OGC Web Services and GML Modelling for Operational Meteorology

6th – 10th February 2006

Communiqué – workshop findings and outcomes

This document summarises the discussions and recommendations of a workshop convened by the UK Met Office to examine the application of OGC Web Services and GML Modelling to Operational Meteorology. The workshop addressed the need to define best practice for development of data standards that span multiple subject domains, jurisdictions and processing technologies. The findings of this workshop will be of use not only to organisations involved in the processing of meteorological data, but any community that requires interoperability along a data processing chain.

Introduction and Scope

Operational Meteorology is a subject domain with an international scope, and data interoperability between practitioners is crucial. However, existing software tooling is generally bespoke and doesn't take advantage of "off-the-shelf" technologies, and so is difficult to maintain and extend. Accordingly, the Met Office[1] is investigating appropriate technologies with the aim of improving the reusability of both software components and data in its information-processing infrastructure. The Met Office has also recognised that due to the international scope of its operations, it may be necessary to take a leadership role in the renewal of relevant international standards.

A number of specific challenges emerged from the early analysis:

1. How to create data product and service specifications for Met Office implementations, such as internal processing chains,
2. How to take these, and predict, test and drive standardisation of data products and service specifications for international exchange of data, and
3. How to maintain metadata at all stages in the processing chain.

The initial starting position was that the above challenges were essentially unsolved, but that the ISO (19100 series) standards provided a theoretical and governance basis for development of appropriate data standards, and that exploiting a Service Orientated Architecture (SOA) based on the Open Geospatial Consortium (OGC)[2] conceptual formalisms provided a practical implementation route.

The OGC is a member-based global organisation that promotes and provides governance for a range of specifications to support interoperability between systems processing geospatial data. There is a working relationship between OGC and the standards bodies: ISO TC211[3] and CEN[4] (the *de jure* bodies for geospatial data standards internationally and in Europe respectively). Many organisations have reported efficiencies in adopting OGC specifications; indeed adoption of these specifications is recommend by the UK e-GIF[5] and will form the cornerstone of the proposed INSPIRE[6] Directive of the EC.

While the twin drivers of rationalisation and conformity both lead to the desire to implement OGC specifications, any attempt at implementation rapidly demonstrates that there is still a lack of clarity in how the meteorological community can and should make use of these specifications. In particular, a fourth key challenge became clear:

- How to reconcile the "coverage" and "feature" views of the world, and in particular to design processing chains that transform information between these paradigms.

To address these challenges, the Met Office has embarked on a programme of activities to deliver interoperability for geospatial information and services; committed in principle to the standards established and proposed by the OGC. In pursuit of this goal, the Met Office sponsored a workshop to:

[1] http://www.metoffice.gov.uk
[2] http://www.opengeospatial.org
[3] http://www.isotc211.org/
[4] http://www.cenorm.be/
[5] http://www.govtalk.gov.uk
[6] http://inspire.jrc.it/

- Establish a 'straw-man' application schema of the Geography Markup Language (GML) for a representative subset of information artefacts used in the operational meteorology domain in such a way that the schema is consistent with wider environmental science initiatives
- Establish a process and methodology for further development of the application schema
- Establish an implementation path to deliver a framework that enables interoperability; focusing on registries / catalogues and services

Attendees represented a number of international organisations (WMO[7], IHO[8], UNESCO/IOC[9], IUGS/CGI[10]), European / UK activities (INSPIRE, MOTIIVE[11], NERC DataGrid[12]), and included OGC specification editors, domain experts, technology experts and software vendors. The document concludes with a list of recommendations and workshop attendees.

This communiqué summarises the output from the workshop, categorised in two main areas:

- **GML Application Schema Development**. We considered the standards used to model geospatial data. Specifically we looked at the tools and standards that can be used to develop data models and how these can be realised in 'application schema' to enable data models to be incorporated into processing systems.
- **Achieving Interoperability**. We considered the methods and approaches to making use of application schema in processing systems. We looked at 'what can be done' within the existing standards landscape, and what changes are required to improve this for the meteorological community.

In the rest of this document we make use of terminology that has a precise meaning within the domain of geospatial modelling. Many of these terms are formally defined in the ISO TC211 series of standards for geographic information, and some are catalogued in the Terminology Repository[13] prototyped under ISO 19104.

