

(19)



**Евразийское
патентное
ведомство**

(11) **033675**(13) **B1**

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ЕВРАЗИЙСКОМУ ПАТЕНТУ

(45) Дата публикации и выдачи патента
2019.11.15

(51) Int. Cl. **G06F 17/00** (2006.01)

(21) Номер заявки
201690498

(22) Дата подачи заявки
2014.09.11

**(54) СИСТЕМА И СПОСОБ АВТОМАТИЗИРОВАННОГО ПРЕОБРАЗОВАНИЯ
ИНТЕРАКТИВНЫХ САЙТОВ И ПРИЛОЖЕНИЙ ДЛЯ ПОДДЕРЖКИ МОБИЛЬНЫХ И
ДРУГИХ СРЕД ОТОБРАЖЕНИЯ**

(31) **61/876,795**

(56) US-A1-20100235411

(32) **2013.09.12**

US-A1-20070016577

(33) **US**

US-A1-20050235248

(43) **2016.09.30**

US-A1-20090049372

(86) **PCT/IB2014/064448**

US-A1-20110216827

(87) **WO 2015/036962 2015.03.19**

US-A1-20110153574

(71)(73) Заявитель и патентовладелец:
ВИКС.КОМ ЛТД. (IL)

US-A-5986654

US-A1-20060184642

US-A1-20070143750

US-A1-20070282869

US-A1-20070294628

US-A1-20080307049

US-A1-20130167012

(72) Изобретатель:
**Бен-Ахарон Рони, Абрахам Надав
(IL)**

(74) Представитель:
**Поликарпов А.В., Соколова М.В.,
Путинцев А.И., Черкас Д.А., Игнатъев
А.В. (RU)**

(57) Система и способ для преобразования компоновки страницы веб-сайта в системе построения веб-сайта, имеющая по меньшей мере один процессор для обработки инструкций, препроцессор, включающий приемник для приема страницы веб-сайта из системы построения веб-сайта, при этом страница имеет исходную компоновку для исходной области отображения и представление компонентов в виде объектной модели, и блоки для определения порядка и набора соотношений между компонентами и для создания целевой компоновки страницы веб-сайта для целевой области отображения, при этом целевая компоновка основана на семантическом анализе и согласовании определенного порядка и набора соотношений, причем исходная область отображения имеет атрибуты, отличные от атрибутов целевой области отображения.

B1**033675****033675****B1**

Перекрестная ссылка на родственные заявки

Заявка на данное изобретение основана на предварительной патентной заявке США № 61/876795, поданной 12 сентября 2013 г., которая, таким образом, включена в полный объем в состав настоящей заявки путем ссылки.

Область техники, к которой относится изобретение

Настоящее изобретение относится к преобразованию интерактивных приложений и, в частности, к мобильным платформам.

Уровень техники

С распространением интернета в последние годы пользователи все больше обращаются к веб-сайтам и интерактивным приложениям с использованием смартфонов, планшетов и других мобильных устройств. Эти устройства постепенно заменяют или дополняют как полнофункциональный настольный персональный компьютер, так и более старые, менее функциональные "телефоны с дополнительными возможностями". Это касается веб-сайтов, присутствующих во всемирной паутине, а также других интерактивных приложений, которые теперь можно получать из многочисленных магазинов приложений, обеспеченных ведущими компаниями, например Apple, Google, Microsoft и Amazon.

Веб-сайты и интерактивные приложения имеют разные форм-факторы дисплея и характеристики в зависимости от того, где они просматриваются, например на настольном ПК, компактном мобильном устройстве и мобильном планшете среднего размера.

Сущность изобретения

В соответствии с предпочтительным вариантом осуществления настоящего изобретения предусмотрен преобразователь, реализуемый посредством системы клиент/сервер, имеющей по меньшей мере один процессор для обработки инструкций, задающих преобразователь. Преобразователь включает приемник для приема страницы веб-сайта из системы построения веб-сайта, при этом упомянутая страница имеет исходную компоновку для исходной области отображения и представление компонентов в виде объектной модели. Преобразователь также включает процессор для определения порядка и набора соотношений между компонентами и для создания целевой компоновки страницы веб-сайта для целевой области отображения, при этом целевая компоновка основана на семантическом анализе и согласовании определенного порядка и набора соотношений, и где исходная область отображения имеет атрибуты, отличные от атрибутов целевой области отображения.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения преобразователь включает дополнительный преобразователь для объединения модификаций в исходной компоновке и целевой компоновке с учетом независимых модификаций, сделанных в целевой компоновке для создания обновленной конфигурации целевой компоновки.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения процессор включает препроцессор для анализа свойств объектов компонентов, и для модификации их пригодности для представления на целевой области отображения, блок создания суперузла для определения местоположения групп компонентов, которые должны оставаться вместе, и для создания иерархии суперузлов на основании местоположения компонентов и соотношений содержания компонентов. Он также включает блок упорядочения для определения порядка для элементов в каждом из суперузлов, блок размещения для размещения элементов в целевой компоновке на основании по меньшей мере одного из определенного порядка и групп, и постпроцессор для регулировки размещенных элементов до отображения.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения компоненты представляют собой по меньшей мере один из атомарных компонентов, компонентов одностраничного контейнера и компонентов многостраничного контейнера.

Дополнительно, в соответствии с предпочтительным вариантом осуществления настоящего изобретения преобразователь реализован на по меньшей мере одном из следующего: клиент, сервер и сторонний сервер.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения препроцессор включает по меньшей мере одно из следующего: обработчик шаблонов для модификации экземпляров компонентов в шаблонах, блок фильтрации компонентов для фильтрации компонентов, непригодных для целевой компоновки, блок адаптации компонентов для адаптации компонентов к целевой компоновке, блок регулировки компонентов для регулировки компонентов к целевой компоновке и блок анализа компонентов для анализа атрибутов компонентов для определения их пригодности к использованию.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения блок создания суперузла включает блок определения местоположения группы перекрытия для определения местоположения групп компонентов, в которых компоненты сильно перекрываются, и для замены группы виртуальным элементом согласно критерию группировки блок определения местоположения заранее определенной группы для определения местоположения групп компонентов согласно рекомендациям по меньшей мере от одного из уровней шаблонов, приложений, страниц или компонентов, и для замены группы виртуальным элементом согласно критерию группировки. Блок создания суперузла так-

же включает блок определения местоположения группы текста поверх изображения для определения местоположения компонентов, где компоненты являются текстовыми компонентами, которые наложены поверх конкретного фонового изображения для замены группы виртуальным элементом согласно критерию группировки, блок определения показателя для определения показателя определенности для корректности блока определения местоположения группы перекрытия, блока определения местоположения заранее определенной группы и блока определения местоположения группы текста поверх изображения, и блок создания узла для создания иерархии суперузлов на основании компонентов и виртуальных элементов.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения блок упорядочения включает блок основного упорядочения для определения последовательности элементов суперузлов, согласно заранее определенному критерию упорядочения, блок определения местоположения частично упорядоченного набора для анализа по меньшей мере одного из семантик, содержания, атрибутов, истории редактирования и геометрии элементов суперузлов, блок определения показателя для определения показателя определенности для корректности блока основного упорядочения и блока определения местоположения частично упорядоченного набора, и интегратор порядка для интеграции последовательности, определенной блоком основного упорядочения, с обнаруженными частично упорядоченными наборами, местоположение которых определено блоком определения местоположения частично упорядоченного набора для создания объединенного модифицированного порядка.

Дополнительно, в соответствии с предпочтительным вариантом осуществления настоящего изобретения постпроцессор включает блок вставки автоматически добавляемых компонентов для вставки по меньшей мере одного из автоматически добавляемых компонентов и компонентов, относящихся к мобильным устройствам, и координатор динамической компоновки для осуществления по меньшей мере одного из создания, модификации и удаления привязок динамической компоновки, где привязки регулируются согласно целевой компоновке.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения блок основного упорядочения включает по меньшей мере одно из следующего: блок упорядочения первичного направления для выстраивания в последовательность элементов суперузлов согласно по меньшей мере одной из строк и столбцов, блока упорядочения первичного направления с разделением и объединением для выстраивания в последовательность элементов суперузлов согласно по меньшей мере одной из строк и столбцов, и для отслеживания любого разделения и объединения строк/столбцов, и блока горизонтального/вертикального расслоения для чередующегося горизонтального и вертикального расслоения элементов суперузлов для создания внутреннего дерева делений и для определения последовательности отображения для элементов.

Дополнительно, в соответствии с предпочтительным вариантом осуществления настоящего изобретения блок размещения включает интерпретатор рекомендации для интерпретации рекомендаций, присоединенных к элементам суперузлов, блок создания разрыва линии для создания разрывов линии согласно рекомендациям, созданным блоком создания узла и блоком упорядочения, блок регулировки размера для применения по меньшей мере одной из регулировок ширины и высоты к элементам суперузлов, и блок изменения размера для изменения размера декоративных изображений.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения блок определения местоположения частично упорядоченного набора включает блок определения местоположения частично упорядоченного набора кластеров для обнаружения частично упорядоченного набора кластеров, причем элементы суперузлов располагаются ближе друг к другу по сравнению со стандартным разнесением между элементами в суперузлах, блок определения местоположения частично упорядоченного семантического набора для обнаружения частично упорядоченного набора семантических соотношений при наличии конкретных комбинаций элементов суперузлов данных типов, близких друг к другу, блок определения местоположения частично упорядоченного набора схем для обнаружения частично упорядоченного набора при наличии установленных схем между элементами суперузлов, блок определения местоположения заранее определенного частично упорядоченного набора для обнаружения частично упорядоченного набора на основании по меньшей мере одного из существующих определений группирования, относящегося к редактированию, информации существующей привязки динамической компоновки и повторного использования шаблонов компонентов для разных наборов компонентов, и блок определения местоположения частично упорядоченного набора на основе информации сеанса редактирования для обнаружения частично упорядоченного набора элементов суперузлов на основании информации, собранной из предыдущих сеансов редактирования.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения по меньшей мере один из автоматически добавляемых компонентов и компонентов, относящихся к мобильным устройствам, включает по меньшей мере одно из виджетов, характерных для целевого устройства, навигационных меню, рекламы и популяризирующего материала.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения преобразователь имеет блок определения местоположения частично упорядоченного набора схем, и при этом блок горизонтального/вертикального расслоения включает блок деления элементов для определе-

ния направления расслоения элементов суперузлов на основании по меньшей мере одного из количества разделителей, размера промежутков, найденных в данном направлении проекции, частично упорядоченного набора схем, местоположение которого определено, и качества выравнивания с осевыми проекциями в обоих направлениях.

Дополнительно, в соответствии с предпочтительным вариантом осуществления настоящего изобретения блок фильтрации компонентов включает блок удаления для удаления компонентов, непригодных для отображения целевой компоновки, и блок сокрытия для сокрытия компонентов, непригодных для отображения целевой компоновки.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения блок адаптации компонентов включает блок модификации для модификации размера и ширины компонентов, блок объединения для объединения компонентов меню, блок обновления содержания для обновления адаптации, относящихся к содержанию, блок создания для создания составных меню, и блок преобразования символов для преобразования символов на основе графики.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения блок регулировки компонентов включает блок создания схемы для создания схемы размера шрифта между исходной компоновкой и целевой компоновкой.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения блок анализа компонентов включает блок сшивания изображения для сшивания компонентов, когда компоненты являются компонентами изображения, в единое изображение, и обработчик декоративных изображений для различения декоративных изображений.

Дополнительно, в соответствии с предпочтительным вариантом осуществления настоящего изобретения дополнительный преобразователь включает блок удаления страницы для удаления страницы из целевой компоновки, когда страница удалена из исходной компоновки, блок добавления страницы для добавления страницы в целевую компоновку, когда страница добавлена в исходную компоновку, блок удаления компонента для удаления компонента из целевой компоновки, когда компонент удален из исходной компоновки, блок добавления компонента для добавления компонента в целевую компоновку, когда компонент добавлен в исходную компоновку, блок модификации компонента для модификации компонента из целевой компоновки, когда компонент модифицирован в исходной компоновке, и мобильный обработчик для обработки модификаций в целевой компоновке, причем модификации не зависят от модификаций в исходной компоновке.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения блок добавления компонента включает блок поиска родителя/предшественника для поиска по меньшей мере одного из ближайшего предшествующего компонента и родительского компонента, ближайших к компоненту, добавленному к исходной компоновке, и блок добавления мобильной компоновки для вставки добавленного компонента в целевую компоновку согласно местоположению по меньшей мере одного из ближайшего предшествующего компонента и родительского компонента.

В соответствии с предпочтительным вариантом осуществления настоящего изобретения предусмотрен способ, реализуемый на вычислительном устройстве, причем способ включает прием страницы веб-сайта из системы построения веб-сайта, при этом страница имеет исходную компоновку для исходной области отображения и представление компонентов в виде объектной модели, и определение порядка и набора соотношений между компонентами и создание целевой компоновки страницы веб-сайта для целевой области отображения, причем целевая компоновка основана на семантическом анализе и согласовании определенного порядка и набора соотношений, где исходная область отображения имеет атрибуты, отличные от атрибутов целевой области отображения.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения способ также включает объединение модификации в исходной компоновке и целевой компоновке с учетом независимых модификаций, сделанных в целевой компоновке для создания обновленной конфигурации целевой компоновки.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения определение и создание включают анализ свойств объектов компонентов и модификацию их пригодности для представления на целевой области отображения, определение местоположения групп компонентов, которые должны оставаться вместе, и создание иерархии суперузлов на основании местоположения компонентов и соотношений содержания компонентов. Они также включают определение порядка для элементов в каждом из суперузлов, размещение элементов в целевой компоновке на основании по меньшей мере одного из определенного порядка и групп, и регулировку размещенных элементов до отображения.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения компоненты представляют собой по меньшей мере один из атомарных компонентов, компонентов одностраничного контейнера и компонентов многостраничного контейнера.

Дополнительно, в соответствии с предпочтительным вариантом осуществления настоящего изобретения анализ и модификация включают по меньшей мере одно из модификации экземпляров компонентов в шаблонах, фильтрации компонентов, непригодных для целевой компоновки, адаптации компонен-

тов к целевой компоновке, регулировки компонентов к целевой компоновки, и анализа атрибутов компонентов для определения их пригодности к использованию.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения определение местоположения и создание включают определение местоположения групп компонентов, в которых компоненты сильно перекрываются, и замену группы виртуальным элементом согласно критерию группировки, определение местоположения групп компонентов согласно рекомендациям по меньшей мере от одного из уровней шаблонов, приложений, страниц или компонентов, и замены группы виртуальным элементом согласно критерию группировки, определение местоположения компонентов, причем компоненты представляют собой текстовые компоненты, которые наложены поверх конкретного фонового изображения, и замену группы виртуальным элементом согласно критерию группировки, определение показателя определенности для корректности определения местоположения групп компонентов, в которых компоненты сильно перекрываются, определение местоположения групп компонентов согласно рекомендациям по меньшей мере от одного из уровней шаблонов, приложений, страниц или компонентов, и определение местоположения компонентов, причем компоненты представляют собой текстовые компоненты, которые наложены поверх конкретного фонового изображения, и создание иерархии суперузлов на основании компонентов и виртуальных элементов.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения определение порядка включает определение последовательности элементов суперузлов согласно заранее определенному критерию упорядочения, и анализ по меньшей мере одного из семантик, содержания, атрибутов, истории редактирования и геометрии элементов суперузлов, определение показателя определенности для корректности определения последовательности элементов суперузлов согласно заранее определенному критерию упорядочения, и упомянутому анализу по меньшей мере одного из семантик, содержания, атрибутов, истории редактирования и геометрии элементов суперузлов, и интеграцию последовательности, определенной путем определения показателя определенности для корректности определения последовательности элементов суперузлов согласно заранее определенному критерию упорядочения, и упомянутому анализу по меньшей мере одного из семантик, содержания, атрибутов, истории редактирования и геометрии элементов суперузлов и создание объединенного модифицированного порядка.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения регулировка включает вставку по меньшей мере одного из автоматически добавляемых компонентов и компонентов, относящихся к мобильным устройствам, и осуществление по меньшей мере одного из создания, модификации и удаления привязок динамической компоновки, причем привязки регулируются согласно целевой компоновке.

Дополнительно, в соответствии с предпочтительным вариантом осуществления настоящего изобретения определение последовательности элементов суперузлов согласно заранее определенному критерию упорядочения включает по меньшей мере одно из: выстраивание в последовательность элементов суперузлов согласно по меньшей мере одной из строк и столбцов, выстраивание в последовательность элементов суперузлов согласно по меньшей мере одной из строк и столбцов и отслеживание любого разделения и объединения строк/столбцов, и перемежение горизонтального и вертикального расслоения элементов суперузлов и создание внутреннего дерева делений, и определение последовательности отображения для элементов.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения размещение включает интерпретацию рекомендаций, присоединенных к элементам суперузлов, создание разрывов линии согласно рекомендациям, созданным путем определения местоположения групп компонентов, которые должны оставаться вместе, и упомянутого создания иерархии суперузлов на основании местоположения компонентов и соотношений содержания компонентов, применение по меньшей мере одной из регулировок ширины и высоты к элементам суперузлов, и изменение размеров декоративных изображений.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения определение последовательности элементов суперузлов согласно заранее определенному критерию упорядочения включает обнаружение частично упорядоченного набора кластеров, причем элементы суперузлов располагаются ближе друг к другу по сравнению со стандартным разнесением между элементами в суперузлах, обнаружение частично упорядоченного набора семантических соотношений при наличии конкретных комбинаций элементов суперузлов данных типов, которые близки друг к другу, обнаружение частично упорядоченного набора при наличии установленных схем между элементами суперузлов, обнаружение частично упорядоченного набора на основании по меньшей мере одного из существующих определений группирования относящегося к редактированию, информации существующей привязки динамической компоновки и повторного использования шаблонов компонентов для разных наборов компонентов и обнаружение частично упорядоченного набора элементов суперузлов на основании информации, собранной из предыдущих сеансов редактирования.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения по меньшей мере один из автоматически добавляемых компонентов и компонентов, относящихся к мо-

бильным устройствам, включает по меньшей мере одно из виджетов, характерных для целевого устройства, навигационных меню, рекламы и популяризирующего материала.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения обнаружение частично упорядоченного набора при наличии установленных схем между элементами суперузлов и когда перемежение горизонтального и вертикального расслоения элементов суперузлов включает определение направления расслоения элементов суперузлов на основании по меньшей мере одного из количества разделителей, размера промежутков, найденных в данном направлении проекции, обнаружения частично упорядоченного набора при наличии установленных схем между элементами суперузлов и качества выравнивания с осевыми проекциями в обоих направлениях.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения фильтрация включает удаление компонентов, непригодных для отображения целевой компоновки, и сокрытие компонентов, непригодных для отображения целевой компоновки.

Дополнительно, в соответствии с предпочтительным вариантом осуществления настоящего изобретения адаптация включает модификацию размера и ширины компонентов, объединение компонентов меню, обновление адаптации, относящихся к содержанию, создание составных меню и преобразование символов на основе графики.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения регулировка включает схему размера шрифта между исходной компоновкой и целевой компоновкой.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения анализ включает сшивание компонентов, когда компоненты являются компонентами изображения, в единое изображение, и различение декоративных изображений.

Кроме того, в соответствии с предпочтительным вариантом осуществления настоящего изобретения дополнительное преобразование включает удаление страницы из целевой компоновки, когда страница удалена из исходной компоновки, добавление страницы в целевую компоновку, когда страница добавлена в исходную компоновку, удаление компонента из целевой компоновки, когда компонент удален из исходной компоновки, добавление компонента в целевую компоновку, когда компонент добавлен в исходную компоновку, модификацию компонента из целевой компоновки, когда компонент модифицирован в исходной компоновке, и обработку модификаций в целевой компоновке, причем модификации не зависят от модификаций в исходной компоновке.

Дополнительно, в соответствии с предпочтительным вариантом осуществления настоящего изобретения добавление компонента включает поиск по меньшей мере одного из ближайшего предшествующего компонента и родительского компонента, ближайших к компоненту, добавленному к исходной компоновке, и вставку добавленного компонента в целевую компоновку согласно местоположению по меньшей мере одного из ближайшего предшествующего компонента и родительского компонента.

Краткое описание чертежей

Предмет изобретения конкретно указан и отдельно заявлен в заключительной части описания изобретения. Однако изобретение, в отношении организации и способа работы, совместно с его задачами, признаками и преимуществами можно наилучшим образом понять, обратившись к нижеследующему подробному описанию, приведенному совместно с прилагаемыми чертежами, на которых показано:

фиг. 1 - схематическая иллюстрация системы для преобразования визуальных приложений между платформами, построенной и действующей в соответствии с настоящим изобретением;

фиг. 2 - схематическая иллюстрация элементов преобразователя начальной компоновки, построенного и действующего в соответствии с настоящим изобретением;

фиг. 3 - схематическая иллюстрация привязки динамической компоновки, разрывающейся при перемещении компонентов;

фиг. 4 - схематическая иллюстрация, показывающая, почему вертикальные линии неуместны на дисплее мобильного устройства;

фиг. 5 - схематическая иллюстрация элементов блока создания суперузла, построенного и действующего в соответствии с настоящим изобретением;

фиг. 6 - иллюстрация необходимого переформатирования текста вследствие изменения размера шрифта и размера поля;

фиг. 7 - изображение, которое выглядит неправильно при изменении размера без сохранения форматного соотношения;

фиг. 8А, 8В и 8С - схематические иллюстрации схемы между визуальным приложением и соответствующей структурой суперузла, осуществляемого в соответствии с настоящим изобретением;

фиг. 9 - схематическая иллюстрация компонентов в контейнерах до перестановки;

фиг. 10 - схематическая иллюстрация компонентов, показанных на фиг. 9, после перестановки;

фиг. 11 - схематическая иллюстрация элементов блока упорядочения, построенного и действующего в соответствии с настоящим изобретением;

фиг. 12 - схематическая иллюстрация расположения четырех абзацев текста с несколькими возможными порядками чтения;

фиг. 13 - схематическая иллюстрация расположения, которое включает два абзаца текста и два изображения, с несколькими возможными порядками чтения;

фиг. 14А-14С - схематические иллюстрации преобразования набора элементов в суперузле в граф элементов и затем порядок элементов, осуществляемого в соответствии с настоящим изобретением;

фиг. 15А-15С - алгоритм, демонстрирующий функциональные возможности блока упорядочения первичного направления с разделением и объединением, построенный и действующий в соответствии с настоящим изобретением;

фиг. 16А-16Н - схематические иллюстрации процесса работы блока упорядочения первичного направления с разделением и объединением, показанного на фиг. 15А-15С, построенного и действующего в соответствии с настоящим изобретением;

фиг. 17 - пример компоновки веб-сайта;

фиг. 18 - схематическая иллюстрация горизонтального и вертикального делений суперузлов, построенных и действующих в соответствии с настоящим изобретением;

фиг. 19 - пример оценивания вертикального и горизонтального делений на основании предыдущего анализа подобия схем, построенного и действующего в соответствии с настоящим изобретением;

фиг. 20 - схематическая иллюстрация предпочтительности вертикального деления суперузлов ввиду того, что компоненты лучше выравниваются по горизонтальным линиям, построенных и действующих в соответствии с настоящим изобретением;

фиг. 21 - схематическая иллюстрация конфигурации взаимосвязанных элементов;

фиг. 22 - схематическая иллюстрация пар компонентов, где компоненты меняются местами;

фиг. 23 - схематическая иллюстрация того, как четыре компонента изображения можно разделить на пары двумя способами;

фиг. 24 - схематическая иллюстрация прямоугольной формы дисплея, оставшейся после исключения автоматически добавляемых компонентов;

фиг. 25 - схематическая иллюстрация элементов дополнительного преобразователя, построенного и действующего в соответствии с настоящим изобретением;

фиг. 26 - схематическая иллюстрация элементов блока добавления компонента, построенного и действующего в соответствии с настоящим изобретением;

фиг. 27 - схематическая иллюстрация функциональных возможностей дополнительного преобразователя, показанного на фиг. 25, построенного и действующего в соответствии с настоящим изобретением;

фиг. 28А-28G - схематические иллюстрации размещения компонентов при преобразовании или дополнительном преобразовании веб-страницы из конфигурации настольной компоновки в конфигурацию мобильной компоновки, построенного и действующего в соответствии с настоящим изобретением;

фиг. 29 - схематическая иллюстрация способов вставки компонентов в мобильную версию веб-страницы; и

фиг. 30 - схематическая иллюстрация удаления компонентов из мобильной версии веб-страницы.

Очевидно, что для простоты и наглядности иллюстрации, элементы, показанные на чертежах, не обязательно изображены в масштабе. Например, размеры некоторых элементов для наглядности могут быть преувеличены относительно других элементов. Кроме того, когда это целесообразно, ссылочные позиции могут повторяться от чертежа к чертежу для указания соответствующих или аналогичных элементов.

Подробное описание настоящего изобретения

В нижеследующем подробном описании, изложены многочисленные конкретные детали для обеспечения полного понимания изобретения. Однако специалистам в данной области техники понятно, что настоящее изобретение можно осуществить на практике без этих конкретных деталей. В других случаях, общеизвестные способы, процедуры и компоненты не были подробно описаны, чтобы не усложнять описание настоящего изобретения.

Заявители установили, что существующие сайты и приложения обычно разрабатываются для настольных РС, где широко применяются дисплеи большого размера с высоким разрешением. При обращении к таким сайтам и приложениям с использованием дисплея меньшего размера возникают многочисленные проблемы, включающие (например) необходимость в чрезмерной прокрутке (в частности, горизонтальной прокрутке); затрудненность навигации; размеры шрифта не адаптированы к мобильным устройствам; конструкции и компоненты отображения, которые не адаптированы к сенсорным экранам мобильных устройств, и т.д.

