                <xsl:element
name="carare:address">
                                                    <xsl:for-each
select="carare:buildingName">
```

```
        <xsl:element name="carare:buildingName">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
        <xsl:if test="@authority !=''">
                <xsl:attribute name="xml:authority">
                        <xsl:value-of select="text()" />
                </xsl:attribute>
        </xsl:if>
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                </xsl:for-
each>
                                                                                <xsl:for-each
select="carare:numberInRoad">
        <xsl:element name="carare:numberInRoad">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
        <xsl:if test="@authority !=''">
                <xsl:attribute name="xml:authority">
                        <xsl:value-of select="text()" />
                </xsl:attribute>
        </xsl:if>
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                </xsl:for-
each>
                                                                                <xsl:for-each
```

```
select="carare:roadName">
        <xsl:element name="carare:roadName">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                </xsl:for-
each>
                                                                                <xsl:for-each
select="carare:townOrCity">
        <xsl:element name="carare:townOrCity">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
        <xsl:if test="@authority !=''">
                <xsl:attribute name="xml:authority">
                        <xsl:value-of select="text()" />
                </xsl:attribute>
        </xsl:if>
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                </xsl:for-
each>
                                                                                <xsl:for-each
select="carare:postcodeOrZipcode">
        <xsl:element name="carare:postcodeOrZipcode">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
```

```
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                    </xsl:attribute>
            </xsl:if>
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                                    </xsl:for-
each>
                                                                                    <xsl:for-each
select="carare:locality">
            <xsl:element name="carare:locality">
            <xsl:if test="@lang !=''">
                    <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                    </xsl:attribute>
            </xsl:if>
            <xsl:if test="@authority !=''">
                    <xsl:attribute name="xml:authority">
                        <xsl:value-of select="text()" />
                    </xsl:attribute>
            </xsl:if>
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                                    </xsl:for-
each>
                                                                                    <xsl:for-each
select="carare:adminArea">
            <xsl:element name="carare:adminArea">
            <xsl:if test="@lang !=''">
                    <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                    </xsl:attribute>
            </xsl:if>
            <xsl:if test="@authority !=''">
```

```
                <xsl:attribute name="xml:authority">
                        <xsl:value-of select="text()" />
                </xsl:attribute>
        </xsl:if>
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                </xsl:for-
each>
                                                                                <xsl:for-each
select="carare:country">
        <xsl:element name="carare:country">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
        <xsl:if test="@authority !=''">
                <xsl:attribute name="xml:authority">
                        <xsl:value-of select="text()" />
                </xsl:attribute>
        </xsl:if>
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                </xsl:for-
each>
                                                                        </xsl:element>
                                                                </xsl:for-each>

                                                                <xsl:for-each
select="carare:geopoliticalArea">
                                                                        <xsl:element
name="carare:geopoliticalArea">
                                                                                <xsl:for-each
select="carare:geopoliticalAreaName">
        <xsl:element name="carare:geopoliticalAreaName">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
```

```
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                    </xsl:attribute>
            </xsl:if>
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                                    </xsl:for-
each>
                                                                                    <xsl:for-each
select="carare:geopoliticalAreaType">
            <xsl:element name="carare:geopoliticalAreaType">
            <xsl:if test="@lang !=''">
                    <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                    </xsl:attribute>
            </xsl:if>
            <xsl:if test="@namespace !=''">
                    <xsl:attribute name="xml:namespace">
                        <xsl:value-of select="text()" />
                    </xsl:attribute>
            </xsl:if>
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                                    </xsl:for-
each>
                                                                                </xsl:element>
                                                                            </xsl:for-each>

                                                                            <xsl:for-each
select="carare:cadastralReference">
                                                                                <xsl:element
name="carare:cadastralReference">
                                                                                    <xsl:if
test="@ref !=''">
            <xsl:attribute name="xml:ref">
            <xsl:value-of select="text()" />
            </xsl:attribute>
                                                                                    </xsl:if>
```

```
                                                                                <xsl:value-of
select="text()" />
                                                                        </xsl:element>
                                                                </xsl:for-each>

                                                                <xsl:for-each
select="carare:historicalName">
                                                                        <xsl:element
name="carare:historicalName">
                                                                                <xsl:if
test="@lang !=''">

        <xsl:attribute name="xml:lang">

        <xsl:value-of select="translate(@lang,$upper,$lower)" />

        </xsl:attribute>
                                                                                </xsl:if>
                                                                                <xsl:if
test="@preferred !=''">

        <xsl:attribute name="xml:preferred">

        <xsl:value-of select="text()" />

        </xsl:attribute>
                                                                                </xsl:if>
                                                                                <xsl:value-of
select="text()" />
                                                                        </xsl:element>
                                                                </xsl:for-each>

                                                        </xsl:element>

                                                </xsl:for-each>

                                                <xsl:for-each
select="carare:spatialReferenceSystem">
                                                        <xsl:element
name="carare:spatialReferenceSystem">
                                                                <xsl:if test="@lang !=''">
                                                                        <xsl:attribute
name="xml:lang">
                                                                                <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                                        </xsl:attribute>
                                                                </xsl:if>
                                                                <xsl:value-of  select="text()"  />

                                                        </xsl:element>
                                                </xsl:for-each>

                                                <xsl:for-each
select="carare:cartographicReference">
                                                        <xsl:element
name="carare:cartographicReference">
                                                                <xsl:for-each
select="carare:spatialFeatureType">
```

```
                                                                    <xsl:element
name="carare:spatialFeatureType">
                                                                        <xsl:if
test="@lang !=''">
            <xsl:attribute name="xml:lang">
            <xsl:value-of select="translate(@lang,$upper,$lower)" />
            </xsl:attribute>
                                                                        </xsl:if>
                                                                        <xsl:value-of
select="text()" />
                                                                    </xsl:element>
                                                                </xsl:for-each>
                                                                <xsl:for-each
select="carare:coordinates">
                                                                    <xsl:element
name="carare:coordinates">
                                                                        <xsl:element
name="carare:x">
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                        <xsl:element
name="carare:y">
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                        <xsl:element
name="carare:z">
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                    </xsl:element>
                                                                </xsl:for-each>
                                                            </xsl:element>
                                                        </xsl:for-each>


                                                        <xsl:for-each select="carare:geometry">
                                                            <xsl:element name="carare:geometry">
                                                                <xsl:for-each
select="carare:boundingBox">
                                                                    <xsl:element
name="carare:boundingBox">
                                                                        <xsl:element
name="carare:minX">
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                        <xsl:element
name="carare:maxX">
```

```
            <xsl:value-of select="text()" />

            </xsl:element>
                                                                                    <xsl:element
name="carare:minY">

            <xsl:value-of select="text()" />

            </xsl:element>
                                                                                    <xsl:element
name="carare:maxX">

            <xsl:value-of select="text()" />

            </xsl:element>
                                                                                        </xsl:element>
                                                                                </xsl:for-each>
                                                                                <xsl:for-each
select="carare:quickpoint">
                                                                                        <xsl:element
name="carare:quickpoint">
                                                                                            <xsl:element
name="carare:x">

