INSPIRE support in MapServer opensource

March 2011

Table of contents

I. INTRODUCTION

The scope of this document is to detail the current status of MapServer opensource project toward INSPIRE requirements based on the View Service implementing rules and requirements.

The main objective is to point out the main obstacles existing in MapServer for building INSPIREd View Services, and to imagine possibilities of bypassing if fast deployment of such services is needed.

INSPIRE support in MapServer has already been a topic of discussion in the community. For more information, see http://lists.osgeo.org/mailman/listinfo/mapserver-inspire.

This document is based on MapServer 5.6.6 version. Some improvements may have been done in the upcoming 6.0.0 version. They will be noted as "Upgrade" in the "Solution" column.

This document has been written on behalf of French Geological Survey, BRGM, which supports and uses daily OpenSource software.

1. Revision History

Date	Release	Editor	Description
29/03/2011	0.1	Guillaume Sueur	Document structure
08/04/2011	1.0	Guillaume Sueur	Document finalization
15/04/2011	1.1	Benjamin Chartier	Review

2. Normative references

The following referenced documents are indispensable for the application of this document.

- Technical Guidance to implement INSPIRE View Services, v3.0
- INSPIRE Metadata Implementing Rules: Technical Guidelines based on EN ISO 19115 and EN ISO 19119, v1.2
- OpenGIS® Catalogue Services Specification 2.0.2 - ISO Metadata Application Profile, Document 07-045
- ISO 19115: 2003: Geographic Information – Metadata
- ISO 19119: 2005, Geographic information – Services
- ISO 19119: 2005 PDAM 1, Geographic information – Services
- ISO 19128: 2005, Geographic information — Web map server interface

3. Status définition

A	Fully supported	MapServer delivers automatically a correct content
B	Support depending on user input	MapServer can deliver a correct content if service or layer metadata are set accordingly to INSPIRE requirements
C	Partially supported	MapServer only supports a part of the recommendation
D	Not Supported	MapServer doesn't support the requirement at all.

4. Solutions definition

U	Upgrade	Upgrade to latest MapServer version
E	Enhancement	MapServer enhancement needed
BP	By-Pass	Use alternative solution

II. GENERAL BACKGROUND

Implementation Requirement 1 An INSPIRE View Service shall implement the minimal mandatory behavior from an [**ISO 19128**] service, extended with the extensions required by the INSPIRE Directive and the Implementing Rules for View services.

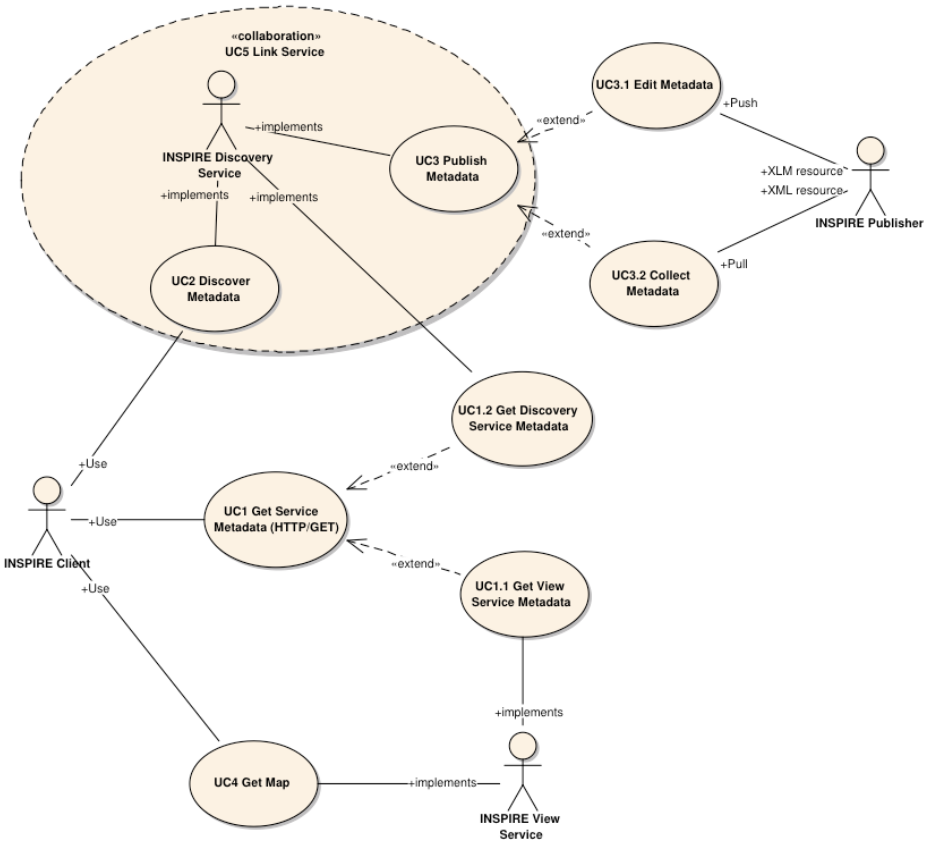

Figure 3: INSPIRE Generic Use Case

Figure 3 illustrates use cases for the creation and publication of metadata, their discovery through a discovery service and viewing of spatial data sets via an INSPIRE View service.

This requirement is generic and covers all the following requirements. On itself, it is not supported by MapServer

IR	STATUS	SOLUTION
1	D	Doesn't apply
Observations	Not only one solution can be proposed here.	

III. VIEW SERVICE OPERATIONS

1. Use of ISO 19128 / aka WMS 1.3.0

Implementation Requirement 2 The use of [ISO 19128] de jure standard as a basis for implementing an INSPIRE View service means that this service shall comply with the "basic WMS" conformance class as defined in this de jure standard.

IR	STATUS	SOLUTION
2	A	
Observations	MapServer is ISO 19128 compliant	

2. Implementation of GetCapabilities and GetMap

Implementation Requirement 3 The following ISO 19128 operations shall be implemented for an INSPIRE View service: GetCapabilities; GetMap.