Поскольку эти сайты можно просматривать на платформах разных размеров, например планшетах и мобильных телефонах, разработчику требуется создавать множественные версии сайта или приложения для поддержки этих множественных сценариев использования и для поддержки внешнего вида и удобства использования первоначального сайта. Эти версии могут отличаться в различных аспектах, включающих используемые визуальную компоновку, компоненты отображения (виджеты), ориентацию сенсорного экрана и далее.

Заявители также установили, что существующие системы для преобразования видов веб-сайта для согласования с подходящей платформой просмотра обычно предназначены для решения задач преобразования веб-сайтов, разработанных с использованием языков разметки, например HTML (языка гипертекстовой разметки) и XML (расширяемого языка разметки). Это могут быть сайты, разработанные непосредственно на данном языке разметки или созданные с использованием генератора кода, который создает файлы на языке разметки для последующего размещения.

Существующие системы также предназначены для разбора первоначального веб-сайта, извлечения из него содержательной информации и создания модифицированного мобильного сайта. Такая технология обычно происходит от систем оптического распознавания символов (OCR) и анализа страниц, которые предназначены для извлечения содержания, а не адаптации содержания и конструкции. Таким образом, существующие системы обычно действуют для извлечения содержания и обеспечения его читаемости на устройстве меньшего размера, устраняя при этом большую часть конструкции и ухудшая внешний вид и удобство использования первоначального сайта.

Очевидно, что анализ существующих веб-сайтов может быть сложным. В частности, некоторые части сайта могут динамически генерироваться, и соотношение между элементами сайта также можно реализовать процедурно. Таким образом, трудно точно понять сайт, если не осуществлять преобразование в ходе работы сайта (например, через консольный браузер, выполняющийся на сервере преобразования). Однако этот последний способ имеет значительные издержки и, в частности, может требовать, чтобы данную страницу можно было адаптировать к размеру мобильного устройства каждый раз, когда она извлекается каждым пользователем вместо того, чтобы преобразовывать один раз и повторно использовать результаты преобразования для всех пользователей.

Другая проблема существующих систем состоит в поддержке изменений и редактирования первоначального веб-сайта после преобразования. Очевидно, что некоторые системы разработаны в попытке сравнения самой недавней версии страницы с предыдущими версиями той же страницы, и стараются определять местоположения различий. Это осуществляется с использованием правил сравнения на основании подобия текста, позиции в исходном файле, используемого шрифта и т.д. Это часто требует, чтобы разработчик вручную создавал связь между мобильной версией и элементами первоначальной настольной версии.

Однако, поскольку редактирование страницы осуществляется отдельно от процесса преобразования, две версии сайта могут значительно отличаться. Например, над сайтом может начать работать новый разработчик (имеющий совершенно другой стиль). Альтернативно, в сайт может быть встроена новая технология или библиотека. Существующие системы могут пытаться сравнивать две версии сайта, которые могут очень сильно отличаться (внутренне) друг от друга, хотя выглядят одинаково. Сравнение двух версий одного и того же веб-сайта может осуществляться эвристически, но не позволяет согласовывать существенно модифицированные элементы сайта.

Заявители установили, что вышеупомянутые ограничения можно преодолеть с использованием объектно-ориентированной системы визуального проектирования. Такая система может использоваться для создания интерактивных приложений и веб-сайтов и может использоваться для поддержки и создания определений параллельной компоновки между разными платформами отображения для данного приложения. Система визуального проектирования также может обеспечивать внутреннюю, согласованную модель объектных данных сайта, с точно указанными атрибутами для всех объектов сайта. Таким образом, система визуального проектирования может обеспечивать множественные визуальные компоновки для одного сайта или приложения с единым набором компонентов.

Заявители также установили, что использование такой системы визуального проектирования может включать собранную информацию истории сеансов редактирования, которую можно использовать для обнаружения изменений компонентов и соотношений компонентов. Таким образом, например, если разработчик дублирует пару объектов (например, изображение и подпись) несколько раз, можно заключить, что такая пара может быть связана и что существует вероятность того, что члены пары остаются близко друг к другу при преобразовании к мобильной версии. Таким образом, контейнерные объекты и т.д. можно преобразовывать, поддерживая при этом свои первоначальные иерархию и соотношения объектов. Другая информация может включать объекты, которые редактируются или перемещаются совместно, объекты, которые исторически сгруппированы, объекты или наборы объектов, которые созданы дублированием ранее существовавших объектов или наборов объектов и планированием по времени редактирования объекта, т.е. применялось ли конкретное изменение последовательно к определенному поднабору объектов.

Системы визуального проектирования обычно имеют уникальный идентификатор (ID) для каждого объекта. Такие ID могут использоваться для согласования разных версий сайта и согласованной поддержки изменений сайта. Кроме того, система визуального проектирования может использовать динамическую компоновку, включающую привязки между отображаемыми компонентами, которые управляют сделанными в их отношении изменениями компоновки. Такие привязки можно использовать для вывода дополнительной информации группирования и компоновки, и также можно использовать для тонкой настройки компоновки на разные размеры дисплеев мобильного устройства. Подсистема преобразования

может фактически автоматически генерировать дополнительные привязки динамической компоновки для дальнейшего их использования в ходе отображения.

Кроме того, с распространением смартфонов полномасштабный и передовой HTML-просмотр, аналогичный доступному на настольном компьютере, также стал обычным в мобильном окружении. Таким образом, необходима система, которая может использовать аналогичную обработку для всех платформ (настольной, планшетной и мобильной) для использования передовых браузеров, доступных на всех платформах.

Очевидно, что визуальные приложения могут быть автономными системами, например PowerPoint Presentation Program, коммерчески доступной от Microsoft Inc., или могут быть включены в более крупную систему редактирования, например Microsoft Word AutoShape Editor, также коммерчески доступную от Microsoft Inc. Такие приложения обычно содержат страницы и компоненты, которые могут дополнительно располагаться в иерархии контейнеров (одностраничных и многостраничных) внутри страницы, содержащей атомарные компоненты. Многостраничный контейнер также может отображать множественные мини-страницы.

Страницы также могут включать приложения списка (например, рассмотренные в патентной заявке США № 14/207,761 под названием "WEBSITE BUILDING SYSTEM INTEGRATING DATA LISTS WITH DYNAMIC CUSTOMIZATION AND ADAPTATION", поданной 13 марта 2014 г. и принадлежащей заявителю) и сторонние приложения. Страницы также могут использовать такие шаблоны, как общие шаблоны страницы или шаблоны компонентов. В частном случае используется главная страница приложения, содержащая компоненты, продублированные во всех остальных стандартных страницах. Расположение компонентов на странице или наборе страниц может именоваться компоновкой. Очевидно, что ниже рассмотрены компоновки, состоящие из прямоугольных компонентов и контейнеров, ориентированных параллельно осям. Другие компоновки также могут включать непрямоугольные компоненты и контейнеры, включающие, в частности, классы объектов, например повернутые или перекошенные объекты и объекты, состоящие из множественных участков, например компонент видеоплеера, который может иметь область отображения видео и область управления видео. Такие участки могут быть соединенными, разъединенными или пересекающимися. Другие компоновки также могут включать объекты, состоящие из произвольной геометрической фигуры.

Очевидно, что обработка таких непрямоугольных многоучастковых объектов может осуществляться с использованием охватывающего прямоугольника для каждого непрямоугольного объекта или путем адаптации геометрических примитивов для обработки протяженных непрямоугольных объектов. Адаптация может включать проекцию на ось, расстояние между фигурами, минимальное/максимальное направленное расстояние между фигурами (т.е. расстояние по вертикали или по горизонтали), обнаружение пересечения фигур, обнаружение пересечения фигур/линий, вычисление участка пересечения и вычисление площади фигуры.

Очевидно также, что каждое приложение может иметь множественные конфигурации компоновки, например, настольную (книжную и альбомную) и мобильную. Некоторые свойства компонентов веб-сайта также могут иметь значение компоновки для каждой конфигурации, например включение фактического компонента (например, конкретный компонент может быть чисто мобильным, или чисто настольным/планшетным). Это может быть контейнерный компонент - все содержащееся в нем совместно подвергается влиянию. Другие свойства могут включать размер (h, w), позицию (x, y), информацию Z-порядка, стиль (шрифт, размер, цвет), цвет, полиморфный тип компонента (например, данная галерея может использовать разные типы галереи для настольных и мобильных компонентов списка), настройки меню/галереи (например, галерея сетки # столбцов/строк), вариант стороннего приложения (для сторонних приложений, которые имеют множественные варианты, отличающиеся размерами дисплея) и ассоциацию вида приложения списка - для данного типа предмета или предмета, связанного с видом, для разных компонентов списка могут быть указаны разные виды.

Также очевидно, что каждая конфигурация компоновки может иметь принятое по умолчанию значение размера экрана (в пикселях). Фактический размер экрана может до некоторой степени отличаться, причем это различие обрабатывается посредством поддержки динамической компоновки подходящей системы визуального проектирования. Конфигурация настольной компоновки может быть главной конфигурацией компоновки, поскольку она имеет наибольшую ширину экрана. Другие конфигурации компоновки могут рассматриваться как теневые конфигурации компоновки главной конфигурации компоновки и могут отличаться от главной конфигурации компоновки шириной и высотой экрана. В частности, ширина теневой конфигурации компоновки может быть меньше, чем ширина главной конфигурации компоновки (например, дисплея мобильного телефона), аналогична (например, для планшета) или больше (например, большого экрана дисплея).

Очевидно также, что преобразование страниц приложений или страниц веб-сайтов от настольной версии к мобильной преследует две противоречивые цели: сохранение читаемости содержания на страницах в сокращенной версии и сохранение визуальной компоновки страницы. Существующие системы, по большей части, призваны извлекать содержание в ущерб конструкции (например, патент США № 7203901 под названием "SMALL FORM FACTOR WEB BROWSING"). Эти системы выводятся из

опубликованных алгоритмов OCR/анализа страницы, которые извлекают текст из участков, представляющих интерес, и не предпринимают попыток (или прикладывают минимальные усилия) для сохранения существующей визуальной конструкции веб-сайта.

Некоторые алгоритмы пытаются до некоторой степени сохранить конструкцию и/или компоновку, сжимая всю страницу или ее элементы. Однако такое сжатие весьма ограничено, поскольку (в частности) содержание текста быстро становится нечитаемым с уменьшением размера шрифта.

Обратимся к фиг. 1, где показана система 100 для преобразования визуальных приложений между платформами согласно варианту осуществления настоящего изобретения.

Система 100 может содержать преобразователь 200 начальной компоновки, дополнительный преобразователь 300 компоновки и базу данных 50. Преобразователь 200 начальной компоновки может также содержать приемник 210 страниц и процессор 250. Преобразователь 200 может преобразовывать главную конфигурацию компоновки из настольной конфигурации компоновки в теневую для мобильного устройства в первый раз. Дополнительный преобразователь 300 может дополнительно преобразовывать конфигурацию компоновки после редактирования приложения главной конфигурации компоновки. Все приложения и конфигурации могут храниться в базе данных 50.

Система 100 может быть установлена как часть системы построения веб-сайта на сервере 150 с возможностью осуществления связи с различными клиентами 5, представляющими различные платформы, например, настольную 5А, смартфон 5В и интерфейс 5С сторонней системы (связь между программами) через подходящую среду передачи данных, например, интернет. На сервере 150 могут храниться разные конфигурации компоновки для каждой из платформ (включая множественные конфигурации компоновки для каждой отдельной платформы). В альтернативном варианте осуществления, система 100 также может выполняться локально на клиенте или может быть адаптирована для предоставления услуг другой системе через подходящий API (интерфейс прикладного программирования).

Очевидно, что преобразователь 200 начальной компоновки может преобразовывать каждую страницу в приложении к достаточно суженной версии, чтобы ее можно было просматривать на суженном экране с использованием вертикальной прокрутки и минимального (при наличии) горизонтального растривания. Очевидно, что, когда система 100 надлежащим образом интегрирована (например, выполняется на подходящем клиенте системы построения веб-сайта), система 100 может осуществлять преобразование в реактивном режиме, в котором новая компоновка вычисляется каждый раз, когда размеры экрана изменяются.

Обратимся к фиг. 2, где показаны элементы процессора 250. Процессор 250 содержит препроцессор 201, блок 230 создания суперузла, блок 240 упорядочения, блок 270 размещения и постпроцессор 202. Препроцессор 201 может содержать обработчик 206 шаблонов, блок 221 фильтрации компонентов, блок 225 адаптации компонентов, блок 227 регулировки компонентов и блок 229 анализа компонентов. Постпроцессор 202 может содержать блок 282 вставки автоматически добавляемых компонентов и координатор 284 динамической компоновки. Очевидно, что результирующая конфигурация целевой компоновки может отображаться блоком 290 отображения посредством подходящей системы построения веб-сайта. Очевидно, что, если система 100 полностью объединена с подходящей системой построения веб-сайта и непосредственно модифицирует ее структуру данных, постпроцессор 202 также может осуществлять фазу постобработки восстановления структуры данных для адаптации модифицированных структур данных системы построения веб-сайта, подлежащих отображению блоком 290 отображения.

Очевидно, что преобразователь 200 начальной компоновки может обрабатывать каждую страницу приложения или веб-сайта по отдельности. В стандартном режиме он может преобразовывать каждую страницу к достаточно более узкой и обычно более длинной версии, чтобы ее можно было просматривать на суженном мобильном экране с использованием вертикальной прокрутки и минимального горизонтального растривания. Очевидно также, что некоторые целевые устройства, например более крупные планшеты и уличные экраны дисплеев, могут иметь более широкий экран, чем первоначальный настольный экран и, таким образом, сужение может не требоваться, но может требоваться адаптация к большей ширине дисплея.

Преобразователь 200 начальной компоновки может обрабатывать трехмерный (координаты x, y экрана и Z-порядок отображения) набор компонентов и может преобразовывать его в упорядоченный набор в математическом смысле. Генерируемый полный порядок может представлять порядок, в котором информация считывается пользователем на подходящей странице. Затем преобразователь 200 начальной компоновки может отображать упорядоченные компоненты на суженном дисплее мобильного устройства.

Преобразователь 200 начальной компоновки может обрабатывать страницу, подлежащую преобразованию, разделяя подходящую страницу на суперузлы на основании анализа компонентов и соотношения содержания, как более подробно описано ниже.

Препроцессор 201 может анализировать пригодность компонентов подходящей страницы для компоновки целевой конфигурации и может, при необходимости, модифицировать существующие компоненты. Хотя можно конфигурировать разные комбинации компоновок, в нижеприведенных примерах рассмотрено преобразование из конфигурации настольной компоновки в конфигурацию мобильного ком-

поновки. Обработчик 206 шаблонов может создавать модифицированные экземпляры элементов в шаблонах, если они существуют. Блок 221 фильтрации компонентов может фильтровать компоненты по их пригодности для конфигурации мобильной компоновки, блок 225 адаптации компонентов может адаптировать компоненты, в особенности для конфигурации мобильной компоновки, блок 227 регулировки компонентов может регулировать компоненты, в особенности для конфигурации мобильной компоновки, и блок 229 анализа компонентов может анализировать компоненты (включая их содержание, геометрию и любые другие атрибуты) для определения их фактического использования. Например, компонентом изображения может быть фоновое изображение или изображение содержания, причем на их обработку влияют другие части системы 100.

Блок 230 создания суперузла может определять местоположение групп компонентов, которые должны оставаться вместе (например, компоненты, которые сильно перекрываются). Блок 230 создания суперузла может также создавать иерархию суперузлов на основании текущей страницы и элементов в иерархии контейнеров (возможно, модифицируя иерархию контейнеров, как дополнительно описано ниже). Блок 240 упорядочения может определять порядок для элементов каждого из суперузлов. Блок 270 размещения может размещать элементы узла в области мобильной компоновки на основании определенного порядка, и постпроцессор 202, при необходимости, может совершать любые окончательные регулировки компоновки. Блок 281 вставки автоматически добавляемых компонентов может вставлять любые автоматически добавляемые компоненты (как более подробно рассмотрено ниже), а также другие компоненты, относящиеся к мобильным устройствам, которые подлежат вставке после упорядочения (например, меню, характерные для мобильных устройств), и координатор 284 динамической компоновки может, при необходимости, модифицировать любые существующие привязки после размещения. Блок 290 отображения может отображать новую компоновку на мобильной платформе, возможно осуществляя окончательную регулировку для конкретной мобильной платформы, как более подробно описано ниже.

Приемник 210 страниц может принимать веб-страницу с конфигурацией настольной компоновки, подлежащей преобразованию, и может пересылать страницу на препроцессор 201 страниц.

Как рассмотрено выше, может существовать иерархия контейнеров, которая может обеспечивать дополнительную информацию о логическом расположении страницы. Иерархия может включать стандартные соотношения удержания, а также параллельные соотношения удержания, например, множественные параллельные мини-страницы, содержащиеся в одном многостраничном контейнере. Препроцессор 201 может анализировать эту иерархию и любые соотношения конкретных контейнеров в начале процесса для сбора информации, которая может быть полезна для процесса. Препроцессор 201 также может определять размер экрана целевого приложения, извлекать навигационные меню (для всего сайта и для отдельной страницы) и объединять их для создания одного или более объединенных меню страницы, например верхних и нижних меню.

Очевидно, что подходящая система визуального проектирования может поддерживать шаблоны, включающие многоуровневые и множественные наследования шаблонов. Такие шаблоны могут включать отдельные страницы, множественные страницы или элементы страниц. Обработчик 206 шаблонов может создавать модифицированные экземпляры элементов шаблонов, поэтому экземпляр шаблона может состоять из компонентов шаблона, к которым применяются локальные модификации. Обработчик 206 шаблонов может также совершать необходимые конкретные модификации для адаптации к конфигурации мобильной компоновки. Например, если унаследованные компоненты достаточно близки к верхней или нижней границе страницы, они могут быть, при необходимости, включены в верхний колонтитул или нижний колонтитул отдельной страницы. Унаследованные компоненты могут логически дублироваться (с модификацией) в каждой странице, которая наследует их.

Также очевидно, что подходящая система визуального проектирования может оставлять отдельные копии унаследованных компонентов и может позволять применять к ним постраничную компоновку (поскольку адаптация к мобильному устройству в каждой странице может различаться). Обработчик 206 шаблонов может использовать эту информацию о первоначальном шаблоне данного набора компонентов для создания группы компонентов на основании шаблона, гарантируя лучшее согласование с первоначальным назначением шаблона для каждой страницы подходящего веб-сайта по отдельности, препроцессор 201 может просматривать элементы с учетом таких параметров, как размеры целевого экрана. Блок 221 фильтрации компонентов может удалять или скрывать компоненты, непригодные для дисплея мобильного устройства. Такие компоненты могут включать, например, вертикальные разделительные линии (но не горизонтальные линии), содержание Adobe Flash (которое не воспроизводится на некоторых мобильных системах) или декоративные рамки. Компоненты, совершенно непригодные для дисплея мобильного устройства, удаляются до дополнительной обработки. Для компонентов, которые не рекомендованы для дисплея мобильного устройства, блок 221 фильтрации компонентов может по своему выбору скрывать их вместо того, чтобы полностью удалять их. В этом случае разработчик веб-сайта имеет возможность повторно вставлять их в конфигурацию мобильной компоновки, чтобы они были видимыми и повторно вставляемыми посредством GUI отображения мобильных модификаций, как более подробно описано ниже. Очевидно, что некоторые компоненты могут быть непригодными или не рекомендованными для мобильной платформы, но все же предоставляют существенную информацию преобразовате-

лю 200 начальной компоновки.

Обратимся к фиг. 4, где показана вертикальная линия Е между компонентами в настольной версии. Например, показано, что вертикальная линия между компонентами совершенно бесполезна в конфигурации мобильной компоновки, поскольку она обычно преобразуется в длинное (и пустое) разнесение между множествами компонентов. Как показано, когда компоненты А, В, С, D изменяют свое взаимное расположение на вертикальную стопку для суженного дисплея мобильного устройства, вертикальная линия Е отнимает место на дисплее и, таким образом, подлежит удалению. Однако вертикальная линия очень полезна, когда сигнализирует преобразователю 200 начальной компоновки, что 4 компонента А, В, С и D должны располагаться как А+В и С+D, а не как А+С и В+D. Чтобы извлечь пользу из этой информации, блок 221 фильтрации компонентов может пометить вертикальную линию как "удалить полностью" или "позже скрыть", но все же добавить невидимую, 0-й ширины, линию, которая обеспечивает необходимую информацию разделения преобразователю 200 начальной компоновки.

Блок 225 адаптации компонентов может использовать средство уменьшения ширины, аналогичное тому, которое может применяться блоком 270 размещения, что подробно рассмотрено ниже (всякий раз, когда такое средство активируется на этой стадии предварительной обработки). Например, блок 225 адаптации компонентов может переключать компоненты на "облегченные" мобильные версии. Например, переключать галереи, которые могут отображать множественные мини-страницы (например, типа гармошки) на галереи, которые отображают по одной содержащейся мини-странице за раз. Блок 225 адаптации компонентов также может обеспечивать конкретную "облегченную" версию данных компонентов для использования в мобильных версиях созданных приложений (посредством отдельной версии или другого вида, специализированного для дисплея мобильного устройства).

Блок 225 адаптации компонентов также может обрабатывать адаптации, связанные с содержанием, например, преобразование символьной графики в векторную масштабируемую графику. Дизайнеры иногда используют в качестве украшений или разделителей текстовые символы, например "//////////", "-----", "=====", "=====" и т.д. После преобразования в аналогичную векторную фигуру их размер можно аккуратно изменять без необходимости в обработке текста.

Блок 225 адаптации компонентов также может объединять компоненты меню в унифицированное, приспособленное для мобильных устройств (малоформатное) меню. Страницы часто содержат множественные навигационные меню, используемые для навигации, как между страницами, так и внутри страницы. Кроме того, некоторые страницы могут содержать множественные по-разному заданные меню, например меню, заданные в самой странице, меню, полученные из шаблонов страницы, используемых в странице (например, меню приложения в целом, содержащиеся в верхнем колонтитуле и нижнем колонтитуле), и меню, автоматически задаваемые на основании структуры приложения (например, меню верхнего уровня навигации по странице). Блок 225 адаптации компонентов может также создавать составные меню (одно или более) и объединять меню в них в зависимости от таких критериев, как близость оцененного меню к данному созданному составному меню (например, объединять все меню, начинающихся в 200 пикселях сверху страницы в единое меню верхнего уровня). Блок 225 адаптации компонентов может вставлять единое (унифицированное и приспособленное для мобильных устройств) или множественные модифицированные меню сразу же или может "постирать" их для включения на более поздней стадии (например, блоком 270 размещения или блоком 281 вставки автоматически добавляемых компонентов). Это особенно значимо, если (одно или более) меню добавляются таким образом, чтобы не являться частью стандартной компоновки компонентов, но как "плавающие", условно активируемые и т.д.

Блок 227 регулировки компонентов может модифицировать параметры компонентов для уменьшения их ширины. Примером этого является модификация компонентов матричной галереи только до 1-2 столбцов. Блок 227 регулировки компонентов также может модифицировать компоненты, заданные с большими декоративными границами в более простую, менее декоративную версию того же компонента.

Блок 227 регулировки компонентов может изменять размеры компонентов для отражения используемого фактического размера. Например, он может стягивать текстовые компоненты согласно фактическому содержанию текста с использованием охватывающих прямоугольников.

Блок 227 регулировки компонента также может "растворять" контейнеры, которые плотно обхватывают внутренний компонент. Например, если контейнер плотно обхватывает подконтейнер без дополнительных компонентов в более крупном контейнере, он может удалять внутренний контейнер, объединяя два контейнера в один. Очевидно, что, таким образом, уровень иерархии сохраняется, поскольку контейнер прозрачен для процесса.

Дизайнеры иногда суммируют множественные секции одного и того же изображения, которые совместно визуальным образом формируют единое изображение. Блок 227 регулировки компонентов может использовать "сшивание изображений" и использовать атрибуты и содержание нескольких таких компонентов изображения для обнаружения, нужно ли сшивать их в единое изображение. Решение может базироваться на истории редактирования, если изображения были совместно созданы и/или модифицированы, на основании подобия размера на краях соседних изображений, на основании длины перекрытия краев соседних изображений по сравнению с разделением между изображениями или на основании использования аналогичных цветов/признаков на границе (обнаруженных путем использования блока анализа со-

держания изображения).

Блок 227 регулировки компонентов может использовать схему размера шрифта. Текст в любом данном приложении или странице может использовать самые разнообразные размеры шрифта. Эти размеры следует отображать в данный (меньший) диапазон для использования на мобильном устройстве, по возможности сохраняя различия в размере шрифта. Некоторые размеры шрифта будут слишком малы, тогда как некоторые будут слишком велики. Отображение не является линейным, т.е. умножением на фиксированный коэффициент. Однако оно является монотонной функцией размера шрифта. Блок 227 регулировки компонентов может собирать диапазон используемых размеров шрифта и отражать этот диапазон в диапазон допустимых для мобильных устройств размеров шрифта. Блок 227 регулировки компонентов может обеспечивать такую схему применительно к системе в целом или конкретно на уровне пользователя, приложения, страницы или компонента.

В схеме размера шрифта для конкретного приложения/страницы блок 227 регулировки компонентов может подсчитывать объем текста (символов) при каждом данном размере шрифта, а затем использовать функцию нормального (гауссова) кумулятивного распределения, поэтому наиболее распространенные размеры символов отражаются в размерах шрифта в центре диапазона допустимых для мобильных устройств размеров шрифта. Очевидно, что такой размер шрифта может дополнительно модифицироваться путем регулировки перемасштабирования конкретного размера шрифта конфигурации мобильной компоновки.

