

A standard for the publication of government news summaries

**FINAL
Version 1.0**

July 7th 2003

Ferry Hendrikx (ferry.hendrikx@ssc.govt.nz)
E-government Unit, State Services Commission

Table of Contents

1	Overview.....	3
1.1	Scope	3
1.2	Who Should Use This Standard	3
1.3	Implementing This Standard	3
2	RSS Specification	5
2.1	RDF Site Summary 1.0 (RSS)	5
2.2	RDF Site Summary 1.0 (RSS) Specification	5
2.2.1	Dublin Core	5
2.2.2	Syndication	5
2.3	NZGLS	5
2.3.1	NZGLS	5
3	RSS Implementation	6
3.1	Header	6
3.2	Channel	6
3.3	Items	7
3.4	The end	7
4	RSS Feed	8
5	FAQ (Frequently Asked Questions)	9
6	A complete example of RSS	10
6.1	Complete example	10
6.2	Changing the example to suit your needs	10
7	Appendix.....	12
7.1	<nzgls:date.valid> (Required)	12
7.2	<nzgls:coverage.spatial> (Optional)	12
7.3	<nzgls:identifier> (Required)	13
7.4	<nzgls:type.agency> (Required)	14
7.5	<nzgls:type.document> (Optional)	14
8	Glossary.....	15

Changes

Change Date	Description	Author
21/03/2003	Minor corrections and changes.	F. Hendrikx
27/03/2003	Rework of the Introduction.	C. Daish/F. Hendrikx
30/04/2003	Minor corrections and changes after feedback from some of the participants in the prototype.	F. Hendrikx
07/07/2003	Updated for final release.	F. Hendrikx

1 Overview

This document proposes a technical standard for the publication of government agency news on the web. "News" in this context means media releases and other event-related content authored by government agencies and intended for public consumption via outlets in various media.

The E-government Unit (EGU) of the State Services Commission proposes the standard to support the development of an e-government "component" that can be used by government agencies in a variety of ways. The component has the following features:

1. A news feed provided in RSS format (the format described in this paper).
2. A news fetch operated by the EGU, which collects and aggregates news stories across participating agencies.
3. A news syndication service operated by the EGU, which allows agencies and other organisations to subscribe to the aggregated news feed.

This document describes the first feature only. The syndication component is currently under development.

1.1 Scope

"Syndication" normally refers the distribution of articles and photographs to several newspapers or magazines for simultaneous or later publication. RSS (RDF Site Summary) has become de facto standard for the syndication of news and current events material on the Internet. RSS is an extensible XML-based schema that allows the syndication of lists of hyperlinks, along with metadata.

Syndication using RSS is achieved by making an RSS feed available on a website at a specific URL address. The feed can be requested just like any other file or resource on the server, which is normally done at regular intervals to get the most recent items on the list. In practice, several feeds are made available by an aggregator, which is a website or program that manages a number of lists and presents them in a single interface.

In this proposal, participating government agencies provide their news content as a feed, which the central server managed by the E-government Unit regularly downloads. The central server acts as an aggregator, displaying the news in a single interface. The central server then makes an aggregated feed available for further consumption. The initial consumer of all news published in this fashion is the government web portal at <http://www.govt.nz/>.

1.2 Who Should Use This Standard

This standard should be used by all agencies wishing to make their news stories available beyond their own website. Specifically agencies can use this system to avoid the manual and often duplicate entry of news stories onto the government portal.

Agencies implementing this standard and making news feeds available will need to notify the E-government Unit so that the aggregator can be appropriately configured to harvest news feeds from agency web servers.

1.3 Implementing This Standard

There are four steps to adopting and implementing the standard:

1. An agency creates news articles in the format described in this document, and creates the appropriate URLs on their web server;

2. The agency notifies the EGU of the URLs by sending email to newsfeed@portal.govt.nz with details of your configuration. The EGU will send the agency a confirmation;
3. The EGU runs a test fetch of the articles and parses the XML code. Details of any errors will be returned to the agency for correction;
4. Once corrections have been made and the agency is happy with operation of the system, the agency sends email to newsfeed@portal.govt.nz nominating a suitable start date and preferred fetch frequency.

Please note there is no formal audit or compliance procedure associated with this standard; rather, agencies can self-certify their compliance. The EGU has configured the XML parser to conform to the standard and will send email detailing any failures encountered, enabling agencies to fine-tune systems where necessary.

