

user may be queried to provide the information needed to establish a description for the new blog item.

[0048] The specification of a descriptor (e.g., attributes, their types and constraints) can be provided with an information source and one or more information source data items. Some examples include, without limitation:

EXAMPLE 1

An Internal Information Source a Blog, Information Sources Items, and Blog Items

- [0049]** blog descriptor
- [0050]** attribute category of type string
- [0051]** attribute frequency of type string (constrained to hourly, daily, weekly, monthly)
- [0052]** blog item descriptor (a set of ‘description schema attributes’)
- [0053]** attribute repairDate of type dateTime
- [0054]** attribute repairLocation of type address
- [0055]** attribute dealership of type string
- [0056]** It should be noted that the exemplary set of types is defined by the service platform. The service platform also defines the ways to constrain types. For example, an attribute of type ‘string’ can be constrained by enumerating the set of allowed values. In the case of a blog, the set of attributes that make up the descriptor can be specified by the user when the blog is created. When an entry is posted to the blog, the user making the post will be queried to specify the values for each attribute part of the descriptor. The set of actual values make up the description of the blog item.
- [0057]** A blog entry:
- [0058]** Subject: poor repair
- [0059]** Summary: paid top \$, got nothing done
- [0060]** Description:
- [0061]** Attribute: repairDate, Value: 2006-03-21
- [0062]** Attribute: repairLocation, Value: Roanoke, Va.
- [0063]** Attribute: dealership, Value: FirstTeam
- [0064]** The RSS feed representation of the blog content will contain the descriptions through an XML namespace based extension model:

```
<item>
  <title>poor repair</title>
  <description>paid top $, got nothing done</description>
  <author>jora</author>
  <guid>jora.autos.41</guid>
  <pubDate>Mon 17-Apr-2006 14:42:12</pubDate>
  <category>blog</category>
  <autos:description>
    xmlns:autos="http://mxm.research.att.com/blog/jora/autos">
      <autos:repairDate>2006-03-21</autos:repairDate>
      <autos:repairLocation>Roanoke, VA</autos:repairLocation>
      <autos:dealership>FirstTeam</autos:dealership>
    </autos:description>
  </item>
```

EXAMPLE 2

Data from an External Information Source

- [0065]** Data from a non-RSS information source, such as, Exchange Server, calendar information. The descriptor of, for example, an RSS feed produced by an

infolet handling information provided by the Exchange server is specified by the infolet.

- [0066]** ExchangeCalendarInfolet item descriptor:
- [0067]** Attribute priority of type string (constrained to low, medium, high)
- [0068]** Attribute meeting of type boolean
- [0069]** Attribute meetingLocation of type address
- [0070]** For each calendar item, the infolet can ‘calculate’ the descriptive attributes from the available exchange calendar information (which is naturally richer information). When the infolet provides the RSS feed corresponding to a user’s Exchange calendar information, the description is calculated and added to the feed:

```
<item>
  <title>poor repair</title>
  <description>paid top $, got nothing done</description>
  <author>jora</author>
  <guid>jora.autos.41</guid>
  <pubDate>Mon 10-Jan-2006 14:42:12</pubDate>
  <category>Exchange calendar</category>
  <calendar:description>
    xmlns:calendar="http://mxm.research.att.com/exchange/calendar">
      <calendar:priority>low</calendar:meetingLocation>
      <calendar:meeting>true</calendar:meetingLocation>
      <calendar:meetingLocation>Florham
        Park,NJ</calendar:meetingLocation>
    </calendar:description>
  </item>
```

EXAMPLE 3

Data from an External RSS Feed

- [0071]** Data from an RSS information source, such as an External information source that is already available in RSS format can be enriched by calculating attributes from the existing information.
- [0072]** Finance news feed item descriptor:
- [0073]** Attribute company of type string
- [0074]** Attribute stock of type string
- [0075]** Attribute eventLocation of type address
- [0076]** When the RSS infolet retrieves the feed, it analyzes the ‘description’ element of each item and looks for whatever seems to be the company name, and extracts it and places it in the description. From the company name, the RSS infolet can figure out the stock symbol for the company and can also add the stock symbol to the description. In a similar way, the event location could be found in the description or retrieved as the company’s headquarters. The resulting feed is shown below:

```
<item>
  <title>GM Will Pay Lower Prices for Delphi Parts (AP)</title>
  <link>http://us.rd.yahoo.com/finance/news/rss/story/*http://us.rd.yahoo.com/finance/news/topnews/*http://biz.yahoo.com/ap/060501/delphi.html?.v=2 </link>
  <description>General Motors Corp. will pay lower prices for parts from Delphi Corp. after it failed to reach an agreement to continue paying higher rates to the auto supplier, Delphi said Monday in a filing with federal regulators.
```