Очевидно также, что при обработке структуры компонента, важны различия между декоративными изображениями и изображениями содержания (последние составляют часть фактических данных для отдельной страницы). Декоративные изображения можно более свободно масштабировать и обрезать, поскольку их содержание не критично для фактического использования приложения. Кроме того, изображения содержания следует анализировать аналогично другим компонентам в описанном ниже алгоритме анализа, тогда как декоративные изображения не следует анализировать. Блок 229 анализа компонентов может распознавать декоративное изображение как таковое на основании любого из следующих: конкретного типа компонента "декоративное изображение" или по конкретным рекомендациям на уровне шаблона/объекта/типа компонента/разработчика. Блок 229 анализа компонентов также может распознавать, охватывают ли изображения область, например изображение охватывает весь (или большую часть) контейнер, в котором оно располагается, является ли изображение фоном для множественных компонентов, или адаптировано ли изображение к его области отображения компонентов, с использованием оператора, который указывает, что изображение служит декорацией, например, "повторять изображение в линиях". Декоративные изображения не всегда являются "тесными" по отношению к компонентам, содержащимся внутри них, часто в силу выбора конкретного задействованного изображения.

После модификации элементов вышеописанным образом, блок 230 создания суперузла может определять местоположение групп компонентов, которые могут сильно перекрываться или быть иным образом связаны и подлежат совместной обработке. Это необходимо, поскольку (например) сильно перекрывающиеся компоненты могут формировать композицию, таким образом, они должны размещаться совместно для сохранения одной и той же внутренней пропорции при изменении размера. Очевидно, что после определения местоположения всех групп блок 230 создания суперузла может преобразовывать результирующую иерархическую структуру компонентов в иерархию суперузлов на основании текущей страницы и/или иерархии контейнеров, поэтому на разных уровнях структуры существуют множественные полные порядки. Для каждой отдельной страницы или мини-страницы (внутри одностраничного или многостраничного контейнера) суперузлы могут именоваться суперузлами страниц. Также очевидно, что каждый суперузел можно рассматривать как отдельную сущность, и, будучи заданы, объекты в ней могут изменять свое взаимное расположение, как более подробно описано ниже.

Очевидно также, что некоторые найденные группы можно преобразовывать в виртуальные суперузлы (как более подробно описано ниже), которые затем можно обрабатывать как единый компонент аналогично суперузлам страниц, как описано выше. Очевидно, что с каждым типом виртуального суперузла может быть связан способ адаптации к мобильному применению, заданный при его создании, например перемасштабирование определенным образом, ограниченное изменение размера, реорганизация и т.д.

Обратимся к фиг. 5, где показаны элементы блока 230 создания суперузла. Блок 230 создания суперузла может содержать блок 232 определения местоположения группы перекрытия, блок 234 определения местоположения группы текста поверх изображения, блок 236 определения местоположения заранее определенной группы, блок 238 создания узла и блок 500 определения показателя, функциональные возможности которых описаны ниже со ссылкой на фиг. 11.

Блок 232 определения местоположения группы перекрытия может определять набор сильно перекрывающихся компонентов, которые обычно могут совместно создавать конкретную конструкцию и должны сохранять ту же пропорцию между компонентами группы при размещении группы в новой компоновке. Например, изображение и текст, образующие логотип, должны сохранять ту же пропорцию и относительное местоположение независимо от их размера и позиции в новой компоновке для поддержки структуры логотипа.

Блок 232 определения местоположения группы перекрытия может циклически перебирать все возможные пары компонентов в суперузле. Для каждой пары пересекающихся компонентов блок 232 определения местоположения группы перекрытия может вычислять величину относительного перекрытия как величину перекрытия по сравнению с меньшим значением из двух охватывающих прямоугольников двух компонентов. Вычисление может осуществляться на основании площади пересечения или путем объединения относительного пересечения для каждой оси (например, с использованием среднего). Если это относительное пересечение превышает данный порог, блок 232 определения местоположения группы перекрытия может рассматривать пару компонентов как перекрывающуюся. Закончив циклический перебор, блок 232 определения местоположения группы перекрытия может объединять перекрывающиеся пары в группы перекрытия согласно общим членам. Например, пары компонентов [a, b] и [b, c] могут объединяться в [a, b, c]. Очевидно, что площадь для каждого созданного виртуального компонента может быть равна площади минимального охватывающего прямоугольника для всех сгруппированных компонентов. Очевидно также, что группы перекрытия обычно представляют детализированные элементы конструкции и, таким образом, подлежат только изменению размера, но не изменению расположения. В частности, текстовый компонент в такой группе должен соответствовать своему первоначальному размеру вместо того, чтобы использовать схему размера шрифта, как описано выше, осуществляемую блоком 225 адаптации компонентов. Таким образом, текст следует масштабировать таким образом, чтобы избежать полного отображения шрифта по схеме размера шрифта или использовать масштабный коэффициент, что обращает эффект от схемы размера шрифта.

Очевидно также, что некоторые компоненты можно исключить из определения группы перекрытия. Они могут включать конкретные компоненты, например (в том числе) горизонтальные линии и контейнеры по ширине экрана и размер текста, превышающий данный предел (например, 25 символов), который интерпретируется как текст, являющийся стандартным абзацем (который не подлежит помечанию как группы перекрытия и изменению размера как композиции), а не текстом логотипа.

Блок 234 определения местоположения группы текста поверх изображения может группировать текстовый компонент, наложенный на конкретное фоновое изображение. Можно выявить случаи, когда текстовый компонент размещен поверх фонового изображения, и оба подлежат совместной обработке. Блок 234 определения местоположения группы текста поверх изображения может искать пару компонентов текста и изображения в текущем суперузле, в котором либо текстовый компонент полностью содержится внутри компонентов изображения, либо на каждой из четырех сторон текстовый компонент достаточно близок (с данным порогом) к охватывающему компоненту изображения.

Если компонент является логотипом, а не текстом поверх изображения, блок 234 определения местоположения группы текста поверх изображения может, альтернативно, предписывать блоку 232 определения местоположения группы перекрытия создать группу перекрытия вместо группы текста поверх изображения. Это может осуществляться на основании оценивания согласно следующим критериям: объем текстовых символов ниже заданного порога, использование декоративных или других весьма уникальных шрифтов или использование базовой линии текста, которая не является стандартной горизонтальной линией (например, текст располагается вдоль кривой).

Блок 236 определения местоположения заранее определенной группы может группировать компоненты согласно рекомендациям на уровне шаблонов, приложений, страниц или компонентов. Это можно рассматривать как постоянную версию группирования, часто осуществляемого с использованием GUI конкретной системы построения, что позволяет совместно манипулировать множественными компонентами.

Очевидно, что после определения местоположения групп и группирования перекрывающихся элементов блок 238 создания узла может объединять и преобразовывать их в виртуальные суперузлы, как более подробно описано ниже, и затем может создавать окончательные суперузлы (включающие виртуальные и страничные суперузлы). Остальные элементы можно рассматривать как разъединенные, что облегчает их упорядочение, как рассмотрено ниже.

Обратимся к фиг. 8А-8С, которые иллюстрируют примеры того, как блок 230 создания суперузла может создавать иерархию суперузлов. Как рассмотрено выше, суперузлы, созданные из страниц и мини-страниц, можно рассматривать как суперузлы страниц и суперузлы, созданные из обнаруженных групп, которые также могут быть виртуальными суперузлами. Для страницы P1 на фиг. 8А страницу (или мини-страницу), содержащую три элемента текста, можно рассматривать как суперузел страницы (PSN1) с 3 элементами. На фиг. 8В страницу P2 можно рассматривать как суперузел страницы (PSN2) с текстовым элементом и виртуальным суперузлом (VSN1), содержащим текстовый элемент и элемент рисунка. Если страницы P1 и P2 являются мини-страницами, содержащимися в странице P3, результирующая структура узла может быть суперузлом страницы (PSN3), имеющим 2 суперузла страниц (PSN1 и PSN2), каждый из которых имеет элементы, проиллюстрированные на фиг. 8С.

Очевидно также, что блок 238 создания узла может задавать плотно обернутые контейнеры как контейнер А, присутствующий внутри контейнера В, где А и В имеют очень близкие размеры (т.е. существуют малые поля на всех сторонах между А и В, где А является только компонентом внутри В). Блок 238 создания узла также может осуществлять перестановку контейнеров на основании семантического

анализа компонентов во множественных контейнерах. Например, как показано со ссылкой на фиг. 9, текстовые компоненты T1, T2 и T3 содержатся в контейнере C1 и компоненты изображения P1, P2 и P3 содержатся в контейнере C2. Блок 238 создания узла может осуществлять семантический анализ для распознавания связанных пар компонентов текста и изображения на основании их типа, близости и отношения к другим компонентам, как дополнительно описано ниже. На основании этого анализа блок 238 создания узла может распознавать, что T1 и P1 образуют связанную пару, а также T2-P2 и T3-P3, и, таким образом, может удалять эти шесть компонентов (T1, T2, T3, P1, P2, P3) из контейнеров C1 и C2 и помещать их во вмещающую страницу, сохраняя информацию об отношениях между ними, как показано со ссылкой на фиг. 10.

Кроме того, блок 238 создания узла может классифицировать компоненты, существенно перекрывающие данный контейнер (например, перекрывая свыше 75% площади), как члены конкретного контейнера в целях генерации суперузла, чтобы иерархия суперузлов лучше отражала визуальное расположение. Это относится к системам визуального проектирования, в которых компоненты, логически не содержащиеся в данном контейнере, все же могут перекрывать контейнер.

После задания структуры суперузла блок 240 упорядочения может создавать порядок для набора компонентов и содержащихся суперузлов страниц и виртуальных суперузлов в каждом суперузле (по отдельности). Очевидно также, что этот основной порядок может дополнительно модифицироваться в результате обнаружений блока определения местоположения частично упорядоченного набора, как более подробно описано ниже. Обратимся к фиг. 11, где показаны элементы блока 240 упорядочения. Блок 240 упорядочения может содержать блок 247 основного упорядочения, блок 250 определения местоположения частично упорядоченного набора (POS) и интегратор 245 порядка. Блок 247 основного упорядочения может также содержать блок 241 упорядочения первичного направления, блок 242 упорядочения первичного направления с разделением и объединением и блок 243 Г/В расслоения. Блок 243 Г/В расслоения может также содержать блок 244 деления элементов, как более подробно описано ниже.

Блок 250 определения местоположения POS может содержать блок 251 определения местоположения POS кластеров, блок 252 определения местоположения POS семантических соотношений, блок 253 определения местоположения POS схем, блок 254 определения местоположения заранее определенного POS и блок 255 определения местоположения POS на основе ESI (информации сеанса редактирования). Очевидно, что блок 250 определения местоположения POS может анализировать семантику, содержание и геометрию компонентов.

Очевидно, что блок 240 упорядочения может пытаться эмулировать порядок, в котором читатель-человек может просматривать элементы на странице (или в конкретном суперузле). Очевидно также, что поскольку страница является 2-мерным объектом (или даже 3-мерным с включенным Z-порядком), этот порядок может не быть столь хорошо задан даже для читателя-человека.

Обратимся к фиг. 12, где показано расположение четырех абзацев текста в квадратном расположении. Исходя из того, что абзацы написаны на английском языке (который предусматривает чтение сверху вниз и слева направо) не ясно, должен ли порядок чтения быть A1, A2, B1, B2 или A1, B1, A2, B2. Оба порядка можно считать правильными. Кроме того, если элементами являются текст и изображения, как показано со ссылкой на фиг. 13, не ясно, чем являются связанные элементы текста и изображения.

Блок 241 упорядочения первичного направления может заранее определять первичное направление (т.е. сортировку сначала строк или сначала столбцов), а затем может пытаться сортировать компоненты на 2-мерной странице на основании первичного направления и затем вторичного направления. Например, блок 241 упорядочения первичного направления, действующий в режиме "сначала строки", может сортировать элементы в суперузле, таким образом, что элемент A предшествует элементу B, если A или B имеет подходящее Y-перекрытие (например, по меньшей мере 25% высоты меньшего из них) и элемент A находится слева (т.е. порядок в строке) или если A и B не имеют достаточного Y-перекрытия и A выше (т.е. порядок среди строк).

Блок 242 упорядочения первичного направления с разделением и объединением (здесь блок 242 упорядочения PDSM) также может заранее определять первичное направление и также может отслеживать любое разделение и объединение строк/столбцов. Это можно считать более общим для столбцов в отличие от строк. Обратимся к фиг. 14A-14C, которые иллюстрируют компоненты (1-10) в суперузле. Очевидно, что компоненты можно упорядочить согласно их порядку в генерируемом графе столбцов в узле. Блок 242 упорядочения PDSM может, по существу, преобразовывать множество прямоугольников (как показано на фиг. 14A) в граф столбцов (который может объединяться, разделяться или продолжаться, как показано на фиг. 14B), который затем используется для генерации упорядочения компонентов (как показано на фиг. 14C). Прямоугольники могут пересекаться.

Обратимся к фиг. 15A-15C и 16A-16G, которые иллюстрируют этапы, осуществляемые блоком 242 упорядочения PDSM для определения графа столбцов и результирующего порядка. Фиг. 16A демонстрирует набор элементов C, которые необходимо обработать. Блок 242 упорядочения PDSM может задавать 2-элементное соотношение между элементами в наборе (этап 400) и затем может "прокладывать" элементы, подлежащие упорядочению (набор C) между фиктивным элементом A верхнего колонтитула и фиктивным элементом B нижнего колонтитула (этап 410), как показано. Затем блок 242 упорядочения

PDSM может создавать два новых набора: набор X генерируемых столбцов (в их текущем состоянии) (этапы 420 и 430) и их элементов и набор Y элементов, подлежащих переработке в столбцы (этапы 440 и 450). Затем блок 242 упорядочения PDSM может сканировать элементы набора Y сверху вниз согласно их у-порядку (этап 470) и для каждого элемента Q в наборе Y выбирать поднабор R элементов (во всех столбцах в X), которые управляют Q . Обратимся к фиг. 16B, где показано, как все элементы A , E и F управляют Q . Затем блок 242 упорядочения PDSM может удалять любые копии из набора R . Как можно видеть на фиг. 16B, помимо того, что все A , E и F управляют Q , A управляет E и E управляет F . В этом сценарии единственным соединением, необходимым блоку 242 упорядочения PDSM, может быть соединение между Q и F (последним управляющим элементом). Очевидно, что, если R пусто, это можно рассматривать как начало висячего столбца, не имеющего соединений, и, таким образом, в X открывается новый столбец с элементом Q , как показано со ссылкой на фиг. 16C. Очевидно также, что, если R имеет более чем один элемент, блок 242 упорядочения PDSM может соединять элемент Q со столбцами, содержащими каждый из элементов $C_1, \dots, C_n$, создавая продолжение столбца (если соединен в конце столбца) или разделение столбца (если соединен в середине столбца) для каждого. Если существует более чем один такой столбец, блок 242 упорядочения PDSM может объединять столбцы, как показано со ссылкой на фиг. 16D. Как показано на фиг. 16D, блок 242 упорядочения PDSM может объединять столбцы G и H в единый столбец, продолженный элементом Q . Обратимся к фиг. 16E, где показан столбец G , который продолжается от элемента, который не является последним элементом в столбце G . В этом сценарии элемент Q создает разделение в столбце G , создавая 2 столбца, продолженных элементами F и Q .

Очевидно, что, когда столбец делится в данной точке, разделенный набор упорядочен слева направо, т.е. множественные столбцы, продолжающие разделенный столбец, имеют порядок, заданный для них согласно x -координате их левого края. То же самое справедливо для объединения столбцов. Также очевидно, что блок 242 упорядочения PDSM может иметь комбинированную ситуацию разделения и объединения, что проиллюстрировано, например, со ссылкой на фиг. 16F. Как показано, элемент Q может создавать разделение для 2 столбцов, содержащих элементы G и H . Блок 242 упорядочения PDSM может непосредственно объединять разделения в единый столбец, продолженный элементом Q .

Таким образом, блок 242 упорядочения PDSM может переносить элементы из Y в X и соединять их с одним или более существующими столбцами в X (или, возможно, создавать новые столбцы) на основании геометрических критериев. Очевидно, что этот процесс сканирования может создавать несколько столбцов в X , которые могут начинаться, разделяться, объединяться или заканчиваться. Процесс заканчивается, когда достигается фиктивный элемент V нижнего колонтитула. Затем блок 242 упорядочения PDSM может сканировать граф (заданный согласно столбцам в множестве X) для вычисления результирующего порядка. Затем блок 242 упорядочения PDSM может размещать элементы согласно порядку в графе сверху вниз и слева направо, например, как проиллюстрировано со ссылкой на фиг. 16G.

Очевидно также, что в случае висячих столбцов блок 242 упорядочения PDSM может размещать их согласно порядку X , как показано со ссылкой на фиг. 16H. Если пары элементов a - c и b - d достаточно близки, чтобы соединиться, они могут формировать полные столбцы, и порядок между ними может быть a - c - b - d , но не a - b - c - d . Если они недостаточно близки, они не соединяются, в результате чего элементы c и d становятся висячими столбцами, и порядок к указанному числовому порядку модифицируется, как описано выше.

Очевидно, что функциональные возможности блока 242 упорядочения PDSM можно надлежащим образом модифицировать для суперузлов, содержание которых располагается справа налево (а не слева направо).

Блок 243 Г/В расслоения может использовать чередующееся горизонтальное и вертикальное расслоение элементов в суперузле для создания внутреннего дерева таких делений и, таким образом, задавать порядок отображения между элементами. Очевидно, что такое внутреннее дерево специфично для данного суперузла и не связано с общим деревом уровня суперузлов. Очевидно, что блок 243 Г/В расслоения лучше всего адаптирован к сайтам в стиле "плитки шоколада", построенных из горизонтальных и вертикальных секций, классическим примером которых является, например, страница с газетной компоновкой, как показано со ссылкой на фиг. 17. Блок 243 Г/В расслоения может брать множество элементов единичного суперузла и создать внутреннее дерево узлов (которые отличаются от суперузлов более высокого уровня, поскольку каждый суперузел в отдельности может содержать такое внутреннее дерево). Узлы могут быть следующих типов.

V-узлы (вертикально размещенные узлы) - множество подузлов/элементов, которые располагаются сверху вниз.

H-узлы (горизонтально размещенные узлы) - множество подузлов/элементов, которые располагаются слева направо (ниже описано расположение справа налево).

Крайевые узлы - узлы, которые представляют оконечные элементы внутреннего дерева - компоненты или содержащиеся суперузлы. Краевой узел может содержать множественные компоненты, которые все еще укладываются в выделенную ширину.

UR-узлы (неразрешаемые узлы) - узлы, которые включают множественные элементы, которые еще не размещены. Это начальное состояние любого узла до его преобразования в V/H/краевой узел.

Каждый узел может содержать множество ссылок элементов, а также координаты и размеры ниже лежащих элементов, и полный охватывающий прямоугольник для элементов узлов. Полная совокупность узлов (четырёх вышеупомянутых типов) внутри единичного суперузла может именоваться внутренним деревом. Очевидно, что блок 243 Г/В расслоения может регулировать процедуру для суперузлов, указанных или обнаруженных как содержащие материал справа налево (R2L) или имеющих компоновку R2L.

Блок 243 Г/В расслоения может при начальном вызове, создавать внутреннее дерево с единственным UR-узлом и помещать ссылки на все элементы в суперузле внутри него. Блок 243 Г/В расслоения может обрабатывать UR узел, беря все элементы в узле. Если охватывающий прямоугольник элементов уместается в допустимой ширине, блок 243 Г/В расслоения может преобразовывать узел в краевой узел. Блок 243 Г/В расслоения также может вызывать блок 244 деления элементов, который может предлагать конкретное деление (горизонтальное или вертикальное) элементов UR-узла на подгруппы.

Затем блок 243 Г/В расслоения может, при необходимости, преобразовывать UR-узел в H-узел или V-узел. Очевидно, что для каждой из подгрупп, возвращаемых блоком 244 деления элементов, блок 243 Г/В расслоения может вычислять охватывающий прямоугольник R для элементов в подгруппе. Если ширина R уместается в допустимой ширине, то он может создавать краевой узел, происходящий от узла N из подгруппы. Иначе, если подгруппа имеет более 1 компонента, блок 243 Г/В расслоения может создавать UR-узел, происходящий от узла N из подгруппы, и обрабатывать его соответственно, как описано выше (т.е. посредством рекурсивного применения блока 243 Г/В расслоения). Если ширина R не укладывается в допустимой ширине, блок 243 Г/В расслоения может создавать краевой узел, происходящий от узла N из подгруппы, содержащей единственный элемент. Затем блок 243 Г/В расслоения может применять уменьшение ширины, например, изменение размеров, переформатирование текста и т.д., как описано выше.

Затем блок 243 Г/В расслоения может рекурсивно сканировать сгенерированное внутреннее дерево (с использованием сканирования сначала по глубине) и выдавать компоненты (например, краевые узлы) согласно порядку сканирования. Блок 243 Г/В расслоения может сканировать каждый H/V-узел согласно его естественному порядку (например, сверху вниз для V-узла). Очевидно, что на этой стадии, поскольку был сгенерирован основной порядок элементов, создавать внутреннее дерево для этого суперузла больше не требуется.

Как рассмотрено выше, блок 244 деления элементов может вычислять деление для группы элементов. Он может вычислять охватывающий прямоугольник для каждого элемента, игнорируя декоративные границы. Затем блок 244 деления элементов может проецировать эти прямоугольники на ось X и Y.

Затем блок 244 деления элементов может делить каждое множество осевых проекций на сегменты, отличающиеся количеством проецируемых на них компонентов. Сегменты с нулевым количеством проецируемых на них компонентов представляют промежутки в данном направлении проекции. Очевидно, что блок 244 деления элементов может обеспечивать конкретные (возможно, не отображаемые) компоненты разделительной линией для помощи в определении правильного направления разделения (например, для использования в качестве рекомендации разработчику по процессу). Таким компонентам также может быть назначен "вес", чтобы отсчитывать их (для отсчета проецируемых прямоугольников) как один или более компонентов.

Если промежутки найдены в обоих направлениях, блок 244 деления элементов может генерировать горизонтальное и вертикальное деления элементов на основании разделителей, ортогональных промежуточным сегментам. Например, как показано со ссылкой на фиг. 18, случай A является горизонтальным делением (на столбцы) с использованием единственного разделителя, и случай B является вертикальным делением (на строки) с использованием двух разделителей.

Блок 244 деления элементов может вычислять показатель качества деления (DQR) для каждого из двух направлений деления с использованием взвешенного среднего максимального количества разделителей и максимального полного размера найденных промежутков, и минимальное количество случаев, когда аналогичные элементы разбиты на отдельные подгруппы посредством деления. Эта информация может быть доступна от блока 253 определения местоположения POS схем (как более подробно описано ниже) на стадии создания суперузлов, который задает отношение подобия между младшими суперузлами. Как показано со ссылкой на фиг. 19, вертикальное деление (на строки) может быть предпочтительно, чтобы блоки услуг/проектов/клиентов (A, B и C) оставались вместе.

Блок 244 деления элементов также может учитывать, в каком из направлений компоненты лучше выравниваются. Например, как показано со ссылкой на фиг. 20, в сценарии A деление должно быть вертикальным (т.е. компоненты располагаются и выравниваются в строках), хотя при осуществлении горизонтального деления (т.е. на столбцы) промежутки больше. Это можно лучше наблюдать при добавлении линий выравнивания в сценарии B.

Блок 244 деления элементов также может добавлять постоянный коэффициент предпочтения к горизонтальному делению (которое уменьшает ширину) против вертикального деления (которое не уменьшает). Затем блок 244 деления элементов может возвращать деление с более высоким DQR.

Если промежутки найдены только в одном направлении, блок 244 деления элементов может возвращать деление в этом направлении. Если промежутков не найдено, возможен случай взаимосвязанных элементов. В таком случае блок 244 деления элементов может создавать разделитель, который пересекает один из элементов. Например, как показано со ссылкой на фиг. 21, элементы делятся с использованием вертикальной разделительной линии А, которая может пересекать только элемент В. Все элементы, помеченные R, могут быть связаны с правой стороной деления (включающей элемент В), и элемент L может быть связан с левой стороной деления.

Очевидно, что в этом сценарии блок 244 деления элементов может находить проекционный сегмент (из проекционных сегментов по обеим осям X и Y), который имеет минимальное количество пересеченных элементов. При наличии только одного такого сегмента блок 244 деления элементов может задавать деление на его основании. При наличии двух или более таких сегментов (в любых направлениях), имеющих одинаковое минимальное количество пересеченных элементов, блок 244 деления элементов может создавать множество делений одним разделителем на основании сегментов, имеющих минимальное количество пересеченных элементов.

Для каждого из этих делений блок 244 деления элементов затем может вычислять DQR на основании взвешенного среднего, критериев для стандартного вычисления DQR (как рассмотрено выше), минимальной площади пересеченного элемента и определенности разреза, т.е. наличия минимального процента площади пересеченного элемента по одну сторону пересекающего разделителя, благодаря чему пересеченный элемент более явно принадлежит одной из двух сторон разделителя. Альтернативно, блок 244 деления элементов может использовать деление с наивысшим DQR.

Затем блок 244 деления элементов может возвращать выбранное деление, при этом каждый пересеченный элемент присоединен к стороне разделителя, которая содержит более значительную площадь пересеченных элементов.

В альтернативном варианте осуществления блок 244 деления элементов может пытаться создавать множественные разделители на основании множественных проекционных сегментов, имеющих одно и то же количество пересеченных элементов, и оценивать деления, содержащие несколько таких пересеченных элементов.

Очевидно, что блок 244 деления элементов может вызывать альтернативное вычисление DQR рекурсивно для каждого из возможных делений и может проверять значение DQR делений, созданных после каждого из пробных потенциальных делений. Очевидно также, что, таким образом, блок 244 деления элементов может находить наилучший единичный элемент для пересечения, чтобы дополнительные деления были успешными.

Очевидно, что параллельно блоку 247 основного упорядочения блок 250 определения местоположения POS может обнаруживать наборы компонентов, которые связаны некоторым образом и, возможно, должны оставаться вместе, когда элементы переупорядочиваются для дисплея мобильного устройства. Например, компонентами могут быть заголовок текста и совпадающий абзац текста.