GML Application Schema Development

The application schema development process extended two "best practice" streams of work:

1. The formalism of UML modelling rules and automated schema development, pioneered by the US Federal Geographic Committee through its Framework Data model and OGC activities. (Adopted by INSPIRE and Australian Harmonised Data Framework programmes)
2. Work on modularisation of data models to support complex observational data and implications for international standardisation, begun within the SEEGrid[14] and NERC DataGrid (NDG) communities.

Other related activities include MarineXML[15], MOTIIVE and the AUKEGGS[16] collaboration. Within the context of this workshop, the relationships between these loosely affiliated activities were explored and, where possible, the activities were harmonised. Worked examples of the GML Application Schemas developed during the workshop will be available in due course[17].

Process and Methodology

FeatureType definitions

ISO 19101 states that a 'feature' is "an abstraction of a real-world phenomena". ISO 19109 extends the definition of features to "their representation in data structures". These definitions may be widely interpreted and lack clarity. The workshop agreed the following rule of thumb "If something has a specific name (or classifier) in a domain of interest, it's probably a candidate feature type". Furthermore, we asserted that all feature instances must have an explicit identifier; i.e. each object can be individually named.

[7] http://www.wmo.int
[8] http://www.iho.shom.fr/
[9] http://ioc.unesco.org/iocweb/index.php
[10] http://www.cgi-iugs.org
[11] https://www.seegrid.csiro.au/twiki/bin/view/Marineweb/MOTIIVE
[12] http://ndg.nerc.ac.uk/
[13] http://www.standardsinaction.org/tc211terms/
[14] https://www.seegrid.csiro.au/twiki/bin/view/Main/WebHome
[15] http://www.marinexml.net
[16] https://www.seegrid.csiro.au/twiki/bin/view/AUKEGGS/WebHome
[17] https://www.seegrid.csiro.au/twiki/bin/view/AUKEGGS/MetOfficeOperationalMeteorologyWorkshopExamples

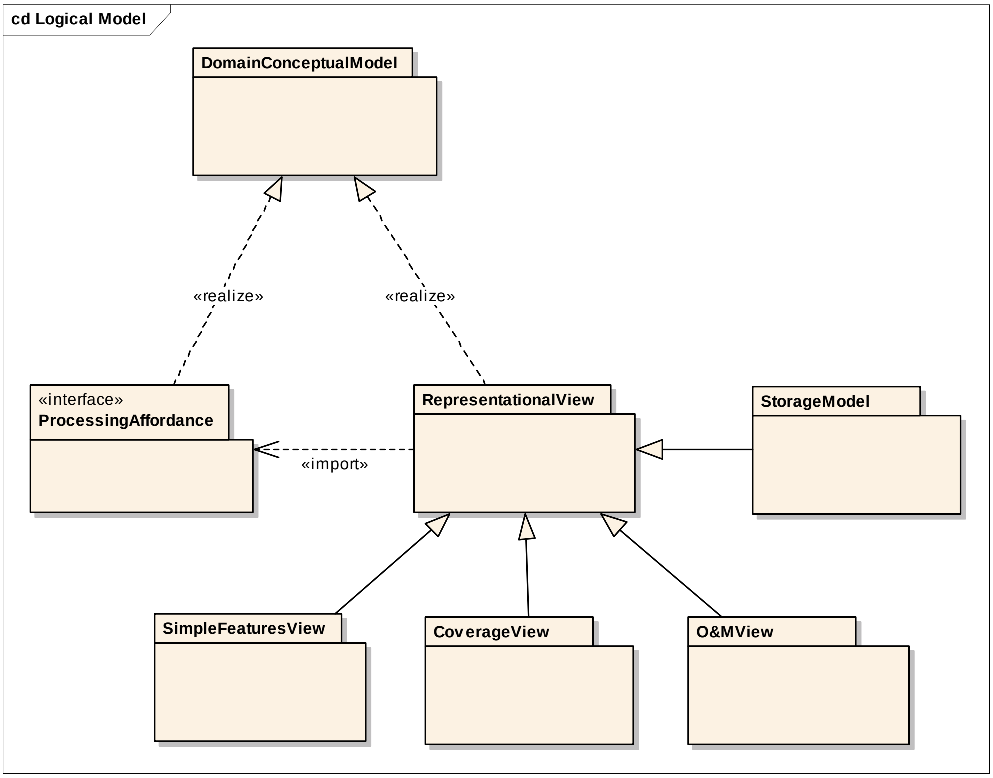

Figure 1: Conceptual framework - representational views and processing affordances