            <xsl:value-of select="text()" />

            </xsl:element>
                                                                                            <xsl:element
name="carare:y">

            <xsl:value-of select="text()" />

            </xsl:element>
                                                                                        </xsl:element>
                                                                                </xsl:for-each>
                                                                                <xsl:for-each
select="carare:entity">
                                                                                        <xsl:element
name="carare:entity">
                                                                                            <xsl:if
test="@lang !=''">

            <xsl:attribute name="xml:lang">

            <xsl:value-of select="translate(@lang,$upper,$lower)" />

            </xsl:attribute>
                                                                                            </xsl:if>

                                                                                            <xsl:value-of
select="text()" />
                                                                                        </xsl:element>
                                                                                </xsl:for-each>
                                                                                <xsl:for-each
select="carare:storedPrecision">
                                                                                        <xsl:element
name="carare:storedPrecision">
                                                                                            <xsl:if
```

```
test="@lang !=''">

            <xsl:attribute name="xml:lang">

            <xsl:value-of select="translate(@lang,$upper,$lower)" />

            </xsl:attribute>
                                                                                                    </xsl:if>

                                                                                                    <xsl:value-of
select="text()" />
                                                                                            </xsl:element>
                                                                                    </xsl:for-each>
                                                                                    <xsl:for-each
select="carare:height">
                                                                                            <xsl:element
name="carare:height">
                                                                                                    <xsl:if
test="@lang !=''">

            <xsl:attribute name="xml:lang">

            <xsl:value-of select="translate(@lang,$upper,$lower)" />

            </xsl:attribute>
                                                                                                    </xsl:if>

                                                                                                    <xsl:value-of
select="text()" />
                                                                                            </xsl:element>
                                                                                    </xsl:for-each>


                                                                                    <xsl:for-each
select="carare:area">
                                                                                            <xsl:element
name="carare:area">
                                                                                                    <xsl:if
test="@lang !=''">

            <xsl:attribute name="xml:lang">

            <xsl:value-of select="translate(@lang,$upper,$lower)" />

            </xsl:attribute>
                                                                                                    </xsl:if>

                                                                                                    <xsl:value-of
select="text()" />
                                                                                            </xsl:element>
                                                                                    </xsl:for-each>

                                                                            </xsl:element>
                                                                    </xsl:for-each>
                                                                    <xsl:for-each select="carare:representations">
                                                                            <xsl:element
name="carare:representations">
                                                                                    <xsl:if test="@lang !=''">
                                                                                            <xsl:attribute
```

```
name="xml:lang">
                                                                                                    <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                                                                </xsl:attribute>
                                                                                            </xsl:if>

                                                                                            <xsl:value-of select="text()" />
                                                                                        </xsl:element>
                                                                                    </xsl:for-each>
                                                                                </xsl:element>
                                                                            </xsl:for-each>
                                                                        </xsl:element>
                                                                    </xsl:for-each>

                                                                    <!-- DIGITAL RESOURCE -->
                                                                    <xsl:for-each select="carare:carare/carare:digitalResource">
                                                                        <xsl:element name="carare:digitalResource">
                                                                            <xsl:for-each select="carare:spatial">
                                                                                <xsl:element name="carare:spatial">
                                                                                    <xsl:for-each select="carare:locationSet">
                                                                                        <xsl:element
name="carare:locationSet">
                                                                                            <xsl:for-each
select="carare:namedLocation">
                                                                                                <xsl:element
name="carare:namedLocation">
                                                                                                    <xsl:if
test="@lang !=''">

                                                                                                    <xsl:attribute name="xml:lang">

                                                                                                    <xsl:value-of select="translate(@lang,$upper,$lower)" />

                                                                                                    </xsl:attribute>
                                                                                                    </xsl:if>
                                                                                                    <xsl:if
test="@preferred !=''">

                                                                                                    <xsl:attribute name="xml:preferred">

                                                                                                    <xsl:value-of select="text()" />

                                                                                                    </xsl:attribute>
                                                                                                    </xsl:if>
                                                                                                    <xsl:if
test="@namespace !=''">

                                                                                                    <xsl:attribute name="xml:namespace">

                                                                                                    <xsl:value-of select="text()" />

                                                                                                    </xsl:attribute>
                                                                                                    </xsl:if>


                                                                                                    <xsl:value-of
select="text()" />

                                                                                                </xsl:element>
```

```
                                                                    </xsl:for-each>

                                                                    <xsl:for-each
select="carare:address">
                                                                        <xsl:element
name="carare:address">
                                                                            <xsl:for-each
select="carare:buildingName">
        <xsl:element name="carare:buildingName">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
        <xsl:if test="@authority !=''">
                <xsl:attribute name="xml:authority">
                        <xsl:value-of select="text()" />
                </xsl:attribute>
        </xsl:if>
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                            </xsl:for-
each>
                                                                            <xsl:for-each
select="carare:numberInRoad">
        <xsl:element name="carare:numberInRoad">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
        <xsl:if test="@authority !=''">
                <xsl:attribute name="xml:authority">
                        <xsl:value-of select="text()" />
```

```
                </xsl:attribute>
        </xsl:if>
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                </xsl:for-
each>
                                                                                <xsl:for-each
select="carare:roadName">
        <xsl:element name="carare:roadName">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                </xsl:for-
each>
                                                                                <xsl:for-each
select="carare:townOrCity">
        <xsl:element name="carare:townOrCity">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
        <xsl:if test="@authority !=''">
                <xsl:attribute name="xml:authority">
                        <xsl:value-of select="text()" />
                </xsl:attribute>
        </xsl:if>
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                </xsl:for-
```

```
each>
                                                                                <xsl:for-each
select="carare:postcodeOrZipcode">
        <xsl:element name="carare:postcodeOrZipcode">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                </xsl:for-
each>
                                                                                <xsl:for-each
select="carare:locality">
        <xsl:element name="carare:locality">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
        <xsl:if test="@authority !=''">
                <xsl:attribute name="xml:authority">
                        <xsl:value-of select="text()" />
                </xsl:attribute>
        </xsl:if>
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                </xsl:for-
each>
                                                                                <xsl:for-each
select="carare:adminArea">
        <xsl:element name="carare:adminArea">
        <xsl:if test="@lang !=''">
```

```
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
        <xsl:if test="@authority !=''">
                <xsl:attribute name="xml:authority">
                        <xsl:value-of select="text()" />
                </xsl:attribute>
        </xsl:if>
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                </xsl:for-
each>
                                                                                <xsl:for-each
select="carare:country">
        <xsl:element name="carare:country">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
        <xsl:if test="@authority !=''">
                <xsl:attribute name="xml:authority">
                        <xsl:value-of select="text()" />
                </xsl:attribute>
        </xsl:if>
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                </xsl:for-
each>
                                                                        </xsl:element>
                                                                </xsl:for-each>