Блок 251 определения местоположения POS кластеров может обнаруживать частично упорядоченный набор кластеров, когда компоненты (любого типа) очень близки друг к другу по сравнению со стандартным разнесением между компонентами в конкретном суперузле. Блок 251 определения местоположения POS кластеров может вычислять среднее расстояние между компонентами в суперузле и затем циклически перебирать все пары компонентов в суперузле, искать пары компонентов, расстояние между которыми составляет данную долю вычисленного среднего расстояния. Затем блок 251 определения местоположения POS кластеров может объединять эти пары компонентов в наборы, обеспечивая максимальное данное расстояние между двумя наиболее удаленными компонентами в каждом таком наборе (указанное как доля среднего расстояния). Блок 251 определения местоположения POS кластеров также может использовать вычисление математического ожидания или медианы вместо вычисления среднего для учета данного количества выбросов, которые могут исказить значение среднего расстояния. В альтернативном варианте осуществления блок 251 определения местоположения POS кластеров может использовать любой известный в технике алгоритм кластеризации. Затем блок 251 определения местоположения POS может проверять извлеченные кластеры и, в случае достаточной плотности (например, максимальное расстояние между членами кластера ниже данного определенного порога), создать частично упорядоченный набор кластеров.

Блок 252 определения местоположения POS семантических соотношений может обнаруживать частично упорядоченный набор семантических соотношений при наличии конкретных комбинаций компонентов данных типов, близких друг к другу, например, изображения и его подписи. Блок 252 определения местоположения POS семантических соотношений может сканировать все возможные пары компонентов в суперузле и для каждой потенциальной пары, проверять, что каждый из компонентов имеет правильный тип (например, один является текстом, а другой является изображением), что компоненты находятся близко друг к другу (расстояние ниже заданного порога) и что между ними нет инвертирующего компонента. Блок 252 определения местоположения POS семантических соотношений может проверять, что для оцененной пары компонентов [A, B] не существует третьего компонента C, который может замещать B, относится к правильному типу для B и быть ближе к A, чем к B. То же самое справедли-

во для соотношения между А и С.

Другая комбинация пар компонентов может представлять собой частично упорядоченный набор из текста и кнопки. Очевидно, что в этом сценарии блок 252 определения местоположения POS семантических соотношений может связывать кнопку, описывающую текстовое поле с фактическим текстовым полем. Очевидно, что соотношение можно определять только согласно позиции.

Другая комбинация может представлять собой частично упорядоченный набор связывающий текст. В этом сценарии, блок 252 определения местоположения POS семантических соотношений может объединять множественные элементы текста, являющиеся продолжением друг друга. Очевидно, что блок 252 определения местоположения POS семантических соотношений может распознавать только один текстовый элемент, находящийся над другим текстовым элементом. Очевидно также, что в данном множестве может существовать любое количество таких элементов, не только 2, которые все связаны друг с другом.

Блок 253 определения местоположения POS схем может обнаруживать частично упорядоченный набор схем при наличии повторяющейся схемы данного количества компонентов (например, пар или троек компонентов), имеющих конкретный тип, свойства и компоновку при данном расстоянии. Блок 253 определения местоположения POS схем также может обнаруживать частично упорядоченные наборы схем в случае переключающихся схем, что проиллюстрировано со ссылкой на фиг. 22. Пары компонентов А и С имеют текст слева и изображение справа, тогда как пары компонентов В и D имеют текст справа и изображение слева. Блок 253 определения местоположения POS схем может обнаруживать такую схему на основании пар компонентов, имеющих конкретные типы и свойства, но с расстоянием по горизонтали являющимся одним из двух вариантов (отрицательные или положительные числа, имеющие одинаковое абсолютное значение) в каждой из пар компонентов. Блок 253 определения местоположения POS схем (на примере 2-элементных схем, которые могут быть унифицированы позже) может сканировать все компоненты в суперузле и для каждого компонента определять местоположения его ближайших соседей на всех четырех сторонах компонента. Блок 253 определения местоположения POS схем может включать соседей с перекрытием компонента вплоть до данного указанного предела. Затем блок 253 определения местоположения POS схем может сохранять список соотношений между компонентами и их соседями и затем сканировать сгенерированный список соотношений и выбирать пары соотношений, которые аналогичны согласно следующим атрибутам: компоненты имеют одинаковые типы (например, компонент [pic,txt] аналогичен другому компоненту [pic,txt]), компоненты имеют одинаковое направление (включая обратные направления, если поддерживаются переключающиеся схемы и аналогичное расстояние (разность не превышает данный порог)).

Очевидно, что два компонента в выбранном соотношении должны иметь разные типы, в противном случае набор из четырех компонентов одного типа, расположенных в виде квадрата, будет генерировать два конфликтующих набора аналогичных пар соотношений. Например, как показано со ссылкой на фиг. 23, четыре компонента изображения могут делиться на пары горизонтально (пара ассоциаций H1, H2) или вертикально (пара ассоциаций V1 и V2).

Блок 253 определения местоположения POS схем может сканировать список выбранных соотношений и объединять их в наборы (например, если $r1=r2$ и $r2=r3$, он может создавать набор $[r1, r2, r3]$). Блок 253 определения местоположения POS схем также может удалять копии.

Очевидно, что единый набор схем (например, набор из нескольких наборов компонентов [txt,txt,pic]) подлежит совместной обработке.

Блок 254 определения местоположения заранее определенного POS может обнаруживать частично упорядоченные наборы, создаваемые на основании конкретных рекомендаций, которые могут обеспечиваться разработчиком приложений или в первоначальном шаблоне приложения. Такие рекомендации могут принимать различные формы. Конкретные составные компоненты, которые включают заранее определенные частично упорядоченные системные рекомендации, например компонент "изображение+текстовая подпись", могут включать их соединение в заранее определенный частично упорядоченный набор. Другая рекомендация может быть явно указанной ассоциацией произвольных компонентов на странице или ассоциацией, связанной с мобильным устройством, выведенной из других форм ассоциаций, доступных в системе визуального проектирования, например группированием для редактирования, поскольку многие системы визуального проектирования имеют возможность связывать компоненты в группы, чтобы их можно было перемещать, изменять или иначе модифицировать совместно. Еще одной ассоциацией могут быть привязки динамической компоновки, поскольку системы визуального проектирования могут поддерживать привязки динамической компоновки (явно указанные или автоматически создаваемые из определения компоновки). Такие привязки могут дополнительно служить рекомендациями для создания частично упорядоченного набора. Еще одной ассоциацией могут быть шаблоны - наборы компонентов, созданные из множественных экземпляров одного и того же многокомпонентного шаблона.

Очевидно, что блок 254 определения местоположения заранее определенного POS может обеспечивать как заранее определенные группы, так и заранее определенные частично упорядоченные наборы. Это разные элементы системы, и блок 254 определения местоположения заранее определенного POS может позволять разработчику указывать заранее определенные групповые рекомендации (используемые

для группирования компонентов в виртуальный суперузел), а также заранее определенные частично упорядоченные наборы/рекомендации упорядочения (используемые для руководства процессом упорядочения).

Блок 255 определения местоположения POS на основе ESI может обнаруживать автоматическое создание частично упорядоченных наборов на основании ESI (при наличии). В частности, блок 255 определения местоположения POS на основе ESI может связывать друг с другом (в один частично упорядоченный набор) компоненты на основании информации, собранной из предыдущих сеансов редактирования, например, наборы компонентов, созданные с использованием дублирования или копирования и вставки, наборов компонентов, совместно сгруппированных и совместно отредактированных (например, когда система визуального проектирования поддерживает только специализированное группирование для редактирования, которое не сохранено в базе данных) и наборов компонентов, которые были последовательно отредактированы.

После того как блок 240 упорядочения создает основной порядок и блок 250 определения местоположения POS определяет любые частично упорядоченные наборы, интегратор 245 порядка может интегрировать основной порядок и обнаруженные частично упорядоченные наборы для создания объединенного модифицированного порядка (который является полным порядком для всех привлеченных элементов), разрешая противоречивые частично упорядоченные наборы вдоль пути.

Очевидно, что некоторые из элементов блока 247 основного упорядочения могут не создавать полный порядок между всеми компонентами, но зато могут делить страницу/суперузел на секции страницы, которые умещаются в допустимую ширину. Это может иметь смысл при необходимости в поддержке основной структуры, например при преобразовании приложения к устройству (например, настольному, большому дисплею), ширина которого аналогична или больше, чем у настольного дисплея. В этом сценарии полное разложение на компоненты, расположенные на линии, может нарушать расположение основных компонентов более низкого уровня, и минимальное "расслоение" может быть визуально превосходящим. С этой целью элементы блока 247 основного упорядочения можно модифицировать для работы на секциях страницы (которые все еще укладываются в доступную ширину), а не на отдельных компонентах.

Затем блок 247 основного упорядочения может использовать эту информацию в качестве руководства, чтобы секции страницы, соединенные посредством существенных соединений POS, не нарезались.

Очевидно, что блок 230 создания суперузла, блок 247 основного упорядочения и блок 250 определения местоположения POS могут дополнительно создавать результаты, которые противоречат друг другу. Например, один из подэлементов блока 250 определения местоположения POS может генерировать определение конкретного частично упорядоченного набора с участием некоторых элементов, и другой элемент может генерировать определение другого и противоречащего частично упорядоченному набору. Интегратор 245 порядка может объединять результаты из разных элементов и также может разрешать любые конфликты для создания единого выхода.

Очевидно также, что для разрешения таких конфликтов блок 230 создания суперузла, блок 247 основного упорядочения и блок 250 определения местоположения POS могут использовать показатель качества. Например, конструкция на основании явных запросов разработчика обычно не имеет максимального показателя качества, поскольку можно предположить, что разработчик полностью осведомлен о композиции своей конструкции. Очевидно, что блок 230 создания суперузла, блок 247 основного упорядочения и блок 250 определения местоположения POS могут сверять свои показатели с показателями качества, поддерживаемыми блоком 500 определения показателя, который может обеспечивать для каждого генерируемого результата показатель определенности того, насколько элемент гарантирует корректность конкретного результата, на основании анализируемых параметров.

Интегратор 245 порядка может собирать результаты из блока 247 основного упорядочения и блока 250 определения местоположения POS и пытаться интегрировать их. Очевидно, что результаты, не противоречащие друг другу, просто комбинировать. Результаты, противоречащие друг другу, могут требовать, чтобы интегратор 245 порядка выбирал, какой из них использовать. Это может осуществляться согласно вычислению комбинированных показателей качества и определенности для конкретного результата. Результаты можно игнорировать, если они не достигают данного (необязательного) порога определенности.

Интегратор 245 порядка также может учитывать дополнительную информацию, например результаты, заданные для компонентов, присутствующих в шаблоне, которые могут иметь более высокую определенность, чем результаты, которые включают только компоненты, заданные в текущей странице (например, поскольку шаблоны лучше построены, чем стандартные страницы). Он также может учитывать результаты, которые могут быть аналогичны результатам, найденным на предыдущих страницах или суперузлах, которые были обработаны ранее, и которым, таким образом, может быть присвоена более высокая определенность.

Интегратор 245 порядка может иметь дополнительные возможности изучения структуры страниц, созданных данным разработчиком, возможно с использованием обратной связи от этого разработчика для оценивания результатов элемента.

Очевидно, что конкретный элемент может возвращать множество результатов, которые следует реализовать совместно или вовсе не реализовать. Например, блок 253 определения местоположения POS схем может находить повторяющуюся схему компонентов. Такую схему следует использовать для всех привлеченных компонентов или вовсе не использовать.

После соответствующего упорядочения компонентов, блок 270 размещения может размещать компоненты и любые автоматически добавляемые компоненты (как более подробно описано ниже) в их надлежащую позицию на странице. При необходимости он может размещать упорядоченные элементы на линиях компонентов для согласования с доступной шириной согласно определенному полному порядку и допустимому свободному месту. Очевидно, что доступная ширина может задаваться шириной суперузла страницы, за минусом любых заранее определенных полей и других ограничений, например, зарезервированных мест. Например, при условии, что ширина главной страницы равна 320px, левое и правое поля равны 10px, и никаких мест не зарезервировано, доступная ширина суперузла главной страницы составляет $320 - (2 \times 10) = 300$ px. Суперузел страницы, который содержится в суперузле главной страницы и занимает максимальную доступную ширину, будет иметь доступную ширину $300 - (2 \times 10) = 280$ px и т.д.

Блок 270 размещения может применять рекомендации по компоновке, присоединенные к компоненту, как часть конфигурации настольной компоновки, при наличии пометки "сохранять в конфигурации мобильной компоновки". Такие компоненты могут функционировать более аналогично автоматически добавляемым компонентам, в том смысле, что они становятся частью основной структуры контейнера, который может стимулировать построение линии компонентов, а не частью самих линий компонентов.

Затем блок 270 размещения может создавать разрывы линии согласно явным рекомендациям (например, "поместить эту группу/POS на отдельной линии"), созданным блоком 230 создания суперузла и блоком 240 упорядочения. Когда добавленный компонент превышает доступную ширину (с учетом зарезервированных мест), блок 270 размещения может пытаться сохранять целостность групп компонентов, т.е. чтобы при необходимости создания разрыва линии до группы, вся группа умещалась на одной и той же линии компонентов. Блок 270 размещения может избегать добавления разрыва линии, чтобы не разобщать компоненты, связанные семантическим соотношением (обнаруженным ранее, например блоком 250 определения местоположения POS).

Блок 270 размещения также может применять средство уменьшения ширины к каждому элементу согласно его типу. Средство уменьшения ширины может включать модификацию компонентов, которые можно адаптировать (например, текстовые компоненты) согласно типу компонента, в том числе перемасштабировать, изменять размер шрифта, переформатировать текст и т.д. Уменьшение ширины также может использовать предписанный способ обработки для каждого виртуального суперузла согласно его типу. Эти средство уменьшения ширины также может включать преобразования, аналогичные осуществляемым блоком 227 регулировки компонентов (как стадия предварительной обработки), которые теперь могут осуществляться с добавлением информации об окончательной позиции и размере компонентов в конфигурацию мобильной компоновки. Блок 270 размещения также может осуществлять увеличение ширины, например, для увеличения компонента, который был размещен вдоль линии компонентов.

Блок 270 размещения обеспечивает конкретное средство изменения размеров для конкретных типов элементов. Например, блок 270 размещения может изменять размеры суперузлов, представляющих текст поверх групп изображений (найденных блоком 234 определения местоположения группы текста поверх изображения) с использованием фонового изображения в качестве виртуального контейнера. Затем текст изменяется в размере для достижения желаемой высоты и ширины. Это включает схему размера шрифта и переформатирование текста, которое может изменять форматное соотношение текстового компонента (например, удлиняя текстовый компонент с более крупными символами внутри), как показано со ссылкой на фиг. 6. Как показано, при переходе от сценария 1 к сценарию 2, размер шрифта увеличивается, тогда как ширина поля уменьшается, требуется переформатирование текста. Блок 234 определения местоположения группы текста поверх изображения также может изменять размер группы, когда фоновое изображение изменяется в размере до следующего размера текста - обычно изменяя форматное соотношение.

Очевидно, что может происходить "визуальная авария", если фактическое содержание изображения плохо выглядит при модификации форматного соотношения (например, изображение человека), например, проиллюстрированного на фиг. 7. Блок 234 определения местоположения группы текста поверх изображения может пытаться решить эту проблему путем проверки содержания изображения на предмет, содержит ли оно подробную информацию, или путем сохранения внутреннего форматного соотношения изображения посредством трансфокации и обрезки изображения вместо изменения его размеров.

Некоторые компоненты в некоторой степени автономны при осуществлении изменения своих собственных размеров. Например, компоненты стороннего приложения (которые могут поддерживать множественные варианты, отличающиеся размерами дисплея) и приложения списка (которые могут иметь множественные ассоциации видов). Такие компоненты осуществляют изменение своих собственных размеров на основании целевого размера, обеспеченного блоком 270 размещения.

Блок 270 размещения также может, при необходимости, изменять размер/повторять декоративное изображение для согласования с размером новой страницы. Если страница превышает данный параметр длины, блок 270 размещения может создавать меню навигации внутри страницы и добавлять его как отдельное меню на страницу или, при необходимости, в любое из существующих унифицированных меню.

Блок 270 размещения может перемасштабировать как высоту, так и ширину, сохраняя форматное соотношение компонента. Некоторые компоненты могут сохранять свою высоту и изменять только свою ширину (например, галереи, модифицированные для использования меньшего количества столбцов). Компоненты, которые должны использовать конкретную высоту при перемасштабировании (например, отображать компоненты, которые имеют внутреннюю прокрутку, и таким образом, оставлять поле над и под ними для прокрутки страницы вместо прокрутки компонента), можно обрабатывать надлежащим образом. Очевидно, что уменьшение ширины можно обрабатывать таким же образом.

Блок 270 размещения также может применять процессы уменьшения/увеличения ширины к компонентам, а также, рекурсивно, к суперузлам и их потомкам. Процессы могут включать изменение размеров фактического компонента, схемы размера шрифта текста, переформатирование текста и уменьшение текста путем его разделения на видимую часть и расширение "показать больше" (которое отображается только по запросу). Другие процессы могут включать переключение компонентов на "облегченные" мобильные версии, модификацию параметров компонентов для изменения их ширины. Обработка также может включать изменение размеров компонентов для отражения используемого фактического размера.

После того как блок 270 размещения размещает подходящие компоненты в их правильных позициях в компоновке мобильной конфигурации, постпроцессор 202 может обрабатывать любые регулировки окончательной компоновки.

Блок 281 вставки автоматически добавляемых компонентов может вставлять автоматически добавляемые компоненты в новую конфигурацию мобильной компоновки, например системные меню или другие виджеты для мобильных устройств, которые могут добавляться во все или некоторые преобразованные приложения (во все или некоторые из их страниц). Такие вставленные автоматически добавляемые компоненты также может включать (например) рекламу и/или другой популяризирующий материал. Такие вставленные автоматически добавляемые компоненты могут быть необходимыми (т.е. всегда вставлять их) или необязательными. В последнем случае их можно вставлять условно на основании параметров или комбинации таких параметров, (например) как тип или профиль пользователя (например, всегда конкретные автоматически добавляемые компоненты для всех пользователей в Европе) или для каждого типа или профиля мобильного устройства (например, вставлять конкретные автоматически добавляемые компоненты для всех пользователей на базе android, имеющих размер экрана 480×320 или более). Они также могут основываться на параметрах или условиях конкретной страницы (например, вставлять конкретные автоматически добавляемые компоненты для страниц, которые не содержат компонентов изображения размером 320×200 или более), поведении пользователя или истории использования приложения, наличия свободного места для вставки автоматически добавляемых компонентов, любых параметров, заданных разработчиком, и параметров, заданных системой построения веб-сайта.

Очевидно, что блок 281 вставки автоматически добавляемых компонентов может размещать такие автоматически добавляемые компоненты различными способами. Он может размещать их как автоматически добавляемые компоненты абсолютной позиции, которые вставляются в данную позицию на странице или в конкретных контейнерах (т.е. суперузлах), обычно с использованием зарезервированного свободного места, как описано ниже. Блок 281 вставки автоматически добавляемых компонентов также может вставлять их как автоматически добавляемые компоненты относительной позиции, вставленные в данную позицию относительно некоторых элементов страницы (например, заранее определенных компонентов, компонентов, расположенных по запросу, и т.д.). Это аналогично вставлению компонентов на уровне конфигурации мобильной компоновки, как более подробно описано ниже.

Блок 281 вставки автоматически добавляемых компонентов также может добавлять автоматически добавляемые компоненты как автоматически добавляемые компоненты свободного места, которые вставляются на основании наличия свободного места в компоновке. Поскольку компоненты иногда перемещаются к отдельной линии компонентов, в ряде случаев существуют дополнительные освобожденные места в конфигурации мобильной компоновки по сравнению с конфигурацией настольной компоновки, которую можно использовать для таких автоматически добавляемых компонентов.

Очевидно, что в случае автоматически добавляемых компонентов абсолютной позиции, блоку 281 вставки автоматически добавляемых компонентов может потребоваться зарезервировать конкретное свободное место в области отображения на мобильном устройстве для этих элементов, выбирая область из прямоугольной области отображения на мобильном устройстве, выделенной странице/суперконтейнере и резервируя ее. Таким образом, область, используемая для линий компонентов, может быть непрямоугольной. Обратимся к фиг. 24, где показано, что область, оставшаяся в главном контейнере С, после исключения зарезервированной области для автоматически добавляемых компонентов А и В является непрямоугольной фигурой.

Очевидно, что подходящая система визуального проектирования также может поддерживать дина-

мическую компоновку, включающую использование явных (указанных разработчиком) привязок, а также неявных (автоматически создаваемых) привязок, как описано в патентной публикации США № 20130219263 под названием "A SERVER BASED WEB SITE DESIGN SYSTEM INTEGRATING DYNAMIC LAYOUT AND DYNAMIC CONTENT", опубликованной 22 августа 2013 г. и принадлежащей заявителю. Таким образом, система визуального проектирования может иметь привязки (неявные и явные), которые могут быть нарушены вследствие процесса перестановки, характерного для адаптации к ширине мобильного устройства.

После соответствующего размещения компонентов обработчик 284 динамической компоновки может модифицировать любые существующие привязки динамической компоновки согласно новой компоновке. Например, обработчик 284 динамической компоновки может удалять привязки для компонентов, которые переместились таким образом, что привязка стала несущественной, как показано со ссылкой на фиг. 3. Как показано, горизонтальная привязка, которая существует между компонентами A и B в сценарии 1, должна нарушаться, когда B перемещается, располагаясь под A в сценарии 2.

Обработчик 284 динамической компоновки также может сохранять привязки, которые могут сохраняться (например, наборы компонентов, которые не были модифицированы, внутри контейнеров, которые не были модифицированы и т.д.). Обработчик 284 динамической компоновки также может модифицировать привязки, которые могут сохраняться, но требовать модификации их параметров (например, изменения длины привязки для компонентов, которые переместились ближе друг к другу). Обработчик 284 динамической компоновки также может создать новые привязки динамической компоновки между близко размещенными компонентами (например, один над другим), включающие как привязки компонента к компоненту, так и привязки компонента к контейнеру. Это может базироваться на критериях автоматического создания привязки, встроенных в систему (например, на основании величины перекрытия и расстояния между компонентами).

Очевидно, что вновь созданные привязки важны, поскольку приложение, адаптированное к мобильным устройствам, все же может быть приложением визуального проектирования в абсолютных координатах. Таким образом, приложение, адаптированное к мобильным устройствам, может потребоваться динамически модифицировать вследствие изменений, например, модификаций содержания компонента, например, объема содержащегося текста. Они могут включать изменения, происходящие из внешнего источника (например, внешнего ввода данных, параллельной пользовательской активности, переключение между записями данных в приложении списка). Обработчику 284 динамической компоновки может потребоваться реализовать изменения для дополнительной адаптации к малым модификациям размера экрана целевого мобильного устройства. Таким образом, обработчик 284 динамической компоновки может модифицировать структуру привязки и также может реализовывать новую структуру привязки для окончательной адаптации.

Очевидно, что результирующая альтернативная компоновка (включающая информацию группирования, POS и упорядочения) может храниться в базе данных 50 совместно с приложением или отдельно для дальнейшего использования, или как основа для дальнейших модификаций, как описано ниже.

Блок 290 отображения может отображать вновь модифицированную компоновку на целевой платформе.

Как рассмотрено выше, дополнительный преобразователь 300 компоновки может объединять модификации, осуществляемые на данной конфигурации настольного уровня и модификации параллельной ей теневой конфигурации мобильной компоновки для создания обновленной конфигурации мобильной компоновки.

Очевидно, что после запуска преобразователя 200 начальной компоновки существуют две (или более) версии приложения - настольная (главная) конфигурация компоновки и одна (или более) мобильная (теневая) конфигурация компоновки. Очевидно, что, говоря о мобильной (или теневой) конфигурации компоновки, можно подразумевать другую дополнительную конфигурацию, например, связанную с планшетом, широкоэкранным дисплеем и т.д. Затем разработчик может применять (отдельные) модификации посредством подходящего инструмента редактирования системы визуального проектирования конфигураций как настольной, так и мобильной компоновки, создавая модифицированную версию каждой. Система визуального проектирования может обеспечивать единый редактор для обеих версий (возможно, ограниченных при редактировании мобильной компоновки) или отдельные редакторы для настольного и мобильного сайта.

Поскольку конфигурация мобильной компоновки выводится из конфигурации настольной компоновки, преобразователь 200 начальной компоновки может применять модификации (например, добавление и удаление страниц и компонентов, а также изменение содержания компонента), чтобы превратить конфигурацию настольной компоновки в конфигурацию мобильной компоновки. Дополнительный преобразователь 300 может объединять любые модификации конфигурации настольной компоновки и любые отдельные модификации конфигурации мобильной компоновки в модифицированную конфигурацию мобильной компоновки для создания окончательной конфигурации мобильной компоновки, подлежащей отображению мобильным пользователям, поддерживая содержание конфигурации мобильной компоновки лучше скоординированным с конфигурацией настольной компоновки. Очевидно, что эта

процедура может быть сложной, поскольку (например), если компонент добавляется к конфигурации настольной компоновки, которая также подлежит добавлению к модифицированной конфигурации мобильной компоновки (компоненты которой могут перемещаться, изменяться в размере или удаляться, в том числе, перемещаться из, в или между контейнерами), дополнительный преобразователь 300 должен определить надлежащую позицию для размещения добавленного компонента в компоновке модифицированной конфигурации, а также компоновке, подлежащей использованию для самих добавленных компонентов.