However, even within these guidelines, there is significant scope for variation in how the feature is defined. Evidence suggests that feature types are developed to meet specific local needs. As these needs change, the feature type is redefined to suit. However, all of these feature types may lack coherence and consistency with each other if they have been developed independently. We concluded that:

- A feature type is an *simplified* view of some 'conceptual model' of the real world (see **Figure 1**)
- The conceptual model defines a single, coherent model from which feature types can be derived. The complete conceptual model may not be implementable
- Each feature type is an implementable representation of a subset of the conceptual model, and may be based on: geometry / topology, semantics / governing equations, governance, sampling regime etc.
- In simple cases, the conceptual model is identical to the implementable representation
- The format in which the information is stored is merely another representation of the conceptual model
- It is possible to have a comprehensive conceptual model to guide interpretation and design of representations, and to only implement locally relevant subsets

This 'separation of concerns' results in the ability to create a suite of consistent Feature Types that can be defined in response to a specific requirement (use case) and improved governance. In general, it will be possible to map data authoritatively from one set of feature types derived from a common conceptual model to other Feature Types, recognising that the mapping may not be complete or lossless. Nonetheless, this will achieve semantic interoperability at the level required for real usage.

A picture is worth a thousand words

Or in this case: "A UML diagram is worth a thousand (pages of) XML schema!" The workshop recognised that the requirements of ISO[18] that application schema development should be driven from UML model, coupled with work undertaken by the SEEGrid community[19], yielded significant benefits both in easing application development and interoperability. However, these benefits depend on two tools:

- The canonical UML template including standard components for the development of an application schema conformant to the standardised ISO 19136 / GML 3.2 encoding rules provided by SEEGrid's "Hollow World" model[20], and
- The "Shape Change" tool[21] which achieves the 'mechanical' conversion of the UML model into XML schema. (While this tooling would benefit from some refactoring to simplify usage, real gains would also relate to modularising the configuration file, thus enabling the importation of externally governed schemas.)

Donor Application Schemas

Application Schemas provide a mechanism to realise Feature Types as objects that can be used within software. The workshop investigated whether existing application schemas (e.g. the Climate Science Modelling Language[22], CSML, of NDG, and the Observations and Measurements model [23], O&M) could be used for operational meteorology.

It quickly became clear that these application schemas encapsulated mutually compatible meta-models, but because they were structured around different aspects of the same problem, looked very different. These meta-models suit different parts of the domain: meteorological 'observations' naturally fit the O&M model, whilst most numerical simulation 'coverages' are well described by CSML. However, neither cover what the Met Office categorise as 'Sensible Weather Objects' (e.g. front, clear-air turbulence zone, lightning flash, jet stream) which are normally some discrete feature 'discovered' via an analysis of a coverage or derived from a 'raw' observation, and these are best mapped directly to the General Feature Model[24].

The workshop recognised that where the only variation between coverage features was the phenomenon (e.g. temperature, humidity), it suggests that it is most appropriate to soft-type the generic coverage feature rather than create one feature type for each different phenomenon! In general, the soft- vs. hard-typing decision point depends on the conceptual basis for feature type discrimination (i.e. are feature types based on differences in structure or underlying phenomena?).

Reconciling different meta-models

The challenge is that alternative schema do not naturally co-exist since they essentially divide the world up using different aspects of the problem, and thus the same data would map into different constructs depending on which viewpoint you started from. This realisation put the problem into stark focus – the single inheritance model used in current application schema forces the data modeller to choose one aspect of the data model for the feature type hierarchy. This generally makes it difficult to reconcile alternative "donor schemas" within an application schema if the feature types are differentiated on different aspects of the model, even if they both have structural or semantic utility.

The reality is that the same real world object needs to behave differently depending on what operations it is being used for. This pattern is common in information system science (polymorphism, supported by operational interfaces) but is not currently supported by the substitution group mechanism of XML schema, hence the difficulties experienced. The solution proposed, and subsequently tested successfully, was to use polymorphism to clearly separate concerns during modelling, and establish the governance arrangement of each interface definition.

