Metia — A Generalized Metadata Driven Framework for the Management and Distribution of Electronic Media

"The synthesis of metadata and media…"

Patrick Stickler
Nokia Research Center, Software Technology Laboratory, Agent Technology Group
Visiokatu 1, FIN 33720 Tampere, Finland
patrick.stickler@nokia.com

Abstract

The Metia Framework defines a set of standard, open and portable models, interfaces, and protocols facilitating the construction of tools and environments optimized for the management, referencing, distribution, storage, and retrieval of electronic media; as well as a set of core software components (agents) providing functions and services relating to archival, versioning, access control, search, retrieval, conversion, navigation, and metadata management.

1 Introduction

Nokia has for many years faced the challenge of producing, managing, and distributing large volumes of customer documentation for a broad range of products. Currently maintained and supported documentation numbers in the millions of pages. Early adoption of SGML[1] and the development of a well defined publishing process provided good solutions for the core business needs; however, as the market shifted more and more towards internet based solutions, there was a proportionate increase in the need to support additional media formats, smaller documentation sets, globally distributed systems, and faster product turn-around cycles; making the present documentation production infrastructure unwieldy and limited in its ability to address these new requirements. More so than simply an overhaul or re-implementation of the current systems and processes, a change in philosophy and methodology was needed in order to meet these new challenges.

Several characteristics and limitations of the previous processes and systems were identified which were seen as the primary obstacles to meeting the needs of documentation production and management for the next generation of products, the most notable being:

- Monolithic, closed systems which are difficult and expensive to extend and maintain.
- Proprietary solutions, often resulting in "bondage" to particular vendors and thus constituting higher risk, depending on the reliability and longevity of the vendor, and

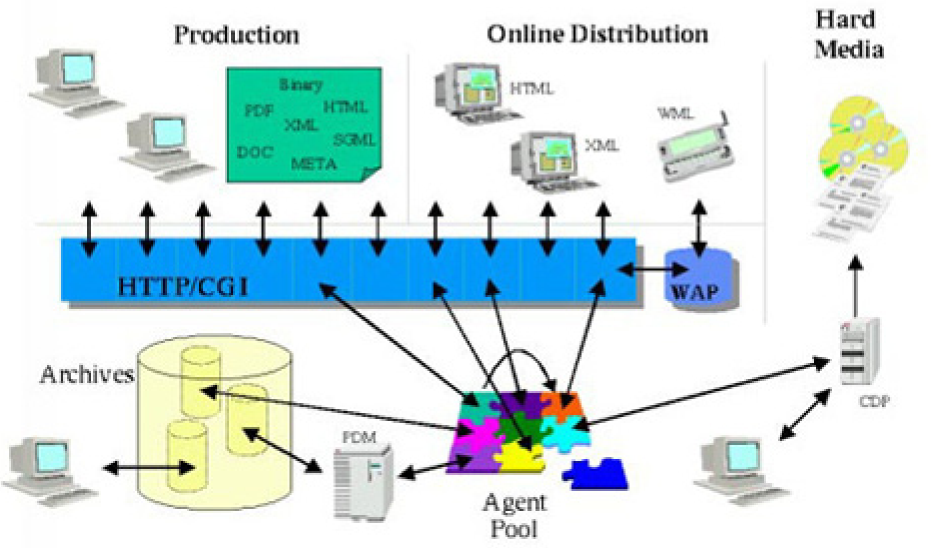

Figure 1: Metia Framework Architecture

oftentimes making integration with other systems difficult or even infeasible.

- Platform specific solutions, limiting portability and easy deployment across multiple sites globally.
- Proprietary encodings which limited tools selection and sometimes choice of distribution media.
- Complex systems with high overhead of installation and support, making some solutions unsuitable and not cost effective for smaller product lines; which resulted in competing solutions which were incompatible, and much wasted effort due to overlap of functionality and redundancy in processes and tools.
- Proprietary tools based solutions rather than generic methodologies and standards based solutions, which limited flexibility and the timely adoption of new technologies into the production and distribution processes.

In addition to the above shortcomings, the nature of documentation production and their distribution was changing, and Nokia was faced with new requirements, such as

- Increasing need for multi-media and single-source to multi-target distribution which requires a more flexible, modular, and scalable system where new media types and new applications of existing content could be easily added while having little to no impact on existing processes and applications.
- Increasing need for distributed, global solutions which both could be deployed identically yet independently in many sites around the world as well as mechanisms for global synchronization and collaboration between systems and services across sites globally.
- Increasing need for scalable solutions which would work easily well for small product lines with less documentation content as well as for large product lines with large volumes of documentation content; with installation, maintenance, and training overhead proportionate to the needs of each installation.
- A common, consistent environment for locating, browsing and viewing electronically encoded information irrespective of media format characteristics.