                                                                <xsl:for-each
select="carare:geopoliticalArea">
```

```
                                                                    <xsl:element
name="carare:geopoliticalArea">
                                                                        <xsl:for-each
select="carare:geopoliticalAreaName">
        <xsl:element name="carare:geopoliticalAreaName">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                        </xsl:for-
each>
                                                                        <xsl:for-each
select="carare:geopoliticalAreaType">
        <xsl:element name="carare:geopoliticalAreaType">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
        <xsl:if test="@namespace !=''">
                <xsl:attribute name="xml:namespace">
                        <xsl:value-of select="text()" />
                </xsl:attribute>
        </xsl:if>
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                        </xsl:for-
each>
                                                                    </xsl:element>
                                                                </xsl:for-each>

                                                                <xsl:for-each
select="carare:cadastralReference">
                                                                    <xsl:element
```

```
name="carare:cadastralReference">
                                                                                <xsl:if
test="@ref !=''">
        <xsl:attribute name="xml:ref">
        <xsl:value-of select="text()" />
        </xsl:attribute>
                                                                                </xsl:if>
                                                                                <xsl:value-of
select="text()" />
                                                                        </xsl:element>
                                                                </xsl:for-each>

                                                                <xsl:for-each
select="carare:historicalName">
                                                                        <xsl:element
name="carare:historicalName">
                                                                                <xsl:if
test="@lang !=''">
        <xsl:attribute name="xml:lang">
        <xsl:value-of select="translate(@lang,$upper,$lower)" />
        </xsl:attribute>
                                                                                </xsl:if>
                                                                                <xsl:if
test="@preferred !=''">
        <xsl:attribute name="xml:preferred">
        <xsl:value-of select="text()" />
        </xsl:attribute>
                                                                                </xsl:if>
                                                                                <xsl:value-of
select="text()" />
                                                                        </xsl:element>
                                                                </xsl:for-each>

                                                        </xsl:element>

                                                </xsl:for-each>

                                                <xsl:for-each
select="carare:spatialReferenceSystem">
                                                        <xsl:element
name="carare:spatialReferenceSystem">
                                                                <xsl:if test="@lang !=''">
                                                                        <xsl:attribute
name="xml:lang">
                                                                                <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                                        </xsl:attribute>
                                                                </xsl:if>
                                                                <xsl:value-of  select="text()"  />
```

```
                                                    </xsl:element>
                                                </xsl:for-each>

                                                <xsl:for-each
select="carare:cartographicReference">
                                                    <xsl:element
name="carare:cartographicReference">
                                                        <xsl:for-each
select="carare:spatialFeatureType">
                                                            <xsl:element
name="carare:spatialFeatureType">
                                                                <xsl:if
test="@lang !=''">
            <xsl:attribute name="xml:lang">
            <xsl:value-of select="translate(@lang,$upper,$lower)" />
            </xsl:attribute>
                                                                </xsl:if>
                                                                <xsl:value-of
select="text()" />
                                                            </xsl:element>
                                                        </xsl:for-each>
                                                        <xsl:for-each
select="carare:coordinates">
                                                            <xsl:element
name="carare:coordinates">
                                                                <xsl:element
name="carare:x">
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                <xsl:element
name="carare:y">
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                <xsl:element
name="carare:z">
            <xsl:value-of select="text()" />
            </xsl:element>
                                                            </xsl:element>
                                                        </xsl:for-each>
                                                    </xsl:element>
                                                </xsl:for-each>


                                                <xsl:for-each select="carare:geometry">
                                                    <xsl:element name="carare:geometry">
                                                        <xsl:for-each
select="carare:boundingBox">
                                                            <xsl:element
```

```
name="carare:boundingBox">
                                                                                <xsl:element
name="carare:minX">
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                <xsl:element
name="carare:maxX">
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                <xsl:element
name="carare:minY">
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                <xsl:element
name="carare:maxX">
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                        </xsl:element>
                                                                        </xsl:for-each>
                                                                        <xsl:for-each
select="carare:quickpoint">
                                                                                <xsl:element
name="carare:quickpoint">
                                                                                <xsl:element
name="carare:x">
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                <xsl:element
name="carare:y">
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                </xsl:element>
                                                                        </xsl:for-each>
                                                                        <xsl:for-each
select="carare:entity">
                                                                                <xsl:element
name="carare:entity">
                                                                                <xsl:if
test="@lang !=''">
        <xsl:attribute name="xml:lang">
        <xsl:value-of select="translate(@lang,$upper,$lower)" />
        </xsl:attribute>
                                                                                </xsl:if>
```

```
                                                                                    <xsl:value-of
select="text()" />
                                                                                </xsl:element>
                                                                            </xsl:for-each>
                                                                            <xsl:for-each
select="carare:storedPrecision">
                                                                                <xsl:element
name="carare:storedPrecision">
                                                                                    <xsl:if
test="@lang !=''">
            <xsl:attribute name="xml:lang">
            <xsl:value-of select="translate(@lang,$upper,$lower)" />
            </xsl:attribute>
                                                                                    </xsl:if>
                                                                                    <xsl:value-of
select="text()" />
                                                                                </xsl:element>
                                                                            </xsl:for-each>
                                                                            <xsl:for-each
select="carare:height">
                                                                                <xsl:element
name="carare:height">
                                                                                    <xsl:if
test="@lang !=''">
            <xsl:attribute name="xml:lang">
            <xsl:value-of select="translate(@lang,$upper,$lower)" />
            </xsl:attribute>
                                                                                    </xsl:if>
                                                                                    <xsl:value-of
select="text()" />
                                                                                </xsl:element>
                                                                            </xsl:for-each>

                                                                            <xsl:for-each
select="carare:area">
                                                                                <xsl:element
name="carare:area">
                                                                                    <xsl:if
test="@lang !=''">
            <xsl:attribute name="xml:lang">
            <xsl:value-of select="translate(@lang,$upper,$lower)" />
            </xsl:attribute>
                                                                                    </xsl:if>
                                                                                    <xsl:value-of
select="text()" />
```

```
                                                                                    </xsl:element>
                                                                                </xsl:for-each>

                                                                            </xsl:element>
                                                                        </xsl:for-each>
                                                                        <xsl:for-each select="carare:representations">
                                                                            <xsl:element
name="carare:representations">
                                                                                <xsl:if test="@lang !=''">
                                                                                    <xsl:attribute
name="xml:lang">
                                                                                        <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                                                    </xsl:attribute>
                                                                                </xsl:if>

                                                                                <xsl:value-of select="text()" />
                                                                            </xsl:element>
                                                                        </xsl:for-each>
                                                                    </xsl:element>
                                                                </xsl:for-each>
                                                            </xsl:element>
                                                        </xsl:for-each>

                                                        <!-- ACTIVITY -->
                                                        <xsl:for-each select="carare:carare/carare:activity">
                                                            <xsl:element name="carare:activity">
                                                                <xsl:for-each select="carare:spatial">
                                                                    <xsl:element name="carare:spatial">
                                                                        <xsl:for-each select="carare:locationSet">
                                                                            <xsl:element
name="carare:locationSet">
                                                                                <xsl:for-each
select="carare:namedLocation">
                                                                                    <xsl:element
name="carare:namedLocation">
                                                                                        <xsl:if
test="@lang !=''">