---

[18] See ISO 19103 ("Conceptual schema language") and ISO 19109 ("Rules for application schema")
[19] refer to the SEEGrid twiki for a discussion on their UML profile of GML: https://www.seegrid.csiro.au/twiki/bin/view/Xmml/UmlGml
[20] https://www.seegrid.csiro.au/twiki/bin/view/Xmml/HollowWorld
[21] https://www.seegrid.csiro.au/twiki/bin/view/Xmml/UmlGml#Shape_Change
[22] http://ndg.nerc.ac.uk/csml
[23] https://www.seegrid.csiro.au/twiki/bin/view/Xmml/ObservationsAndMeasurements, OGC document 05-087r3 (pending publication)
[24] ISO 19109 "Rules for Application Schema"

Packaging according to Governance

An organisation creating an application schema will seek to import XML schema definitions for externally governed parts of the information model (e.g. GML, O&M etc). This allows the resulting application schema to define only concepts unique to that organisation's domain (and normally coinciding with the governance mandate for that organisation).

Correct packaging of the UML model (and hence the derived W3C XML schema documents) will be the key to re-use. We decided that the information model should be packaged (in the UML sense) according to responsibility for governance. For example: parts of the application schema pertinent to ALL operational meteorology (and thus to be eventually governed by WMO) should be packaged separately from parts pertinent only to the UK Met Office.

A challenge identified is the case where an external definition is adopted, but there is no published normative model or schema. It is difficult to incorporate such definitions, leading to extra work, scope for semantic inconsistencies and alternative implementations, yet an implementation is required to make progress. We conclude that if an organisation does not intend to claim a role for acting as authority for part of its data model, but is merely providing an implementation of an external definition it should package the model accordingly and make it clear that the external model implementation is an interim solution. Furthermore, it should publicise the local implementation, bringing it to the notice of the body one would hope should govern those interim definitions in the longer term.

Operational interfaces for feature types – the 'processing affordance' pattern

The General Feature Model supports the concept of 'operations'. However, implementation of features using W3C XML Schema only allows definition of a static XML document; defining only the structure and properties. There is no mechanism allowing the description of operations that a feature can invoke / be invoked on it.

The concept of 'processing affordance' is proposed to enable description of operational interfaces. The semantics of processing affordance are equivalent to those of an interface in Object Oriented programming, or 'mix-ins' in Aspect Oriented programming. The 'interface' defines a declaration of intent, describing a series of operations that can be invoked. Where a feature supports the 'interface', it implies that that feature is able to provide sufficient information to execute the operation described in the interface. Or expressed mathematically: the feature has the attributes p,q,r to support $f(p,q,r)$. Furthermore, a feature may be able to support multiple interfaces, thus enabling polymorphism.

While the properties of features can be defined in W3C XML schema, only limited definition of relationships between feature types (and other objects) is allowed (i.e. single inheritance, association). This means the capability to make associations between features and the processing affordance (interface or interfaces) must be provided by other means. One possible methodology is to define the processing affordance as an object in a registry (i.e. a catalogue with appropriate governance) and associate it with feature instances or feature types. However, this is essentially a partial ontology and the effectiveness of its implementation in a registry context needs to be tested.

The impact of this proposal is likely to be an extension to the GML profile of UML, adding an implementation of interface realization. The rules defined in ISO 19118 and ISO 19136 mapping UML to GML will have to be updated with the implication that further changes will need to be applied to the Shape Change UML-GML tool.

We also noted that CSML describes two concepts which may benefit from refactoring under the 'processing affordance' pattern (see Figure 2). First, there are semantics associated with environmental science (e.g. the physical/environmental phenomenon described by the feature). Second, there are packaging strategies for multi-dimensional data. The latter may be modelled as re-usable geometric 'affordances' (e.g. a 'Grid' or 'Profile' geometry are important to identify because they afford different processing operations).

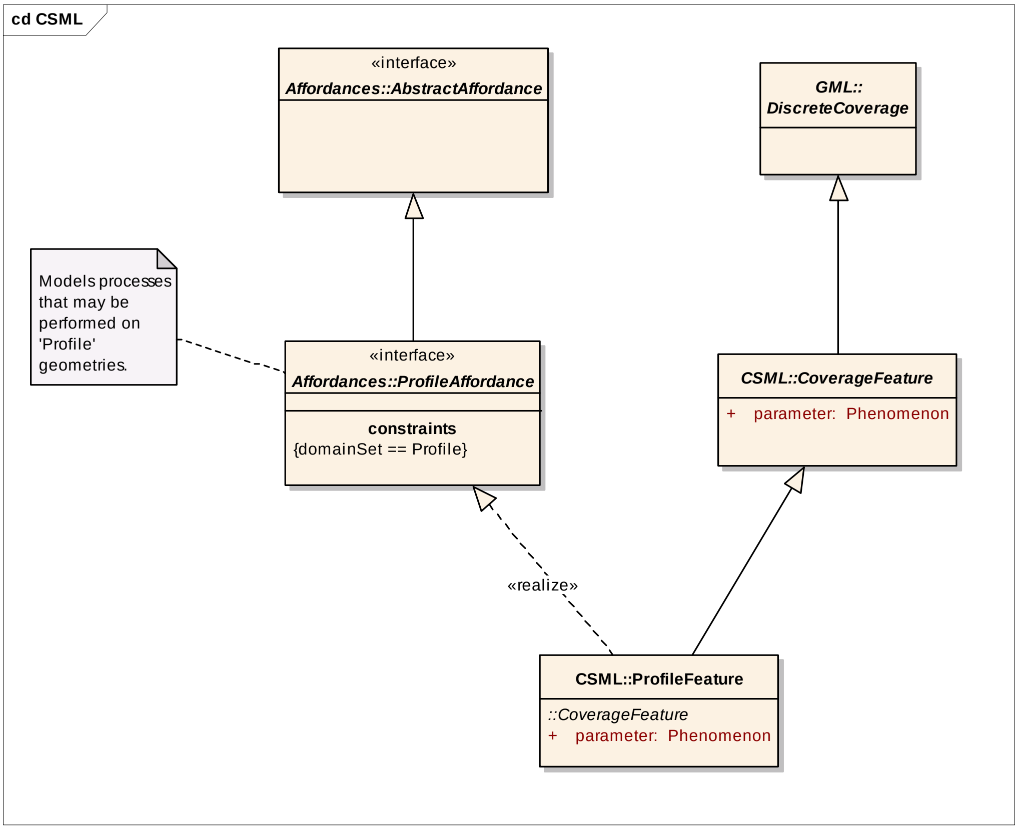

Figure 2: Re-factoring CSML under the 'processing affordance' pattern

Achieving Interoperability

Interoperability cannot be achieved by definition of an application schema alone. A set of well defined service interfaces are vital to ensure that data can be accessed in an implementation agnostic fashion[25]. A key enabler of interoperability is the registry. The registry provides the capability to publish (and govern) application schemas, phenomena dictionaries, controlled vocabularies, service bindings etc. for all to see and use. Furthermore, it is the registry that enables associations between objects to be expressed. This leads to true interoperability; for example, allowing a user to discover an object of interest, browse by navigating associations and execute a chain of processes on some dataset to derive added value.

Catalogues and registries implement both information models and governance structures – registry instances may be factored along governance lines.

Registries and Repositories

ebRIM

The ebXML Registry Information Model (ebRIM) v3.0 (developed under the auspices of OASIS[26]) has been adopted as a general purpose registry model by an overwhelming majority of OGC Catalog service implementers. We reviewed the design criteria and usage of this model within the Oceans Portal design and MOTIIVE Feature Type Catalogue work plans. After some consideration we believe that the ebXML Registry Information Model is the only identified standards-based offering that provides sufficient functionality to deliver the use cases we considered for the operational oceanography and coastal zone management communities, and the operational meteorology domain.

[25] See the ISO definition of interoperability in ISO 19101, section 6.4.1.
[26] http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=regrep, also see the SEEGrid twiki for a useful introduction to ebRIM https://www.seegrid.csiro.au/twiki/bin/view/Compsrvices/EbXMLRim

ebRIM implementation

The EU MOTIIVE and the Australian Oceans Portal project are collaborating to deliver an ebRIM registry/repository implementation, focusing on delivering a feature type catalogue (but service bindings, data standards driven query models, presentation resources and processing chains will also be exposed within the registry).

The Met Office has proposed to initiate a parallel track to develop a second reference implementation based on the OGC Catalogue Services for Web (CSW) for (at least) the same set of use cases. Additionally, deployment of phenomenon dictionaries may also be investigated. This activity is seen as a key strand for the Met Office's geospatial interoperability programme and of great importance within the wider meteorological community and WMO.

As a result of multiple reference implementation tracks, a number of interoperability tests across MOTIIVE/ Oceans Portal, MarineXML and Met Office registries could be scheduled for late 2006, perhaps forming the basis of an OGC interoperability experiment.

