|  |  |  |  |
| --- | --- | --- | --- |
| Segment ID | Segment status | Source segment | Target segment |
| 1ae993321-be11-418c-a979-489d5ef5f72c | Translated (0%) | The Animation Framework | Qt Core 5.7 | The Animation Framework | Qt Core 5.7 |
| 23645739b-182e-4915-9216-fbde1dc1571e | Translated (100%) | Qt 5.7 | Qt 5.7 |
| 36a6a39d5-36b3-479b-934b-5f741d9c2047 | Translated (CM) | Qt Core | Qt Core |
| 42b822dfc-393e-42a8-b1f9-b05f140ce0d1 | Translated (0%) | The Animation Framework | The Animation Framework |
| 55418f10f-ef92-4f9f-99d1-f8e3a3ac130a | Translated (100%) | Qt 5.7.1 Reference Documentation | Справочная документация Qt 5.7.1 |
| 69bfb3cd0-3462-4f2d-aea7-d8a4d3196e1a | Translated (99%) | Contents | Содержание |
| 788e5a582-1472-4bc1-9eda-9b78d657805e | Translated (0%) | The Animation Architecture | Архитектура Анимации |
| 83fac1934-afd2-4a41-ba49-7fa01041ee04 | Translated (70%) | Classes in the Animation Framework | Классы в Animation Framework |
| 9ef46e7b7-35a9-4ef3-8b04-140a8acc4676 | Translated (94%) | Animating Qt Properties | Анимация Свойств Qt |
| 10d653abdc-5e63-4a6f-a4ae-d73190ec0184 | Translated (0%) | Animations and the Graphics View Framework | Анимация и Graphics View Framework |
| 1163d0c3d7-feb5-4a61-91fc-d773ee163f5c | Translated (0%) | Easing Curves | Переходные характеристики |
| 12ec3e28ad-2a15-4a90-a0f9-f4a73b2339b7 | Translated (0%) | Putting Animations Together | Объединяем анимацию |
| 13039a6047-0c3c-45bf-8b41-270887f6134d | Translated (0%) | Animations and States | Анимации и состояния |
| 14e2394b8b-8986-43cd-89a8-b802b89ba9dd | Translated (100%) | The Animation Framework | The Animation Framework |
| 1556dfbd19-c9ac-4df9-b7b5-a66493583ec9 | Translated (0%) | The animation framework aims to provide an easy way for creating animated and smooth GUIs. | Анимационный фреймворк призван обеспечить простой способ создания анимированных и плавных графических интерфейсов. |
| 1656dfbd19-c9ac-4df9-b7b5-a66493583ec9 | Translated (0%) | By animating Qt properties, the framework provides great freedom for animating widgets and other <41>QObject</41>s. | Анимируя свойства Qt, платформа предоставляет большую свободу для анимации виджетов и других объектов <41>QObject</41>. |
| 1756dfbd19-c9ac-4df9-b7b5-a66493583ec9 | Translated (0%) | The framework can also be used with the Graphics View framework. | Фреймворк также можно использовать с фреймворком Graphics View. |
| 1856dfbd19-c9ac-4df9-b7b5-a66493583ec9 | Translated (0%) | Many of the concepts available in the animation framework are also available in <44>Qt Quick</44>, where it offers a declarative way of defining animations. | Многие из концепций, доступных в рамках анимации, также доступны в <44>Qt Quick</44>, где он предлагает декларативный способ определения анимации. |
| 1956dfbd19-c9ac-4df9-b7b5-a66493583ec9 | Translated (0%) | Much of the knowledge acquired about the animation framework can be applied to <47>Qt Quick</47>. | Большая часть знаний, полученных о структуре анимации, может быть применена к <47>Qt Quick</47>. |
| 203dce6b34-941a-4a73-95ab-df1a0ddd8449 | Translated (0%) | In this overview, we explain the basics of its architecture. | В этом обзоре мы объясняем основы его архитектуры. |
| 213dce6b34-941a-4a73-95ab-df1a0ddd8449 | Translated (0%) | We also show examples of the most common techniques that the framework allows for animating <50>QObject</50>s and graphics items. | Мы также показываем примеры наиболее распространенных методов, которые платформа позволяет анимировать объекты <50>QObject</50> и графические элементы. |
| 227bfa8dbd-1edb-42ef-8c1f-2adea4590f2b | Translated (100%) | The Animation Architecture | Архитектура Анимации |
| 23464384e3-3699-4bbd-b586-fd67c96bedb2 | Translated (0%) | We will in this section take a high-level look at the animation framework's architecture and how it is used to animate Qt properties. | В этом разделе мы целиком рассмотрим архитектуру фреймворка анимации и то, как он используется для анимации свойств Qt. |
| 24464384e3-3699-4bbd-b586-fd67c96bedb2 | Translated (0%) | The following diagram shows the most important classes in the animation framework. | На следующей диаграмме показаны наиболее важные классы в Animation framework. |