IR	STATUS	SOLUTION
3	A	
Observations	The GET method is recommended for these operations. It is the standard MapServer behavior.	

The common requests parameters for View Service operations include a LANGUAGE parameter that is NOT supported by MapServer. All language requisites will be treated in the relevant LANGUAGE REQUIREMENTS section IV.

3. View Service Metadata

Implementation Requirement 4 The metadata response parameters shall be provided through the service Capabilities, as defined in the WMS Standard [ISO 19128, Section 7.2.4]. These capabilities are mandatory and defined when a WMS is set up. They consist of service information, supported operations and parameters values. The extended capabilities section shall be used to fully comply with the INSPIRE View Service metadata requirements (see section 4.2.3.3.1).

If View Service metadata are correctly provided throught the GetCapabilities operation in MapServer, the content of its response doesn't contain INSPIRE specific metadata yet. These metadata will be described in the following requirements.

IR	STATUS	SOLUTION

4	C	
Observations	This is a global requirement again, for which not only one solution can be proposed. Specific solutions will be provided for each subsequent items.	

4. GetCapabilities Operation

Implementation Requirement 5 The operation for implementing INSPIRE "Get View Service Metadata" operation is the GetCapabilities operation. The parameters defined within the [ISO 19128] standard shall be used to convey relevant information in order to get the expected responses as described in [INS NS, Annex III, Section 2.2] of the Regulation on INSPIRE Network Services.

As of GetCapabilities operations, the core parameters are standard WMS 1.3.0 / ISO 19128 parameters plus one optional LANGUAGE parameter. As this parameter is said to be optional, we could conclude that MapServer already fulfilled these needs, as long as we are aware that omitting the LANGUAGE parameter should force the use of the default language, which is not implemented yet in MapServer. Again, this will be part of the LANGUAGE REQUIREMENTS section. But as MapServer is not able to parse this parameter yet, which is necessary to proceed to the language support requirements, we can't consider MapServer supports this requirement.

IR	STATUS	SOLUTION
5	C	E
Observations	See LANGUAGE REQUIREMENTS section for a full approach.	

5. GetCapabilities Response.

Two different scenarios are proposed in the View Service Technical Guidance for publishing the View Service metadata. They are not mutually exclusive and both can be implemented.

The first scenario involves adding a reference to an external online INSPIRE metadata specific resource. This leads to IR 6 :

Implementation Requirement 6 The `<inspire_common:MetadataURL>` element within the extended INSPIRE capabilities of an [ISO 19128] – WMS 1.3.0 `<wms:Capability>` element shall be used to reference the INSPIRE service metadata available through an INSPIRE Discovery Service. Mandatory [ISO 19128] – WMS 1.3.0 metadata elements shall be mapped to INSPIRE metadata elements to implement a consistent interface.

This external metadata resource should then contain all mandatory INSPIRE metadata elements, all the ones coming from a standard WMS 1.3.0 content being mapped to corresponding INSPIRE metadata elements.

The second scenario involves mapping all INSPIRE metadata to WMS 1.3.0 elements and adding extended capabilities.

Implementation Requirement 7 **INSPIRE metadata are mapped to WMS capabilities elements to its full extent. It is mandatory to use the mapping provided in this Technical Guideline (described in Section 4.2.3.3.1.1 to 4.2.3.3.1.16. INSPIRE metadata elements that cannot be mapped to available [ISO 19128] – WMS1.3.0 elements are implemented as Extended Capabilities. Metadata are published through a service's capabilities document and can be harvested by an INSPIRE Discovery service.**

In any way, a language section must be added to the capabilities document :

Implementation Requirement 8 **Regardless of the scenario chosen to be implemented, a language section shall be added in the extended capability of the service to fulfil the language requirements of the Network Services Regulation [INS NS].**