                                                                                        <xsl:attribute name="xml:lang">

                                                                                        <xsl:value-of select="translate(@lang,$upper,$lower)" />

                                                                                        </xsl:attribute>
                                                                                        </xsl:if>
                                                                                        <xsl:if
test="@preferred !=''">

                                                                                        <xsl:attribute name="xml:preferred">

                                                                                        <xsl:value-of select="text()" />

                                                                                        </xsl:attribute>
                                                                                        </xsl:if>
                                                                                        <xsl:if
test="@namespace !=''">

                                                                                        <xsl:attribute name="xml:namespace">
```

```
        <xsl:value-of select="text()" />
        </xsl:attribute>
                                                                                </xsl:if>
                                                                                <xsl:value-of
select="text()" />
                                                                        </xsl:element>
                                                                </xsl:for-each>

                                                                <xsl:for-each
select="carare:address">
                                                                        <xsl:element
name="carare:address">
                                                                                <xsl:for-each
select="carare:buildingName">
        <xsl:element name="carare:buildingName">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
        <xsl:if test="@authority !=''">
                <xsl:attribute name="xml:authority">
                        <xsl:value-of select="text()" />
                </xsl:attribute>
        </xsl:if>
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                </xsl:for-
each>
                                                                                <xsl:for-each
select="carare:numberInRoad">
        <xsl:element name="carare:numberInRoad">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
```

```
                </xsl:attribute>
        </xsl:if>
        <xsl:if test="@authority !=''">
                <xsl:attribute name="xml:authority">
                        <xsl:value-of select="text()" />
                </xsl:attribute>
        </xsl:if>
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                </xsl:for-
each>
                                                                                <xsl:for-each
select="carare:roadName">
        <xsl:element name="carare:roadName">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                </xsl:for-
each>
                                                                                <xsl:for-each
select="carare:townOrCity">
        <xsl:element name="carare:townOrCity">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
        <xsl:if test="@authority !=''">
                <xsl:attribute name="xml:authority">
```

```
                        <xsl:value-of select="text()" />
                    </xsl:attribute>
            </xsl:if>
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                                    </xsl:for-
each>
                                                                                    <xsl:for-each
select="carare:postcodeOrZipcode">
            <xsl:element name="carare:postcodeOrZipcode">
            <xsl:if test="@lang !=''">
                    <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                    </xsl:attribute>
            </xsl:if>
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                                    </xsl:for-
each>
                                                                                    <xsl:for-each
select="carare:locality">
            <xsl:element name="carare:locality">
            <xsl:if test="@lang !=''">
                    <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                    </xsl:attribute>
            </xsl:if>
            <xsl:if test="@authority !=''">
                    <xsl:attribute name="xml:authority">
                        <xsl:value-of select="text()" />
                    </xsl:attribute>
            </xsl:if>
            <xsl:value-of select="text()" />
```

```
        </xsl:element>
                                                                                </xsl:for-
each>
                                                                                <xsl:for-each
select="carare:adminArea">
        <xsl:element name="carare:adminArea">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
        <xsl:if test="@authority !=''">
                <xsl:attribute name="xml:authority">
                        <xsl:value-of select="text()" />
                </xsl:attribute>
        </xsl:if>
        <xsl:value-of select="text()" />
        </xsl:element>
                                                                                </xsl:for-
each>
                                                                                <xsl:for-each
select="carare:country">
        <xsl:element name="carare:country">
        <xsl:if test="@lang !=''">
                <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                </xsl:attribute>
        </xsl:if>
        <xsl:if test="@authority !=''">
                <xsl:attribute name="xml:authority">
                        <xsl:value-of select="text()" />
                </xsl:attribute>
        </xsl:if>
```

```
		<xsl:value-of select="text()" />
		</xsl:element>
</xsl:for-each>
</xsl:element>
</xsl:for-each>

<xsl:for-each select="carare:geopoliticalArea">
<xsl:element name="carare:geopoliticalArea">
<xsl:for-each select="carare:geopoliticalAreaName">
		<xsl:element name="carare:geopoliticalAreaName">
		<xsl:if test="@lang !=''">
			<xsl:attribute name="xml:lang">
				<xsl:value-of select="translate(@lang,$upper,$lower)" />
			</xsl:attribute>
		</xsl:if>
		<xsl:value-of select="text()" />
		</xsl:element>
</xsl:for-each>
<xsl:for-each select="carare:geopoliticalAreaType">
		<xsl:element name="carare:geopoliticalAreaType">
		<xsl:if test="@lang !=''">
			<xsl:attribute name="xml:lang">
				<xsl:value-of select="translate(@lang,$upper,$lower)" />
			</xsl:attribute>
		</xsl:if>
		<xsl:if test="@namespace !=''">
			<xsl:attribute name="xml:namespace">
				<xsl:value-of select="text()" />
			</xsl:attribute>
		</xsl:if>
		<xsl:value-of select="text()" />
```

```
        </xsl:element>
                                                                                    </xsl:for-
each>
                                                                                </xsl:element>
                                                                            </xsl:for-each>

                                                                            <xsl:for-each
select="carare:cadastralReference">
                                                                                <xsl:element
name="carare:cadastralReference">
                                                                                    <xsl:if
test="@ref !=''">
        <xsl:attribute name="xml:ref">
        <xsl:value-of select="text()" />
        </xsl:attribute>
                                                                                    </xsl:if>
                                                                                    <xsl:value-of
select="text()" />
                                                                                </xsl:element>
                                                                            </xsl:for-each>

                                                                            <xsl:for-each
select="carare:historicalName">
                                                                                <xsl:element
name="carare:historicalName">
                                                                                    <xsl:if
test="@lang !=''">
        <xsl:attribute name="xml:lang">
        <xsl:value-of select="translate(@lang,$upper,$lower)" />
        </xsl:attribute>
                                                                                    </xsl:if>
                                                                                    <xsl:if
test="@preferred !=''">
        <xsl:attribute name="xml:preferred">
        <xsl:value-of select="text()" />
        </xsl:attribute>
                                                                                    </xsl:if>
                                                                                    <xsl:value-of
select="text()" />
                                                                                </xsl:element>
                                                                            </xsl:for-each>

                                                                        </xsl:element>

                                                                    </xsl:for-each>

                                                                    <xsl:for-each
select="carare:spatialReferenceSystem">
```

```
                                                    <xsl:element
name="carare:spatialReferenceSystem">
                                                        <xsl:if test="@lang !=''">
                                                            <xsl:attribute
name="xml:lang">
                                                                <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                            </xsl:attribute>
                                                        </xsl:if>
                                                        <xsl:value-of  select="text()"  />