2 RSS Specification

2.1 RDF Site Summary 1.0 (RSS)

RSS is a lightweight multipurpose extensible metadata description and syndication format. RSS is an XML application that conforms to the W3C's RDF Specification and is extensible via XML-namespace and/or RDF based modularisation (see below).

An RSS file, at a minimum, is a document describing a "channel" consisting of URL-retrievable items. Each item consists of a title, link, and brief description. While items have traditionally been news headlines, RSS has seen much repurposing in its short existence.

2.2 RDF Site Summary 1.0 (RSS) Specification

The RDF Site Summary 1.0 (RSS) specification can be found at:

- <http://web.resource.org/rss/1.0/spec>

This specification defines the core RSS schema. Additional elements that are not in the core schema are added by the use of modules. Each "imported" module extends the namespace of the core schema. This **namespace-based modularisation** affords RSS 1.0 extensibility. The only modules that ship "in the box" with RSS 1.0 are Dublin Core, Content and Syndication.

Consult the appropriate module documentation for further information:

2.2.1 Dublin Core

- <http://purl.org/rss/1.0/modules/dc/>

2.2.2 Syndication

- <http://purl.org/rss/1.0/modules/syndication/>

2.3 NZGLS

The core set, together with the modules above does not give us all the elements required to describe our (Government) data set (for example, there is no "agency" element).

Defining our own Government specific module and "importing" this into the core RSS schema easily remedies this problem, and provides for "future-proofing". We can always extend the schema should we need new elements. Our Government specific module is called NZGLS and was defined as part of the E-government programme:

2.3.1 NZGLS

- <http://www.nzgls.govt.nz/standard/>

Please note that the appendix lists the elements and syntax of the NZGLS module.

3 RSS Implementation

The news import files should be formatted as RDF 1.0, with a single channel. The file must be encoded as UTF-8. See Unicode (<http://www.unicode.org/>) for further information.

All of the RSS core schema elements along with the 3 modules (defined above) and the NZGLS extensions (see below) can be used within an RSS file. They may be used as required (and within the limits of the standard).

The following sample RDF blocks should be used as a guide only. Please note that the RDF blocks given below are examples and may contain information that is fictional or only partially complete.

3.1 Header

Each RDF document should start as follows:

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

<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns="http://purl.org/rss/1.0/"
  xmlns:nzgls="http://www.nzgls.govt.nz/standard/"
  xmlns:dc="http://purl.org/rss/1.0/modules/dc/"
  xmlns:syn="http://purl.org/rss/1.0/modules/syndication/"
>
```

You may not need all of the *xmlns* (XML namespace) includes. If this is the case, simply leave out those includes that are not required (the top three modules are **always required**).

3.2 Channel

A channel declaration should follow the header. The URL specified in the *rdf:about* attribute of the channel should be unique within an RSS file. This URL is commonly the place where the RSS file can be found (it may also be the home page of the content being described).

```
<channel rdf:about="http://www.ssc.govt.nz/news/news.rss">
  <title>State Services Commission News</title>
  <link>http://www.ssc.govt.nz/news/</link>
  <description>The State Services Commission News</description>

  <dc:creator>news@ssc.govt.nz</dc:creator>
  <dc:language>en-nz</dc:language>
  <dc:publisher>State Services Commission</dc:publisher>

  <syn:updatePeriod>daily</syn:updatePeriod>
  <syn:updateFrequency>2</syn:updateFrequency>
```

The channel **must** contain a *title* that describes the source of this news. The *link* element **must** also be present and must point to a place where a textual version of this news can be found. This is typically a URL that points to a HTML page. The optional *description* can be used to describe this channel. In the example above the description contains the name of the news source.

The Dublin Core elements (dc) allow the creator to add specific information about the channel. The examples above show the use of the *creator*, *language* and the *publisher* elements. These elements are **optional** but are highly recommended. As a general rule, NZGLS elements should be used in preference to Dublin Core elements for government specific information. In this instance, Dublin Core elements are ideal because the information is not government specific and they are more widely understood by software.

The Syndication elements (syn) allow the feed to hint to the aggregator how often the content should be fetched. In this example, the feed has hinted that it should be fetched twice daily. The aggregator may ignore this information because this it is considered to be a hint only. These elements are **optional** but are highly recommended.

The channel must contain a list of links of the news items that are present in the file. This is represented using an RDF sequence of links to the actual news items. There should be a link at this point for every news item that will appear in the file (after the closing channel element).