In addition to several other initiatives, including the adoption of XML[2], the development of a modular documentation process for content reuse, and a general commitment to open standards and tools wherever possible, the Metia Framework was developed to guide and facilitate a broad range of solutions which avoid the shortcomings of previous systems and processes as well as meet the needs and challenges of the changing marketplace and the next generation of Nokia customer documentation.

2 Framework Overview

The Metia Framework serves as the foundation for the realization of the Nokia Customer Documentation strategy, upon which company wide tools and services operate. The Metia Framework addresses the common requirements of all Nokia business units, while also allowing custom extensibility by specific business units for special needs.

The Metia Framework is specifically designed to embody the following qualities and characteristics:

open The framework is based on open standards and proven technologies wherever possible, and all framework specific properties and characteristics are fully documented.

scalable Environments based on the framework should function equally well with both few and many agents, on a single machine or across a distributed network, and on both small and large systems; where performance issues are primarily tied to the properties and capabilities of the individual agents and/or systems and network bandwidth, and not to properties of the framework itself.

modular All agents within a given environment interact efficiently and effectively with one another with little to no specialized configuration and with no special knowledge of the implementation details of particular agents.

portable Agents conforming to the framework can be implemented on a broad range of platforms using practically any tools, programming languages, or other means. The core software components provided by the framework itself are implemented in Java[3], providing maximal portability to different platforms and environments.

distributed Agents are not limited to data or the services of other agents running on the same machine, but may interact (often transparently) with agents running on any machine which is accessible over the network.

reusable The framework provides for maximal use and reuse of existing software components and agents, where more complex agents are implemented using the services of more specialized agents. This allows refinement and extension of processes with little to no modification to any existing implementation.

extensible Additional agents may be added to any environment based on the framework with little to no impact to and/or reconfiguration of any existing agents.

The Metia Framework architecture is based on a standard HTTP[4] web server (see figure 1 above).

One of the goals of the framework is to be media neutral, such that the particular encoding of any data is not relevant to storage by or interchange between agents. This does not mean that specific encoding or other media constraints may not exist for any given environment implementing the framework, depending on the operating system(s), tools, and processes used, only that the framework itself aims not to impose any such constraints itself.

Non-agent systems, processes, tools, or services which are utilized by an agent can still be accessed via proprietary means if necessary or useful for any operations or processes outside of the scope of the framework. Thus, framework based tools and services can co-exist freely with other tools and services utilizing the same resources.

2.1 Benefits to the Information Consumer

- Common interface to all information, regardless of media type
- Maximum flexibility for consumer access to information:
  - selection of tools (browsers, style sheets, parsers)
  - selection of distribution media (CD-ROM, online)
  - selection of media format (PDF, XML, HTML, Word, CGM, GIF, etc.)
- Increased ease of use through integration of documentation and software products in a consistent manner
- Open standards simplify installation, maintenance and customization
- Simplifies integration with consumer's own documentation and information

2.2 Benefits to Information Producer

- Provides a consistent and well defined end-to-end model
- Evolutionary, allows full utilization of existing systems and resources
- Extensible:
  - allows customization to meet any special needs
  - allows all needs to be met with minimal effort
  - allows full utilization of existing internal and third party tools
- Conforms to open standards, minimizing risks of dead-end systems
- Provides for live content updates
- Supports distributed and fault tolerant systems
- Scales from single user, single document browsing to full scale information distribution and management network
- Allows maximum flexibility in selection of tools and components
- Allows quick response to emerging needs and opportunities

The Metia Framework brings together both existing, legacy systems as well as new solutions into a common, interoperable environment; maximizing the investment in current systems while reducing the cost and risk of evolving and/or new solutions.

3 Framework Components

The Metia Framework is comprised of a number of components, each defining a core area of functionality needed in the construction of a complete production and distribution environment. Each framework component is defined separately by its own specification, in addition to the top level framework specification.

3.1 Media Attribution and Reference Semantics (MARS):

MARS is a metadata specification framework and core standard vocabulary and semantics facilitating the portable management, referencing, distribution, storage and retrieval of electronic media.