                                                    </xsl:element>
                                                </xsl:for-each>

                                                <xsl:for-each
select="carare:cartographicReference">
                                                    <xsl:element
name="carare:cartographicReference">
                                                        <xsl:for-each
select="carare:spatialFeatureType">
                                                            <xsl:element
name="carare:spatialFeatureType">
                                                                <xsl:if
test="@lang !=''">
            <xsl:attribute name="xml:lang">
            <xsl:value-of select="translate(@lang,$upper,$lower)" />
            </xsl:attribute>
                                                                </xsl:if>
                                                                <xsl:value-of
select="text()" />
                                                            </xsl:element>
                                                        </xsl:for-each>
                                                        <xsl:for-each
select="carare:coordinates">
                                                            <xsl:element
name="carare:coordinates">
                                                                <xsl:element
name="carare:x">
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                <xsl:element
name="carare:y">
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                <xsl:element
name="carare:z">
            <xsl:value-of select="text()" />
            </xsl:element>
                                                            </xsl:element>
```

```
                                                                    </xsl:for-each>
                                                                </xsl:element>
                                                            </xsl:for-each>


                                                            <xsl:for-each select="carare:geometry">
                                                                <xsl:element name="carare:geometry">
                                                                    <xsl:for-each
select="carare:boundingBox">
                                                                        <xsl:element
name="carare:boundingBox">
                                                                            <xsl:element
name="carare:minX">
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                            <xsl:element
name="carare:maxX">
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                            <xsl:element
name="carare:minY">
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                            <xsl:element
name="carare:maxX">
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                        </xsl:element>
                                                                    </xsl:for-each>
                                                                    <xsl:for-each
select="carare:quickpoint">
                                                                        <xsl:element
name="carare:quickpoint">
                                                                            <xsl:element
name="carare:x">
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                            <xsl:element
name="carare:y">
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                        </xsl:element>
                                                                    </xsl:for-each>
                                                                    <xsl:for-each
select="carare:entity">
                                                                        <xsl:element
```

```
name="carare:entity">
                                                                                <xsl:if
test="@lang !=''">
        <xsl:attribute name="xml:lang">
        <xsl:value-of select="translate(@lang,$upper,$lower)" />
        </xsl:attribute>
                                                                                </xsl:if>

                                                                                <xsl:value-of
select="text()" />
                                                                        </xsl:element>
                                                                </xsl:for-each>
                                                                <xsl:for-each
select="carare:storedPrecision">
                                                                        <xsl:element
name="carare:storedPrecision">
                                                                                <xsl:if
test="@lang !=''">
        <xsl:attribute name="xml:lang">
        <xsl:value-of select="translate(@lang,$upper,$lower)" />
        </xsl:attribute>
                                                                                </xsl:if>

                                                                                <xsl:value-of
select="text()" />
                                                                        </xsl:element>
                                                                </xsl:for-each>
                                                                <xsl:for-each
select="carare:height">
                                                                        <xsl:element
name="carare:height">
                                                                                <xsl:if
test="@lang !=''">
        <xsl:attribute name="xml:lang">
        <xsl:value-of select="translate(@lang,$upper,$lower)" />
        </xsl:attribute>
                                                                                </xsl:if>

                                                                                <xsl:value-of
select="text()" />
                                                                        </xsl:element>
                                                                </xsl:for-each>

                                                                <xsl:for-each
select="carare:area">
                                                                        <xsl:element
name="carare:area">
                                                                                <xsl:if
test="@lang !=''">
```

```
            <xsl:attribute name="xml:lang">

            <xsl:value-of select="translate(@lang,$upper,$lower)" />

            </xsl:attribute>
                                                                                                                                                                    </xsl:if>

                                                                                                                                                                    <xsl:value-of
select="text()" />
                                                                                                                                                        </xsl:element>
                                                                                                                                            </xsl:for-each>

                                                                                                                                </xsl:element>
                                                                                                                    </xsl:for-each>
                                                                                                                    <xsl:for-each select="carare:representations">
                                                                                                                                <xsl:element
name="carare:representations">
                                                                                                                                            <xsl:if test="@lang !=''">
                                                                                                                                                        <xsl:attribute
name="xml:lang">
                                                                                                                                                                    <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                                                                                                                        </xsl:attribute>
                                                                                                                                            </xsl:if>

                                                                                                                                            <xsl:value-of select="text()" />
                                                                                                                                </xsl:element>
                                                                                                                    </xsl:for-each>
                                                                                                        </xsl:element>
                                                                                            </xsl:for-each>
                                    </xsl:element>
                        </xsl:for-each>

            </carare:carare>
</carare:carareWrap>
</xsl:template>

</xsl:stylesheet>
```

Temporal:

```
<?xml version="1.0" encoding="utf-8"?>

<xsl:stylesheet version="2.0"
            xmlns:carare="http://www.carare.eu/carareSchema"
            xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
   <xsl:strip-space elements="*" />
<xsl:output method="xml" version="1.0" encoding="utf-8" indent="yes"/>

<xsl:variable name="lower">abcdefghijklmnopqrstuvwxyz</xsl:variable>
<xsl:variable name="upper">ABCDEFGHIJKLMNOPQRSTUVWXYZ</xsl:variable>

<xsl:template match="carare:carareWrap">
<carare:carareWrap>
            <carare:carare>
```

```
<!-- COLLECTION INFORMATION -->
<xsl:for-each select="carare:carare/carare:collectionInformation">
<xsl:element name="carare:collectionInformation">
<xsl:for-each select="carare:coverage">
<xsl:element name="carare:coverage">
<xsl:for-each select="carare:temporal">
<xsl:element name="carare:temporal">
<xsl:for-each select="carare:timeSpan">
<xsl:element name="carare:timeSpan">
<xsl:for-each
select="carare:startDate">
<xsl:element
name="carare:startDate">
<xsl:if
test="@lang !=''">
<xsl:attribute name="xml:lang">
<xsl:value-of select="translate(@lang,$upper,$lower)" />
</xsl:attribute>
</xsl:if>
<xsl:value-of
select="text()" />
</xsl:element>
</xsl:for-each>
<xsl:for-each
select="carare:endDate">
<xsl:element
name="carare:endDate">
<xsl:if
test="@lang !=''">
<xsl:attribute name="xml:lang">
<xsl:value-of select="translate(@lang,$upper,$lower)" />
</xsl:attribute>
</xsl:if>
<xsl:value-of
select="text()" />
</xsl:element>
</xsl:for-each>
<xsl:for-each
select="carare:dimension">
<xsl:element
name="carare:dimension">
<xsl:for-each
select="carare:measurementUnit">
<xsl:element name="carare:measurementUnit">
<xsl:if test="@lang !=''">
<xsl:attribute name="xml:lang">
```

```
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                    </xsl:attribute>
            </xsl:if>
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                                    </xsl:for-
each>
                                                                                    <xsl:for-each
select="carare:value">
            <xsl:element name="carare:value">
            <xsl:if test="@lang !=''">
                    <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                    </xsl:attribute>
            </xsl:if>
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                                    </xsl:for-
each>
                                                                                    <xsl:for-each
select="carare:type">
            <xsl:element name="carare:type">
            <xsl:if test="@lang !=''">
                    <xsl:attribute name="xml:lang">
                        <xsl:value-of select="translate(@lang,$upper,$lower)" />
                    </xsl:attribute>
            </xsl:if>
            <xsl:value-of select="text()" />
            </xsl:element>
                                                                                    </xsl:for-
each>
                                                                                        </xsl:element>
                                                                                </xsl:for-each>
                                                                                <xsl:for-each
select="carare:dateRangeQualifier">
                                                                                        <xsl:element
name="carare:dateRangeQualifier">
                                                                                            <xsl:if
```