```
<items>
  <rdf:Seq>
    <rdf:li rdf:resource="http://www.ssc.govt.nz/disability-mentoring-day" />
    <rdf:li rdf:resource="http://www.ssc.govt.nz/appt-ce-natlib-oct02" />
    ...
    ...
  </rdf:Seq>
</items>
</channel>
```

3.3 Items

The actual news content should be formatted as follows: Each *item* **must** include a **title**, **link**, **nzgls:date.valid**, **nzgls:identifier** and **nzgls:type.agency** at the minimum. Further fields can be added from *NZGLS*, *dc* or *content* as required (within the limitations of the specification).

Please note that the content of the description field should consist of pure text and should not contain any HTML or XML mark-up.

```
<item rdf:about="http://www.ssc.govt.nz/disability-mentoring-day">
  <title>Applications invited for Disability Mentoring Day</title>
  <link>http://www.ssc.govt.nz/disability-mentoring-day</link>
  <description>The State Services Commission, through its Mainstream Employment Programme, is pleased to host the first New Zealand Disability Mentoring Day: Career Development for the 21 Century. Disability Mentoring Day will be celebrated in Wellington on Tuesday 12 November 2002. Applications are now invited from people with disabilities from departments and other State sector organisations to be mentors for the day, and from tertiary students with disabilities who would like to take part as mentees.</description>
  <nzgls:date.valid>start=2002-10-07; end=2002-11-12T19:00:00+12:00</nzgls:date.valid>
  <nzgls:identifier>20021007-0015-SSC</nzgls:identifier>
  <nzgls:type.agency>State Services Commission</nzgls:type.agency>
</item>

<item rdf:about="http://www.ssc.govt.nz/appt-ce-natlib-oct02">
  <title>Appointment of National Librarian announced</title>
  <link>http://www.ssc.govt.nz/appt-ce-natlib-oct02</link>
  <description>The State Services Commissioner, Michael Wintringham, announced today the appointment of Penny Carnaby as Chief Executive of National Library of New Zealand and National Librarian...</description>
  <nzgls:date.valid>2002-10-09</nzgls:date.valid>
  <nzgls:identifier>20021009-0016-SSC</nzgls:identifier>
  <nzgls:type.agency>State Services Commission</nzgls:type.agency>
  <nzgls:type.document>News</nzgls:type.document>
</item>
```

3.4 The end

Finally, the RDF closing statement should mark the end of the document:

