

These computer program instructions may also be stored in a computer-readable memory that can direct a computer or other programmable data processing apparatus to function in a particular manner, such that the instructions stored in the computer-readable memory produce an article of manufacture including instruction means which implement the function specified in the flowchart and/or block diagram block or blocks.

The computer program instructions may also be loaded onto a computer or other programmable data processing apparatus to cause a series of operational steps to be performed on the computer or other programmable apparatus to produce a computer-implemented process such that the instructions which execute on the computer or other programmable apparatus provide steps for implementing the functions specified in the flowchart and/or block diagram block or blocks.

While preferred embodiments of the present invention have been described, additional variations and modifications in those embodiments may occur to those skilled in the art once they learn of the basic inventive concepts. Therefore, it is intended that the appended claims shall be construed to include preferred embodiments and all such variations and modifications as fall within the spirit and scope of the invention.

What is claimed is:

1. A method of dynamically adding content to a displayed work unit view in a task-based interface for a Web-based application, the task-based interface provided in a content aggregation framework, comprising:

responsive to a request to execute a selected task, initially creating a work unit view to display the selected task in the content aggregation framework, comprising:

locating a template page definition that corresponds to the selected task;

invoking each of at least one content-creating software entity identified in the template page definition, the located template page definition further specifying a layout of the work unit view to display the selected task; and

rendering, in the work unit view according to the specified layout, content created by each of the at least one invoked content-creating software entity;

displaying the work unit view in the content aggregation framework;

responsive to a request for dynamic invocation, from the displayed work unit view, of an additional content-creating software entity that is not identified in the template page definition, dynamically updating the work unit view to include content resulting from dynamically invoking the additional content-creating software entity, comprising:

creating a container, the container comprising a work unit view area and a row for the resulting content;

rendering, in the work unit view area of the container, the currently-displayed work unit view;

invoking the additional content-creating software entity; and

rendering, in the row of the container, the content resulting from invoking the additional content-creating software entity; and

displaying the dynamically-updated work unit view, as a replacement for the currently-displayed work unit view, using the container.

2. The method according to claim 1, wherein the row in the container is selectably closable to cause removing, from the

dynamically-updated work unit view, the content resulting from invoking the additional content-creating software entity.

3. The method according to claim 2, wherein an instance of the additional invoked content-creating software entity is destroyed responsive to closing the row in the dynamically-updated work unit view.

4. The method according to claim 2, wherein resources associated with the additional invoked content-creating software entity are programmatically released responsive to closing the row in the dynamically-updated work unit view.

5. The method according to claim 2, wherein the row is selectably closable by selecting, from the dynamically-updated work unit view, a graphical control corresponding to the row.

6. The method according to claim 1, further comprising saving the new container and associated state information, along with contextual information associated with the additional invoked content-creating software entity, at least temporarily, for subsequent recall and display of the dynamically-updated work unit view.

7. The method according to claim 1, wherein the content-creating software entities are portlets.

8. The method according to claim 1, wherein the content aggregation framework is a portal.

9. The method according to claim 1, further comprising: updating contextual information used by the additional invoked content-creating software entity when initially creating the resulting content, responsive to receiving an update trigger;

rendering, in the row of the container, an updated version of the resulting content, the updated version created by subsequently invoking the additional content-creating software entity using the updated contextual information; and

re-displaying the dynamically-updated work unit view using the updated version of the resulting content rendered in the row of the container.

10. The method according to claim 1, wherein: the template page definition further specifies a number of columns to use when dynamically adding content to the row in the work unit view; and

the rendering, in the row of the container, the content resulting from invoking the additional content-creating software entity comprises rendering, in the row of the container, the resulting content into a first-available one of the specified number of columns within the row.

11. The method according to claim 1, further comprising saving, at least temporarily, contextual information associated with each of the at least one content-creating software entity identified in the template page definition and contextual information associated with the additional invoked content-creating software entity, thereby enabling each of the content-creating software entities to restore itself upon subsequent recall for display of the dynamically-updated work unit view.

12. The method according to claim 11, wherein the saving operates responsive to a request to close the displayed dynamically-updated work unit view.

13. A system for dynamically adding content to a displayed work unit view in a task-based interface for a Web-based application, the task-based interface provided in a content aggregation framework, comprising:

a computer comprising a processor; and instructions which are executable, using the processor, to perform: