# Automating geospatial metadata using ESRI's ArcGIS and Microsoft's .NET

James K Batcheller, Robert I Dunfey, Femke Reitsma and Bruce M Gittings
Edinburgh Earth Observatory - School of GeoSciences - University of Edinburgh
Drummond Street - Edinburgh EH8 9XP - Scotland, U.K
Tel. + 44 131 650 2565 Fax + 44 131 650 2524
{jk.batch|femke.reitsma|bruce}@ed.ac.uk

## INTRODUCTION

Geospatial metadata has long played an important role in the management and location of geospatial datasets (Kim, 1999; Tsou, 2002; Limbach et al., 2004). Often employed by institutions to organise, maintain and document their geographic resources internally, metadata may also provide a vehicle for exposing marketable data assets externally when contributed to on-line geospatial exchange initiatives such as the UK's public sector metadata service *Gigateway* (Batcheller and Gittings, 2006). In spite of the numerous benefits afforded, obstacles to the production of such metadata are numerous (Mathys 2004). Perceptions of it being a tedious yet arduous task, coupled with an assignment of low priority even where the advantages are appreciated all too often result in what may be referred to as the *metadata bottleneck* (Liddy *et al.*, 2002). The current work proposes an approach aimed at reducing the effort associated with geospatial metadata generation through the customisation of a proprietary GIS. By coupling data preparation, management and documentation approaches with such a bespoke application, it is intended to mitigate impediments to geospatial metadata generation whilst promoting a system of data administration that safeguards the data it supports. Geospatial metadata has long been advocated to facilitate the management of data collections; the current approach takes this one step further, using metadata standard elements to coordinate data filing and in the process, contribute to metadata production.

## APPROACH

The prototype was designed to integrate a systematic data management model with data initialisation and documentation processes, the aim being to conflate the component workflows whilst facilitating the automatic creation of appropriate metadata. Developing the tool within an existing GIS suite complete with metadata support offers a means by which data creation and editing can be bound more closely to that of its metadata, mitigating the data – metadata disconnect and minimising the risk of inconsistency. ESRI's ArcGIS was chosen due to its extensible ArcObjects-based architecture of modular programming components with which software can be rapidly deployed. Further, by providing a "framework for the implementation of a custom metadata environment" (Vermeij, 2001), its ArcCatalog application offers an extensive pre-existing toolkit with which to develop. The platform for development used was Microsoft's .NET, chosen both for its support for solution extensibility and in its tight integration with XML technologies (Stephens and Hochgurtel, 2002). The personal geodatabase was selected as the test data storage model due its positioning between (legacy) hybrid single-user file-based data stores and integrated multi-user database strategies (Batcheller *et al.*, 2007). A Qualified Dublin Core profile with geospatial refinements was also defined, providing a concise set of twenty-three elements against which the prototype could be evaluated (Table 1).

| Core Element Name | Element refinement | Description |
|---|---|---|
| Title | - | Title |
| | Alternative | Alternative title |
| Description | Abstract | A brief narrative summary of the dataset |
| Language | | Language |
| Subject | Keywords | Main dataset theme(s) |
| Date | Created | Date of creation |
| | Modified | Last date of update |
| | Period.name | Name of a specific interval. Used here to define frequency of dataset update |
| Creator | | Originating person / organisation |
| Publisher | | Distributing person / organisation |
| Contributor | | Contributing person / organisation |
| Format | | Digital manifestation of resource |
| Type | Dataset | Nature of content |
| Rights | Access Rights | Access restrictions |
| Coverage | Spatial.Box.name | Name of geographic extent of dataset |
| | Spatial.Box.projection | Spatial reference system of dataset |
| | Spatial.Box.northlimit | Limits of dataset extent in coordinates |
| | Spatial.Box.eastlimit | |
| | Spatial.Box.southlimit | |
| | Spatial.Box.westlimit | |
| Identifier | | Online linkage to dataset |
| Relation | | A reference to a related resource |
| Source | | A reference to a resource from which the present resource is derived. |

***Table 1.*** Qualified Dublin Core element set used to document the test dataset and evaluate the metadata tool. Fifteen core elements are qualified by an additional eight element refinements, providing twenty three fields in total.

## TOOL EXECUTION

On selecting a dataset within ArcCatalog the prototype is initiated via a standard button interface, presenting the user with a form on which metadata elements may be edited and which also functions as the principal mechanism through which the utility is controlled. Any pre-existing metadata items held with the dataset are immediately collected on form load; elements may be manually edited, added if empty or selected for overwrite using the prototype’s routines. In addition, all operations may be selected to run simultaneously, individually or in various combinations, allowing full control over what routines are executed. The operations the prototype performs are illustrated in Figure 1.

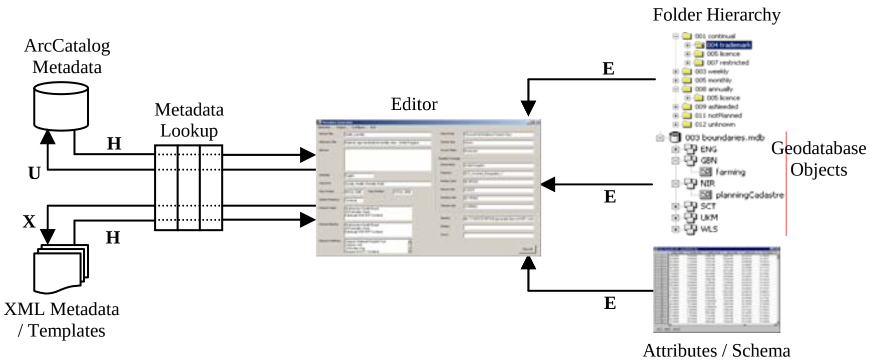


***Figure 1.*** Flow diagram of the operations performed by the current metadata tool, generating elements from a number of separate sources for review on the main editing form.
H: Harvesting; E: Extraction; X: Export; U: Update.

## ROUTINES

### Element Harvesting

Harvesting routines are run using XPath expressions, defining from whence to retrieve pre-formed metadata from both internal, ArcCatalog-specific XML (stored alongside the dataset in question) as well as from external user-defined XML templates . In the case of the latter, system variables read from the underlying operating system (such as workstation domain and username) can be used to determine the appropriate templates to query. XPath expressions are encoded in a lookup table interpreted by the tool and which may be readily adapted to a variety of metadata conventions as well as used as a fundamental crosswalk for metadata output.

### Element Extraction

Custom routines are used to extract further information from the dataset, its data content as well as the dataset's location within a refined folder hierarchy.

Folder & Geodatabase Hierarchy

Metadata entities are used to organise the very data they describe, providing a nomenclature with which datasets may be tagged, categorised and stored. Personal geodatabases, their contents and the folders in which they are held are labelled according to appropriate metadata vocabulary terms by which they may be unambiguously characterised (Table 2), facilitating the logical, hierarchical management of data stores whilst contributing towards the automated compilation of their corresponding metadata records. The hierarchy initially employed by the prototype is illustrated in Figure 2.

| Container | Name | ISO Code List |
|---|---|---|
| Primary tier | Date Period | 19115:MD_MaintenanceFrequencyCode |
| Secondary tier | Access Rights | 19115:MD_RestrictionCode |
| Personal geodatabase | Subject Keyword | 19115:MD_TopicCategoryCode |
| Feature dataset | Coverage Spatial Box Name | 3166-2 |
| Feature class | Subject Keyword | 19115:MD_TopicCategoryCode |

***Table 2.*** Prototype folder hierarchy in which datasets are tagged and filed, employing entities from code lists (vocabularies) of commonly used ISO standards.

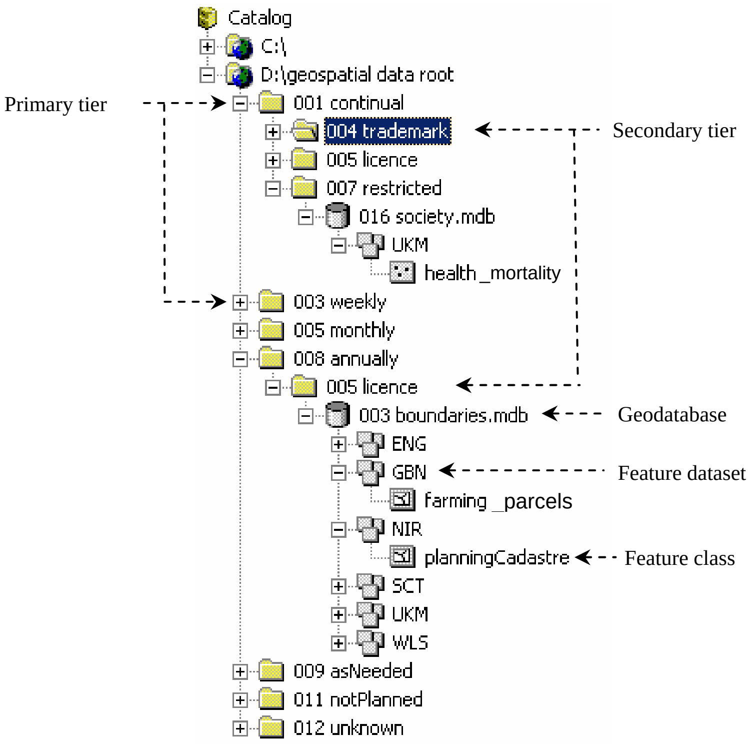
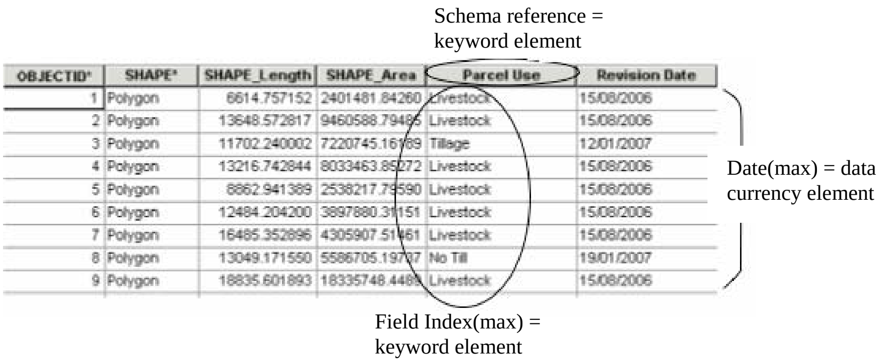