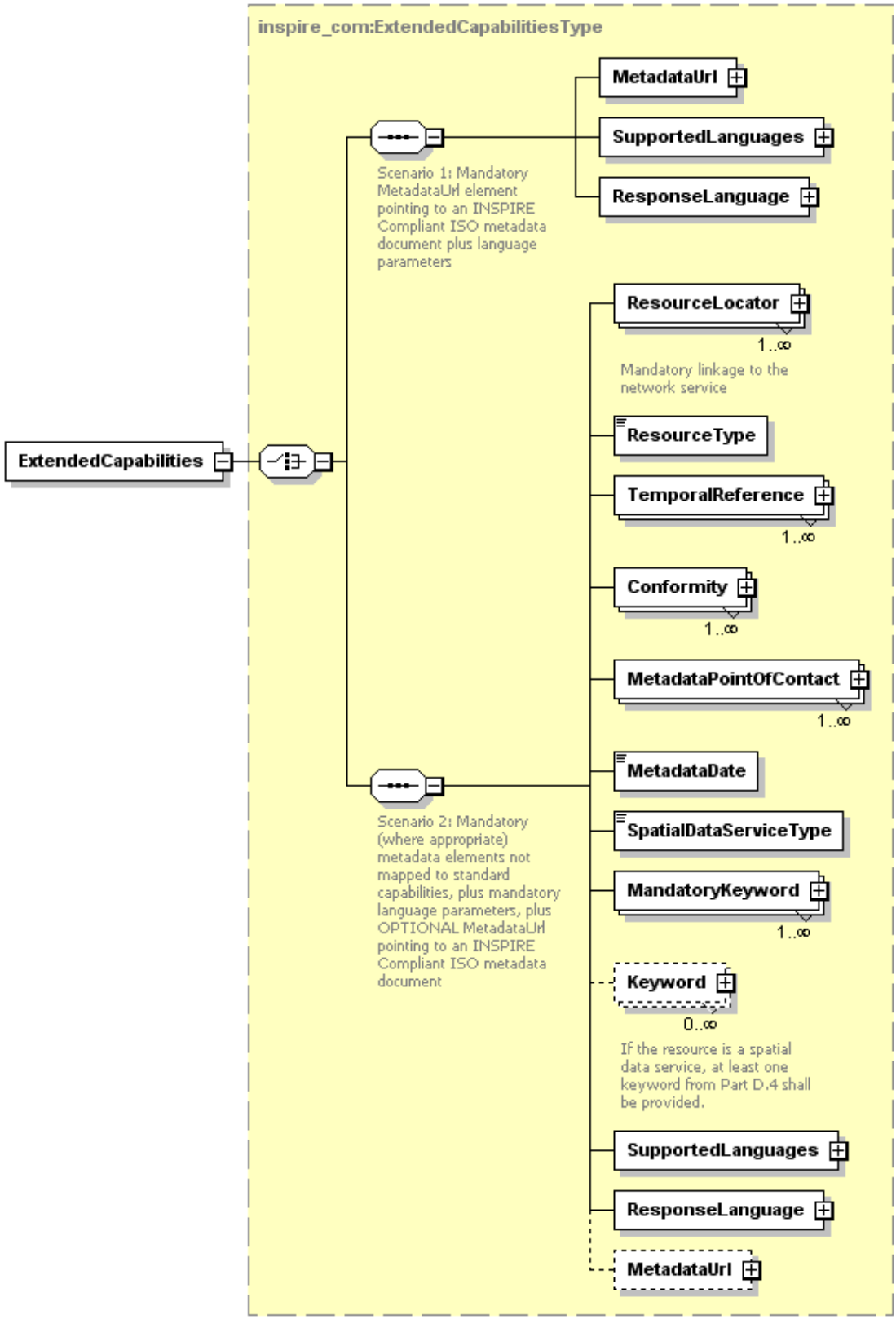

Figure 5: Illustration of the Extended Capabilities for Scenario 1 and 2 for INSPIRE View Services

As the choice of the scenario doesn't even rely on the View Service Provider but on the Member State itself, and as both scenarios may be implemented, we can consider that MapServer should be able to propose both as well.

IR	STATUS	SOLUTION
6	D	E
7	D	E
8	D	E
Observations	The getCapabilities document provided by MapServer doesn't fulfill INSPIRE needs. Language section must be added, as well as either a <inpire_common:MetadataURL> element pointing to an external document (which may need to be provided as well but may be provided by some other piece of software); either all mandatory INSPIRE extended Capabilities which will be described in the following items. The implementation of these two scenarios will lead to refactor the Capabilities process, depending on some switch to notify MapServer to generate an INSPIREd Capabilities document, and eventually which scenario to choose from.	

6. View Service Metadata

Implementation Requirement 9 Regardless of the scenario chosen to be implemented View Service Metadata shall be published in an INSPIRE Discovery Service. This is required to support a) the INSPIRE View Link service operation and b) discovery of View services by client applications such as the INSPIRE geoportal

This is a generic requirement which doesn't suppose any comment in itself but emphasizes the fact that all further INSPIRE metadata elements are mandatory to bring up an INSPIREd View Service compatible with both View Link Service and Discovery Service.

Implementation Requirement 10 An INSPIRE View service shall contain the INSPIRE metadata elements set out in the Metadata Regulation [INS MD] as shown in Table 3.

Here is Table 3 and all the following IR will detail these elements.

Table 3: Mapping between INSPIRE metadata elements and [ISO 19128] WMS elements

INSPIRE Metadata elements (Mandatory - Conditional)	**ISO 19128 elements of <WMS_Capabilities>**
Resource Title (M)	wms:Title
Resource Abstract (M)	wms:Abstract
Resource Type (M)	inspire_common:ResourceType (ExtendedCapabilities)
Resource Locator (C)	inspire_common:ResourceLocator (ExtendedCapabilities)
Coupled Resource (C)	wms:MetadataURL (Layer property)
Spatial Data Service Type (M)	inspire_common:SpatialDataServiceType (ExtendedCapabilities)
Keyword (M)	wms:Keyword; inspire_common:Keyword
Geographic Bounding Box (M)	wms:EX_GeographicBoundingBox (Layer property)
Temporal Reference (M)	inspire_common:TemporalReference (ExtendedCapabilities)
Spatial Resolution (C)	wms:Abstract
Conformity (M)	inspire_common:Conformity (ExtendedCapabilities)
Conditions for Access and Use (M)	wms:Fees
Limitations on Public Access (M)	wms:AccessConstraints
Responsible Organisation (M)	wms:ContactInformation
Metadata Point of Contact (M)	inspire_common:MetadataPointOfContact (ExtendedCapabilities)
Metadata Date (M)	inspire_common:MetadataDate (ExtendedCapabilities)
Metadata Language (M)	inspire_common:SupportedLanguages (ExtendedCapabilities)

The elements coming from ISO-19128 / WMS 1.3.0 specs are currently supported by MapServer, but sometimes not with the same prerequisites than the ones involved by the INSPIRE. That's why we will add specific comments on them even when an IR does not specifically cover them.

Beside these, IR10 introduces a set of new INSPIRE dedicated elements which disserve a first global approach of their implementation in MapServer.

It will be needed to be able to add and process these elements in the mapfile. They could be prefixed by some "inspire_" prefix and would be read and processed only in the case of the INSPIRE profile being asked (regardless to the way that profile would be asked, which is another discussion). For all of these elements, the implementation can be analogous to what has be done for wms_ metadata.

The use of these elements will lead to add a specific namespace to the xml document :

```
      xmlns:inspire_common="http://inspire.ec.europa.eu/schemas/common/1.0"
xmlns:inspire_vs="http://inspire.ec.europa.eu/schemas/inspire_vs/1.0"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation=http://inspire.ec.europa.eu/schemas/inspire_vs/1.0
http://inspire.ec.europa.eu/schemas/inspire_vs/1.0/inspire_vs.xsd
```

All of the inspire_common elements will have to be rendered inside a inspire_vs:ExtendedCapabilities element.