```
</rdf:RDF>
```

4 RSS Feed

The news feed must be available as an (HTTP) URL that can be fetched over the Internet. The URL can point to any type of page, as long as it is retrievable via HTTP. You may make your news content available using a static or dynamic (PHP, ASP, Perl, etc) page. The URL **must** always remain the same.

Some example (fictitious) URLs:

- <http://www.doc.govt.nz/doc-news.rdf>
- <http://www.ssc.govt.nz/news/news.rss>

The content within your feed **must** always contain the news items for the **last 7 days** (at a minimum). In this way, news items will never be lost, even if there is a service disruption. If there was no news content for this period of time, the aggregator will expect to find an "empty" feed. This does not mean a file of zero bytes, but rather a feed with no <items> in it.

An alternative solution is to always provide the 10 (for example) latest items in your feed. Even if no content was added recently, your feed will always contain the most recent news items.

On fetching a feed, the aggregator will save new items and update items that were previously already published. If an item was previously published, it will be updated in the aggregator, taking the version present in the feed as the authoritative version. The only exception to this rule is if the aggregator editor had edited a given item. In this instance, the item will no longer consider the source feed as authoritative (and will therefore no longer synchronise that given item with the source feed).

5 FAQ (Frequently Asked Questions)

5.1 Which fields are required within an item?

At the *very minimum* you require the following fields:

- <title>
- <link>
- <nzxls:date.valid>
- <nzxls:identifier>
- <nzxls:type.agency>

Of course, you may add any further elements as required, and the <description> element will make your items a lot more useful!

5.2 What is the <nzxls:identifier> field used for?

This field provides each item with a unique identifier. This is used when checking for updates to a given item. It should consist of the date, followed by a sequence number and finally an abbreviated agency or unit name. These 3 parameters ensure that the identifier remains unique.

5.3 What happens to items with a future start date?

An item with a future start date in the <nzxls:date.valid> element will not get published until that date and time has been reached. Nor will the item be available in the aggregated output until that time.

5.4 Do items with an expired end date get deleted?

An item with an expired end date in the <nzxls:date.valid> element is simply not shown on the most recent list of articles. It is moved to the archive section of the aggregator. Expired items will also not appear in the aggregated output anymore.

5.5 What can I do if I need an item expired today?

You can change the expiry date of the item in your feed. When the item on the aggregator is synchronised with the item in the feed, it will immediately acquire the new expiry date.

5.6 How long should an item keep appearing within my feed?

An item should appear for at least 7 days in your feed. This so that items are never missed, even if there are service disruptions or "acts of God". A simpler solution that you may consider is to ensure that the 10 (for example) most recent items are always present in your feed.

5.7 Why are <nzxls> elements used rather than <dc> elements?

The NZGLS metadata standard is the official New Zealand Government standard for creating discovery-level metadata (see <http://www.nzxls.govt.nz/>). The standard is based closely on two well-established standards: the Dublin Core (DC) Metadata Element Set and the Australian Government Locator Service (AGLS). NZGLS elements should be used in preference to Dublin Core elements for government specific information.

5.8 How much content should go into a <description>?

This is up to each feed provider to decide. A basic rule of thumb is to provide sufficient information such that the item is complete in itself; the reader should be able to determine what the item is about and its main points. A description can contain the entire content, partial content or only a summary of the content.

5.9 Which encoding scheme should we use?

You should use UTF-8 encoding. This means that all characters with an ASCII value higher than 127 (0x7F) are encoded as double byte sequences or as "&#{NUMBER};". Please note that special characters like & (ampersand) should also be encoded using the "&#{NUMBER};" notation.

6 A complete example of RSS

This section outlines a complete example of RDF Site Summary 1.0 (RSS) as specified in this document. It uses only the **required** elements, ignoring all optional elements for the sake of clarity and simplicity.

6.1 Complete example

This example is constructed from the examples used in this document previously.

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

<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns="http://purl.org/rss/1.0/"
  xmlns:nzgls="http://www.nzgls.govt.nz/standard/"
>

<channel rdf:about="http://www.ssc.govt.nz/news/news.rss">
  <title>State Services Commission News</title>
  <link>http://www.ssc.govt.nz/news/</link>
  <description>The State Services Commission News</description>
  <items>
    <rdf:Seq>
      <rdf:li rdf:resource="http://www.ssc.govt.nz/disability-mentoring-day" />
      <rdf:li rdf:resource="http://www.ssc.govt.nz/appt-ce-natlib-oct02" />
    </rdf:Seq>
  </items>
</channel>

<item rdf:about="http://www.ssc.govt.nz/disability-mentoring-day">
  <title>Applications invited for Disability Mentoring Day</title>
  <link>http://www.ssc.govt.nz/disability-mentoring-day</link>
  <description>The State Services Commission, through its Mainstream Employment Programme, is pleased to host the first New Zealand Disability Mentoring Day: Career Development for the 21 Century. Disability Mentoring Day will be celebrated in Wellington on Tuesday 12 November 2002.</description>
  <nzgls:date.valid>2002-10-07</nzgls:date.valid>
  <nzgls:identifier>20021007-0015-SSC</nzgls:identifier>
  <nzgls:type.agency>State Services Commission</nzgls:type.agency>
</item>

<item rdf:about="http://www.ssc.govt.nz/appt-ce-natlib-oct02">
  <title>Appointment of National Librarian announced</title>
  <link>http://www.ssc.govt.nz/appt-ce-natlib-oct02</link>
  <description>The State Services Commissioner, Michael Wintringham, announced today the appointment of Penny Carnaby as Chief Executive of National Library of New Zealand and National Librarian...</description>
  <nzgls:date.valid>2002-10-09</nzgls:date.valid>
  <nzgls:identifier>20021009-0016-SSC</nzgls:identifier>
  <nzgls:type.agency>State Services Commission</nzgls:type.agency>
</item>

</rdf:RDF>
```

6.2 Changing the example to suit your needs

You can adapt the above example to your own situation by following these simple steps.

In the `<channel>` section of the above example:

- Change the `<title>` element to the name of your agency.
- Change the `<link>` element to the URL of your agency news, or failing that, the URL of your agency's website.
- Change the `<description>` element to something appropriate for your agency.

