the appended data item in the data list, which the facility adds to the object descriptor in step **609** (FIG. **6**). These steps then conclude.

FIG. **8** is a flow diagram showing the Add Visual Element Descriptor routine, which is preferably called by the facility in step **611** (FIG. **6**) to add a visual element descriptor to the object list, if necessary. In steps **801–803**, the facility loops through the visual element descriptors in the visual element list. In step **802**, the facility determines whether the visual element descriptor to be added to the visual element list matches the visual element descriptor in the visual element list. This determination preferably involves determining whether these two items contain the same visual element-specific information (e.g., the visual element type in position) and references to all of the same data items in the data list. In step **802**, if the visual element descriptor to be added to the visual element list matches the visual element descriptor in the visual element list, then the facility returns a reference to this visual element descriptor in the visual element list, which is added to the page descriptor in step **612** (FIG. **6**), else the facility continues at step **803**. In step **803**, the visual element descriptor loop ends. If more visual element descriptors remain, then the facility continues at step **801**, else the facility continues at step **804**. In step **804**, because the visual element descriptor to be added does not match any visual element descriptor in the visual element list, the facility appends the visual element descriptor to be added to the visual element list. The facility then returns a reference to the appended visual element descriptor in the visual element list, which the facility adds to the page descriptor in step **612** (FIG. **6**). These steps then conclude. Those skilled in the art will recognize the general correspondence between the steps shown in FIG. **8** and those shown in FIG. **7**.

A document stored by the facility in a compressed document set may preferably be decompressed by locating the document descriptor for the document in the document list and following the references in the document descriptor to the pages of the document, the references to the visual element descriptors and the pages descriptors and the references in the visual element descriptors to the data items, copying each of these list entries into the decompressed document and establishing links between them.

A single page of a document stored by the facility in a compressed document set may similarly be decompressed by finding the document in the document list, finding the page descriptor for the appropriate page among the page descriptors referred to by the document descriptor, and following the references in the page descriptor to the visual element descriptors and from the visual element descriptors to the data items, copying each visual element descriptor or data item referred to by a traversed reference into the compressed document and establishing links between them.

While this invention has been shown and described with reference to preferred embodiments, it will be understood by those skilled in the art that various changes or modifications in form and detail may be made without departing from the scope of the invention. For example, computer documents of all types comprised of constituent data structures make it be compressed using the facility. Such documents need not be arranged in pages or have any of the other exemplary features discussed above in conjunction with the sample document data structures.

I claim:

1. A method in a computer system for compressing a level of a multiple-level source data structure representing one or more documents while generating a destination data structure from the source data structure, the source data structure made up of constituent data structures each at one of the levels of the source data structure, the method comprising the steps of:

selecting a level of the source data structure at which the source data structure is to be compressed, two or more of the constituent data structures at the selected level of the source data structure containing the same data, each constituent data structure at the selected level of the source data structure being referred to by a reference in a constituent data structure at the next higher level of the source data structure;

identifying all unique constituent data structures at the selected level of the source data structure that do not contain the same data;

storing the identified unique constituent data structures in the destination data structure; and

for each constituent data structure at the next higher level of the source data structure:

replacing a reference in the constituent data structure at the next higher level that refers to a constituent data structure at the selected level of the source data structure with a reference that refers to a constituent data structure of the selected level stored in the destination data structure that contains the same data as the constituent data structure at the selected level of the source data structure, and

storing the constituent data structure of the next higher level in the destination data structure.

2. A method in a computer system for compressing a data structure representing one or more computer documents, the data structure comprised of a multiplicity of constituent data structures connected by links, the method comprising the steps of:

identifying a plurality of matching constituent data structures;

deleting all but a selected one of the identified constituent data structures; and

replacing links to the deleted constituent data structures with links to the selected constituent data structure.

3. The method of claim **2** wherein each constituent data structure has contents, and wherein the identifying step determines that all of the constituent data structures identified as matching have the same contents.

4. The method of claim **3** wherein the contents of the identified constituent data structures include indications of links to one or more other constituent data structures, and wherein the determining step further determines that the indications of links in the identified constituent data structures indicate that the identified constituent data structures are each linked to the same other constituent data structures.

5. The method of claim **3** wherein the data structure comprises an acyclic directed graph of constituent data structures.

6. A method in a computer system for decompressing data representing a portion of a document, comprising the steps of:

retrieving a compressed data structure representing one or more documents, the compressed data structure comprising a plurality of constituent data structures connected by references, the constituent data structures