Очевидно, что в этом сценарии любые модификации конфигурации мобильной компоновки не объединяются обратно в конфигурацию настольной компоновки. Хотя редактирование настольного сайта может изменять мобильный сайт, редактирование мобильного сайта не влияет на настольный сайт. Также очевидно, что изменения, касающиеся только компоновки (т.е. изменения позиции и размера) в отношении либо конфигурации настольной компоновки, либо конфигурации мобильной компоновки, могут не влиять на другую конфигурацию компоновки в отличие, например, от изменений содержания или удаления компонента (которые влияют на конфигурацию мобильной компоновки, если осуществляются в конфигурации настольной компоновки). Очевидно также, что система 100 может ограничивать мобильное редактирование, чтобы компоненты можно было удалять (или скрывать) из конфигурации мобильной компоновки, но нельзя было добавлять или редактировать их содержание, за исключением, возможно, некоторых мобильно-ориентированных компонентов, как рассмотрено более подробно ниже. Таким образом (например), мобильный редактор может не позволять перемещать компонент в конфигурации мобильной компоновки с одной страницы на другую, поскольку это может приводить к добавлению компонентов на страницу.

Обратимся к фиг. 25, где показаны элементы дополнительного преобразователя 300. Дополнительный преобразователь 300 может содержать блок 310 удаления страницы, блок 320 добавления страницы, блок 330 удаления компонента, блок 340 добавления компонента, блок 350 модификации компонента и мобильный обработчик 360. Блок 340 добавления компонента может дополнительно содержать идентификатор 342 добавленного компонента, блок 344 поиска родителя/предшественника и блок 346 добавления мобильной компоновки, как показано со ссылкой на фиг. 26. Функциональные возможности этих элементов более подробно описаны ниже.

Обратимся к фиг. 27, где показаны функциональные возможности дополнительного преобразователя 300. Преобразователь 200 начальной компоновки может преобразовывать DP1 (настольную страницу 1), создавая результирующую конфигурацию мобильной компоновки MP2. Затем разработчик может редактировать MP1 с использованием локального мобильного редактора для создания MP2. После этого, разработчик может редактировать DP1 с использованием локального настольного редактора для создания DP2. Затем дополнительный преобразователь 300 может создавать обновленную конфигурацию мобильной компоновки, объединяя изменения, произведенные в DP2, а также в MP2.

Очевидно, что это автоматическое объединение может быть уместно, только если фаза настольного редактирования ($DP1 \Rightarrow DP2$) осуществляется после фазы мобильного редактирования ($MP1 \Rightarrow MP2$) (как показано на фиг. 26). В обратном случае (сначала осуществляется настольное редактирование) преобразователь 200 начальной компоновки может автоматически повторно преобразовывать модифицированную конфигурацию настольной компоновки DP2 в конфигурацию мобильной компоновки MP2 по завершении редактирования DP2. Таким образом, разработчик осуществляет изменения $MP1 \Rightarrow MP2$ на уже модифицированной MP1, что требует ручного объединения изменений в ходе (позже) мобильного редактирования.

Очевидно также, что целью выполнения одного и того же сайта на множественных платформах является, по возможности, полное совпадение содержания в обоих сайтах. Однако, как рассмотрено выше, не все настольные компоненты совместимы с мобильными вследствие (например) использования выпадающих меню и вертикальных линий.

Также очевидно, что основным требованием к интерактивным приложениям на основе всемирной паутины (например, веб-сайтам) является совместимость поисковых машин. Таким образом, обе конфигурации настольной и мобильной компоновки должны представлять одну и ту же общую структуру страницы поисковой машине, и одни и те же страницы должны быть доступны в обеих конфигурациях компоновки. В противном случае, страница может индексироваться агентом поисковой машины, считывающим страницы в конфигурации настольной компоновки, но не будет доступна, когда пользователь достигнет ее непосредственно посредством поисковой машины на мобильном устройстве (генерирующей "пропавшую веб-страницу" или аналогичное условие ошибки). В обратном случае страница, доступная в конфигурации мобильной компоновки, но не в (индексированной) конфигурации настольной компоновки, не будет найдена поисковой машиной. Некоторые поисковые машины должны поддерживать разные структуры страницы для разных версий сайта, но использование и наличие такого варианта является внутренним свойством конкретной поисковой машины и служит слабой опорой.

Очевидно, что, хотя настольный и мобильный сайты могут выглядеть по-разному, они могут по мере возможности совместно использовать одни и те же данные. В противном случае пользователь может

достигать мобильной страницы путем поиска (например) данного текстового выражения, которое присутствует в содержании настольной страницы, только чтобы обнаружить, что мобильная страница отличается и не содержит искомого выражения. В частности, мобильная страница не должна содержать компонент, которого нет на настольной странице. Можно применять некоторые исключения, например, добавление нового меню, адаптированного к мобильному устройству для замены существующих меню, или удаление текстовых компонентов только в конфигурации мобильной компоновки.

Также очевидно, что время от времени разработчику может потребоваться намеренно производить конкретные модификации конфигурации мобильной компоновки, чтобы она отличалась от конфигурации настольной компоновки, в противоречие политике поисковой машины. Например, система, отображающая сообщение сводок о новых предметах, может отображать более длинные сводки в конфигурации настольной компоновки и более короткие сводки в конфигурации мобильной компоновки. Конечно, разработчик может модифицировать размер, позицию и упорядочение компонента, и такие изменения, в общем случае, не должны влиять на обработку мобильного сайта поисковой машиной.

Для разрешения этого конфликта или, по меньшей мере, его сглаживания дополнительный преобразователь 300 может ограничивать объем модификаций, которые разработчик может осуществлять в конфигурации мобильной компоновки. Например, разработчика можно ограничивать модификациями, которые могут влиять только на компоновку конфигурации мобильной компоновки, но не на фактическое содержание. Такое ограничение может обеспечиваться мобильным редактором системы до обработки различных модификаций дополнительным преобразователем 300.

Модификации настольной компоновки могут включать добавление страниц, удаление страниц, добавление компонентов, удаление компонентов, изменение атрибутов компонентов, перемещение и изменение размеров компонентов и изменение содержания компонента (например, текста внутри абзаца текста). Как упомянуто выше, изменения, касающиеся только компоновки (например, перемещение и изменение размеров), не влияют на конфигурацию мобильной компоновки.

Как рассмотрено выше, система визуального проектирования может иметь внутренний уникальный ID для каждого компонента, который не изменяется при модификации или перемещении компонента. Очевидно, что дополнительный преобразователь 300 может использовать эти ID для отслеживания компонентов, модифицированных от одной версии страницы к другой. Поскольку дополнительный преобразователь 300 предполагает, что все модификации осуществляются посредством редактора системы визуального проектирования и полностью отслеживаются, при оценивании модифицированной версии, дополнительный преобразователь 300 может иметь (для каждого модифицированного компонента) подробный список модифицированных атрибутов и изменений значений (например, для компонента X стиль был изменен с S1 на S2 и ширина с W1 на W2).

При удалении существующей настольной страницы блок 310 удаления страницы также может удалять страницу из конфигурации мобильной компоновки. Очевидно, что это означает, что конкретные модификации конфигурации мобильной компоновки применительно к этой странице теряются. Восстановить их можно только посредством функциональных возможностей "отмены" (например, отмены на уровне сеанса в ходе текущего сеанса или возврата к предыдущей версии на уровне базы данных).

При добавлении новой страницы в настольную конфигурацию блок 320 добавления страницы может предписывать преобразователю 200 начальной компоновки преобразовывать всю страницу в конфигурацию мобильной компоновки, как описано выше. Очевидно, что преобразованная страница может оказаться в надлежащей позиции в конфигурации мобильной компоновки. Блок 320 добавления страницы может обновлять любые мобильные навигационные меню для включения во вновь вставленную страницу.

При удалении компонента из существующей конфигурации настольной компоновки блок 330 удаления компонента может удалять соответствующий компонент из соответствующей конфигурации мобильной компоновки. Блок 330 удаления компонента также может предписывать мобильному обработчику 360 обновлять отображение мобильной компоновки, закрывая промежутки в данной линии компонентов или удаляя всю линию компонентов, как описано ниже в описании мобильного обработчика 360.

Очевидно, что при добавлении одного или более новых компонентов к настольной странице новые компоненты могут создавать новый иерархический порядок компонентов для модифицированной настольной страницы, включающий вновь добавленные компоненты. Чтобы изменять конфигурацию настольной компоновки только в отсутствие изменений конфигурации мобильной компоновки, дополнительный преобразователь 300 может предписывать преобразователю 200 начальной компоновки повторно преобразовывать настольную страницу ввиду отсутствия необходимости в объединении конфигурации настольной компоновки и модификаций конфигурации мобильной компоновки.

Очевидно, что блок 320 добавления компонента должен находить надлежащее место в конфигурации мобильной компоновки для добавления компонента, добавленного в конфигурацию компоновки. Эта задача может быть особенно сложна, поскольку компоненты могут перемещаться (в том числе, между контейнерами), изменяться в размере, изменять свое местоположение или удаляться из конфигурации мобильной компоновки. Кроме того, компонент конфигурации настольной компоновки может перемещаться или удаляться. Таким образом, блок 320 добавления компонента должен находить надлежащую

позицию для добавления компонентов согласно предшественнику и/или родителю, один из которых всегда существует.

Для каждого компонента X , добавленного в конфигурацию мобильной компоновки, идентификатор 342 добавленного компонента может идентифицировать добавленный компонент (например, посредством сравнения ID компонента, редактирования истории сеансов или доступа к нижележащей базе данных). Идентификатор 342 компонента может предписывать блоку 221 фильтрации компонентов использовать его функцию фильтрации компонентов для определения, подходит ли компонент для дисплея мобильного устройства в целом (например, не является пустым компонентом, не относится к типу, не согласующемуся с мобильным устройством и т.д.).

Затем блок 344 поиска родителя/предшественника может осуществлять поиск родителя/предшественника для определения местоположения ближайшего предшествующего компонента $PD(X)$ и/или родительского компонента $PT(X)$, которые являются ближайшими к добавленному компоненту X и которые также существуют в конфигурации мобильной компоновки (т.е. не были удалены из конфигурации мобильной компоновки явно или вследствие своего типа). Предшественник определяется согласно порядку, указанному блоком 240 упорядочения на основании ближайшего предшественника, который появляется в конфигурации мобильной компоновки. Родитель определяется согласно иерархии контейнеров страниц на основании родительского контейнерного компонента ближайшего уровня, который появляется в конфигурации мобильной компоновки. Очевидно, что некоторые компоненты могут быть удалены (или скрыты) в конфигурации мобильной компоновки вручную и, таким образом, исключены из поиска родителя/предшественника. Также очевидно, что ближайший предшествующий компонент может быть пустым, например, если X является первым компонентом в своей странице/контейнере (за исключением компонентов, не наблюдаемых на мобильном устройстве), он не имеет предшественника. Однако очевидно также, что родительский компонент не может быть пустым, поскольку поиск всегда достигает вмещающей страницы, которая служит контейнером верхнего уровня. После того как блок 344 поиска родителя/предшественника определяет предшественника и родителя (при наличии), блок 346 добавления мобильной компоновки может размещать добавленный компонент X после ближайшего предшествующего компонента в позиции, несущей то же соотношение с предшествующими/родительскими компонентами в конфигурации мобильной компоновки, что и X с теми же предшествующими/родительскими компонентами в конфигурации настольной компоновки, и понижают все компоненты, которые располагаются после предшественника.

Очевидно, что при обработке множественных добавленных компонентов конфигурации настольной компоновки ($X[1]$, $X[2]$, ...) существует два основных способа реализации вышеописанной процедуры - поочередный способ или кластеризованный способ.

В поочередном способе добавленные компоненты конфигурации настольной компоновки $X[i]$ обрабатываются согласно их порядку (заданному, например, блоком 240 упорядочения). Для каждого компонента $X[i]$ блок 344 поиска родителя/предшественника осуществляет поиск предшественника/родителя, который учитывает ранее добавленные компоненты $X[j]$ ($1 \leq j < i$). Таким образом, каждый компонент $X[i]$ обрабатывается с учетом других добавленных компонентов.

В кластеризованном способе блок 344 поиска родителя/предшественника может осуществлять поиск родителя/предшественника отдельно для каждого из добавленных компонентов конфигурации настольной компоновки $X[i]$, игнорируя наличие других добавленных компонентов конфигурации настольной компоновки $X[j]$ (для $j \neq i$). Таким образом, блок 344 поиска родителя/предшественника осуществляет поиск родителя/предшественника для каждого $X[i]$, как если бы это был только компонент, добавленный в конфигурацию настольной компоновки. После этого компоненты $X[i]$ кластеризуются согласно общей комбинации предшествующего компонента $PD(X[i])$ и родительского компонента $PT(X[i])$. Компоненты в каждом кластере можно группировать друг с другом в виде единой виртуальной страницы (например, в виде виртуального суперузла, возможно, содержащего в себе другие суперузлы). Затем блок 340 добавления компонента может запускать преобразователь 200 начальной компоновки на каждой такой виртуальной странице в отдельности, рекурсивно выполняя полный процесс преобразования (включающий предварительную обработку, анализ, привязки и т.д., как описано выше). Процесс может преобразовывать каждую виртуальную страницу в преобразованную виртуальную страницу, которую блок 346 добавления мобильной компоновки затем помещает как целое в конфигурацию мобильной компоновки на основании общего предшественника и родителя для кластера.

Обратимся к фиг. 28A-28G которые иллюстрируют разные сценарии добавления и редактирования компонентов и их обработку блоком 340 добавления компонента.

Обратимся к фиг. 28A, где показана настольная страница $DP1$, содержащая два компонента, A и B . Преобразователь 200 начальной компоновки может преобразовывать эту страницу в мобильную страницу $MP1$. Если новый компонент X затем добавляется между A и B в $DP1$ для создания страницы $DP2$, блок 344 поиска родителя/предшественника осуществляет поиск родителя/предшественника для X , находя, что X следует за A внутри вмещающей страницы (в конфигурации настольной компоновки). Таким образом, блок добавления мобильной компоновки вставляет X после A во вмещающую страницу в $MP1$.

для создания MP2.

В другом сценарии, как показано со ссылкой на фиг. 28В, мобильный редактор редактирует мобильную страницу MP1, чтобы А и В обменивались вертикальными позициями, создавая MP2. После этого разработчик вставляет новый компонент Х между А и В в DP1 для создания страницы DP2. Блок 340 добавления компонента может согласовывать изменения, произведенные в отношении DP2 и MP2. Блок 344 поиска родителя/предшественника может осуществлять поиск родителя/предшественника для компонента Х, находя, что Х следует за А (его ближайшим предшественником) внутри вмещающей страницы (его ближайшего родителя в конфигурации настольной компоновки). Таким образом, блок 346 добавления мобильной компоновки может вставлять Х в MP2 во вмещающую страницу, следующую за компонентом А (который теперь находится под В, а не над В), создавая обновленную мобильную страницу MP3.

В еще одном сценарии, как показано со ссылкой на фиг. 28С, мобильный редактор редактирует мобильную страницу MP1 для удаления (сокрытия) компонента В, который располагается под А и над С (создавая MP2). Затем настольный редактор редактирует DP1 для вставки компонента Х после В и до С (создавая DP2). Затем блок 340 добавления компонента может согласовывать изменения, произведенные в отношении DP2 и MP2. Блок 344 поиска родителя/предшественника может осуществлять поиск родителя/предшественника для добавленного компонента Х и может определять, что предшественником Х является А, а не В, поскольку В помечен как "удаленный в мобильной конфигурации". Таким образом, блок добавления мобильной компоновки вставляет Х после А (и до С) в MP2 для создания MP3.

В еще одном сценарии, как показано со ссылкой на фиг. 28D, страница DP1 содержит контейнер А, который содержит внутренний контейнер В и другой компонент D, которые все преобразуются в MP1. Затем мобильный редактор редактирует мобильную страницу MP1 для удаления (сокрытия) внутреннего контейнера В (создавая MP2). Затем настольный редактор редактирует DP1 для вставки компонента С во внутренний контейнер В (создавая DP2). Затем блок 340 добавления компонента может согласовывать изменения, произведенные в отношении DP2 и MP2. В этом сценарии при осуществлении поиска родителя/предшественника для добавленного компонента С, блок 344 поиска родителя/предшественника может определять, что родителем является А, а не В, поскольку В помечен как "удаленный в мобильной конфигурации". Блок 344 поиска родителя/предшественника может дополнительно определять, что С не имеет предшественников, поскольку он является первым компонентом в А (без учета В). Таким образом, блок 346 добавления мобильной компоновки может вставлять С внутрь А (вверху и до D) в MP2 для создания MP3.

В еще одном сценарии, как показано со ссылкой на фиг. 28Е, мобильный редактор редактирует мобильную страницу MP1, чтобы А и В обменивались вертикальными позициями, создавая MP2. После этого разработчик вставляет два новых сильно перекрывающихся компонента X1 и X2 между компонентами А и В в DP1 для создания страницы DP2. Блок 340 добавления компонента может согласовывать изменения, произведенные в отношении DP2 и MP2 одним из двух возможных способов, генерируя либо MP3 (для поочередного способа), либо MP4 (для кластеризованного способа).

Очевидно, что, если используется поочередный способ и поскольку X2 геометрически располагается "позже" X1, блок 344 поиска родителя/предшественника может сначала осуществлять поиск родителя/предшественника X1, определяя, что А является предшественником X1. Затем блок 344 поиска родителя/предшественника может осуществлять поиск родителя/предшественника для X2 (с учетом X1) и может определять, что X1 является предшественником X2. Таким образом, затем блок 346 добавления мобильной компоновки может размещать компонент X1 после А и размещать компонент X2 после X1, таким образом создавая MP3. Очевидно, что согласно этому способу компоненты X1 и X2 анализируются и перемещаются как отдельные объекты и, таким образом, их композиции разделяются.

Если используется кластеризованный способ, блок 344 поиска родителя/предшественника может осуществлять поиск родителя/предшественника для обоих X1 и X2 без учета X1 или X2 (для поиска) и может определять, что оба имеют общего родителя (главную страницу) и общего предшественника (компонент А). Таким образом, блок 340 добавления компонента может совместно кластеризовать X1 и X2, может создавать виртуальный суперузел, содержащий X1 и X2, и может предписывать преобразователю 200 начальной компоновки создавать версию, готовую для мобильного использования композиции X1 и X2. Очевидно, что эта преобразованная композиция будет размещаться ниже А, создающего MP4. Поскольку X1 и X2 сильно перекрываются, они, вероятно, изменяются в размере для сохранения их композиции (как описано выше).

В еще одном сценарии, как показано со ссылкой на фиг. 28F, настольный редактор редактирует первоначальную настольную страницу DP1, содержащую два компонента (А и В) путем добавления 4 новых компонентов - двух пар семантически связанных компонентов текста и изображения (T1/I1 и T2/I2), создавая новую страницу DP2. Если DP2 преобразуется в мобильную с использованием поочередного способа (как описано выше), блок 344 поиска родителя/предшественника может определять, что T1 является предшественником T2. Таким образом, генерируемая (вертикальная) последовательность компонентов представляет собой А-T1-T2-I1-I2-В (как показано в MP2), нарушая соединения текстовой подписи с изображением. Если DP2 преобразуется в мобильную с использованием кластеризованного

способа (как описано выше), блок 344 поиска родителя/предшественника может определять, что T1/T2/I1/I2 все имеют общего предшественника (А) и родителя (главную страницу). Таким образом, блок 346 добавления мобильной компоновки может кластеризовать эти 4 компонента в новую виртуальную страницу и может запускать преобразователь 200 начальной компоновки. Поскольку T1 семантически связан с I1 и T2 с I2, эти пары будут оставаться вместе. Таким образом, созданная (вертикальная) последовательность представляет собой А-T1-I1-T2-I-B (как в МР3), сохраняя семантическую информацию.

В еще одном сценарии, как показано со ссылкой на фиг. 28G, настольная страница DP1 содержит три компонента (А, В и С) и преобразуется в мобильную. Мобильный редактор редактирует результирующую мобильную страницу MP1, удаляя компонент С и меняя порядок расположения А и В для создания MP2. Затем настольный редактор редактирует DP1, добавляя два перекрывающихся компонента X1 и X2 между В и С и дополнительный компонент X3 ниже С для создания DP2. Блок 340 добавления компонента выполнен с возможностью использования кластеризованного способа, и, таким образом, все три добавленных компонента (X1, X2 и X3) кластеризуются совместно, поскольку блок 344 поиска родителя/предшественника определил, что В является предшественником всех 3 новых компонентов (поскольку С удален в мобильной версии). Таким образом, блок 346 добавления мобильной компоновки может размещать все 3 новых компонента ниже В и выше А, сохраняя композицию X1 и X2 (как показано в МР3).

Очевидно, что размещение добавленных компонентов блоком 346 добавления мобильной компоновки можно реализовать в ограниченной версии, в которой компоненты добавляются в настольную версию или добавляются только в мобильную версию вертикально между двумя вертикально разделенными компонентами.

В другом варианте осуществления блок 340 добавления компонента может добавлять "кластер" добавленных компонентов куда-либо в "линию" компонентов, присутствующую в мобильной версии, чтобы не ограничивать вставку новых компонентов местом под данным компонентом в мобильной версии.

В еще одном варианте осуществления блок 340 добавления компонента может присоединять добавленный компонент В к контейнеру конфигураций настольной компоновки, который содержит его, но не в качестве предшественника. Очевидно, что в этом сценарии компонент В остается присоединенным к содержащему его контейнеру С и располагается после последнего элемента в С, который существует в конфигурациях настольной и мобильной компоновки и который предшествует компоненту В. Этот подход может использоваться, пока контейнер С (и совпадающий суперузел) не пуст или не скрыт в конфигурации мобильной компоновки.

В еще одном варианте осуществления блок 340 добавления компонента может добавлять компоненты в конфигурацию мобильной компоновки путем горизонтального разделения конфигурации мобильной компоновки на сегменты и добавления добавленного компонента или кластера компонентов после сегмента, содержащего предшественник, а не непосредственно после предшественника.

Очевидно, что при наличии изменений атрибутов, например стиля, шрифта или размера текста настольного компонента, включающих изменения общего свойства, например изменение количества строк или столбцов в компоненте галереи, блок 350 модификации компонента может модифицировать компоненты в конфигурации мобильной компоновки. Поскольку конфигурация мобильной компоновки задает визуальное приложение, подлежащее динамической компоновке, она может принимать такие изменения и соответственно перерегулировать размеры и позиции компонентов.

Также очевидно, что, хотя изменения размера и позиции настольного компонента влияют на конфигурацию настольной компоновки, они не влияют на конфигурацию мобильной компоновки, пока конфигурация мобильной компоновки не будет явно повторно сгенерирована, как описано ниже. Это возможно, поскольку система 100 может допускать различия во внешнем виде (конструкции) между множественными конфигурациями компоновки, при условии, что содержание не изменяется.

В отсутствие изменений в данных/содержании настольного компонента блок 350 модификации компонента может соответственно обновлять конфигурацию мобильной компоновки. Аналогично с модификацией атрибутов такое изменение данных может приводить к некоторому переформатированию компонентов в конфигурации мобильной компоновки, обрабатываемой с использованием динамической компоновки.

Очевидно, что любая модификация конфигурации мобильной компоновки может осуществляться пользователем обычно посредством специализированного отображения GUI (мобильного редактора) или отображения мобильных модификаций с конфигурацией компоновки предварительного просмотра для мобильного устройства. Такой мобильный редактор может позволять разработчику осуществлять лишь определенные операции, например добавление компонентов, характерных для мобильных устройств, удаление компонентов (сокрытие их), перемещение компонентов или только модификацию компонентов. Таким образом, мобильный редактор может взаимодействовать с мобильным обработчиком 360, который обрабатывает только изменения на мобильной стороне.

Мобильный обработчик 360 может использовать несколько способов размещения добавленных (только мобильных) компонентов. Мобильный обработчик 360 дает возможность вставлять новый компонент только между двумя компонентами на линии компонентов или на новой линии компонентов, сле-

дующей за или предшествующей существующей, как показано со ссылкой на фиг. 29. Новый компонент X можно вставлять между A и B или новый компонент Y можно вставлять ниже P и выше A, B и C. Он также может вставлять разрывы линии до/после добавленного компонента.

Вставленный компонент (X или Y) в этом случае может быть привязан к предшествующему компоненту (или контейнеру) и может быть повторно вставлен в эту позицию, если повторно сгенерирована конфигурация мобильной компоновки и запрошено "сохранение добавленных мобильных компонентов". Если привязка недоступна (например, привязанный компонент удален в конфигурации настольной компоновки), мобильный обработчик 360 может искать ближайший предыдущий компонент или контейнер, как упомянуто выше, для добавления компонентов в конфигурацию настольной компоновки.

Мобильный обработчик 360 также может допускать размещение нового (только мобильного) компонента X в произвольной позиции на странице. Затем мобильный обработчик 360 может искать компонент привязки путем поиска компонента, наиболее пересекающегося с X. В отсутствие такого компонента мобильный обработчик 360 может искать ближайшего соседа (со всех 4 сторон) на основании взвешенного среднего расстояния и перекрытия и с использованием порогов расстояния и перекрытия. Если такого компонента, который использует контейнер, содержащий X, не найдено, то мобильный обработчик 360 может использовать начало страницы в качестве привязки, которая может задаваться как привязка в отношении расстояния и смещения от ближайшего края привязки.

Во избежание несогласованности с политиками поисковой машины дополнительный преобразователь 300 может вовсе избегать предложения этой операции. Альтернативно, дополнительный преобразователь 300 может ограничиваться добавлением компонентов, которые не влияют на просмотр страницы поисковой машиной, например компонентов декоративного типа (например, линий, декоративных фигур), компонентов без содержания (например, меню навигации внутри страницы, которые не добавляют дополнительную информацию пункта меню), компонентов с содержанием, которое не поступает на поисковые машины (например, фоновые изображения), и компонентов, которые относятся только к мобильному окружению (например, кнопки, обеспечивающие функциональные возможности для осуществления телефонного вызова, отправки текстового/MMS сообщения, навигации к данному местоположению или переключения на настольную версию сайта).

Также очевидно, что мобильный редактор может помещать компоненты, не рекомендованные для мобильного использования, первоначально в отображение "удаленного компонента" и требовать второго этапа для фактического включения их в конфигурацию мобильной компоновки.