***Figure 2.*** Illustration of current data storage hierarchy which yields five metadata values. Container tags are based on specific metadata entity terms provided above in Table 2.

Data and Dataset

Additional dataset properties not formally treated as items of metadata within ArcCatalog but which are nevertheless programmatically accessible may also be extracted. The current Dublin Core profile's Alternative Title is thus generated; other exploitable properties include spatial resolution and certain elevation values.

Feature attribute instances and attribute schemas may also be leveraged to contribute towards metadata production. Providing predictable feature catalogue-based schemas[1] are adhered to, metadata items may be extracted through the use of indexing techniques, functions performed against the attribute values of a specific field or by direct referencing of feature type definitions (Figure 3).



Schema reference = keyword element

| OBJECTID* | SHAPE* | SHAPE_Length | SHAPE_Area | Parcel Use | Revision Date |
|---|---|---|---|---|---|
| 1 | Polygon | 6614.757152 | 2401481.84260 | Livestock | 15/08/2006 |
| 2 | Polygon | 13648.572817 | 9460588.79485 | Livestock | 15/08/2006 |
| 3 | Polygon | 11702.240002 | 7220745.16189 | Tillage | 12/01/2007 |
| 4 | Polygon | 13216.742844 | 8033463.85272 | Livestock | 15/08/2006 |
| 5 | Polygon | 6862.941389 | 2538217.79590 | Livestock | 15/08/2006 |
| 6 | Polygon | 12484.204200 | 3897880.31151 | Livestock | 15/08/2006 |
| 7 | Polygon | 16485.352896 | 4305907.51461 | Livestock | 15/08/2006 |
| 8 | Polygon | 13049.171550 | 5586705.19737 | No Till | 19/01/2007 |
| 9 | Polygon | 18835.601893 | 18335748.4485 | Livestock | 15/08/2006 |

Date(max) = data currency element

Field Index(max) = keyword element

***Figure 3.*** Elements are extracted by referencing a predictable attribute schema; specific attribute fields may yield elements via indexing or custom functions.

## EDITING, OUTPUT AND VALIDATION

The tool centres on a form interface through which routines are initiated and metadata items are edited. Extraction and harvesting routines can either be performed automatically or individually executed once the form has loaded; similarly, elements can be interactively deselected to prevent being overwritten. On form completion, records may be output to ArcCatalog-native format for storage alongside the data, as well as exported to XML files conforming to those standards depicted in the metadata crosswalk file. Further validation routines may be incorporated via on-form spell-checking and domain lookups or may involve more stringent XML schema-based validation supported by the Microsoft XML Core Services which accompany the .NET platform.

## RESULTS

Of the total twenty three metadata standard elements outlined above, twenty were completed using the proposed approach (Table 3); the compound element “keyword” comprising of four sub-elements retrieved through the various extraction methods.