In the `<rdf:Seq>` section of the above example:

- Ensure there is an `<rdf:li>` element for each news `<item>` in the file. This URL in this element should match the URL with the `<item>` itself.

Each news `<item>` should contain:

- A `<title>`.
- A `<link>` (URL) where the news article can be found.
- A short `<description>` of the news item. This description should be pure text only and contain no HTML or XML mark-up.
- An `<nzgls:date>` for when this `<item>` should be published.
- An `<nzgls:identifier>` that is unique for each `<item>`.
- Your agency name in the `<nzgls:type.agency>` element.

7 Appendix

This appendix lists details of the NZGLS elements used in the example above. For each element you will find the its required syntax, and its default values should you not specify them.

7.1 **<nzgls:date.valid>** (Required)

This element is used to define a date (and time) range for when this item is valid. Date ranges should follow the DCMI Period specification for defining the start and end dates. It should be noted that this element controls an item's validity period; the validity of an item does not guarantee that it will be published.

This element is required, but at a minimum, it is sufficient to only specify the start date. Please note that the default end date and time is 7 days *from* the start date at 23:59 local time. An item with a start date in the future, but with no end date will also get an end date of 7 days from the start date (rounded up till 23:59).

A date should always be expressed as YYYY-MM-DD at a minimum. The time (if specified) **must** always be expressed as HH:MM:SS+12:00. The +12:00 is needed in New Zealand because the W3C specifies that the time will always be in UTC unless the time offset is also included.

A fully specified date and time is therefore defined as:

YYYY-MM-DDTHH:MM:SS+1200

Please note that the separating 'T' between the date and time is required when specifying a date *and* time. Some example values for date and time are:

2002-10-15
2002-10-15T11:25:00+12:00

See <http://www.w3.org/TR/NOTE-datetime> for further information about date and time formatting, and see <http://dublincore.org/documents/2000/07/28/dcmi-period/> for more detailed information about using DCMI-Period.

Syntax: <nzgls:date.valid>start={DATE}; end={DATE}</nzgls:date.valid>

Requirement: Required

Required Attribute(s): none

Default: See description above.

Examples:

```
<nzgls:date.valid>2002-10-07</nzgls:date.valid>
```

```
<nzgls:date.valid>start=2002-10-07</nzgls:date.valid>
```

```
<nzgls:date.valid>start=2002-10-07; end=2002-11-12</nzgls:date.valid>
```

```
<nzgls:date.valid>start=2002-10-07; end=2002-11-12T19:00:00+12:00</nzgls:date.valid>
```

```
<nzgls:date.valid>
```

```
start=2002-10-07T09:00:00+12:00; end=2002-11-12T19:00:00+12:00
```

```
</nzgls:date.valid>
```

7.2 **<nzgls:coverage.spatial>** (Optional)

This element is used to define the spatial location for a given item, and will be used for the regionalisation of content delivery. The aggregator will separate content into broad regions, using regional council boundaries.

The regional councils, in geographical order are: Northland, Auckland, Waikato, Bay of Plenty, Gisborne, Hawke's Bay, Taranaki, Manawatu (-Wanganui), Wellington, Marlborough, Nelson, Westland, Canterbury, Otago and Southland.

This element is optional; any item without this element will be assumed to be valid for the entire country. This element may be up to 128 characters long.

If an item is valid for multiple regions, then each region should be listed separated by commas. An item that applies to both a particular region and the whole country can use the region "New Zealand" followed by the specific region name.

Syntax: <nzgls:coverage.spatial>{REGION NAME}</nzgls:coverage.spatial>

Requirement: Optional

Required Attribute(s): none

Examples:

```
<nzgls:coverage.spatial>Canterbury</nzgls:coverage.spatial>
<nzgls:coverage.spatial>Canterbury, Wellington</nzgls:coverage.spatial>

<nzgls:coverage.spatial>New Zealand</nzgls:coverage.spatial>
<nzgls:coverage.spatial>New Zealand, Wellington</nzgls:coverage.spatial>
```

7.3 <nzgls:identifier> (Required)

This element is used to define a unique identifier for each record. This element is required; any item without this element will be ignored.

The aggregator detects duplicates and updates to previously published news items using the unique identifier. A news item may have changed entirely, but as long as the identifier remains valid, it can be successfully updated.

The identifier can be made up of up to 64 characters, from the set [0-9a-zA-Z-+ _]. The identifier **must** consist of an YYYYMMDD date (optionally hyphen separated), followed by a sequence number and an abbreviated agency name, all separated by hyphens.