Добавленные компоненты могут быть перечислены в отображении мобильных модификаций, как рассмотрено ниже, для упрощения удаления конкретных компонентов, добавляемых только для мобильных устройств.

Мобильный обработчик 360 также может удалять компонент из конфигурации мобильной компоновки (что можно реализовать фактическим удалением компонента или просто его сокрытием). Очевидно, что при удалении компонента из конфигурации мобильной компоновки мобильный обработчик 360 может не изменять взаимное расположение компонентов на одной и той же линии. При удалении всей линии компонентов мобильный обработчик 360 может перемещать компоненты ниже, вверх, как можно ближе, как показано со ссылкой на фиг. 30. При удалении компонентов A, B или C (но не всех трех) оставшиеся на линии компоненты (A, B или C) не подвергаются влиянию и не перемещаются. При удалении всех трех компонентов (A, B и C) компонент Q и следующие за ним компоненты перемещаются вверх, приближаясь к компоненту P. Удаленный(е) компонент(ы) может(гут) быть перечислен(ы) в отображении мобильных модификаций, как рассмотрено ниже, для упрощения повторной вставки, при необходимости, скрытых компонентов в приложение на более поздней стадии.

Также очевидно, что компоненты в конфигурации мобильной компоновки могут перемещаться пользователем посредством стандартного GUI системы редактирования визуальных данных. Мобильный обработчик 360 может оставлять модификации без изменения, пока страница не будет повторно создана, и может перечислять их в отображении мобильных модификаций, как рассмотрено ниже, для упрощения отмены конкретных изменений.

Мобильный редактор также может допускать вывод компонентов (вправо или влево) за пределы узкой полосы отображения мобильного устройства. В этом сценарии компонент может оставлять некоторое минимальное горизонтальное перекрытие с "полоской" и может отображаться обрезанным до размера полосы. В противном случае этот вариант может становиться эквивалентным варианту "скрывать компонент". Этот вариант может использоваться для демонстрации только части компонента/контейнера или для временного размещения компонентов сбоку для помощи в изменении размещения компонентов в мобильной конфигурации.

Мобильный редактор также может позволять пользователю выборочно изменять атрибуты, свойства и стили компонента в конфигурации мобильной компоновки. Мобильный редактор может помечать такие измененные атрибуты, как "измененные на мобильные", благодаря чему изменение конкретного атрибута в конфигурации настольной компоновки может не влиять (и преодолевать) на изменение, произведенное в отношении этого атрибута в конфигурации мобильной компоновки.

Одно конкретное изменение, которое может быть желательно как изменение размера шрифта, отно-

сится к конфигурации мобильной компоновки. Это может быть необходимо, если блок 227 регулировки компонентов генерирует размер, отличающийся от необходимого разработчику. Таким образом, конкретное изменение размера шрифта (например, коэффициент подлежащий применению к размеру шрифта, генерируемому преобразователем 200 начальной компоновки) может добавляться и применяться к отображаемому тексту для каждого поля. Мобильный редактор может изменять размер шрифта с использованием локального атрибута "масштабный коэффициент" и (например) кнопок для 5%-ного увеличения/уменьшения размера шрифта, которое может влиять на этот атрибут. Очевидно, что мобильный редактор может все же ограничивать новые размеры шрифта данным диапазоном размеров шрифта, пригодных для мобильных устройств. Любая модификация атрибутов (включая изменение размера шрифта) может быть перечислена в отображении мобильных модификаций, как рассмотрено ниже.

Как рассмотрено выше, дополнительный преобразователь 300 может обеспечивать отдельные GUI, отображающие список модификаций, специфичный для данной конфигурации мобильной компоновки (например, добавленные компоненты, удаленные компоненты и модифицированные компоненты). Это может быть полезно для просмотра конкретных модификаций конфигурации мобильной компоновки и различения между ними и другими модификациями компоновки, производимыми преобразователем 200 начальной компоновки. Это также может быть полезно для просмотра разных изменений, произведенных в отношении разных конфигураций мобильной компоновки и чтобы иметь возможность отменять конкретные модификации конфигурации мобильной компоновки, например, для повторной вставки конкретного "удаленного" компонента.

Очевидно, что такой дисплей может делиться согласно странице, в которой осуществлялась модификация, т.е. текущих страниц, другой(их) страницы() или любого шаблона или главной странице или согласно типу модификации, т.е. вставлению, удалению или модификации компонента.

Дисплей также можно использовать для сортировки этих модификаций, например, согласно метке времени, местоположению на странице или типу компонента, подвергающемуся влиянию. Мобильный редактор может поддерживать операцию отмены для выбранного набора модификаций. Затем мобильный обработчик 360 может отменять любые модификации, например повторно вставлять удаленные компоненты в качестве обратной модификации позиции компонента.

Дополнительный преобразователь 300 может также поддерживать варианты регенерации конфигурации мобильной компоновки. В случае активации дополнительный преобразователь 300 может предписывать преобразователю 200 начальной компоновки снова запускаться на нижележащей конфигурации настольной компоновки (которая может быть модифицирована после последнего запуска преобразователя 200 начальной компоновки), и конкретные изменения конфигурации мобильной компоновки могут повторно применяться.

Очевидно, что регенерация может применяться к страницам, которые следует повторно генерировать (текущей странице, конкретным страницам, всему приложению), и изменениям конфигурации мобильной компоновки, которые следует повторно применять после повторной генерации конфигурации мобильной компоновки. Их можно выбирать по категориям (например, только сокрытие компонента), они могут включать конкретные выбранные модификации или все модификации. Также очевидно, что если проход регенерации осуществляется без повторного применения изменений конфигурации мобильной компоновки к данной странице или страницам, соответствующие изменения конфигурации мобильной компоновки отвергаются.

Блок 290 отображения может отображать новую обновленную конфигурацию мобильной компоновки, как описано выше.

Вышеописанную систему можно объединять с подходящей системой построения веб-сайта различными способами. Например, система 100 может быть встроена в фактический код системы построения веб-сайта, либо полностью на стороне клиента, либо полностью на стороне сервера, либо на обеих. Система 100 также может работать на той же серверной платформе, что и система построения веб-сайта или на отдельном автономном сервере или наборе серверов. Система 100 может работать в режиме онлайн (всякий раз при просмотре страницы системы построения веб-сайта), в режиме оффлайн (для создания сохраненных преобразованных версий системы построения веб-сайта) или в комбинации (где некоторые стадии активируются в режиме оффлайн, а другие в режиме онлайн). Она может действовать непосредственно на структурах данных системы построения веб-сайта или обеспечивать интерфейс (например, веб-услугу), вызываемый системой построения веб-сайта для изоляции системы 100 от содержания системы построения веб-сайта.

Очевидно, что в различных вариантах осуществления настоящего изобретения разные элементы системы 100 могут по-разному делить свои функциональные возможности и могут осуществлять вышеупомянутые функциональные возможности в порядке, отличном от вышеописанного. Очевидно также, что система 100 может быть частично реализована с использованием частичных функциональных возможностей.

Таким образом, конфигурацию настольной компоновки конкретного приложения можно надлежащим образом преобразовывать в конфигурацию мобильной компоновки (или любые другие целевые компоновки разных размеров) с учетом соотношения между компонентами.

Кроме того, после начальных преобразований модификации могут осуществляться в отношении конфигураций настольной и мобильной компоновки для одной и той же страницы и могут отражаться во вновь генерируемой мобильной странице.

Если конкретно не указано обратное, как явствует из вышеприведенных рассмотрений, следует понимать, что на протяжении описания изобретения рассмотрения, использующие такие термины, как "обработка", "расчет", "вычисление", "определение" и т.п., относятся к действию и/или процессам компьютера, вычислительной системы, системы клиент/сервер или аналогичного электронного вычислительного устройства, которое манипулирует и/или преобразует данные, представленные как физические, например электронные, величины в регистрах и/или блоках памяти вычислительной системы в другие данные, аналогично представленные как физические величины в блоках памяти, регистрах или других таких устройствах хранения, передачи или отображения информации вычислительной системы.

Варианты осуществления настоящего изобретения могут включать устройство для осуществления описанных операций. Это устройство может быть специально построено для желаемых целей или может содержать компьютер общего назначения, выборочно активируемый или переконфигурируемый компьютерной программой, сохраненной на компьютере. Полученное устройство согласно инструкциям программного обеспечения может превращать компьютер общего назначения в рассмотренные здесь элементы, отвечающие изобретению. Инструкции могут задавать устройство, отвечающее изобретению, в операции с компьютерной платформой, для которой это желательно. Такая компьютерная программа может храниться на машиночитаемом носителе данных, например, но без ограничения, дисках любого типа, включая флоппи-диски, оптические диски, магнитооптические диски, блоки постоянной памяти (ROM), компакт-диски с возможностью только чтения (CD-ROM), блоки оперативной памяти (RAM), блоки электрически программируемой постоянной памяти (EPROM), блоки электрически стираемой и программируемой постоянной памяти (EEPROM), магнитные или оптические карты, флэш-память, флэш-диск или носители любого другого типа, пригодные для хранения электронных инструкций и способные подключаться к системной шине компьютера.

Представленные здесь процессы и дисплеи, по существу, не связаны с одним конкретным компьютером или другим устройством. Различные системы общего назначения могут использоваться с программами в соответствии с изложенными здесь принципами или могут быть удобны для построения более специализированного устройства для осуществления желаемого способа. Желаемая структура для этих различных систем следует из данного описания. Кроме того, варианты осуществления настоящего изобретения не описаны со ссылкой на какой-либо конкретный язык программирования. Очевидно, что для реализации описанных здесь принципов изобретения могут использоваться различные языки программирования.

Хотя здесь проиллюстрированы и описаны определенные признаки изобретения, специалисты в данной области техники могут предложить многочисленные модификации, замены, изменения и эквиваленты. Таким образом, следует понимать, что нижеследующая формула изобретения призвана охватывать все подобные модификации и изменения, отвечающие истинной сущности изобретения.

ФОРМУЛА ИЗОБРЕТЕНИЯ

1. Система для преобразования компоновки страницы веб-сайта в системе построения веб-сайта, содержащая

процессор аппаратного обеспечения;

препроцессор, выполняющийся на упомянутом процессоре аппаратного обеспечения, для анализа объектных свойств, использования и связей элементов системы построения веб-сайта для упомянутой страницы веб-сайта, хранимой в упомянутой системе построения веб-сайта, при этом упомянутая страница имеет исходную компоновку для исходной области отображения на первой платформе;

блок создания суперузлов, выполняющийся на упомянутом процессоре аппаратного обеспечения и имеющий множество блоков определения групп, каждый из которых имеет критерии группирования и выполняется на упомянутом процессоре аппаратного обеспечения, для группирования, на основании выходных данных упомянутого препроцессора, упомянутых элементов системы построения веб-сайта, которые должны оставаться вместе при отображении этих элементов в целевой компоновке, и для замены упомянутой группы виртуальным элементом согласно критериям группирования и создания иерархии суперузлов элементов для упомянутой страницы из выходных данных упомянутых блоков определения местоположения групп;

блок упорядочения, выполняющийся на упомянутом процессоре аппаратного обеспечения, для определения последовательности упомянутых элементов упомянутых суперузлов согласно заранее заданным критериям упорядочения, связанным с порядком чтения;

блок размещения, выполняющийся на упомянутом процессоре аппаратного обеспечения, для размещения упомянутых элементов в пределах области целевой компоновки в целевой компоновке на второй платформе на основании по меньшей мере одного из упомянутой определенной последовательности, зарезервированных мест и упомянутых групп, при этом блок размещения помещает максимально воз-

можное количество элементов в линию элементов в упомянутой целевой компоновке,

при этом упомянутая исходная область отображения имеет атрибуты, отличные от атрибутов упомянутой целевой области отображения; и

обратный преобразователь, выполняющийся на упомянутом процессоре аппаратного обеспечения, для объединения модификаций в упомянутой исходной компоновке и упомянутой целевой компоновке с учетом независимых модификаций, сделанных в упомянутой целевой компоновке для создания обновленной конфигурации целевой компоновки, причем упомянутая обновленная целевая компоновка отражает изменения, сделанные в упомянутой исходной компоновке,

при этом упомянутый процессор аппаратного обеспечения реализует упомянутые препроцессор, блок создания суперузлов, блок упорядочения, блок размещения и обратный преобразователь.

2. Система по п.1, которая также содержит постпроцессор, выполняющийся на упомянутом процессоре аппаратного обеспечения, для регулировки упомянутых размещенных элементов до отображения.

3. Система по п.1, которая реализована по меньшей мере на одном из следующего: клиент, сервер и сторонний сервер.

4. Система по п.1, в которой упомянутый препроцессор содержит по меньшей мере одно из следующего:

обработчик шаблонов для модификации экземпляров упомянутых элементов в шаблонах;

блок фильтрации компонентов для фильтрации упомянутых элементов, непригодных для упомянутой целевой компоновки;

блок адаптации компонентов для адаптации упомянутых элементов к упомянутой целевой компоновке;

блок регулировки компонентов для регулировки упомянутых элементов к упомянутой целевой компоновке;

блок анализа компонентов для анализа атрибутов упомянутых элементов для определения их пригодности к использованию.

5. Система по п.1, в которой упомянутый блок создания суперузлов содержит по меньшей мере один из следующих блоков:

блок определения группы перекрытия для определения групп упомянутых элементов, в которых упомянутые элементы сильно перекрываются, и для замены упомянутой группы виртуальным элементом согласно критерию группировки;

блок определения заранее заданной группы для определения групп упомянутых элементов согласно рекомендациям по меньшей мере от одного из уровней шаблонов, приложений, страниц или элементов и для замены упомянутой группы виртуальным элементом согласно критерию группировки;

блок определения группы текста поверх изображения для определения элементов, которые представляют собой текстовые компоненты, которые накладываются поверх конкретного фонового изображения, для замены упомянутой группы виртуальным элементом согласно критерию группировки;

блок определения показателя для определения показателя определенности для корректности упомянутого блока определения группы перекрытия, упомянутого блока определения заранее заданной группы и упомянутого блока определения группы текста поверх изображения;

блок создания узла для создания иерархии упомянутых суперузлов на основании упомянутых элементов и упомянутых виртуальных элементов.

6. Система по п.1, в которой упомянутый блок упорядочения содержит по меньшей мере один из следующих:

блок основного упорядочения для определения упомянутой последовательности упомянутых элементов упомянутых суперузлов согласно заранее заданному критерию упорядочения;

блок определения частично упорядоченного набора для анализа по меньшей мере одного из семантики, содержания, атрибутов, истории редактирования и геометрии упомянутых элементов упомянутых суперузлов;

блок определения показателя для определения показателя определенности для корректности упомянутого блока основного упорядочения и упомянутого блока определения местоположения частично упорядоченного набора;

интегратор порядка для интеграции последовательности, определенной упомянутым блоком основного упорядочения, с частично упорядоченными наборами, обнаруженными упомянутым блоком определения частично упорядоченного набора, для создания объединенного модифицированного порядка.

7. Система по п.2, в которой упомянутый постпроцессор содержит по меньшей мере один из:

блок вставки автоматически добавляемых компонентов для вставки по меньшей мере одного из автоматически добавляемых элементов и элементов, относящихся к мобильным устройствам; и

координатор динамической компоновки для осуществления по меньшей мере одного из создания, модификации и удаления привязок динамической компоновки, причем упомянутые привязки регулируются согласно упомянутой целевой компоновке.

8. Система по п.6, в которой упомянутый блок основного упорядочения содержит по меньшей мере один из следующих блоков:

блок упорядочения первичного направления для выстраивания в последовательность упомянутых элементов упомянутых суперузлов согласно по меньшей мере одной из строк и столбцов;

блок упорядочения первичного направления с разделением и объединением для выстраивания в последовательность упомянутых элементов упомянутых суперузлов согласно по меньшей мере одной из строк и столбцов и для отслеживания любого разделения и объединения строк/столбцов;

блок горизонтального/вертикального расслоения для чередующегося горизонтального и вертикального расслоения упомянутых элементов упомянутых суперузлов для создания внутреннего дерева делений и для определения последовательности отображения для упомянутых элементов.

9. Система по п.5, в которой упомянутый блок размещения содержит по меньшей мере один из:

интерпретатор рекомендации для интерпретации рекомендаций, присоединенных к упомянутым элементам упомянутых суперузлов;

блок создания разрыва линии для создания разрывов линии согласно упомянутым рекомендациям, созданным упомянутым блоком создания узла и упомянутым блоком упорядочения;

блок уменьшения ширины для уменьшения ширины упомянутых элементов, содержащий по меньшей мере один из блока изменения масштаба, блока изменения шрифта и блока переформатирования текста, при этом упомянутый блок уменьшения ширины сохраняет визуальный дизайн;

блок регулировки размера для применения по меньшей мере одной из регулировок ширины и высоты к упомянутым элементам упомянутых суперузлов; и

блок изменения размера для изменения размера декоративных изображений.

10. Система по п.6, в которой упомянутый блок определения частично упорядоченного набора содержит по меньшей мере один из следующих блоков:

блок определения частично упорядоченного набора кластеров для обнаружения частично упорядоченного набора кластеров, где упомянутые элементы упомянутых суперузлов располагаются ближе друг к другу по сравнению со стандартным разнесением между упомянутыми элементами в упомянутых суперузлах;

блок определения частично упорядоченного семантического набора для обнаружения частично упорядоченного набора семантических соотношений при наличии конкретных комбинаций упомянутых элементов упомянутых суперузлов данных типов, которые близки друг к другу;

блок определения частично упорядоченного набора схем для обнаружения частично упорядоченного набора при наличии установленных схем между упомянутыми элементами упомянутых суперузлов;

блок определения заранее заданного частично упорядоченного набора для обнаружения частично упорядоченного набора на основании по меньшей мере одного из существующих определений группирования, относящегося к редактированию, информации существующей привязки динамической компоновки и повторного использования шаблонов компонентов для разных наборов элементов; и

блок определения частично упорядоченного набора на основе информации сеанса редактирования для обнаружения частично упорядоченного набора упомянутых элементов упомянутых суперузлов на основании информации, собранной из предыдущих сеансов редактирования.

11. Система по п.1, в которой упомянутый обратный преобразователь содержит

блок удаления страницы для удаления страницы из упомянутой целевой компоновки, когда упомянутая страница удалена из упомянутой исходной компоновки;

блок добавления страницы для добавления страницы в упомянутую целевую компоновку, когда упомянутая страница добавлена в упомянутую исходную компоновку;

блок удаления компонента для удаления элемента из упомянутой целевой компоновки, когда упомянутый элемент удален из упомянутой исходной компоновки;

блок добавления компонента для добавления элемента в упомянутую целевую компоновку, когда упомянутый элемент добавлен в упомянутую исходную компоновку;

блок модификации компонента для модификации элемента в упомянутой целевой компоновке, когда упомянутый элемент модифицирован в упомянутой исходной компоновке; и

мобильный обработчик для обработки модификаций в упомянутой целевой компоновке, причем упомянутые модификации не зависят от модификаций в упомянутой исходной компоновке.

12. Система по п.11, в которой упомянутый блок добавления компонента содержит

блок поиска родителя/предшественника для поиска по меньшей мере одного из ближайшего предшественствующего элемента и родительского элемента, ближайших к элементу, добавленному к упомянутой исходной компоновке; и

блок добавления мобильной компоновки для вставки упомянутого добавленного элемента в упомянутую целевую компоновку согласно местоположению упомянутого по меньшей мере одного из ближайшего предшественствующего элемента и родительского элемента.

13. Способ преобразования компоновки страницы веб-сайта в системе построения веб-сайта, содержащий этапы, на которых

анализируют объектные свойства, использование и связи элементов системы построения веб-сайта для текущей страницы веб-сайта, хранимой в упомянутой системе построения веб-сайта, при этом упомянутая страница имеет исходную компоновку на первой платформе;

определяют согласно критериям группирования, на основании выходных данных упомянутого анализа группы упомянутых элементов системы построения веб-сайта, которые должны оставаться вместе при отображении этих элементов в целевой компоновке, заменяют упомянутую группу виртуальным элементом согласно критериям группирования и создают иерархию суперузлов элементов для упомянутой страницы из выходных данных упомянутого определения групп;

определяют последовательность упомянутых элементов упомянутых суперузлов согласно заранее заданным критериям упорядочения, связанным с порядком чтения;

размещают упомянутые элементы в пределах области целевой компоновки на второй платформе на основании по меньшей мере одного из упомянутой определенной последовательности, зарезервированных мест и упомянутых групп и помещают максимально возможное количество элементов в линию элементов в упомянутой целевой компоновке;

при этом упомянутая исходная область отображения имеет атрибуты, отличные от атрибутов упомянутой целевой области отображения; и

объединяют модификации в упомянутой исходной компоновке и упомянутой целевой компоновке с учетом независимых модификаций, сделанных в упомянутой целевой компоновке для создания обновленной конфигурации целевой компоновки, причем упомянутая обновленная целевая компоновка отражает изменения, сделанные в упомянутой исходной компоновке.

14. Способ по п.13, в котором дополнительно регулируют упомянутые размещенные элементы до отображения.

15. Система по п.1 и способ по п.13, в которых упомянутые элементы представляют собой по меньшей мере один из атомарных компонентов, компонентов одностраничного контейнера и компонентов многостраничного контейнера.

16. Способ по п.13, в котором упомянутый анализ содержит по меньшей мере один из этапов, на которых

модифицируют экземпляры упомянутых элементов в шаблонах;

фильтруют упомянутые элементы, непригодные для упомянутой целевой компоновки;

адаптируют упомянутые элементы к упомянутой целевой компоновке;

регулируют упомянутые элементы к упомянутой целевой компоновке;

анализируют атрибуты упомянутых элементов для определения их пригодности к использованию.

17. Способ по п.13, в котором упомянутые определение групп и создание иерархии суперузлов содержат по меньшей мере один из этапов, на которых

определяют группы упомянутых элементов, в которых упомянутые элементы сильно перекрываются, и заменяют упомянутую группу виртуальным элементом согласно критерию группировки;

определяют группы упомянутых элементов согласно рекомендациям по меньшей мере от одного из уровней шаблонов, приложений, страниц или компонентов и заменяют упомянутую группу виртуальным элементом согласно критерию группировки;

определяют элементы, которые представляют собой текстовые компоненты, которые накладываются поверх конкретного фонового изображения и заменяют упомянутую группу виртуальным элементом согласно критерию группировки;

определяют показатель определенности для корректности упомянутого определения групп упомянутых элементов, в которых упомянутые элементы сильно перекрываются, определяют группы упомянутых элементов согласно рекомендациям по меньшей мере от одного из уровней шаблонов, приложений, страниц или компонентов и определяют компоненты, которые представляют собой текстовые компоненты, которые накладываются поверх конкретного фонового изображения; и

создают иерархию суперузлов на основании упомянутых элементов и упомянутых виртуальных элементов.

18. Способ по п.13, в котором упомянутое определение последовательности содержит по меньшей мере один из этапов, на которых

определяют упомянутую последовательность упомянутых элементов упомянутых суперузлов согласно заранее заданному критерию упорядочения;

анализируют по меньшей мере одно из семантики, содержания, атрибутов, истории редактирования и геометрии упомянутых элементов упомянутых суперузлов;

определяют показатель определенности для корректности упомянутого определения последовательности упомянутых элементов упомянутых суперузлов согласно заранее заданному критерию упорядочения и упомянутому анализу по меньшей мере одного из семантики, содержания, атрибутов, истории редактирования и геометрии упомянутых элементов упомянутых суперузлов;

интегрируют последовательность, определенную путем определения показателя определенности для корректности упомянутого определения последовательности упомянутых элементов упомянутых суперузлов согласно заранее определенному критерию упорядочения и упомянутому анализу по меньшей мере одного из семантики, содержания, атрибутов, истории редактирования и геометрии упомянутых элементов упомянутых суперузлов и создают объединенный модифицированный порядок.

19. Способ по п.14, в котором упомянутая регулировка содержит по меньшей мере один из этапов,

на которых

вставляют автоматически добавляемые компоненты и элементы, относящиеся к мобильным устройствам; и

осуществляют по меньшей мере одно из создания, модификации и удаления привязок динамической компоновки, причем упомянутые привязки регулируются согласно упомянутой целевой компоновке.

20. Способ по п.18, в котором упомянутое определение последовательности упомянутых элементов упомянутых суперузлов согласно заранее заданному критерию упорядочения содержит по меньшей мере один из этапов, на которых

выстраивают в последовательность упомянутые элементы упомянутых суперузлов согласно по меньшей мере одной из строк и столбцов;

выстраивают в последовательность упомянутые элементы упомянутых суперузлов согласно по меньшей мере одной из строк и столбцов и отслеживают любое разделение и объединение строк/столбцов; и

перемежают горизонтальное и вертикальное расслоение упомянутых элементов упомянутых суперузлов и создают внутреннее дерево делений и определяют последовательность отображения для упомянутых элементов.

21. Способ по п.13, в котором упомянутое размещение содержит по меньшей мере один из этапов, на которых

интерпретируют рекомендации, присоединенные к упомянутым элементам упомянутых суперузлов;

создают разрывы линии согласно упомянутым рекомендациям, созданным путем упомянутого определения местоположения групп упомянутых компонентов, которые должны оставаться вместе, и упомянутого создания иерархии суперузлов на основании местоположения упомянутых компонентов и соотношений содержания упомянутых компонентов;

применяют по меньшей мере одну из регулировок ширины и высоты к упомянутым элементам упомянутых суперузлов;

изменяют размеры декоративных изображений.