Service Implementations

The OGC Web Service specifications provide a basis for deploying a functional 'Spatial Data Infrastructure' (SDI). However, it is clear that the communities represented at this workshop require a number of changes to the specifications before they can provide all the capabilities required.

Data access query model (OGC Change Request: 05-022r1)

We assert that it is necessary to provide guidance on the types of queries that are supported by a data service implementing a WFS interface. This guidance binds appropriate vocabularies to the instances of feature types accessible via the service. It also allows for certain queries or processing functions to be optimised or scripted to access data using a specific set of parameters.

The Data Access Query Model (DAQM) builds on the OGC Filter Specification and allows the definition of pre-defined queries. The corollary is that a service may explicitly announce that it rejects queries that do not follow the advertised query model. Furthermore, if desired it allows a query based on feature type A to deliver a response containing feature type B.

At some point in the future, DAQM syntax may be standardised, however the initial requirement is to ensure that such considerations can be built into service design. Furthermore, DAQM may provide a mechanism to implement operational interfaces (refer to the 'processing affordance' pattern). However, this assertion needs to be robustly tested before this can be adopted as best practice.

Service coherence model

We note the intent of the OGC Service interface specifications to support data access on the basis of Feature Types. We also note that the document OGC 05-008 (OGC Web Services Common Specification) does not address the relationships between the interfaces. We assert that there is utility in formally modelling the meta-classes associated with OGC service interfaces to show relationships between Feature Type (definition) instances, Feature instances, Filter artefacts, Styled Layer Descriptors, Web Processing etc.

Such a "service coherence model" can be used to explicitly and consistently define how new services relate to existing services and hence what sort of processing chains are possible. This model will form the basis of designs to propagate metadata effectively through processing chains, and should be part of the OGC Reference Model, preferably extending or complementing OGC 05-008.

With such a model in place, it will become easy to understand the relationships between "convenience APIs" such as Web Map Service, Web Coverage Service, Sensor Observation Service, WPS etc and general feature access (WFS), and thus to improve and maintain consistency between semantics of such services.

Referencing compact (binary) encodings from GML

An instance of a GML coverage feature is likely to contain both domain and range descriptions. These may both be encoded as XML elements. Even allowing for the use of XML arrays to reduce the bulk, the range set (and potentially the domain description) for many coverages is simply too large to manage within the XML document.

We believe that the 'metadata' for the coverage feature should be described using GML (including the domain description where possible to allow for spatial indexing). For most coverages, the range set should be referenced from the GML document using xlink:href to point to a URI where the range set can be found (e.g. a Web Coverage Service, JPIP[27] stream, relational database etc.). xlink:role should be used to guide the interpretation of content found at the URI, enabling a convenient description of the range set encoding. The content of the range set therein can be logically inserted into the GML document. In the case of complex geometries (e.g. numerical simulations on curvilinear grids), the domain characterisation can be as large as the range, and may benefit also from the xlink 'by-reference' pattern.

The 'deferred binding' semantics of xlink:href imply that it is up to the GML parser to decide how to deal with the content. This ensures that only parsers capable of handling the encoded range set (or understanding how to access the content if a URN is used) are required to interpret the content.

The GML 'content substitution' pattern enabled through xlink may find general utility for integrating legacy data, and is applicable not only to binary file-based data, but also relational.

Coverage Services within the environmental science community

GALEON IE[28] is an OGC interoperability experiment supporting open access to atmospheric and oceanographic modelling and simulation outputs. The issues identified as a result of that experiment are also relevant to the meteorological community. We support and endorse the recommendations made within the report from Woolf, Feb 2006[29] regarding changes to OGC / TC 211 specifications. Of particular note are: (i) modifications to ISO 19111 to support spatio-parametric coordination reference systems (e.g. pressure-based vertical coordinates), (ii) adoption of CSML schemas for irregular rectified grids, and (iii) WCS to recognise CF-netCDF as a supported format.