```
test="@lang !=''">

            <xsl:attribute name="xml:lang">

            <xsl:value-of select="translate(@lang,$upper,$lower)" />

            </xsl:attribute>
                                                                                                                                    </xsl:if>
                                                                                                                                    <xsl:if
test="@namespace !=''">

            <xsl:attribute name="xml:lang">

            <xsl:value-of select="text()" />

            </xsl:attribute>
                                                                                                                                    </xsl:if>


                                                                                                                                    <xsl:value-of
select="text()" />
                                                                                                                                            </xsl:element>
                                                                                                                            </xsl:for-each>
                                                                                                                    </xsl:element>
                                                                                                            </xsl:for-each>
                                                                                                            <xsl:for-each select="carare:periodName">
                                                                                                                    <xsl:element
name="carare:periodName">
                                                                                                                            <xsl:if test="@lang !=''">
                                                                                                                                    <xsl:attribute
name="xml:lang">
                                                                                                                                            <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                                                                                                    </xsl:attribute>
                                                                                                                            </xsl:if>
                                                                                                                            <xsl:if test="@preferred !=''">
                                                                                                                                    <xsl:attribute
name="xml:preferred">
                                                                                                                                            <xsl:value-of
select="text()" />
                                                                                                                                    </xsl:attribute>
                                                                                                                            </xsl:if>

                                                                                                                            <xsl:value-of  select="text()"  />

                                                                                                                    </xsl:element>
                                                                                                            </xsl:for-each>

                                                                                                            <xsl:for-each select="carare:displayDate">
                                                                                                                    <xsl:element
name="carare:displayDate">
                                                                                                                            <xsl:if test="@lang !=''">
                                                                                                                                    <xsl:attribute
name="xml:lang">
                                                                                                                                            <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                                                                                                    </xsl:attribute>
                                                                                                                            </xsl:if>
                                                                                                                            <xsl:if test="@preferred !=''">
```

```
                                                                                <xsl:attribute
name="xml:preferred">
                                                                                        <xsl:value-of
select="text()" />
                                                                                </xsl:attribute>
                                                                        </xsl:if>

                                                                        <xsl:value-of  select="text()"  />

                                                                </xsl:element>
                                                        </xsl:for-each>

                                                        <xsl:for-each select="carare:scientificDate">
                                                                <xsl:element
name="carare:scientificDate">
                                                                        <xsl:if test="@lang !=''">
                                                                                <xsl:attribute
name="xml:lang">
                                                                                        <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                                                </xsl:attribute>
                                                                        </xsl:if>

                                                                        <xsl:value-of  select="text()"  />

                                                                </xsl:element>
                                                        </xsl:for-each>

                                                        <xsl:for-each
select="carare:scientificDateMethod">
                                                                <xsl:element
name="carare:scientificDateMethod">
                                                                        <xsl:if test="@lang !=''">
                                                                                <xsl:attribute
name="xml:lang">
                                                                                        <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                                                </xsl:attribute>
                                                                        </xsl:if>

                                                                        <xsl:value-of  select="text()"  />

                                                                </xsl:element>
                                                        </xsl:for-each>
                                                </xsl:element>
                                        </xsl:for-each>
                                </xsl:element>
                        </xsl:for-each>
                </xsl:element>
        </xsl:for-each>


        <!-- HERITAGE ASSET -->
        <xsl:for-each select="carare:carare/carare:heritageAssetIdentification">
                <xsl:element name="carare:heritageAssetIdentification">
                        <xsl:for-each select="carare:characters">
                                <xsl:element name="carare:characters">
                                        <xsl:for-each select="carare:temporal">
```

```
                                                    <xsl:element name="carare:temporal">
                                                        <xsl:for-each select="carare:timeSpan">
                                                            <xsl:element name="carare:timeSpan">
                                                                <xsl:for-each
select="carare:startDate">
                                                                    <xsl:element
name="carare:startDate">
                                                                        <xsl:if
test="@lang !=''">
            <xsl:attribute name="xml:lang">
            <xsl:value-of select="translate(@lang,$upper,$lower)" />
            </xsl:attribute>
                                                                        </xsl:if>

                                                                        <xsl:value-of
select="text()" />
                                                                    </xsl:element>
                                                                </xsl:for-each>
                                                                <xsl:for-each
select="carare:endDate">
                                                                    <xsl:element
name="carare:endDate">
                                                                        <xsl:if
test="@lang !=''">
            <xsl:attribute name="xml:lang">
            <xsl:value-of select="translate(@lang,$upper,$lower)" />
            </xsl:attribute>
                                                                        </xsl:if>

                                                                        <xsl:value-of
select="text()" />
                                                                    </xsl:element>
                                                                </xsl:for-each>
                                                                <xsl:for-each
select="carare:dimension">
                                                                    <xsl:element
name="carare:dimension">
                                                                        <xsl:for-each
select="carare:measurementUnit">
            <xsl:element name="carare:measurementUnit">
            <xsl:if test="@lang !=''">
                        <xsl:attribute name="xml:lang">
                                    <xsl:value-of select="translate(@lang,$upper,$lower)" />
                        </xsl:attribute>
            </xsl:if>
```

```
		<xsl:value-of select="text()" />
		</xsl:element>
	</xsl:for-each>
	<xsl:for-each select="carare:value">
		<xsl:element name="carare:value">
		<xsl:if test="@lang !=''">
			<xsl:attribute name="xml:lang">
				<xsl:value-of select="translate(@lang,$upper,$lower)" />
			</xsl:attribute>
		</xsl:if>
		<xsl:value-of select="text()" />
		</xsl:element>
	</xsl:for-each>
	<xsl:for-each select="carare:type">
		<xsl:element name="carare:type">
		<xsl:if test="@lang !=''">
			<xsl:attribute name="xml:lang">
				<xsl:value-of select="translate(@lang,$upper,$lower)" />
			</xsl:attribute>
		</xsl:if>
		<xsl:value-of select="text()" />
		</xsl:element>
	</xsl:for-each>
		</xsl:element>
	</xsl:for-each>
	<xsl:for-each select="carare:dateRangeQualifier">
		<xsl:element name="carare:dateRangeQualifier">
			<xsl:if test="@lang !=''">
		<xsl:attribute name="xml:lang">
		<xsl:value-of select="translate(@lang,$upper,$lower)" />
```

```
        </xsl:attribute>
                                                                                    </xsl:if>
                                                                                    <xsl:if
test="@namespace !=''">

        <xsl:attribute name="xml:lang">

        <xsl:value-of select="text()" />

        </xsl:attribute>
                                                                                    </xsl:if>