IR	STATUS	SOLUTION
9, 10	D	E
Observations	Need a switch to declare INSPIRE support is requested. Maybe when finding INSPIRE_PROFILE	

Title

MapServer correctly handles this element as it is required at the MAP level (inside Web Object) to turn on the WMS Service

Abstract

This element can be provided by MapServer but is not mandatory. This is the kind of element for which MapServer behaves correctly only if its user is aware of INSPIRE implementation rules. Ideally, switching to INSPIREd Capabilities should make this element mandatory, and MapServer would display an alert comment if not declared in the MAPFILE.

Resource Type

Implementation Requirement 11 Within the scope defined by the INSPIRE directive the value of the Resource Type shall be fixed to 'service' for spatial data services. As the Resource Type is not supported by [ISO 19128] – WMS 1.3.0, an extension shall be used to map this to an <inspire_common:ResourceType> element within an <inspire_vs:ExtendedCapabilities> element.

This is the first INSPIRE metadata to come and probably the easiest to add as its content is fixed to "service" in our context. The implementation in MapServer should then consider to add a

<inspire_common:ResourceType>service</inspire_common:ResourceType>

element as soon as the INSPIRE profile is required.

IR	STATUS	SOLUTION

11	D	E
Observations		

Resource Locator

The resource Locator is the INSPIRE equivalent to the WMS OnlineResource elements. But instead of mapping them, it is required for consistency with INSPIRE Discovery services.

Implementation Requirement 12 **An extension shall be used to map Resource Locator to an `<inspire_common:ResourceLocator>` element within an `<inspire_vs:ExtendedCapabilities>` element.**

IR	STATUS	SOLUTION
12	D	E
Observations	ResourceLocator to be added with the same content than OnlineResource.	

Coupled Resource

Implementation Requirement 13 **Coupled Resource shall be mapped to the `<MetadataURL>` elements of the Layer elements of the service capabilities. If linkage to the data sets or series on which the service operates are available, then the linkage to these resources shall be provided as stated by the INSPIRE Metadata Technical Guidance [INS MDTG].**

Implementation Requirement 14 **Each of the `<MetadataURL>` elements shall be populated with a URL that allows access to an unambiguous metadata record. The URL shall be either an HTTP/GET call on the GetRecordById operation of the Discovery Service or a direct link to the ISO 19139 metadata document.**

IR	STATUS	SOLUTION
13, 14	C	E
Observations	If MapServer is already able to use and publish MetadataURL, it is still unable to use several elements of that kind, which yet could be useful for pointing to different metadata format for instance, or various metadata flavours. But for a first implementation of INSPIRE support in MapServer it can be postponed.	

Spatial Data Service Type

Implementation Requirement 15 **For the Spatial Data Service Type as defined by the INSPIRE Metadata Regulation [INS MD] ('view') an extension shall be used to map this to an `<inspire_common:SpatialDataServiceType>` element within an `<inspire_vs:ExtendedCapabilities>` element. For an INSPIRE View Service the Spatial Data Service Type shall have a fixed value "view" according to INSPIRE Metadata Regulation [INS MD Part 3].**

IR	STATUS	SOLUTION
15	D	E
Observations	Add an inspire_common:SpatialDataServiceType in the Capabilities as soon as the INSPIRE profile is required.	

Keyword

Implementation Requirement 16 **The INSPIRE Metadata Regulation [INS MD] mandates that in the case of spatial data services at least one keyword from the "Classification of Spatial data Services" (Part D.4 from INS MD] shall be provided.**

Implementation Requirement 17 **If additional keywords are provided they shall be mapped with the `<wms:KeywordList>` element, the individual keywords shall be mapped to the `<wms:Keyword>` element, the referenced vocabulary shall be mapped to the 'vocabulary' attribute of the `<wms:Keyword>` element.**

Implementation Requirement 18 **The keywords shall be mapped to the capabilities extension `<inspire_common:Keyword>` and `<inspire_common:MandatoryKeyword>` within an `<inspire_vs:ExtendedCapabilities>` element.**

This is one of the most difficult element to implement, with the whole Language requirement. The complexity of the relationship between those keyword, mandatoryKeyword, KeywordList and their possible vocabularies makes it hard to fit into the standard METADATA block of the WEB object in the mapfile. Thus, solutions have to be found to allow the mandatory elements at least to be published. A first humble approach would be to first implement an inspire_keyword in the METADATA, which would be integrated as needed in the Capabilities document. Ability of publishing more than one keyword from more than the "Classification of Spatial data Services" could then be added later.

IR	STATUS	SOLUTION
16	D then B	E
17	D then B	E

18	D then B	E
Observations	Start with one inspire_keyword metadata in which the user will have to put a keyword from "Classification of Spatial data Services. Even after a full implementation of this requirement, its complete respect will depend on the user input.	

Geographic bounding box

Implementation Requirement 19 Geographic Bounding Box shall be mapped to the `EX_GeographicBoundingBox` element of Layer elements.

IR	STATUS	SOLUTION
19	A	
Observations	Same as WMS.	

Temporal Reference

Implementation Requirement 20 To be compliant with the INSPIRE Metadata Regulation [INS MD] and with [ISO 19115] one of following dates shall be used: date of publication, date of last revision, or the date of creation. Date of last revision is preferred. The date shall be expressed in conformity with the [INS MD]

Implementation Requirement 21 As the Temporal Reference is not directly supported by [ISO 19128] – WMS 1.3.0 an extension shall be used to map this to an `<inspire_common:TemporalReference>` element within an `<inspire_vs:ExtendedCapabilities>` element.

That is a rather straightforward requirement, which only means to add some metadata. Within a global TemporalReference element, one should find some of the following items: DateOfPublication, DateOfLastRevision, DateOfCreation.

IR	STATUS	SOLUTION
20 21	D then B D then B	E E
Observations	As each date must be set following INSPIRE MD convention (ie YYYY-MM-DD), complete respect if this requirement will depend on user input, unless MapServer can apply a pattern verification of these fields.	