By preference, the date should be the date that the item was published, but this is not mandatory. The sequence number is simply present to ensure uniqueness and can be either a simple sequence number or some kind of internal reference id.

The identifier **must** remain the same for all instances of the same item; otherwise the item in question may be published multiple times.

Syntax: <nzgls:identifier>{IDENTIFIER}</nzgls:identifier>

Requirement: Required

Required Attribute(s): none

Examples:

```
<nzgls:identifier>20021015-0001-SSC</nzgls:identifier>
<nzgls:identifier>20021015-0002-SSC</nzgls:identifier>
<nzgls:identifier>20021016-0003-SSC</nzgls:identifier>

<nzgls:identifier>20030411-269059-EW</nzgls:identifier>
<nzgls:identifier>20030411-269061-EW</nzgls:identifier>
```

A hyphen separated date is also accepted:

```
<nzgls:identifier>2003-02-12-0078-SSC</nzgls:identifier>
<nzgls:identifier>2003-02-12-0079-SSC</nzgls:identifier>
```

7.4 **<nzgls:type.agency>** (Required)

This element is used to define the Agency who published this item. This element is required; any item without this element will be ignored. This element may be up to 128 characters long.

This is a required element, even though it is highly likely that all the items within a single feed will contain the same information in this element. This element is required so that an aggregated output feed will contain the correct values for this element.

Syntax: <nzgls:type.agency>{AGENCY NAME}</nzgls:type.agency>

Requirement: Required

Required Attribute(s): none

Examples:

<nzgls:type.agency>State Services Commission</nzgls:type.agency>

7.5 **<nzgls:type.document>** (Optional)

This element is used to define the type of this particular item. This element is optional; any item without this element will be assumed to be of type "News".

The type should be one of:

- News
- Consulting
- Newsletter

The above list of types does not preclude you from adding other types as needed. For example, agencies wishing to convey information about a vacancy could define the type "Vacancy". At the moment the aggregator will silently ignore any types it does not recognise.

Syntax: <nzgls:type.document>{TYPE}</nzgls:type.document>

Requirement: Optional

Required Attribute(s): none

8 Glossary

HTML

HTML is the *lingua franca* for publishing hypertext on the World Wide Web. It is a non-proprietary format based upon SGML, and can be created and processed by a wide range of tools. See W3C (<http://www.w3.org/MarkUp/>) for more information.

HTTP

The Hypertext Transfer Protocol is the main protocol used for communications between web servers and browsers.

NZGLS

The NZGLS metadata standard is the official New Zealand Government standard for creating discovery-level metadata (<http://www.nzgls.govt.nz/>). The standard is based closely on two well-established standards: the Dublin Core (DC) Metadata Element Set and the Australian Government Locator Service (AGLS).

PHP

"PHP: Hypertext Preprocessor" (or Personal Home Page tools) is a scripting language that embeds instructions along with HTML in each page. When a user requests a given page, the web server will execute the instructions and send the results to the user's browser. See (<http://www.php.net/>) for more information.

RDF

Resource Description Framework is a universal format for data on the Web. Using a simple relational model, it allows structured and semi-structured data to be mixed, exported and shared across different applications. See W3C (<http://www.w3.org/RDF/>) for more information.

RSS

RDF Site Summary (or Really Simple Syndication) is a lightweight multipurpose extensible metadata description and syndication format. RSS is an XML application, conforms to the W3C's RDF Specification and is extensible via XML-namespace and/or RDF based modularisation.

URL

The Uniform Resource Locator is the address of a resource, or file, available on the Internet. The URL contains the protocol of the resource (e.g. <http://> or <ftp://>), the domain name for the resource, and the hierarchical name for the file.

XML

Extensible Markup Language (XML) is a simple, very flexible text format derived from SGML. Originally designed to meet the challenges of large-scale electronic publishing, XML is also playing an increasingly important role in the exchange of a wide variety of data on the Web and elsewhere. See W3C (<http://www.w3.org/XML/>) for more information.

XML namespace (xmlns)

In general, a namespace uniquely identifies a set of names so that there is no ambiguity when objects having different origins but the same names are mixed together. In XML, a namespace is commonly given the name of a Uniform Resource Identifier (URI) - such as a Web site's address - both because the namespace may be associated with the site or page of that URI (for example, a company name) and because a URI is conveniently likely to be a unique name.