                                                                                    <xsl:value-of
select="text()" />
                                                                                        </xsl:element>
                                                                                </xsl:for-each>
                                                                            </xsl:element>
                                                                        </xsl:for-each>
                                                                        <xsl:for-each select="carare:periodName">
                                                                            <xsl:element
name="carare:periodName">
                                                                                <xsl:if test="@lang !=''">
                                                                                    <xsl:attribute
name="xml:lang">
                                                                                        <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                                                    </xsl:attribute>
                                                                                </xsl:if>
                                                                                <xsl:if test="@preferred !=''">
                                                                                    <xsl:attribute
name="xml:preferred">
                                                                                        <xsl:value-of
select="text()" />
                                                                                    </xsl:attribute>
                                                                                </xsl:if>

                                                                                <xsl:value-of  select="text()"  />

                                                                            </xsl:element>
                                                                        </xsl:for-each>

                                                                        <xsl:for-each select="carare:displayDate">
                                                                            <xsl:element
name="carare:displayDate">
                                                                                <xsl:if test="@lang !=''">
                                                                                    <xsl:attribute
name="xml:lang">
                                                                                        <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                                                    </xsl:attribute>
                                                                                </xsl:if>
                                                                                <xsl:if test="@preferred !=''">
                                                                                    <xsl:attribute
name="xml:preferred">
                                                                                        <xsl:value-of
select="text()" />
                                                                                    </xsl:attribute>
                                                                                </xsl:if>
```

```
                                                                        <xsl:value-of  select="text()"  />

                                                                </xsl:element>
                                                        </xsl:for-each>

                                                        <xsl:for-each select="carare:scientificDate">
                                                                <xsl:element
name="carare:scientificDate">
                                                                        <xsl:if test="@lang !=''">
                                                                                <xsl:attribute
name="xml:lang">
                                                                                        <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                                                </xsl:attribute>
                                                                        </xsl:if>

                                                                        <xsl:value-of  select="text()"  />

                                                                </xsl:element>
                                                        </xsl:for-each>

                                                        <xsl:for-each
select="carare:scientificDateMethod">
                                                                <xsl:element
name="carare:scientificDateMethod">
                                                                        <xsl:if test="@lang !=''">
                                                                                <xsl:attribute
name="xml:lang">
                                                                                        <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                                                </xsl:attribute>
                                                                        </xsl:if>

                                                                        <xsl:value-of  select="text()"  />

                                                                </xsl:element>
                                                        </xsl:for-each>
                                                </xsl:element>
                                        </xsl:for-each>
                                </xsl:element>
                        </xsl:for-each>
                </xsl:element>
        </xsl:for-each>

        <!-- DIGITAL RESOURCE -->
        <xsl:for-each select="carare:carare/carare:digitalResource">
                <xsl:element name="carare:digitalResource">
                        <xsl:element name="carare:temporal">
                                <xsl:for-each select="carare:timeSpan">
                                        <xsl:element name="carare:timeSpan">
                                                <xsl:for-each select="carare:startDate">
                                                        <xsl:element name="carare:startDate">
                                                                <xsl:if test="@lang !=''">
                                                                        <xsl:attribute
name="xml:lang">
                                                                                <xsl:value-of
select="translate(@lang,$upper,$lower)" />
```

```
                                                                    </xsl:attribute>
                                                                </xsl:if>

                                                                <xsl:value-of select="text()" />

                                                            </xsl:element>
                                                        </xsl:for-each>
                                                        <xsl:for-each select="carare:endDate">
                                                            <xsl:element name="carare:endDate">
                                                                <xsl:if test="@lang !=''">
                                                                    <xsl:attribute
name="xml:lang">
                                                                        <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                                    </xsl:attribute>
                                                                </xsl:if>

                                                                <xsl:value-of select="text()" />

                                                            </xsl:element>
                                                        </xsl:for-each>
                                                        <xsl:for-each select="carare:dimension">
                                                            <xsl:element name="carare:dimension">
                                                                <xsl:for-each
select="carare:measurementUnit">
                                                                    <xsl:element
name="carare:measurementUnit">
                                                                        <xsl:if      test="@lang
!=''">
                                                                            <xsl:attribute
name="xml:lang">

            <xsl:value-of select="translate(@lang,$upper,$lower)" />

            </xsl:attribute>
                                                                        </xsl:if>

                                                                        <xsl:value-of
select="text()" />
                                                                    </xsl:element>
                                                                </xsl:for-each>
                                                                <xsl:for-each select="carare:value">
                                                                    <xsl:element
name="carare:value">
                                                                        <xsl:if      test="@lang
!=''">
                                                                            <xsl:attribute
name="xml:lang">

            <xsl:value-of select="translate(@lang,$upper,$lower)" />

            </xsl:attribute>
                                                                        </xsl:if>

                                                                        <xsl:value-of
select="text()" />
                                                                    </xsl:element>
                                                                </xsl:for-each>
```

```
                                                    <xsl:for-each select="carare:type">
                                                        <xsl:element
name="carare:type">
                                                            <xsl:if      test="@lang
!=''">
                                                                <xsl:attribute
name="xml:lang">
            <xsl:value-of select="translate(@lang,$upper,$lower)" />
            </xsl:attribute>
                                                            </xsl:if>

                                                            <xsl:value-of
select="text()" />
                                                        </xsl:element>
                                                    </xsl:for-each>
                                                </xsl:element>
                                            </xsl:for-each>
                                            <xsl:for-each select="carare:dateRangeQualifier">
                                                <xsl:element
name="carare:dateRangeQualifier">
                                                    <xsl:if test="@lang !=''">
                                                        <xsl:attribute
name="xml:lang">
                                                            <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                        </xsl:attribute>
                                                    </xsl:if>
                                                    <xsl:if test="@namespace !=''">
                                                        <xsl:attribute
name="xml:lang">
                                                            <xsl:value-of
select="text()" />
                                                        </xsl:attribute>
                                                    </xsl:if>

                                                    <xsl:value-of select="text()" />
                                                </xsl:element>
                                            </xsl:for-each>
                                        </xsl:element>
                                    </xsl:for-each>
                                    <xsl:for-each select="carare:periodName">
                                        <xsl:element name="carare:periodName">
                                            <xsl:if test="@lang !=''">
                                                <xsl:attribute name="xml:lang">
                                                    <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                </xsl:attribute>
                                            </xsl:if>
                                            <xsl:if test="@preferred !=''">
                                                <xsl:attribute name="xml:preferred">
                                                    <xsl:value-of select="text()" />
                                                </xsl:attribute>
                                            </xsl:if>

                                            <xsl:value-of select="text()" />
                                        </xsl:element>
```

```
                                        </xsl:for-each>

                                        <xsl:for-each select="carare:displayDate">
                                                <xsl:element name="carare:displayDate">
                                                        <xsl:if test="@lang !=''">
                                                                <xsl:attribute name="xml:lang">
                                                                        <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                                </xsl:attribute>
                                                        </xsl:if>
                                                        <xsl:if test="@preferred !=''">
                                                                <xsl:attribute name="xml:preferred">
                                                                        <xsl:value-of select="text()" />
                                                                </xsl:attribute>
                                                        </xsl:if>