Spatial Resolution

Spatial resolution of the data set is not mapped to any normative element at this date. Its content should be added to the <abstract> element content, but it is still at a recommendation level only, not a requirement yet.

Conformity

Implementation Requirement 22 **The INSPIRE Metadata Regulation [INS MD] requires that metadata shall include information on the degree of conformity with the implementing rules provided in Art. 7.1 (Interoperability of spatial data sets and services) of the INSPIRE Directive [Directive 2007/2/EC].**

Implementation Requirement 23 **An extension shall be used to map this to an `<inspire_common:Conformity>` element within an `<inspire_vs:ExtendedCapabilities>` element.**

IR	STATUS	SOLUTION
22, 23	D then B	E
Observations	Conformity must be one of the following terms : “conformant”, “not conformant” or “not evaluated” which should be the default value and would be the estimated value if this element is not set. Note that the final structure is: <Conformity><Degree>notEvaluated</Degree></Conformity>	

Conditions for access and use.

Implementation Requirement 24 **This metadata element shall be mapped to the `<wms:Fees>` element of the capabilities. If no conditions apply to the access and use of the resource, "no conditions apply" shall be used. If conditions are unknown "conditions unknown" shall be used.**

IR	STATUS	SOLUTION
24	B	
Observations	The ISO 19128 “fees” element does exist in MapServer capabilities, but for full compliance, its content should be “no conditions apply”, “conditions unknown” or the specific conditions specified by the user. I guess that if wms_fees is not present in the Mapfile, writing a wms:Fees element with “conditions unknown” would be a plus.	

Limitation on Public access

No requirement here, but the recommendation to use None or something from MD_RestrictionCode codelist (ISO-19115, annex B, Data dictionnary).

Ideally, MapServer should add the AccessConstraint element with "None" content when this metadata is not explicitly set by the user.

Responsible organisation

Implementation Requirement 25 **Responsible Party as described in the INSPIRE Metadata Regulation [INS MD] shall be mapped to the `<wms:ContactOrganization>` element of the `<wms:ContactPersonPrimary>` within the `<wms:ContactInformation>` element.**

Implementation Requirement 26 **The value domain of the Responsible Party role shall comply with the INSPIRE Metadata Regulation [INS MD, Part D6]. The Responsible Party Role shall be mapped to the `<wms:ContactPosition>` of the `<wms:ContactInformation>` element.**

IR	STATUS	SOLUTION
25 26	A B	
Observations	Full compliance will only be obtained if user follows Responsible Party Role implementation rules.	

Metadata Point of Contact

Implementation Requirement 27 **INSPIRE is more demanding than [ISO 19115] by mandating both the name of the organisation, and a contact e-mail address. The role of the responsible party serving as a metadata point of contact is out of scope of the Metadata Regulation [INS MD], but this property is mandated by [ISO 19115]. Its value shall be defaulted to "pointOfContact".**

Implementation Requirement 28 **Since only one `<wms:ContactInformation>` element is allowed in [ISO 19128] – WMS 1.3.0 (to which Responsible Organisation is mapped), an extension shall be used to map this to an `<inspire_common:MetadataPointOfContact>` element within an `<inspire_vs:ExtendedCapabilities>` element.**

IR	STATUS	SOLUTION
27 28	D D	E E
Observations	INSPIRE introduces here a second "contact organization" which is not included in ISO 19128. Two inspire_ metadata could be added in the Mapfile : Inspire_mpoc_name Inspire_mpoc_email In the capabilities, another element should be added in	

	order to keep compliance with ISO-19115 : a Role element which would be set to “PointOfContact”.

Metadata Date

Implementation Requirement 29 **As the Metadata Date is not supported by [ISO 19128] – WMS 1.3.0, an extension shall be used to map this to an `<inspire_common:MetadataDate>` element within an `<inspire_vs:ExtendedCapabilities>` element. The date shall be expressed in conformity with the [INS MD].**

IR	STATUS	SOLUTION
29	D then B	E
Observations	Another date to be added here, with same requirements as for IR 20.	

7. Operations Metadata

There are three distinct operations to be exposed by the service:

GetCapabilities operations metadata

Implementation Requirement 30 **GetCapabilities operation metadata shall be mapped to the**

`<wms:GetCapabilities>` element.

IR	STATUS	SOLUTION
30	A	
Observations		

GetMap operation metadata

Implementation Requirement 31 **GetMap operation metadata shall be mapped to the `<wms:GetMap>` element. Either PNG or GIF format (without LZW compression) with transparency shall be supported by the View service [INS NS, Annex III, Part B].**

IR	STATUS	SOLUTION
31	A	
Observations		

LinkView Service operation metadata

Not a requirement, but a Recommendation:

The LinkView Service, which allows a Public Authority or a Third Party to declare a view Service for the viewing of its resource through the Member State View Service while maintaining the viewing capability at the Public Authority or the Third Party location, shall be implemented through the "Discover Metadata" operation of the Discovery Service.

Implementation Recommendation 7 The use of the "Discover Metadata" operation of the INSPIRE Discovery service is recommended for implementing the Link View Service operation.

This operation is so beyond the scope of MapServer Inspire support.

8. Layers Metadata.

Implementation Requirement 32 The description of a layer shall use elements defined for the service capabilities in the [ISO 19128] standard. This description shall specify the role of some parameters for the INSPIRE View Service as stated in the Regulation on INSPIRE Network Services [INS NS].

The good news here are that all required Metadata are mapped to existing ISO 19128 elements. This means that unless current lack in MapServer implementation, most of the job is already done.

IR	STATUS	SOLUTION
32	A ?	
Observations	Complete compliance to be checked in further sub-items.	

Title

Implementation Requirement 33 It is mapped with `<wms:Title>`. The harmonised title of a layer for an INSPIRE spatial data theme is defined by [INS DS] and shall be subject to multilingualism (translations shall appear in each mono-lingual capabilities localised documents).

IR	STATUS	SOLUTION
33	A	
Observations	Multilinguism aspects will be treated in LANGUAGE REQUIREMENTS section VI.	

