

[0043] A collection **300** of objects incorporating aspects of the present invention is shown in **FIG. 3**. The collection **300** has a data object **302** which relates to the actual object information. Additionally, the collection **300** has a meta information object **304**, which, in this particular example, comprises general information including some standard properties, such time or object size. In other embodiments, the meta information object **304** may not include standard properties, and instead the standard properties would be stored in yet another object (not shown).

[0044] The collection **300** also has a version-specific property **306** as shown in **FIG. 3**. The version-specific property **306** comprises three types of information. It has a meta information section **308**, which typically has a name for the property such as the name of the third party application used to create the property, and possibly other information, such as the length of the property, its address location on the disk, among others.

[0045] The version-specific property **306** also has a version information section **310**. The version information section **310** comprises information related to the version of the specific application that was used most recently to create the version-specific property **306**, such as the version of the third party application used to create the property. That is, since such version information may be relevant to the program module that is evaluating the property, a section is dedicated to this type of information.

[0046] The version-specific property **306** also comprises a mask information section **312** which is used to provide information to the server system as to what is a relevant update. The mask information section **312** essentially dictates to other applications or to the server system the protocol or policy in which events may or may not invalidate the version-specific property **306**.

[0047] In a particular embodiment, if the version-specific property **306** was created by a virus scan application, then the meta information section **308** may hold the name of the virus scan application used to create the version-specific property **306**. Additionally, the version information section **310** would include information related to the particular virus definition file used when the version-specific property **306** was created. Version information in this particular instance is important because virus scan applications are frequently updated to include recently detected viruses. Therefore virus scan applications should not only determine whether an

object has been scanned previously, but whether it has been scanned with an updated virus definition file.

[0048] With respect to the mask information section **312**, when a virus scan application creates the version-specific property **306**, the mask information section **312** may include the specific events that would cause an invalidation of the version-specific property **306**, wherein the events may be uniquely pertinent to a virus scan application. Such events that may cause an invalidation of property **306** in this particular instance may include modifying the data object **302**, either by adding new data or by erasing data. In other embodiments, other predetermined events may cause the invalidation of version-specific property, such as modification of another property associated with the data object **302**, among others.

[0049] In this particular example, predetermined events that may specifically not be included in the mask information section **312** and therefore do not cause the version-specific property to be invalidated may include read only access events, changing the name of the object, or backing up the object collection **300**. Indeed, if the event does not cause the version-specific property to be invalidated than the property remains associated with the object **302**, even in such cases as when the name of the object is changed.

[0050] The version-specific property **306** is created by a third party application, e.g., an application separate from the particular object store. In creating the version-specific property, the third party application supplies the name, version and mask information to the server system, where the XML object is stored. The server system then creates and associates the property with the particular data object. Alternatively, the version specific property may be created and stored on a remote computer system, as long as the property is associated with the data object **302**.

[0051] In an embodiment of the invention, the version specific property is an extension of the HTTP, as part of WebDAV, i.e., the World Wide Web Distributed Authoring and Versioning protocol (DAV). In essence, the version-specific property is a new type of DAV property and has the same live/dead degree of freedom as other DAV properties, i.e., where a live property is managed at a server and dead property is managed at the client. In order to define a new version specific property in DAV the document type definitions (DTD) shown in Table 1 may be implemented. Of course these samples could also be written as schemas.

1 Name: versionspecificproperty

Namespace: DAV:

Purpose: Specifies that a property is version-specific and defines its characteristics.

Description: The versionspecificproperty XML element specifies when a version-specific property is invalidated due to updates to the containing element and whether it should be preserved in COPY operations. If this XML element is not included in the request body then the server can assume that the property is not version-specific.

<!ELEMENT versionspecificproperty (copybehavior?, isreplicator?, invalidationrule?, contents)>

2 Name: copybehavior

Namespace: DAV:

Purpose: Specifies whether the property should be preserved across a COPY operation.

Description: The copybehavior XML element specifies whether a version-specific property is preserved across a COPY operation or not. If this XML element is not included in the request body then the property will be omitted in a COPY operation.