22. Способ по п.18, в котором упомянутый анализ по меньшей мере одно из семантики, содержания, атрибутов, истории редактирования и геометрии упомянутых элементов упомянутых суперузлов содержит по меньшей мере одно из:

обнаружение частично упорядоченного набора кластеров, где упомянутые элементы упомянутых суперузлов располагаются ближе друг к другу по сравнению со стандартным разнесением между упомянутыми элементами в упомянутых суперузлах;

обнаружение частично упорядоченного набора семантических соотношений при наличии конкретных комбинаций упомянутых элементов упомянутых суперузлов данных типов, которые близки друг к другу;

обнаружение частично упорядоченного набора при наличии установленных схем между упомянутыми элементами упомянутых суперузлов;

обнаружение частично упорядоченного набора на основании по меньшей мере одного из существующих определений группирования, относящегося к редактированию, информации существующей привязки динамической компоновки и повторного использования шаблонов компонентов для разных наборов компонентов;

обнаружение частично упорядоченного набора упомянутых элементов упомянутых суперузлов на основании информации, собранной из предыдущих сеансов редактирования.

23. Система по п.7 и способ по п.19, в которых упомянутый по меньшей мере один из автоматически добавляемых компонентов и компонентов, относящихся к мобильным устройствам, включает по меньшей мере одно из виджетов, характерных для целевого устройства, навигационных меню, рекламы и популяризирующего материала.

24. Способ по п.13, в котором упомянутое изменение модификаций в упомянутой исходной компоновке и упомянутой целевой компоновке содержит

удаление страницы из упомянутой целевой компоновки, когда упомянутая страница удалена из упомянутой исходной компоновки;

добавление страницы в упомянутую целевую компоновку, когда упомянутая страница добавлена в упомянутую исходную компоновку;

удаление элемента из упомянутой целевой компоновки, когда упомянутый элемент удален из упомянутой исходной компоновки;

добавление элемента в упомянутую целевую компоновку, когда упомянутый элемент добавлен в упомянутую исходную компоновку;

модификацию элемента из упомянутой целевой компоновки, когда упомянутый элемент модифицирован в упомянутой исходной компоновке; и

обработку модификаций в упомянутой целевой компоновке, причем упомянутые модификации не

зависят от модификаций в упомянутой исходной компоновке.

25. Способ по п.24, в котором упомянутое добавление элемента содержит поиск по меньшей мере одного из ближайшего предшествующего элемента и родительского элемента, ближайших к элементу, добавленному к упомянутой исходной компоновке; и

вставку упомянутого добавленного элемента в упомянутую целевую компоновку согласно местоположению упомянутого по меньшей мере одного из ближайшего предшествующего элемента и родительского элемента.

26. Система по п.5, в которой упомянутый блок создания узла выполняет по меньшей мере одно из перемещения и объединения контейнеров согласно семантическому анализу.

27. Способ по п.17, в котором упомянутое создание иерархии суперузлов включает по меньшей мере одно из перемещения и объединения контейнеров согласно семантическому анализу.

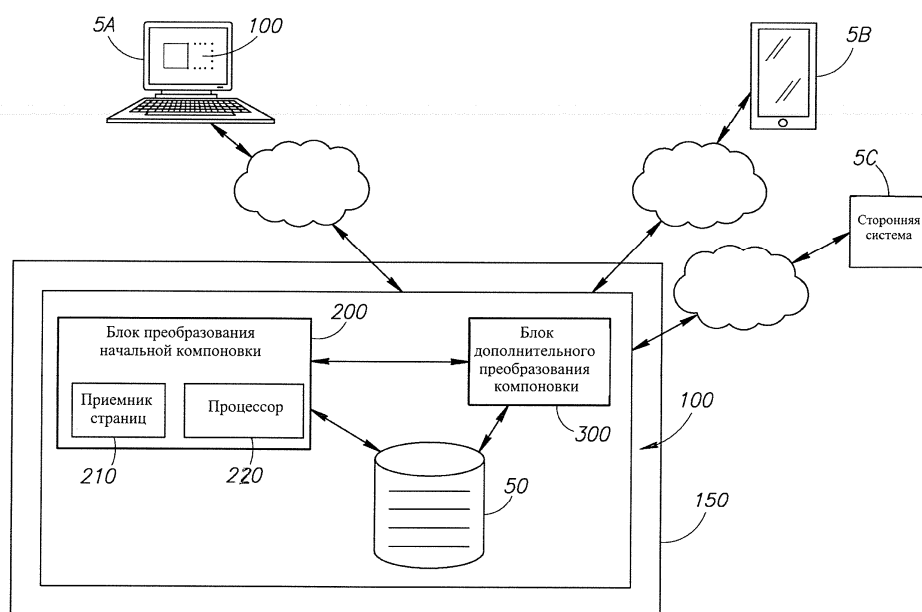
28. Система по п.12, в которой упомянутый блок поиска родителя/предшественника содержит блок кластеризации для кластеризации элементов согласно их близости к упомянутому предшествующему элементу и упомянутому родительскому элементу.

29. Способ по п.25, в котором упомянутый поиск включает кластеризацию компонентов согласно их близости к упомянутому предшествующему компоненту и упомянутому родительскому компоненту.

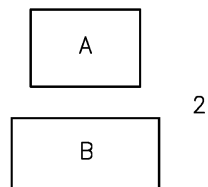
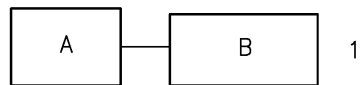
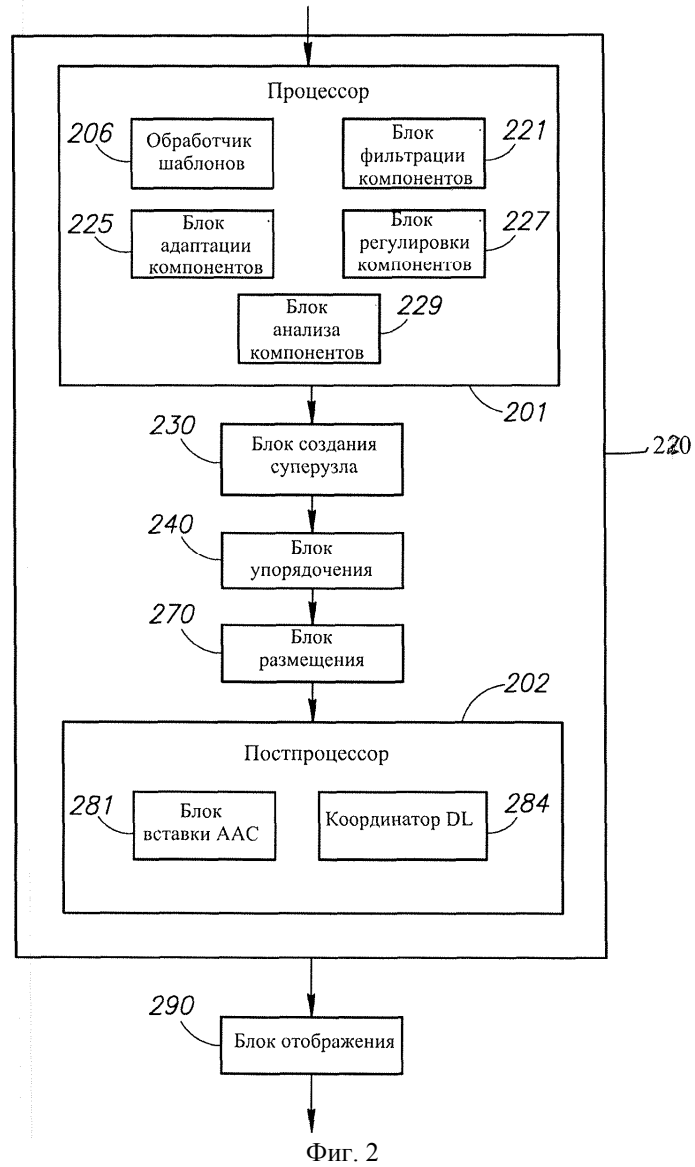
30. Способ по п.13, в котором упомянутый способ выполняют рекурсивно согласно упомянутой исходной компоновке.

31. Система по п.11, в которой упомянутый мобильный обработчик выполняет по меньшей мере одно из закрытия промежутков в заданной линии компонентов и удаления всей линии компонентов, когда упомянутый блок удаления компонента удаляет элемент из упомянутой целевой компоновки.

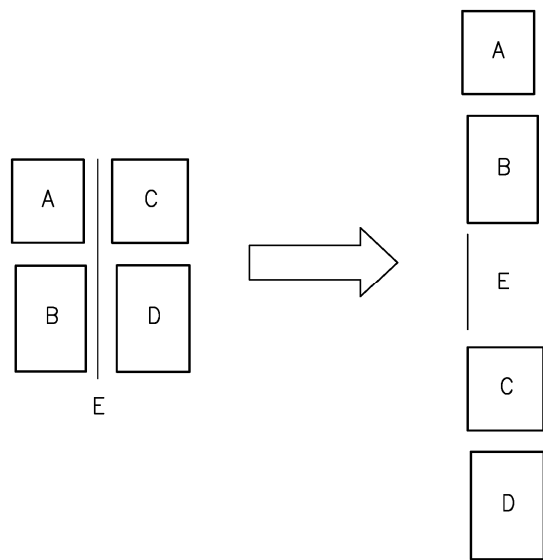
32. Способ по п.24, в котором упомянутая обработка модификаций в упомянутой целевой компоновке включает по меньшей мере одно из закрытия промежутков в заданной линии компонентов и удаления всей линии компонентов, когда упомянутый элемент удаляют из упомянутой исходной компоновки.