                                                        <xsl:value-of select="text()" />
                                                </xsl:element>
                                        </xsl:for-each>

                                        <xsl:for-each select="carare:scientificDate">
                                                <xsl:element name="carare:scientificDate">
                                                        <xsl:if test="@lang !=''">
                                                                <xsl:attribute name="xml:lang">
                                                                        <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                                </xsl:attribute>
                                                        </xsl:if>

                                                        <xsl:value-of select="text()" />
                                                </xsl:element>
                                        </xsl:for-each>

                                        <xsl:for-each select="carare:scientificDateMethod">
                                                <xsl:element name="carare:scientificDateMethod">
                                                        <xsl:if test="@lang !=''">
                                                                <xsl:attribute name="xml:lang">
                                                                        <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                                </xsl:attribute>
                                                        </xsl:if>

                                                        <xsl:value-of select="text()" />
                                                </xsl:element>
                                        </xsl:for-each>
                                </xsl:element>
                        </xsl:element>
                </xsl:for-each>

                <!-- ACTIVITY -->
                <xsl:for-each select="carare:carare/carare:activity">
                        <xsl:element name="carare:activity">
                                <xsl:for-each select="carare:temporal">
                                        <xsl:element name="carare:temporal">
                                                <xsl:for-each select="carare:timeSpan">
                                                        <xsl:element name="carare:timeSpan">
                                                                <xsl:for-each select="carare:startDate">
                                                                        <xsl:element name="carare:startDate">
```

```
                                                                        <xsl:if test="@lang !=''">
                                                                                <xsl:attribute
name="xml:lang">
                                                                                        <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                                                </xsl:attribute>
                                                                        </xsl:if>

                                                                        <xsl:value-of  select="text()"  />

                                                                </xsl:element>
                                                        </xsl:for-each>
                                                        <xsl:for-each select="carare:endDate">
                                                                <xsl:element name="carare:endDate">
                                                                        <xsl:if test="@lang !=''">
                                                                                <xsl:attribute
name="xml:lang">
                                                                                        <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                                                </xsl:attribute>
                                                                        </xsl:if>

                                                                        <xsl:value-of  select="text()"  />

                                                                </xsl:element>
                                                        </xsl:for-each>
                                                        <xsl:for-each select="carare:dimension">
                                                                <xsl:element
name="carare:dimension">
                                                                        <xsl:for-each
select="carare:measurementUnit">
                                                                                <xsl:element
name="carare:measurementUnit">
                                                                                        <xsl:if
test="@lang !=''">
        <xsl:attribute name="xml:lang">
        <xsl:value-of select="translate(@lang,$upper,$lower)" />
        </xsl:attribute>
                                                                                        </xsl:if>

                                                                                        <xsl:value-of
select="text()" />
                                                                                </xsl:element>
                                                                        </xsl:for-each>
                                                                        <xsl:for-each
select="carare:value">
                                                                                <xsl:element
name="carare:value">
                                                                                        <xsl:if
test="@lang !=''">
        <xsl:attribute name="xml:lang">
        <xsl:value-of select="translate(@lang,$upper,$lower)" />
```

```
            </xsl:attribute>
                                                                                    </xsl:if>

                                                                                    <xsl:value-of
select="text()" />
                                                                                </xsl:element>
                                                                            </xsl:for-each>
                                                                            <xsl:for-each
select="carare:type">
                                                                                <xsl:element
name="carare:type">
                                                                                    <xsl:if
test="@lang !=''">

            <xsl:attribute name="xml:lang">

            <xsl:value-of select="translate(@lang,$upper,$lower)" />

            </xsl:attribute>
                                                                                    </xsl:if>

                                                                                    <xsl:value-of
select="text()" />
                                                                                </xsl:element>
                                                                            </xsl:for-each>
                                                                        </xsl:element>
                                                                    </xsl:for-each>
                                                                    <xsl:for-each
select="carare:dateRangeQualifier">
                                                                        <xsl:element
name="carare:dateRangeQualifier">
                                                                            <xsl:if test="@lang !=''">
                                                                                <xsl:attribute
name="xml:lang">
                                                                                    <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                                                </xsl:attribute>
                                                                            </xsl:if>
                                                                            <xsl:if        test="@namespace
!=''">
                                                                                <xsl:attribute
name="xml:lang">
                                                                                    <xsl:value-of
select="text()" />
                                                                                </xsl:attribute>
                                                                            </xsl:if>

                                                                            <xsl:value-of  select="text()"  />

                                                                        </xsl:element>
                                                                    </xsl:for-each>
                                                                </xsl:element>
                                                            </xsl:for-each>
                                                            <xsl:for-each select="carare:periodName">
                                                                <xsl:element name="carare:periodName">
                                                                    <xsl:if test="@lang !=''">
                                                                        <xsl:attribute name="xml:lang">
                                                                            <xsl:value-of
```

```
select="translate(@lang,$upper,$lower)" />
                                                                </xsl:attribute>
                                                        </xsl:if>
                                                        <xsl:if test="@preferred !=''">
                                                                <xsl:attribute name="xml:preferred">
                                                                        <xsl:value-of select="text()" />
                                                                </xsl:attribute>
                                                        </xsl:if>

                                                        <xsl:value-of select="text()" />
                                                </xsl:element>
                                        </xsl:for-each>

                                        <xsl:for-each select="carare:displayDate">
                                                <xsl:element name="carare:displayDate">
                                                        <xsl:if test="@lang !=''">
                                                                <xsl:attribute name="xml:lang">
                                                                        <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                                </xsl:attribute>
                                                        </xsl:if>
                                                        <xsl:if test="@preferred !=''">
                                                                <xsl:attribute name="xml:preferred">
                                                                        <xsl:value-of select="text()" />
                                                                </xsl:attribute>
                                                        </xsl:if>

                                                        <xsl:value-of select="text()" />
                                                </xsl:element>
                                        </xsl:for-each>

                                        <xsl:for-each select="carare:scientificDate">
                                                <xsl:element name="carare:scientificDate">
                                                        <xsl:if test="@lang !=''">
                                                                <xsl:attribute name="xml:lang">
                                                                        <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                                </xsl:attribute>
                                                        </xsl:if>

                                                        <xsl:value-of select="text()" />
                                                </xsl:element>
                                        </xsl:for-each>

                                        <xsl:for-each select="carare:scientificDateMethod">
                                                <xsl:element name="carare:scientificDateMethod">
                                                        <xsl:if test="@lang !=''">
                                                                <xsl:attribute name="xml:lang">
                                                                        <xsl:value-of
select="translate(@lang,$upper,$lower)" />
                                                                </xsl:attribute>
                                                        </xsl:if>

                                                        <xsl:value-of select="text()" />
                                                </xsl:element>
                                        </xsl:for-each>
                                </xsl:element>
                        </xsl:for-each>
```

```
                        </xsl:element>
                    </xsl:for-each>

            </carare:carare>
</carare:carareWrap>
</xsl:template>

</xsl:stylesheet>
```