Abstract

Implementation Requirement 34 Text describing the layer. Subject to multilingualism. It shall be mapped with the `<wms:Abstract>` element.

IR	STATUS	SOLUTION

34	A	
Observations	Multilinguism aspects will be treated in LANGUAGE REQUIREMENTS section VI.	

Keywords

Implementation Requirement 35, It shall be mapped to the `<wms:KeywordList>` element.

IR	STATUS	SOLUTION
35	C	
Observations	Same requirements on vocabulary exposure than in Service metadata are recommended. Same implementation should be used.	

BoundingBox

Implementation Requirement 36 This Layer metadata element shall be mapped to the `<wms:BoundingBox>` element. The minimum bounding rectangle of the area covered by the Layer in all supported CRS shall be given.

IR	STATUS	SOLUTION
36	D	U -> 6.0
Observations	All supported CRS shall be given. wms_extent usage may become mandatory as computation of extent for some types of layers is not implemented (Postgis mainly). Supported in MapServer 6.0	

Unique Resource Identifier

Implementation Requirement 37 The [INS MD] Regulation defines a Unique Resource Identifier as a value uniquely identifying an object within a namespace. The code property shall be specified at a minimum, and a codeSpace (namespace) property may be provided.

Implementation Requirement 38 To be able to map the concept of a responsible `body/codeSpace` and local `identifier/code` to [ISO 19128]), `AuthorityURL` and `Identifier` elements shall be used. The authority name and explanatory URL shall be defined in a separate AuthorityURL element, which may be defined once and inherited by subsidiary layers. Identifiers themselves are not inherited.

The URI is not currently supported by MapServer. It involves adding two new metadata at the layer's level. Thus, the possibility of reusing an already declared AuthorityURL seems harder to implement and handle than letting the user repeating it for each of the layers.

IR	STATUS	SOLUTION
37, 38	D	E
Observations	Implementation recommendation mention one should use UUID as specified by IETF for URI.	

Name

Implementation Requirement 39 **Name shall be mapped with the `<wms:Name>` element. The harmonised name of a layer shall comply with the Layer requirements of the [INS DS, Article 14]**

IR	STATUS	SOLUTION
39	B	
Observations	It will be of the user responsibility to follow harmonized names convention of INSPIRE	

Coordinate reference systems

Implementation Requirement 40 **It is mandatory to use geographical coordinate system based on ETRS89 in continental Europe and ITRS outside continental Europe.**

IR	STATUS	SOLUTION
40	B	
Observations	Maybe MapServer can generate some warnings if the mandatory reference systems are not declared.	

Styles

Implementation Requirement 41 **A Style shall be composed of a Title and a Unique Identifier.**

Implementation Requirement 42 **An `<inspire_common:DEFAULT>` style for each theme shall be as defined in the "Portrayal" section of the [INS DS, Article 14].**

Implementation Requirement 43 **For layers with no associated default style, the INSPIRE Generic Conceptual Model [INS GCM] defines simple styles shall be used in data portrayal, derived from Symbology Encoding Implementation**

Specification [OGC SEIS]: Point: grey square, 6 pixels; Curve: black solid line, 1 pixel; Surface: black solid line, 1 pixel, grey fill.

Implementation Requirement 44 If no style is specified in the request or the style parameter is empty, the <inspire_common:DEFAULT> style shall be used in layer rendering.

Implementation Requirement 45 A legend shall be provided for each style and supported language defined in the View Service.

Implementation Requirement 46 Style shall be mapped to the <wms:Style> element. The human- readable name shall be mapped to the <wms:Title> element and the Unique Identifier shall be mapped to the <wms:Name> element.

We have several requirements coming up for STYLES here. The global mechanism is already supported by MapServer, but the language part (IR45) may be the most difficult step to achieve.
Further analysis assumes the usage of CLASSGROUP and GROUP inside CLASS elements in order to expose several different styles into the getCapabilities document.

IR	STATUS	SOLUTION
41	A, but they are the same, generated from the GROUP strings inside CLASS blocks	E
42	B, as it depends on how the user declares his own CLASSES GROUPS	
43	C, as usage of the required defaults styles should be implemented.	E
44	B, as it needs the CLASSGROUP to be set to "INSPIRE:DEFAULT"	
45	D, languages not supported, see LANGUAGE REQUIREMENTS section VI	E
46	A	
Observations	A global approach of the STYLEs management must be done, taking care of MapServer 6 recent evolutions.	

An ideal refactoring of the style part would consider, in INSPIRE profile, the default style to be named INSPIRE:DEFAULT to be able to get rid of the CLASSGROUP thing when only one style is defined. It make it simpler for mapfile migration too.

The biggest pain will be to implement multilingualism in this perimeter. That would suppose to have some “inspire_style_*name*_name” and “inspire_style_*name*_tiltle” for each of the supported languages.

LegendURL

Implementation Requirement 47 As the capabilities document is a mono-lingual document, internationalized legend may be placed in a different capabilities document for each value of the LANGUAGE parameter. It shall be mapped with the `<wms:LegendURL>` element.

IR	STATUS	SOLUTION
47	C	E
Observations	LANGUAGE parameter handling must be done.	

Dimension pairs

Implementation Requirement 48 In other cases such as time and elevation, `<wms:Dimension>` shall be used according to [INS NS].

IR	STATUS	SOLUTION
48	A	
Observations	Currently, only TIME dimension is supported in MapServer. For this only dimension, support is conform.	

Category Layer

Implementation Requirement 49 A containing Category Layer itself includes a Name by which a map portraying all of the nested layers can be requested at once. If a metadata description of this category composition exists then the MetadataURL for the Category Layer shall be provided.

IR	STATUS	SOLUTION
49	C	E
Observations	Currently, the category Layer comes from the GROUP attribute in a LAYER block of the mapfile. Thus, no metadataURL is provided at this level.	