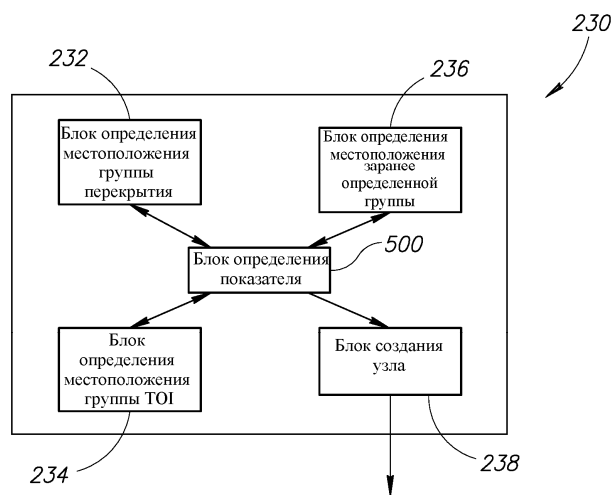
Фиг. 1



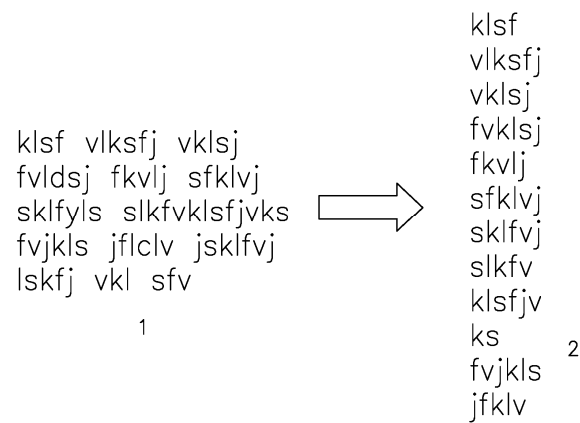
Фиг. 3



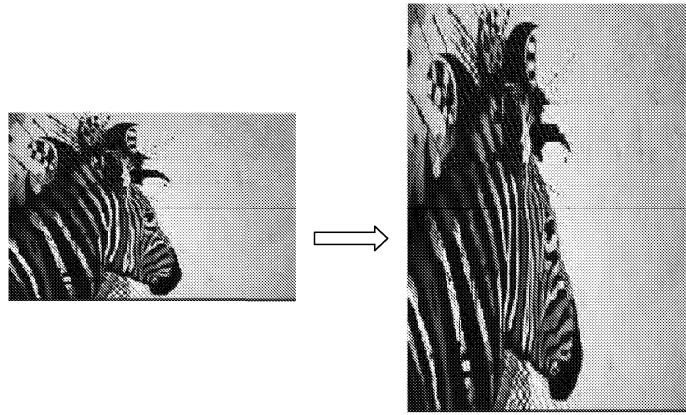
Фиг. 4



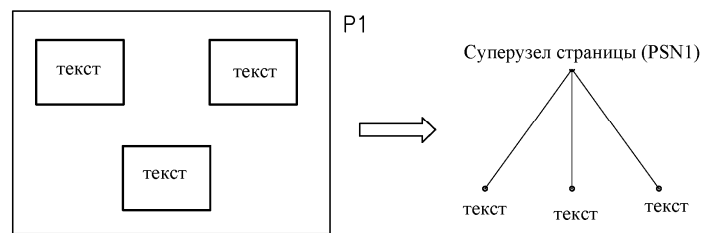
Фиг. 5



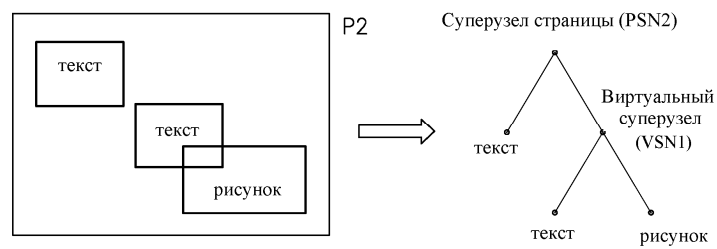
Фиг. 6



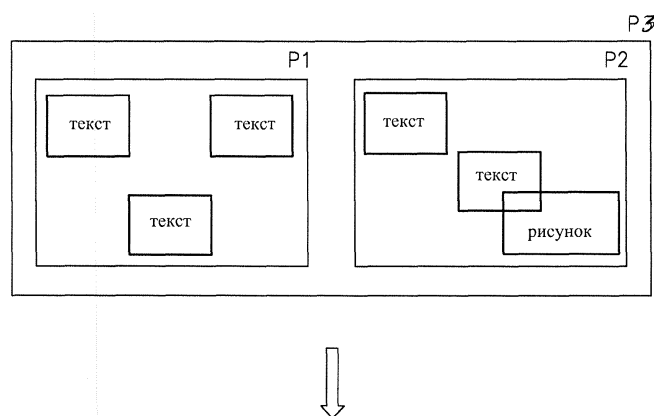
Фиг. 7



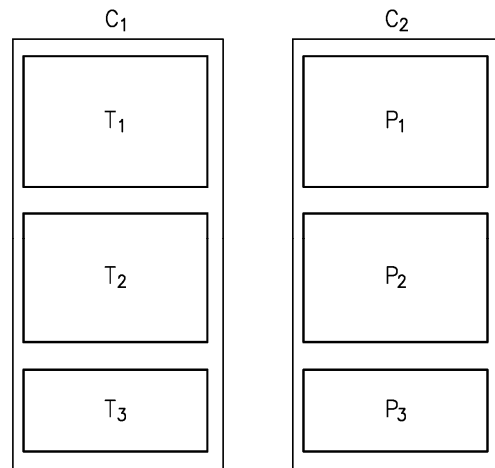
Фиг. 8А



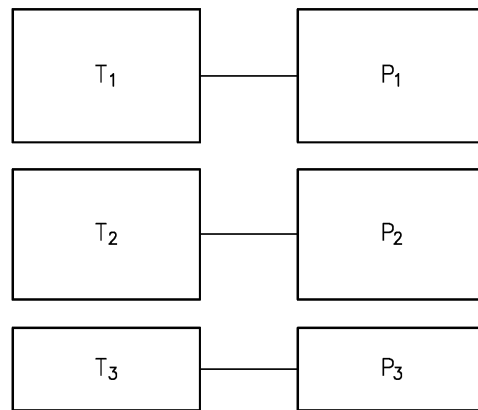
Фиг. 8В



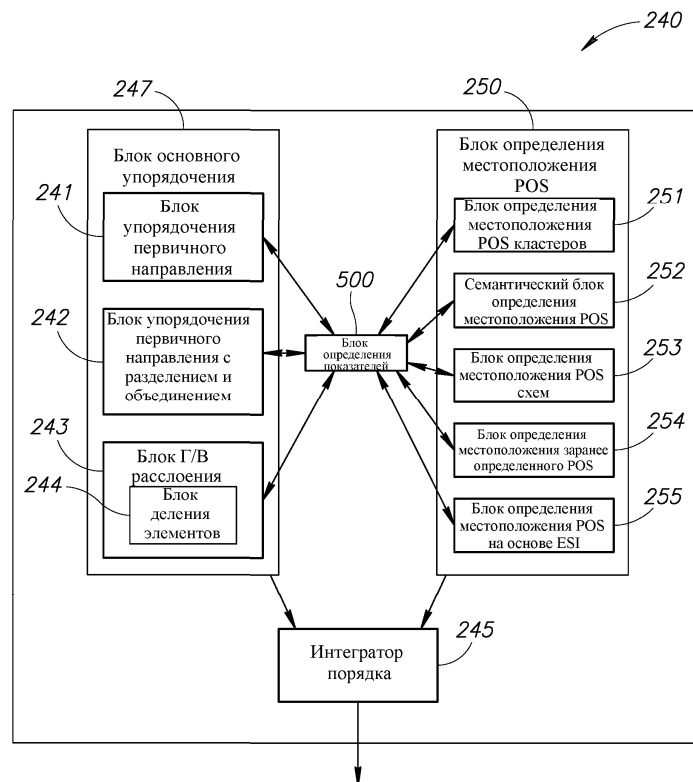
Фиг. 8С



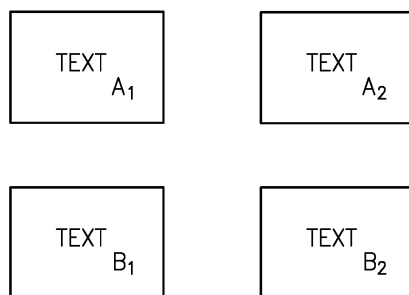
Фиг. 9



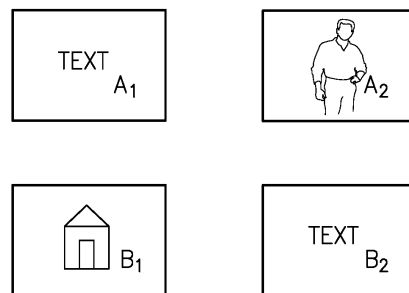
Фиг. 10



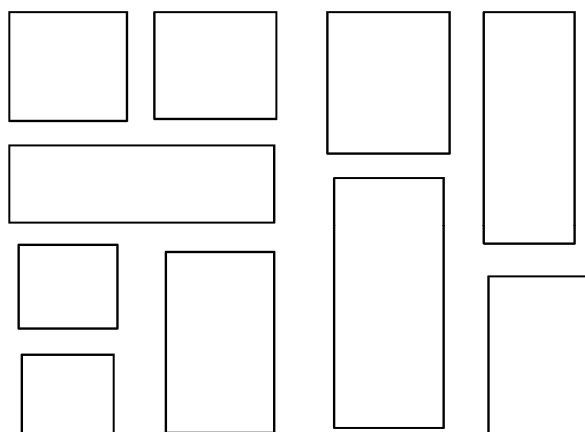
Фиг. 11



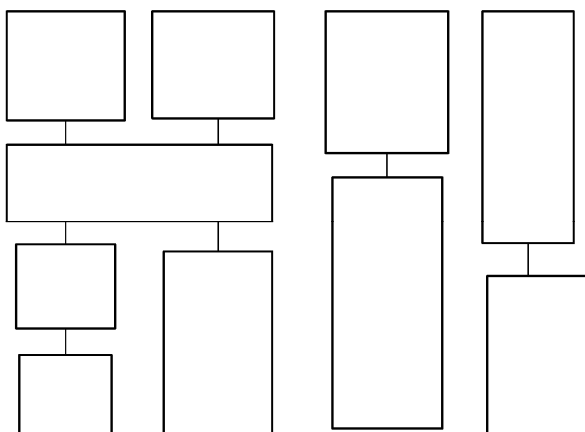
Фиг. 12



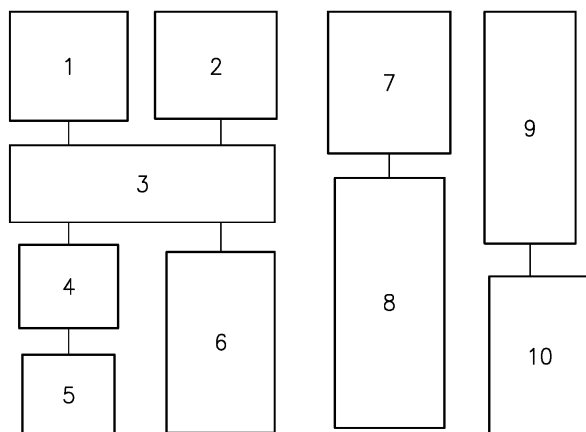
Фиг. 13



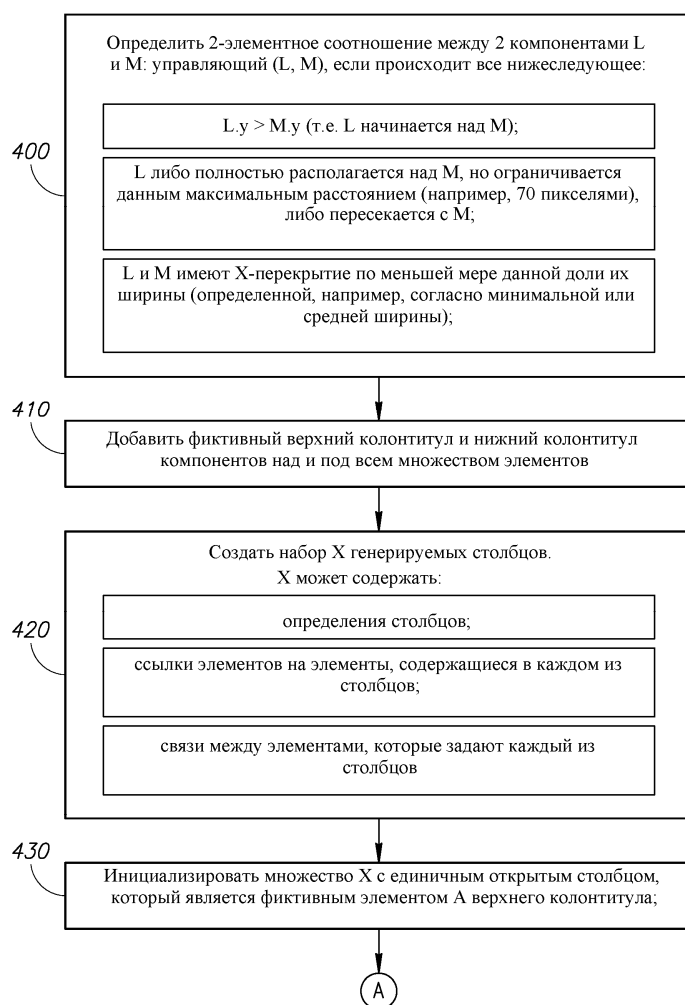
Фиг. 14А



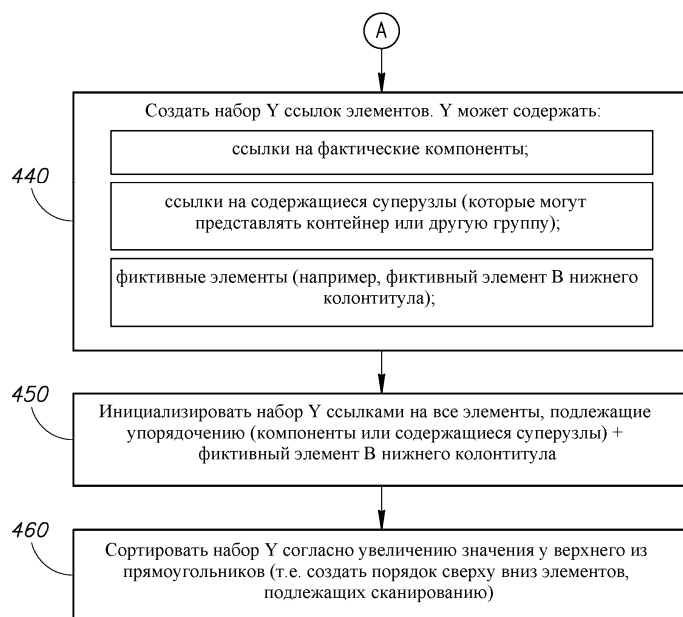
Фиг. 14В



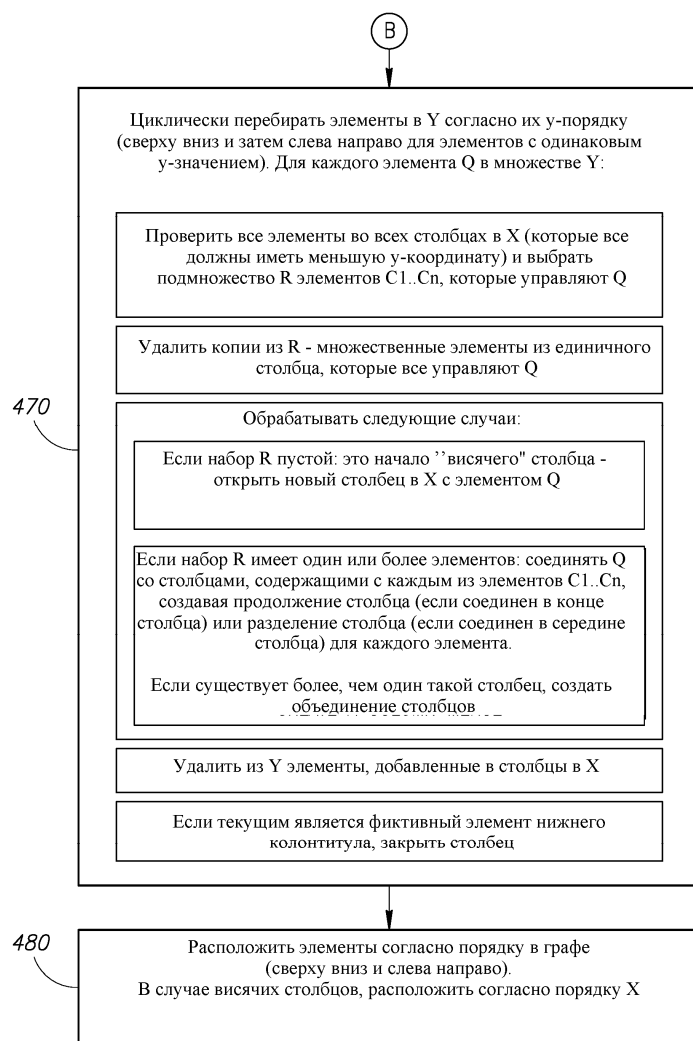
Фиг. 14С



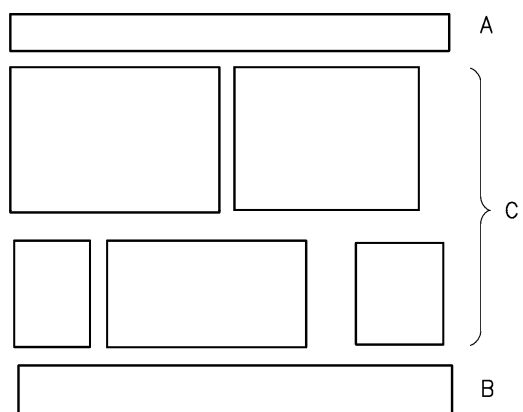
Фиг. 15А



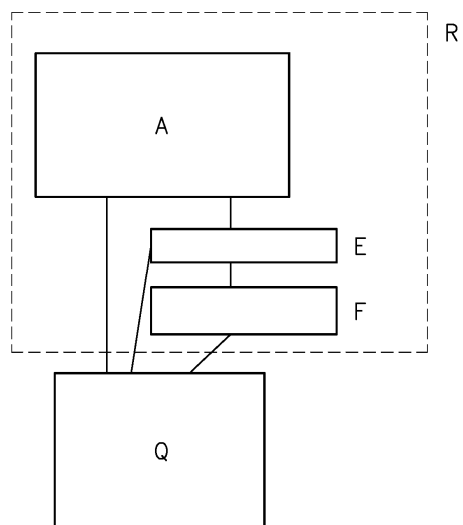
Фиг. 15В



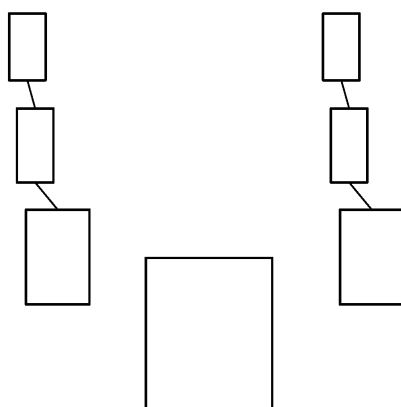
Фиг. 15С



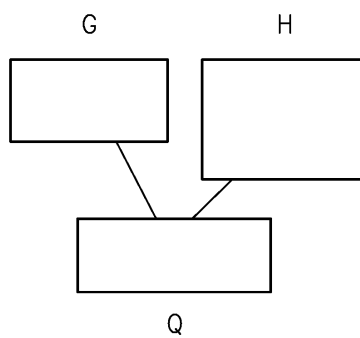
Фиг. 16А



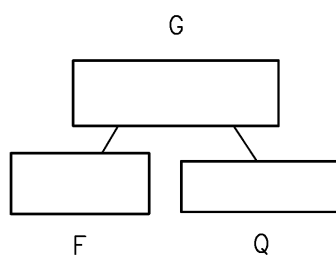
Фиг. 16В



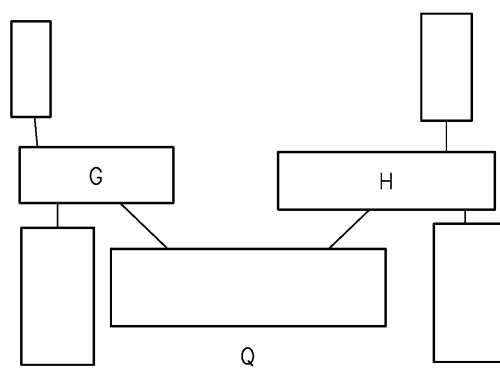
Q
Фиг. 16С



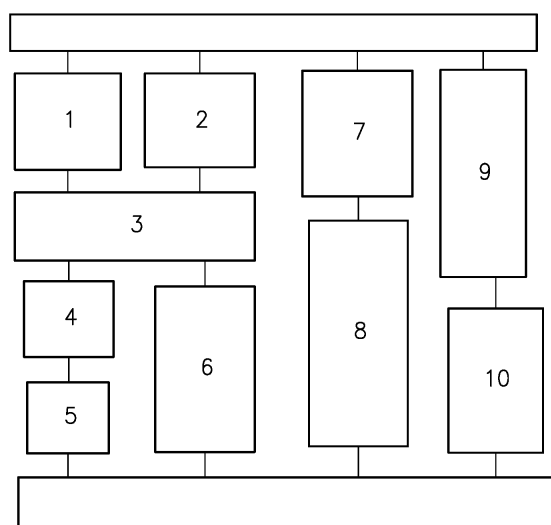
Фиг. 16D



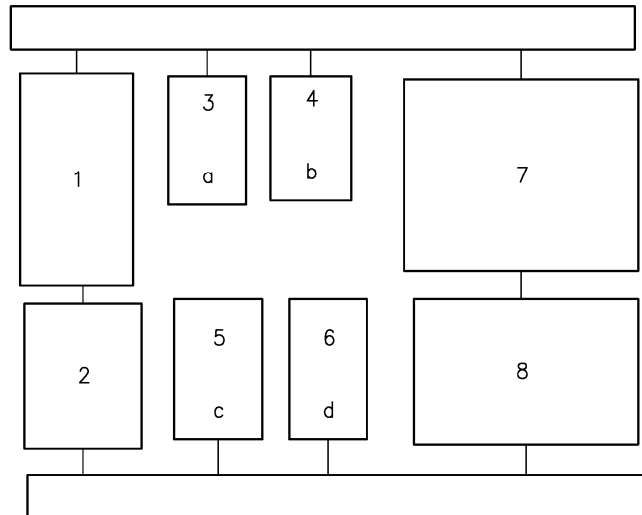
Фиг. 16E



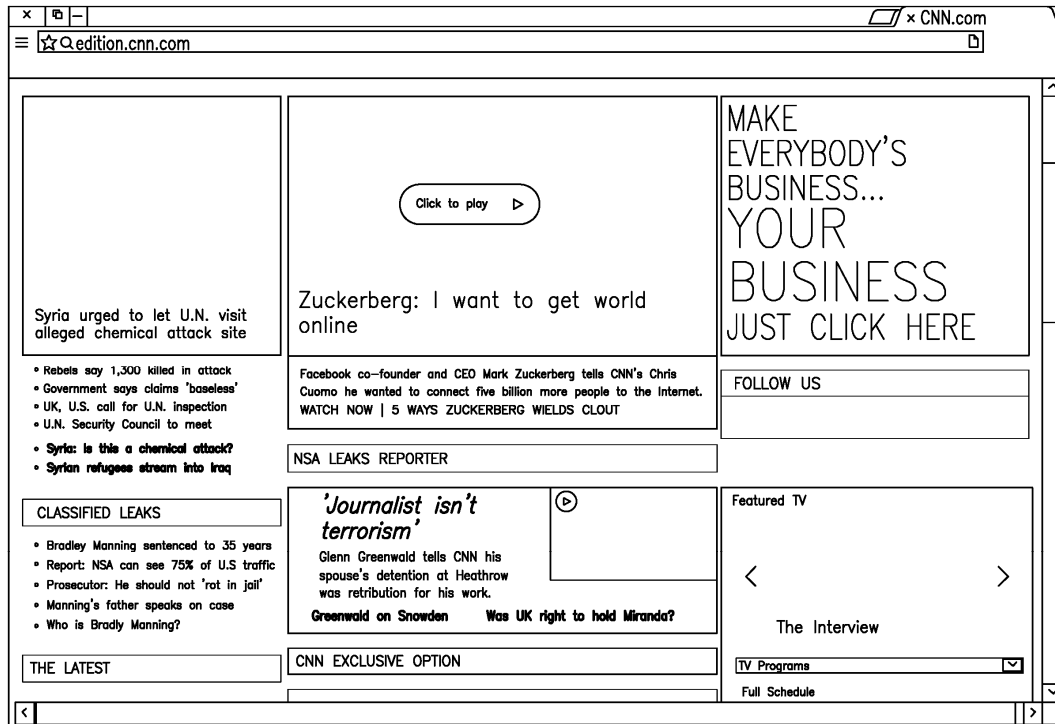
Фиг. 16F



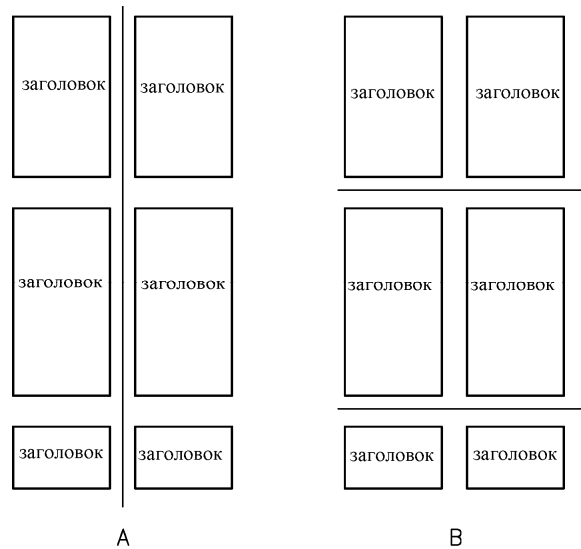
Фиг. 16G



Фиг. 16H



Фиг. 17



Фиг. 18

A

SERVICES

YOU HAVE AN IDEA ABOUT WHERE YOU WANT YOUR BUSINESS TO GO, WE LOOK FORWARD TO TRANSFORMING YOUR BUSINESS...

[MORE INFO](#)

B

PROJECTS

WE WORKED WITH PLANKTOP, A LOCAL FASHION RETAILER.... WHILE RETAINING THE LOCAL FEEL OF THEIR ONLINE STORE...

[MORE INFO](#)

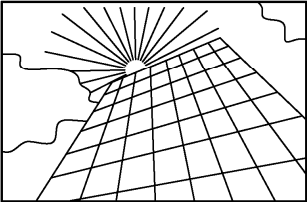
C

CLIENTS

WE WORKED WITH A VARIETY OF BUSINESSES VARYING CORPORATIONS... TO HELPING SMALL AND MEDIUM SIZE CLIENTS DISCOVER...

[MORE INFO](#)

THE WORLD CAN BE YOURS!

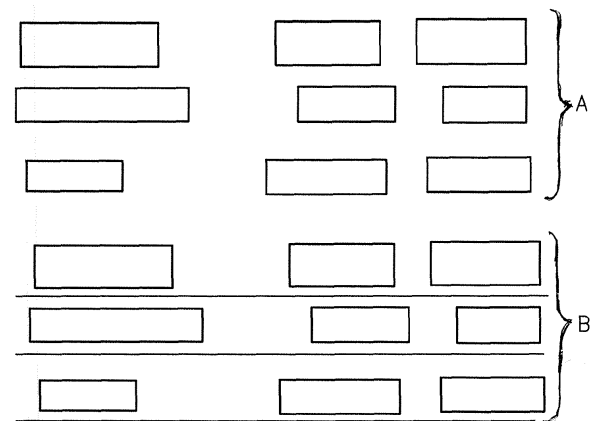


OUR GOAL AT XYZ CONSULTING IS TO PROVIDE BOOST TO YOUR SAMLL BUS...
OUR CONSULTING DESIGN SOLUTIONS...
CHECK OUT OUR CUSTOM SOLUTIONS &..

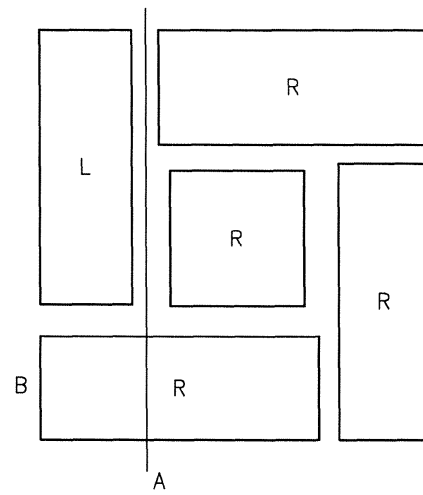
TESTIMONIAL

"I'M A TESTIMONIAL. CLICK ME TO EDIT AND ADD TEXT THAT SAY SOMETHING NICE ABOUT YOU AND YOUR SERVICES."

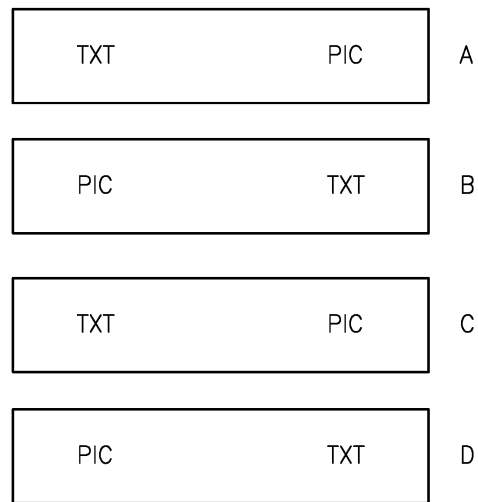
Фиг. 19



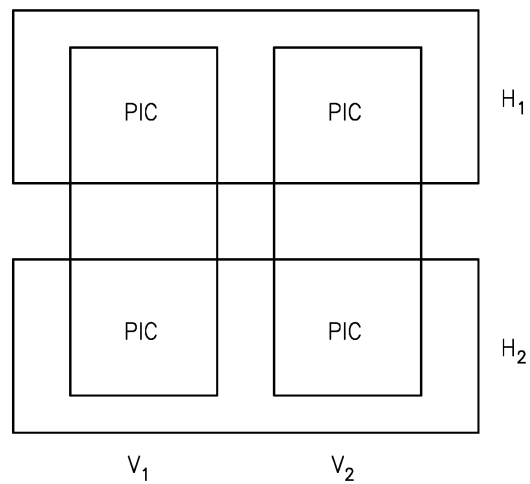
Фиг. 20



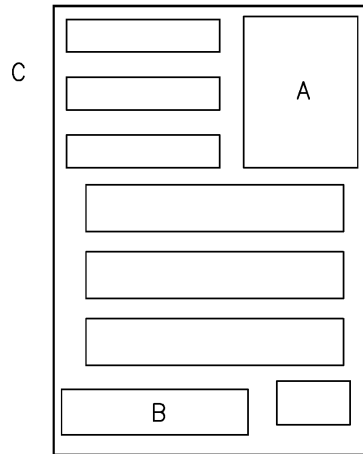
Фиг. 21



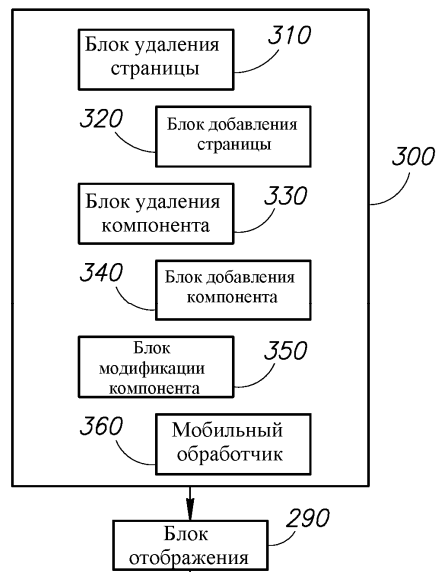
Фиг. 22



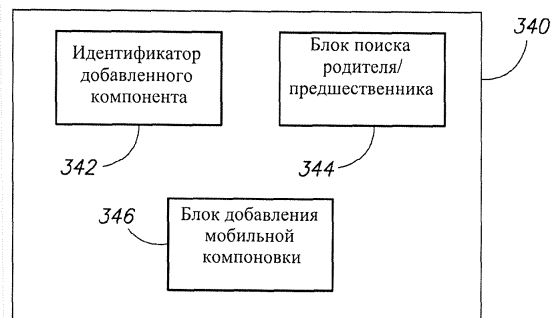
Фиг. 23



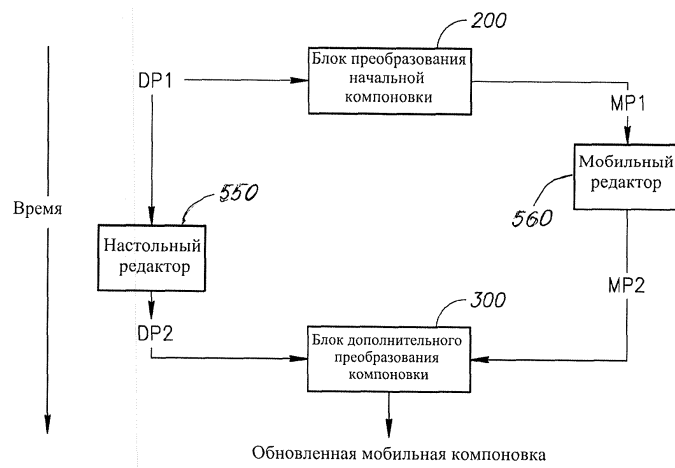
Фиг. 24



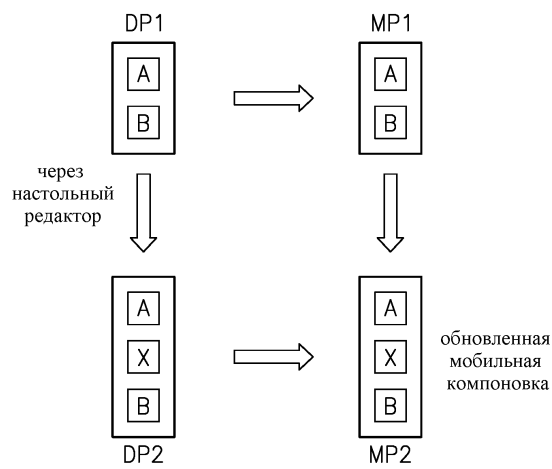
Фиг. 25



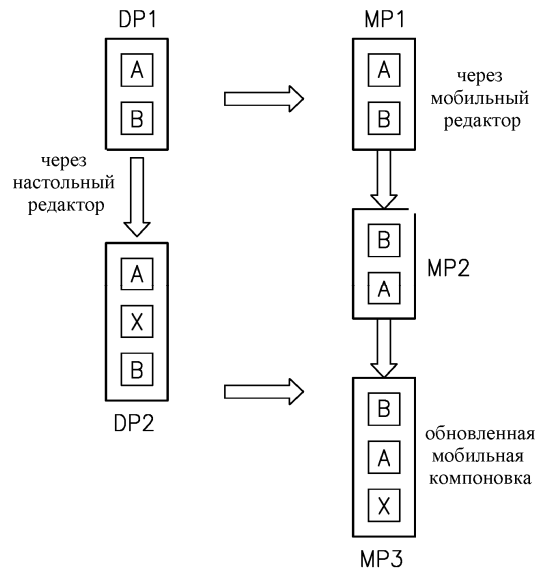
Фиг. 26



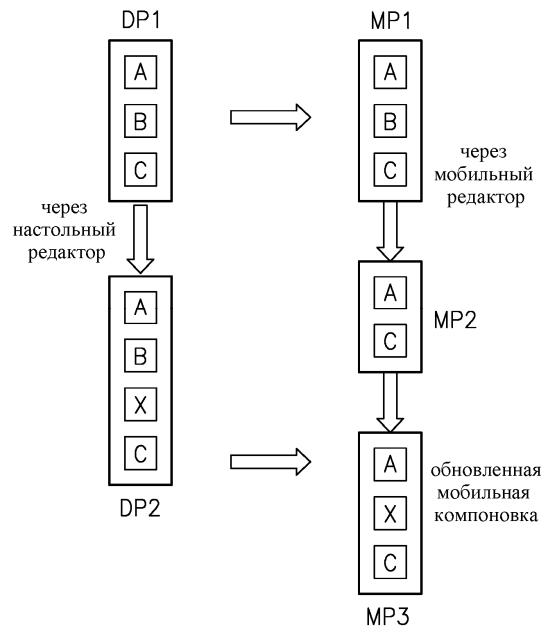
Фиг. 27



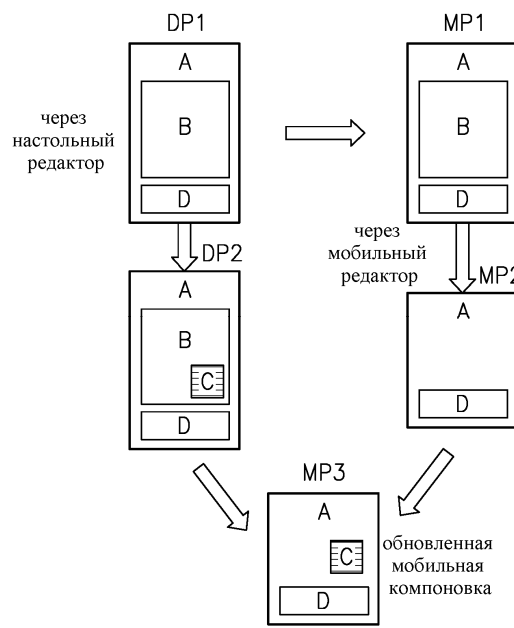
Фиг. 28А



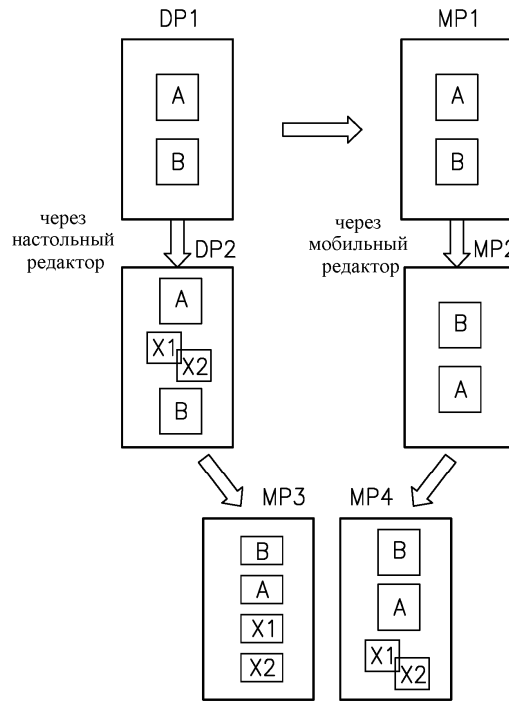
Фиг. 28В



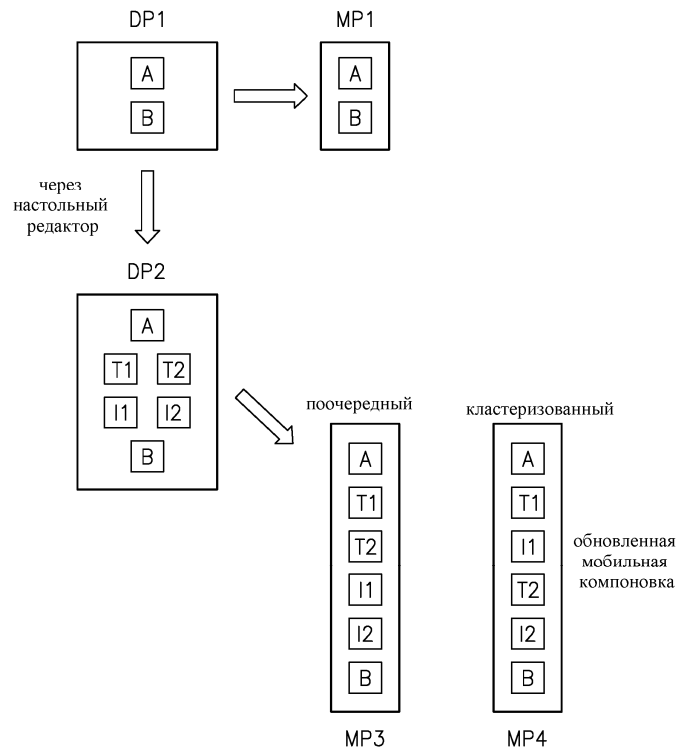
Фиг. 28С



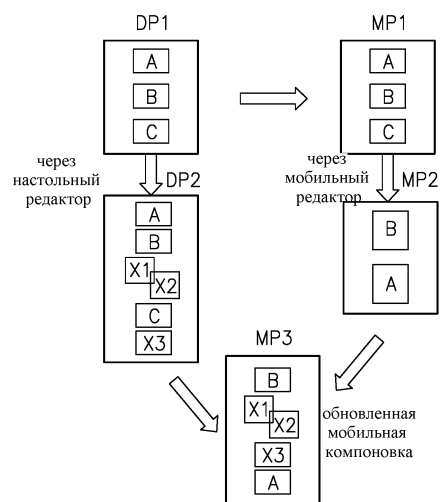
Фиг. 28D



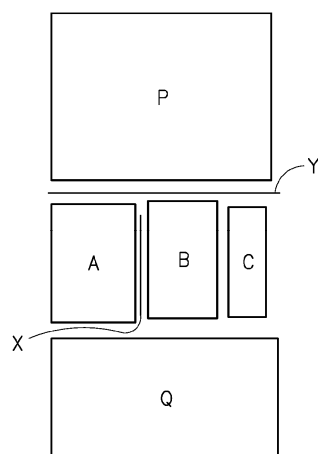
Фиг. 28Е



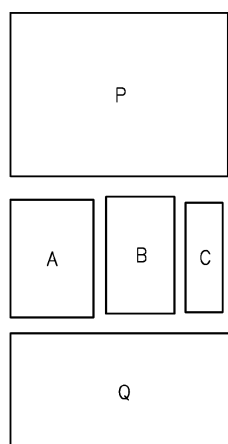
Фиг. 28F



Фиг. 28G



Фиг. 29



Фиг. 30