MARS is designed specifically for the definition of metadata for use by automated systems and for the consistent, platform independent communication between software components storing, exchanging, modifying, accessing, searching, and/or displaying various types of information such as documentation, images, video, etc. It is designed with considerations for automated processing and storage by computer systems in mind, not particularly for direct consumption by humans; though mechanisms are provided for associating with any given metadata property one or more presentation labels for use in user interfaces, reports, forms, etc.

MARS aims to fulfill the following two goals:

1. To define a framework within which metadata can be explicitly defined and efficiently and reliably processed by automated systems.
2. To define a core metadata vocabulary of properties and values for automated systems used for storing, exchanging, operating on, and/or displaying electronic media.

Extensibility of the core vocabulary is of course of paramount importance, as MARS cannot address all of the needs of all groups, systems, processes, products fully and still serve as a manageable standard; nor can it foresee all possible needs and applications in the future; however, it remains possible and beneficial both to define as rigorously as possible a framework for metadata and a core vocabulary and then enable extensions and enhancements to that core as needed, within the constraints of that framework.

It is important to note that the core vocabulary defined by MARS is data-centric and not use-centric, in that the metadata properties defined therein apply primarily to characteristics or attributes of the data itself, and not how, where, or by whom the data is used or referenced. Processes such as for Product Data Management (PDM), Configuration Management (CM), and Work Flow Management (WFM) are not directly addressed in the core MARS vocabulary insofar as these processes define uses of the data and not characteristics of the data itself.

The core vocabulary is specifically designed to meet the needs of organization and management processes applied to large volumes of technical and user documentation, though the framework and most if not all of the core vocabulary is applicable to many other applications as well.

3.1.1 Scoping Identity Model

MARS is made up of sets of metadata properties grouped into modules. Each module corresponds to a particular function or purpose which the properties contained in that module share. The properties defined in the Identity module are the heart of the Metia Framework.

As the module name implies, these properties are use to encode the unique identity of data entities, both abstract and concrete. The identity properties are scoping, meaning that they define a hierarchy of levels, corresponding to Media Object, Instance, Component, and Item (see diagram below).

The "identifier" property identifies an abstract *media object.*

The four properties "edition", "language", "coverage", and "encoding" together, along with the "identifier" property, identify an abstract *media instance.*

The "component" property, together with the higher scoped properties, identifies an abstract *media component.*

The "item" property, together with the higher scoped properties, identifies a concrete *storage item.*

It is important to note that the Identity properties differ from all other properties in that some value is required in order to fully identify any discrete body of data. Tools operating on MARS metadata are permitted to presume that a specified default value (as defined in the MARS specification) is valid if no other value is provided.

Filenames, URLs, and other system specific means of identification are typically fragile, frequently non-portable, and do not necessarily follow any formal model or methodology, hampering interoperability between disparate systems. Using sets of standard metadata properties such as those defined in the MARS Identity module provides a platform, system, and process independent means of defining the identity of documentation entities. It also allows systems to operate on one or more levels of scope, such as media object or instance, using user and/or environment information to resolve abstract references to physical data items.

Identity properties may only have single values. This is a special constraint and follows logically from the fact that if multiple values are allowed, there is no way to ensure that the same values are always used or that new values are not added, essentially changing the identity of the data. To change an Identity value is to change the data's identity. It is similar in effect to changing a filename in a file system.

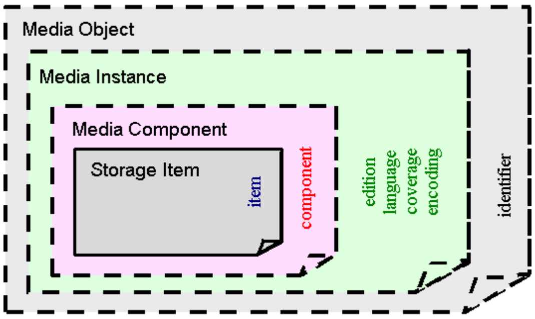

Figure 2: Metia Scoping Identity Model

3.2 Generalized Media Archive (GMA):

The GMA defines an abstract archival model for the storage and management of data based solely on Media Attribution and Reference Semantics (MARS) metadata; providing a uniform, consistent, and implementation independent model for information storage and retrieval, versioning, and access control.

The GMA is a central component of the Metia Framework and serves as the common archival model for all managed media objects controlled, accessed, transferred or otherwise manipulated by Metia Framework agencies.