[1] The ISO 19109:2005 Geographic Information – Rules for Application Schema standard for example permits the definition of conceptual data models to define the logical structure of a particular application’s data, commonly instantiated using feature catalogues that define permissable feature types

| Routine | Element (abridged) |
|---|---|
| Harvesting - pre-formed metadata | Title; Language; Date Created; Format; Dataset Type; Projection; Spatial Box Coordinates; Identifier – 11 total |
| Harvesting – external templates | Creator; Publisher; Contributor – 3 total |
| Element Extraction – hierarchy | Date Period; Access Rights; Spatial Box Name; Keyword (x2); - 3 ½ total |
| Element Extraction – Dataset | Alternative Title – 1 total |
| Element Extraction – Data | Date Modified; Keyword (x2) 1 ½ total |

***Table 3.*** Breakdown of metadata items retrieved and the corresponding routines used. Qualified Dublin Core elements not completed include Abstract, Relation and Source.

## DISCUSSION

It could be argued that what is presented here is not so much the automatic generation of metadata but the transfer of effort from metadata authoring to data preparation and management. While this is certainly, but not exclusively, the case, it is put forward as a sound model for metadata management as it promotes a considered approach to data storage as well as sound data preparation. Furthermore, it enables the release of authoring resources which may be redirected towards more intellectually challenging metadata tasks such as descriptive metadata creation and quality control – a conspicuous advantages in cases where data documenters and data authors or managers are distinct. It can also serve to safeguard metadata quality – contingent on appropriate dataset categorisation and data preparation – as the majority of elements are no longer entered manually and susceptible to human error. And while the data storage strategy proposed herein may be quite reasonably viewed as contrived; the opinion held here is that data management, by definition, should adhere to a predictable, formal schema to best allow data categorisation and subsequent retrieval. In all, it is contended that the current approach has the potential to offer a significant net saving of time for applications reliant on the production of metadata, despite the potentially high initial investment.

## CONCLUSIONS AND OUTLOOK

The metadata management framework outlined above supports users in reducing the effort involved in documenting data, ensuring that a minimum amount of elements are automatically generated according to relevant metadata standards and best practice. Considering the parallels with the digital library and internet cataloguing arenas, where resource volumes make it “unrealistic to depend on traditional humanly-generated metadata approaches” (Greenberg *et al.*, 2006 p3), efforts to streamline metadata creation though automation begin to take on more importance. The future of generating useful metadata involves increasing computational support to minimise human effort; advances in representing the semantics of metadata may well have particular relevance for automating its collection and exploitation. In conclusion, while the approach presented was one bound to a particular proprietary solution, the objective was not to laud one offering above all others but to highlight the potential contribution a dataset’s ambient computing infrastructure can make in automating the creation of geospatial metadata.

## BIBLIOGRAPHY

Batcheller, J. K. and Gittings, B. M., 2006. Avenues for developing the UK's National Geospatial Metadata Service. Proceedings of the GIS Research UK 14th Annual Conference, University of Nottingham, Nottingham, UK, pp.259-262.

Batcheller, J. K., Gittings, B. M. and Dowers, S., 2007. The Performance of Vector Oriented Data Storage Structures in ESRI's ArcGIS. Transactions in GIS 11(1): 47-65.

Greenberg, J., Spurgin, K. and Crystal, A., 2006. Functionalities for automatic metadata generation applications: a survey of metadata experts' opinions. International Journal of Metadata, Semantics and Ontologies 1(1): 3-20.

Kim, T. J., 1999. Metadata for geo-spatial data sharing: A comparative analysis. The Annals of Regional Science 33: 171-181.

Liddy, E.D., Allen, E., Harwell, S., Corieri, S., Yilmazel, O., Ozgencil, N.E., Diekema, A., McCracken, N.J., Silverstein, J. and Sutton, S.A., 2002. Automatic metadata generation and evaluation. Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Tampere, Finland, pp.401–402.

Limbach, T., Krawczyk, A. and Surowiec, G., 2004. Metadata Lifecycle Management with GIS Context. 10th EC GI & GIS Workshop, ESDI State of the Art, Warsaw, Poland.

Mathys, T. (2004). The Go-Geo! Portal Metadata Initiatives. In: Proceedings of the GIS Research UK 12th Annual Conference, University of East Anglia, Norwich, UK, pp.148-154

Stephens, R., Hochgurtel B., 2002 Visual Basic .NET and XML. John Wiley & Sons Inc, New York. ISBN 047112060X

Tsou, M.-H., 2002. An Operational Metadata Framework for Searching, Indexing, and Retrieving Distributed Geographic Information Services on the Internet. In: Egenhofer, M. and Mark, D. (Eds.) Lecture Notes in Computer Science Vol. 2478, Springer-Verlag, Berlin, pp. 313-332.

Vermeij, B., 2001 Implementing European Metadata Using ArcCatalog - ArcUser Online. http://www.esri.com/news/arcuser/0701/metadata.html. Last accessed: 17th January 2007.