IV. GETMAP OPERATIONS

Version

Implementation Requirement 50 **The mandatory VERSION parameter. The value "1.3.0" shall be used for GetMap requests that comply with the [ISO 19128] standard.**

IR	STATUS	SOLUTION
50	A	
Observations	That's more a client requirement than a server's.	

Request

Implementation Requirement 51 **The mandatory REQUEST parameter is defined in [ISO 19128, Section 6.9.2]. To invoke the GetMap operation, the value "GetMap" shall be used to comply with the [ISO 19128] standard.**

IR	STATUS	SOLUTION
51	A	
Observations		

Layers

Implementation Requirement 52 **The mandatory LAYERS parameter lists the map layer(s) to be returned by this GetMap request. The value of the LAYERS parameter shall be a comma-separated list of one or more valid INSPIRE harmonized layer names.**

IR	STATUS	SOLUTION
52	A	
Observations		

Styles

Implementation Requirement 53 **The mandatory STYLES parameter lists the style in which each layer is to be rendered. The value of the STYLES parameter shall be a comma-separated list of one or more valid INSPIRE style names. A client may request the default Style using a null value (as in "STYLES=").**

IR	STATUS	SOLUTION
53	A	
Observations		

CRS

Implementation Requirement 54 **The CRS request parameter states what Layer CRS applies to the BBOX request parameter. Values must be CRS that are defined in the INSPIRE Annex I, theme 1, Coordinate Reference System.**

IR	STATUS	SOLUTION
54	B	
Observations	Users will have to specify the correct CRS	

Bounding Box

Implementation Requirement 55 **The mandatory BBOX parameter allows a Client to request a particular Bounding Box. The value of the BBOX parameter in a GetMap request shall be a list of comma-separated real numbers in the form "minx,miny,maxx,maxy". These values specify the minimum X, minimum Y, maximum X, and maximum Y values of a region in the Layer CRS of the request. The units, ordering and direction of increment of the X and Y axes shall be as defined by the Layer CRS. The four bounding box values indicate the outside limits of the region.**

IR	STATUS	SOLUTION
55	A	
Observations		

Width, Height

Implementation Requirement 56 **The mandatory WIDTH and HEIGHT parameters specify the size in integer pixels of the map to be produced.**

IR	STATUS	SOLUTION
56	A	
Observations		

Format

Implementation Requirement 57 **The mandatory FORMAT parameter states the desired format of the map. The [INS NS, Annex III, Part B, Section 2] Image format states that at least one of "image/png" or "image/gif" must be supported and therefore advertised in the GetCapabilities operation.**

IR	STATUS	SOLUTION
57	A	
Observations		

Transparent

Implementation Requirement 58 The optional TRANSPARENT parameter specifies whether the map background is to be made transparent or not. The service is required to implement this.

IR	STATUS	SOLUTION
58	A	
Observations		

Exceptions

Implementation Requirement 59 The default value shall be "XML" if this parameter is absent from the request. Other valid values are INIMAGE and BLANK.

IR	STATUS	SOLUTION
59	A	
Observations		

V. LINK VIEW SERVICE OPERATION

IR 60 and 61 don't apply to MapServer as they rely to a Discovery Service.

Cascading

Implementation Requirement 62 **In the case where it is more preferable to collate maps in a View Service (for example: the Member State View Service collates maps that are served locally with maps that are served remote by a Third Party), the Member State's View Service shall include the service's layer metadata in his own service metadata (capabilities document).**

Implementation Requirement 63 **The "cascaded" attribute of the `<wms:Layer>` element shall be used to indicate that the layer is hosted by a remote View Service.**

Implementation Requirement 64 **Every time a map from a View Service is cascaded through another View Service the value of the "cascaded" attribute shall be incremented by 1. The actual collation of maps is out-of-scope for this Technical Guideline.**

IR	STATUS	SOLUTION
62 63 64	A A D	
Observations	Cascaded attribute is always 1 (Boolean) in MapServer.	

Transparency

Implementation Requirement 65 **To support collation with other maps for both supported image formats (GIF and PNG), the transparency parameter (TRANSPARENT) of the WMS GetMap request shall be set to "true" and the background parameter (BGCOLOR) for all layers shall be set to the same colour.**

IR	STATUS	SOLUTION
65	A	
Observations		

VI. LANGUAGE REQUIREMENTS

This topic is certainly the most painful to implement as it applies to all the natural languages fields, such as layer title, style title and so on.

Implementation Requirement 66 **A network service metadata response shall contain a list of the natural languages supported by the service. This list shall contain one or more languages that are supported.**

Implementation Requirement 67 **A client may specify a specific language in a request. If the requested language is contained in the list of supported languages, the natural language fields of the service response shall be in the requested language. Iff the requested language is not supported by the service, then this parameter shall be ignored.**

Implementation Requirement 68 **The name of this parameter shall be "LANGUAGE". The parameter values are based on ISO 639-2/B alpha 3 codes as used in [INS MDTG].**

Implementation Requirement 69 **If a client request specifies an unsupported language, or the parameter is absent in the request, the above fields shall be provided in the service default language.**

Implementation Requirement 70 **The Extended Capabilities shall indicate the response language used for the GetCapabilities-Response: Depending on the requested language the value of the `<inspire_common:ResponseLanguage>` corresponds to the current used language. If a supported language was requested, `<inspire_common:ResponseLanguage>` shall correspond to that requested language. If an unsupported language was requested or if no specific language was requested `<inspire_common:ResponseLanguage>` shall correspond to the service default language `<inspire_common:DefaultLanguage>`**

Implementation Requirement 71 **The Extended Capabilities shall contain the list of supported languages indicated in `<inspire_common:SupportedLanguages>`.**

This list of supported languages shall consist of 1. exact one element `<inspire_common:DefaultLanguage>` indicating the service default language, and 2. zero or more elements `<inspire_common:SupportedLanguage>` to indicate all additional supported languages.

Regardless of the response language, the list of supported languages is

invariant for each GetCapabilities-Response.

Implementation Requirement 72 The Extended Capabilities shall use the XML Schema as defined in the INSPIRE online schema repository.