The GMA provides a uniform, generic, and abstract organizational model and functional interface to a potentially wide range of actual archive implementations; independent of operating system, file system, repository organization, versioning mechanisms, or other implementation details. This abstraction facilitates the creation of tools, processes, and methodologies based on this generic model and interface which are insulated from the internals of the GMA compliant repositories with which they interact.

The GMA defines specific behavior for basic storage and retrieval, access control based on user identity, versioning, automated generation of variant instances, and event processing.

The identity of individual storage items is based on MARS Identity metadata properties and all interaction between a client and a GMA implementation must be expressed as MARS metadata property sets.

3.3 Portable Media Archive (PMA):

The PMA is a physical organization model of a file system based data repository conforming to and suitable for implementations of the Generalized Media Archive (GMA) abstract archival model.

The PMA defines an explicit yet highly portable file system organization for the storage and retrieval of information based on Media Attribution and Reference Semantics (MARS) metadata. The PMA uses the MARS Identity metadata property values themselves as directory and/or file names, avoiding the need for a secondary referencing mechanism and thereby simplifying the implementation, maximizing efficiency, and producing a mnemonic organizational structure.

This specification only defines the physical organization of a file system, and not the processes or algorithms for accessing, manipulating, or otherwise interacting with or operating on that file system. Different GMA implementations based on the PMA model may interact with the data in different ways.

Any GMA may use a physical organization model other than the PMA. The PMA physical archival model is not a requirement of the GMA abstract archival model. However, the PMA may nevertheless be employed by such implementations both as a data interchange format between disparate GMA implementations as well as a format for storing portable backups of a given archive.

3.4 Registry Service Architecture (REGS):

REGS is a generic architecture for dynamic query resolution agencies based on the Metia Framework and Media Attribution and Reference Semantics (MARS), providing a unified interface model for a broad range of search and retrieval tools.

REGS provides a generic means to interact with any number of specialized search and retrieval tools using a common set of protocols and interfaces based on the Metia Framework; namely MARS metadata semantics and either a POSIX or CGI compliant interface. As with other Metia Framework components, this allows for much greater flexibility in the implementation and evolution of particular solutions while minimizing the interdependencies between the tools and their users (human or otherwise).

Being based on MARS metadata allows for a high degree of automation and tight synchronization with the archival and management systems used in the same environment, with each registry service deriving its own registry database directly from the metadata stored in and maintained by the various archives themselves; while at the same time, each registry service is insulated from the implementation details of and changes in the archives from which it receives its information.

Every registry service shares a common architecture and fundamental behavior, differing primarily only in the actual metadata properties required for their particular application.

A typical environment is expected to employ numerous Registry Services interacting with numerous Media Archives which may or may not be implemented using the PMA file system model, all organized by and integrated via MARS metadata.

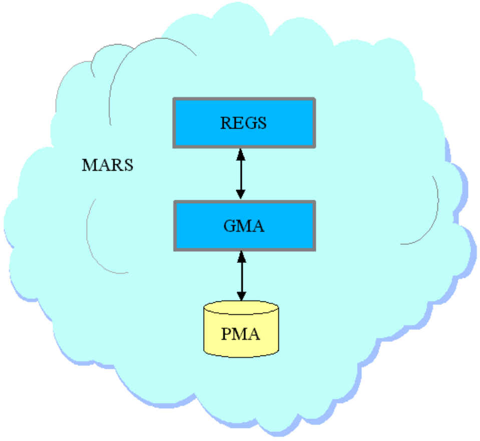

Figure 3: The relationships between the framework components, in the simplest case.

4 Current Deployment

Version 1.1 of the top level Metia Framework specification and the specifications for the GMA, PMA, and REGS components, as well as version 2.1 of MARS are approved and currently in use within Nokia.

A Java SDK (Software Development Kit) for the Metia Framework was released in the Spring of 2001, providing ready-made software components and utilities for constructing framework agents and other necessary components in Java, including two implementations of a GMA, one based on a relational database and another based on the PMA.

A complete documentation management system based on the Metia Framework – including version control, configuration management, and workflow management – is in the final stages of pilot testing and will enter full use beginning the Fall of 2001.