Summary of Workshop Recommendations

1. Formal UML modelling rules and automated xml schema generation form best practice for application schema development.
   a. Some effort in refactoring and modularising the Shape Change tool would have real benefit.
2. Information models informing application schema should be packaged (in the UML sense) according to governance, and where it is necessary to implement a package that ought to be in another domain, the external body responsible for that domain should be notified, and the package clearly marked as an interim solution.
3. With respect to features, it is recommend that:
   a. If something has a specific name (or classifier) in a domain of interest, it's probably a candidate feature.
   b. Feature types should be implementable representations of subsets of complete conceptual feature models (whether or not the conceptual model is implementable).
   c. All feature instances must have an explicit identifier; i.e. each object can be individually named.
   d. When only the phenomenon differs between coverage features (e.g. temperature, humidity), it is suggested that it is most appropriate to soft-type the generic coverage feature rather than create multiple feature types to cover multiple phenomena.
4. The concept of 'processing affordance' enables the description of operational interfaces for features. Interfaces should define a declaration of intent, describing a series of operations that can be invoked, and features should be clearly associated with these interfaces. Where a feature supports the 'interface', it implies that that feature is able to provide sufficient information to execute the operation described in the interface.
   a. This proposal requires an extension of the GML profile of UML to implement interface realization.
   b. Utility would be provided if CSML could be refactored to provide geometric affordances.

[27] http://www.jpeg.org/jpeg2000/j2kpart9.html
[28] http://www.opengeospatial.org/initiatives/?iid=173
[29] http://galeon-wcs.jot.com/WikiHome/Implementation+Progress+Page/GALEON-NERC-report.doc

5. The single-inheritance model used in current application schema limits their reuse, even in situations where multiple inheritance would have structural or semantic utility. It is suggested that polymorphism should be used to clearly separate concerns, allowing multiple inheritance of relevant feature characteristics and resultant interfaces. Implementation of polymorphism in this context remains unresolved.
6. Because XML schema limits relationships to single inheritance and associations, necessary relationships must be provided by other means. One possible methodology is to define processing affordance as an object in a registry and associate it with feature instances or feature types. This may enable multiple inheritance relationships (i.e. polymorphism) to be expressed.
7. Registries are key to providing interoperability in an implementation agnostic fashion. At the time of writing the ebXML Registry Information Model (ebRIM) is the only standards-based offering that provides sufficient functionality for the use cases considered.
   a. The Met Office will develop a reference implementation based on the OGC catalogue services for the web (CSW) for the same set of use cases as the MOTIIVE and Australian Oceans Portal project, and investigate the deployment of phenomenon dictionaries. This work will be done with a view to future interoperability experiments.
8. For efficiency and in order to deploy interfaces which support the processing affordance concept, it will be necessary to provide guidance on the types of queries that are supported by a data service implementing a WFS interface. Such guidance could be provided by the Data Access Query Model (OGC Change Request 05-022r1).
9. Significant utility could be gained by formally modelling the metaclasses associated with OGC service interfaces to show relationships between the objects exposed and the various interface types (e.g. WMS, WCS etc).
10. In some cases volumes of data associated with domain and range sets of coverages may be too large to handle within GML. In such cases the ‘metadata’ should remain in GML, but xlink:href should be used to point at a URI where the range set can be found and xlink:role should be used to guide the interpretation of the content found at the URI. If necessary, the domain should also be characterised in the same way.
11. We support the recommendations of Woolf 2006 as part of the GALEON interoperability experiment that;
    a. Modifications to ISO19111 are required to support relevant spatio-parametric coordinate systems.
    b. The CSML application schema is used to provide support for irregular rectified grids.
    c. The OGC Web Coverage Service should support CF compliant NetCDF as a supported format.

Workshop Attendees

Name	Organisation	Affiliation(s)
Rob Atkinson	Social Change Online	SEEGrid, MOTIIVE
Dave Burggraf	Galdos	
Simon Cox	CSIRO	SEEGrid, IUGS/CGI
Adam Flaherty	UK Met Office	
Roy Lowry	British Oceanographic Data Centre	IOC, NERC Data Grid, MarineXML
Bryan Lawrence (chair)	NCAS/British Atmospheric Data Centre	NERC Data Grid
Graham Mallin	UK Met Office	
Keiran Millard	HR Wallingford	MarineXML, MOTIIVE
Peter Parslow	UK Hydrographic Office	IHO, MOTIIVE
Andrew Robson	Social Change Online	MOTIIVE
Gil Ross	UK Met Office	WMO, INSPIRE
John Swift	UK Met Office	
Jeremy Tandy (editor)	UK Met Office	WMO, SIMDAT
Peter Trevelyan	UK Met Office	
Andrew Woolf	CCLRC e-science	NERC Data Grid, INSPIRE, MOTIIVE
Bruce Wright	UK Met Office	