IR	STATUS	SOLUTION
66 67 68 69 70 71 72	D D D then B D D D D	E E E E E E E
Observations	One can add SUPPORTED_LANGUAGES and DEFAULT_LANGUAGE in WEB object METADATA. Compliance with ISO codes will be let to user input.	

Implementation Requirement 73 If any portrayal rules require language support for rendered text - e.g. by further amendments for Annex II or Annex III - INSPIRE View Services shall implement the common concept as stated in Section4.3.2.

IR	STATUS	SOLUTION
73	D	
Observations	That's a real last but not least IR. It states thatt if someday GetMap results depend on any selected language, the capabilities will have to expose the specific access point to that language. Given that LANGUAGE parameter is not in the perimeter of the GetMap operation, it becomes an infrastructure specification which allows to broadcast several GetMap entries : http://someHOST.example/**eng**/GetMap? For the moment, let's say wisely "not required yet"…	

VII. MAPSERVER IMPLEMENTATION GUIDELINES.

This chapter's scope is to consider the preceding notes and requirements from a MapServer point of view and to give clues and milestones in order to achieve the implementation of Inspire requirements in MapServer.

1. How to know INSPIRE profile is requested?

As nothing differs in a GetCapabilities INSPIRE request from a standard WMS GetCapabilitiesG request (the language parameter is optional, and thus can't be considered as a flag), the service master will have to specify in the mapfile that this service must be exposed following INSPIRE rules. This can be achieved by adding a metadata into the WEB Object:

"INSPIRE_PROFILE" "ON"

2. How to choose between the 2 available scenarios?

The content of the capabilities is very different depending on the chosen scenario. While scenario 1 only implies to add MetadataURL element and Language sections (which is not few however), scenario 2 injects in the capabilities document all the INSPIRE MD not mapped to any ISO-19128 elements. That's why we can consider adding another metadata to the WEB object:

"INSPIRE_SCENARIO" "1"

In another hand, this can be mixed with the first item:

"INSPIRE_PROFILE" "1"

which allows to declare in the same metadata to activate INSPIRE profile and which scenario to choose.

3. How to generate an INSPIRE Scenario 1 valid capabilities?

The good news with scenario 1 is that most of the INSPIRE MD are delegated to a specific external document, available from a GeoNetwork catalog for instance. That is probably the easiest and fastest approach to make MapServer compatible with INSPIRE requirements, as implementing both the scenarios is not a requirement in itself[1]. This will lead to:

- Add convenient INSPIRE namespace in the XML :

```
xmlns:inspire_common="http://inspire.ec.europa.eu/schemas/common/1.0" xmlns:inspire_vs="http://inspire.ec.europa.eu/schemas/inspire_vs/1.0"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation=http://inspire.ec.europa.eu/schemas/inspire_vs/1.0
http://inspire.ec.europa.eu/schemas/inspire_vs/1.0/inspire_vs.xsd
```

- Add another entry in the WEB object's metadata :

« inspire_metadataURL » « some url »

[1] However, the choice between S1 and S2 has to be made by the Member State, which means MapServer will have to support both in order to be usable in any Member State.

and place this in an inspire extended capabilities element :

```
<inspire_vs:ExtendedCapabilities>
<inspire_common:MetadataUrl
xsi:type="inspire_common:resourceLocatorType">
<inspire_common:URL>http://discoveryServiceURL.be?Service=CSW&amp;amp;Request
=GetRecordById&amp;amp;Version=2.0.2&amp;amp;id=qc7f8260-3pe9-66e0-9031-
d15699a4fda4&amp;amp;outputSchema=http://www.isotc211.org/2005/gmd&amp;amp;elementS
etName=full</inspire_common:URL>
<inspire_common:MediaType>application/vnd.ogc.csw.GetRecordByIdResponse_x
ml</inspire_common:MediaType>
</inspire_common:MetadataUrl>
</inspire_vs:ExtendedCapabilities>
```

4. Language support:

This is the toughest part of INSPIRE support in MapServer. Not only it requires handling language parameter in the GetCapabilities Requests, but also it requires exposing specific urls and translated terms.
A straightforward approach would be to have some SUPPORTED_LANGUAGES item in the WEB metadata and then make the user add “wms_[language_code]_title” for the layers and styles. But it can become quickly hard to maintain such a mapfile.
Can we imagine managing translations (which mainly apply to titles and abstracts for Service and Layer, and to title for styles) outside the Mapfile? We could have, like FONTSET, an external folder giving all the required translations:

MAPFILE

LANGUAGE_FOLDER → LOCALE

MapServer could generate and prepopulate these external folders with entries for each of the declared supported languages. Let’s say we have a layer with “Rivers” for title and“Hydrography physical waters” for abstract, and “ENG, FRE” for supported languages. We would have, inside LOCALE folder:

eng.txt:

“Rivers” “Rivers”

“Hydrography physical waters” “Hydrography physical waters”

fre.txt

“Rivers” “Rivières”

“Hydrography physical waters” “Réseau hydrographique”

A simple command-line tool could help in these language files management, to update keys for instance.
All the terms used in the mapfile would be considered as the default language entries. All the files for supported languages would be populated with these default entries, giving the user the responsibility of translating them, with no impact on MapServer behavior (like “no input for this language” issue).

5. Overall considerations

Implementation of INSPIRE ViewServices requirements for scenario 1 should not be too difficult to achieve. It involves adding few metadata at the map level, specify correctly few others at the layer level (mainly STYLE handling, where MapServer’s default behavior publishes a “default” layer, not an

“INSPIRE:DEFAULT”). It involves generating a specific GetCapabilities, but Scenario 1 makes it very close to the ISO-19128 one.

The main task then is to add language support. Having it handled beside the MapFile makes sense to me, as it allows people to build the mapfile as usual, with only few concerns on language issues (having just to declare supported and default languages at the map level). The default language files generated would allow INSPIRE conformance, letting to the user the responsibility to put in there the correct translations.