Several research, pilot and system deployment projects will be conducted during the course of 2001[5] and 2002 to further apply the framework in areas relating to:

- navigation and organization methodologies
- search and retrieval methodologies
- dynamic partitioning and content optimization in the browsing environment
- semantic web applications

5 Future Work

The Metia Framework is still in its infancy, and though much progress has been made during the past year and it has reached a stage where it provides a solid foundation for the systems and solutions currently being developed within Nokia, there is still much to do before we have addressed the complete set of challenges now facing us. The following are the most immediate issues that should be examined over the next 12-18 months:

The current ontology is based on the particular perspectives and needs of Nokia and may not be sufficiently generic for industry wide or cross-industry applications.

New technologies such as WebDAV[6], LDAP[7], SyncML[8], and SOAP[9] should be investigated, as to how they might augment or enhance the framework.

"Global Environment" issues such as media object identifiers and user identity which are reliable, authoritative, non-proprietary, global, and persistent such that information references/links/etc. utilizing such identifiers remain valid over a long period of time and across a broad range of platforms, systems, and environments should be explored further.

We need to do practical work relating to improved serialization/encoding methods for metadata property sets and develop a more effective set of tools and processes for metadata specification and management.

We should explore the use of technologies such as Topic Navigation Maps[10] or XTM[11], RDF[12], RDF Schemas[13], XML Schemas[14], and similar to provide better solutions for navigation, organization, configuration, search and retrieval.

Work is needed in the area of vocabularies and ontologies to provide better classification models and search solutions to facilitate effective use of modular documentation components and multiple user and/or context specific dynamic views into a common information space. MARS should be extended and refined to include a core set of ontologies relevant for the basic processes falling within the scope of the framework.

References

[1] ISO. 1986. ISO 8879:1986 Information processing -- Text and office systems -- Standard Generalized Markup Language (SGML). Geneva: ISO.

[2] Bray, Tim, Jean Paoli, C. M. Sperberg-McQueen, and Eve Maler. 2000. Extensible Markup Language (XML) 1.0 (Second Edition).

http://www.w3.org/TR/2000/REC-xml-20001006: World Wide Web Consortium (W3C).

[3] Joy, Bill, Guy Steele, James Gosling, and Gilad Bracha. 2000. *The Java Language Specification, Second Edition (The Java Series)*. Addison-Wesley Pub Co., June 5.

[4] IETF Network Working Group. 1999. RFC2616: Hypertext Transfer Protocol -- HTTP/1.1. http://www.ietf.org/rfc/rfc2616.txt: The Internet Society, June.

[5] Helin, Riikka. 2000. *NCDE 2001 Release Plan*, Nokia. November, Draft, Company Confidential.

[6] IETF Network Working Group. 1999. RFC2518: HTTP Extensions for Distributed Authoring -- WEBDAV. http://andrew2.andrew.cmu.edu/rfc/rfc2518.html: The Internet Society, February.

[7] IETF Network Working Group. 1995. RFC1777: Lightweight Directory Access Protocol. http://idm.internet.com/RFC/rfc-1777.html: The Internet Society, March.

[8] SyncML Consortium. 2000. SyncML Sync Protocol, version 1.0. http://www.syncml.org/docs/syncml_protocol_v10_20001207.pdf: SyncML Consortium, December 7.

[9] Box, Don, David Ehnebuske, Gopal Kakivaya, Andrew Layman, Noah Mendelsohn, Henrik Frystyk Nielsen, Satish Thatte, and Dave Winer. 2000. Simple Object Access Protocol (SOAP) 1.1. http://www.w3.org/TR/2000/NOTE-SOAP-20000508: World Wide Web Consortium (W3C), May 8.

[10] ISO. 1999. ISO/IEC 13250 Topic Maps: Information Technology -- Document Description and Markup Languages. Geneva: ISO, December 3.

[11] XTM Authoring Group. 2000. XTM: XML Topic Maps (XTM) 1.0 Core Deliverables. http://www.topicmaps.org/xtm/1.0/core.html: TopicMaps.Org, 4 Dec.

[12] Lassila, Ora, and Ralph R. Swick. Eds. 1999. Resource Description Framework (RDF) Model and Syntax Specification. http://www.w3.org/TR/1999/REC-rdf-syntax-19990222: World Wide Web Consortium (W3C).

[13] Brickley, Dan, and R.V. Guha. Eds. 2000. Resource Description Framework (RDF) Schema Specification 1.0. http://www.w3.org/TR/2000/CR-rdf-schema-20000327: World Wide Web Consortium (W3C).

[14] Connolly, Dan, and Henry Thompson. Eds. 2000. *XML Schema*. http://www.w3.org/XML/Schema: World Wide Web Consortium (W3C).