



## Компоновка

### Введение в компоновку

Последнее обновление: 16.01.2016



Чтобы перейти уже непосредственно к созданию красивых интерфейсов и их компонентов, сначала необходимо познакомиться с компоновкой. Компоновка (layout) представляет собой процесс размещения элементов внутри контейнера. Возможно, вы обращали внимание, что одни программы и веб-сайты на разных экранах с разным разрешением выглядят по-разному: где-то лучше, где-то хуже. В большинстве своем такие программы используют жестко закодированные в коде размеры элементов управления. WPF уходит от такого подхода в пользу так называемого "резинового дизайна", где весь процесс позиционирования элементов осуществляется с помощью компоновки.

Благодаря компоновке мы можем удобным нам образом настроить элементы интерфейса, позиционировать их определенным образом. Например, элементы компоновки в WPF позволяют при ресайзе - сжатии или растяжении масштабировать элементы, что очень удобно, а визуально не создает всяких шероховатостей типа незаполненных пустот на форме.

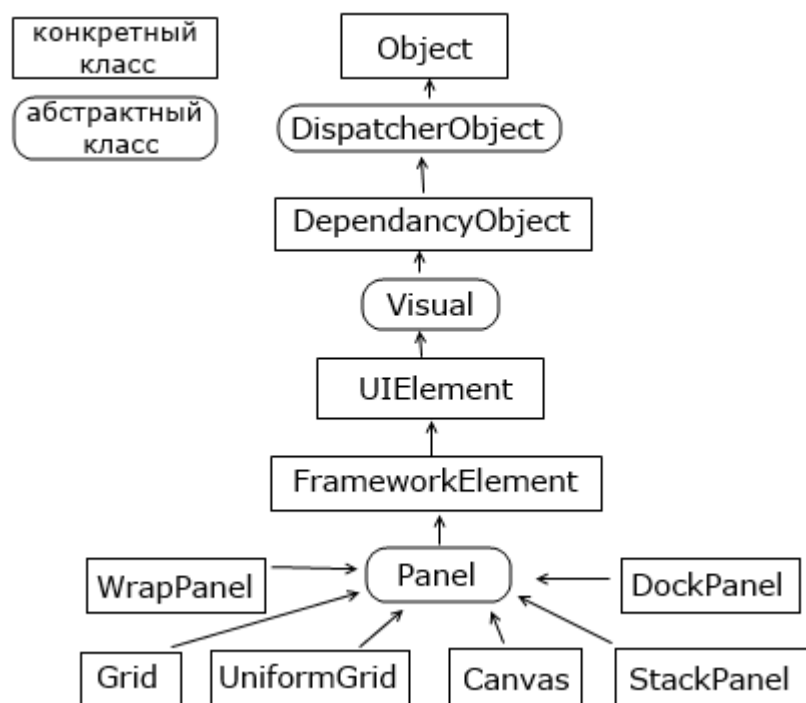
В WPF компоновка осуществляется при помощи специальных контейнеров. Фреймворк предоставляет нам следующие контейнеры: **Grid**, **UniformGrid**, **StackPanel**, **WrapPanel**, **DockPanel** и **Canvas**.

Различные контейнеры могут содержать внутри себя другие контейнеры. Кроме данных контейнеров существует еще ряд элементов, такие как **TabPanel**, которые могут включать другие элементы и даже контейнеры компоновки, однако на саму компоновку не столь влияют в отличие от выше перечисленных. Кроме того, если нам не хватает стандартных контейнеров, мы можем определить свои с нужной нам функциональностью.

Контейнеры компоновки позволяют эффективно распределить доступное пространство между элементами, найти для него наиболее

предпочтительные размеры.

Все выше перечисленные контейнеры компоновки наследуются от абстрактного класса `Panel`, а само дерево наследования можно представить следующим образом:



Наглядным примером применения компоновки может служить сделанное в первой главе приложение калькулятора, где мы не указываем явным образом размеры кнопок, а полагаемся на контейнер `Grid`, который и осуществляет всю расстановку и масштабирование дочерних элементов.

В WPF при компоновке и расположении элементов внутри окна нам надо придерживаться следующих принципов:

- Нежелательно указывать явные размеры элементов (за исключением минимальных и максимальных размеров). Размеры должны определяться контейнерами.
- Нежелательно указывать явную позицию и координаты элементов внутри окна. Позиционирование элементов всецело должно быть прерогативой контейнеров. И контейнер сам должен определять, как элемент будет располагаться. Если нам надо создать сложную систему компоновки, то мы можем вкладывать один контейнер в другой, чтобы добиться максимально удобного

расположения элементов управления.

## Процесс компоновки

Процесс компоновки проходит два этапа: измерение (measure) и расстановка (arrange). На этапе измерения контейнер производит измерение предпочтительного для дочерних элементов места. Однако не всегда контейнер имеет достаточно места, чтобы расставить все элементы по их предпочтительным размерам, поэтому их размеры приходится усекаать. Затем происходит этап непосредственной расстановки дочерних элементов внутри контейнера.

Теперь рассмотрим контейнеры компоновки подробнее.

[Назад](#) [Содержание](#) [Вперед](#)



ТАКЖЕ НА METANIT.COM

### Добавление данных в MySQLi

2 месяца назад • 1 коммент...

Добавление данных в БД MySQL в языке PHP с помощью библиотеки ...

### Первое графическое приложение на ...

2 месяца назад • 1 коммент...

Первое графическое приложение на Rust для Windows 10, проект Rust ...

### Google представил новую ОС - Fuchsia

3 месяца назад • 2 коммент...

Google представил новую ОС - Fuchsia для устройства Google ...

### Удаление данных в MySQLi

2 месяца назад • 1 коммент...

Удаление данных в БД MySQL в языке PHP с помощью библиотеки ...

### Оператор let. Привязка знач

23 дня назад • 1 коммент...

Оператор let. Привязка значений в языке программирования Rust

1 Комментарий metanit.com  Политика конфиденциальности Disqus

 1 Войти

 Рекомендовать 5  Твитнуть  Поделиться

Лучшее в начале



Присоединиться к обсуждению...

ВОЙТИ С ПОМОЩЬЮ

ИЛИ ЧЕРЕЗ DISQUS ?



Имя



Ilya Velsh • 4 years ago

DependAncyObject na kartinke

^ | v • Ответить • Поделиться ›

 Подписаться  Добавить Disqus на свой сайтДобавить DisqusДобавить  Do Not Sell My Data

#### Помощь сайту WebMoney

- R378451176208
- Z280152397659

#### ЮMoney/Яндекс-Деньги

- 410011174743222

#### PayPal

- metanit22@mail.ru

#### Перевод на карту

- Номер карты: 4048415020898850

[Вконтакте](#) | [Twitter](#) | [Канал сайта на youtube](#) | [Помощь сайту](#)

Контакты для связи: metanit22@mail.ru