| 253df2be80-f5a2-41b6-a5a0-b541ac0e4543 | Translated (0%) | The animation framework foundation consists of the base class <57>QAbstractAnimation</57>, and its two subclasses <60>QVariantAnimation</60> and <63>QAnimationGroup</63>. | Animation framework foundation состоит из базового класса <57>QAbstractAnimation</57> и двух его подклассов <60>QVariantAnimation</60> и <63>QAnimationGroup</63>. |
| 263df2be80-f5a2-41b6-a5a0-b541ac0e4543 | Translated (0%) | <66>QAbstractAnimation</66> is the ancestor of all animations. | <66>QAbstractAnimation</66> является предком всех анимаций. |
| 273df2be80-f5a2-41b6-a5a0-b541ac0e4543 | Translated (0%) | It represents basic properties that are common for all animations in the framework; notably, the ability to start, stop, and pause an animation. | Он представляет основные свойства, общие для всех анимаций во фреймворке; в частности, возможность запуска, остановки и приостановки анимации. |
| 283df2be80-f5a2-41b6-a5a0-b541ac0e4543 | Translated (0%) | It is also receives the time change notifications. | Он также получает уведомления об изменении времени. |
| 2919821382-cc3e-424b-b2f7-804b4432daa5 | Translated (0%) | The animation framework further provides the <69>QPropertyAnimation</69> class, which inherits <72>QVariantAnimation</72> and performs animation of a Qt property, which is part of Qt's <75>meta-object system</75>. | Animation framework также предоставляет класс <69>QPropertyAnimation</69>, который наследует <72>QVariantAnimation</72>и выполняет анимацию Qt property, которое является частью <75>метаобъектной системы</75> Qt. |
| 3019821382-cc3e-424b-b2f7-804b4432daa5 | Translated (0%) | The class performs an interpolation over the property using an easing curve. | Класс выполняет интерполяцию по свойству, используя переходную характеристику. |
| 3119821382-cc3e-424b-b2f7-804b4432daa5 | Translated (0%) | So when you want to animate a value, you can declare it as a property and make your class a <78>QObject</78>. | Поэтому, когда вы хотите анимировать значение, вы можете объявить его как свойство и сделать свой класс <78>QObject</78>. |
| 3219821382-cc3e-424b-b2f7-804b4432daa5 | Translated (0%) | Note that this gives us great freedom in animating already existing widgets and other <81>QObject</81>s. | Обратите внимание, что это дает нам большую свободу в анимации уже существующих виджетов и других объектов <81>QObject</81>. |
| 33650d4cb8-66d0-40ff-8e2e-cb211f69b744 | Translated (0%) | Complex animations can be constructed by building a tree structure of <84>QAbstractAnimation</84>s. | Сложные анимации могут быть построены путем построения древовидной структуры <84>QAbstractAnimation</84>. |
| 34650d4cb8-66d0-40ff-8e2e-cb211f69b744 | Translated (0%) | The tree is built by using <87>QAnimationGroup</87>s, which function as containers for other animations. | Дерево построено с помощью <87>QAnimationGroup</87>, которые функционируют как контейнеры для других анимаций. |
| 35650d4cb8-66d0-40ff-8e2e-cb211f69b744 | Translated (0%) | Note also that the groups are subclasses of <90>QAbstractAnimation</90>, so groups can themselves contain other groups. | Обратите также внимание, что группы являются подклассами <90>QAbstractAnimation</90>, поэтому сами группы могут содержать другие группы. |
| 365af53704-01c9-4492-96fa-5543aa6943f8 | Translated (0%) | The animation framework can be used on its own, but is also designed to be part of the state machine framework (See the <93>state machine framework</93> for an introduction to the Qt state machine). | Animation framework может использоваться сам по себе, но также предназначен для того, чтобы быть частью фреймворка state machine (см. <93>фреймворк state machine</93> для введения в Qt state machine). |
| 375af53704-01c9-4492-96fa-5543aa6943f8 | Translated (0%) | The state machine provides a special state that can play an animation. | State machine предоставляет специальную структуру, которая может воспроизводить анимацию. |
| 385af53704-01c9-4492-96fa-5543aa6943f8 | Translated (0%) | A <96>QState</96> can also set properties when the state is entered or exited, and this special animation state will interpolate between these values when given a <99>QPropertyAnimation</99>. | <96>QState</96> также может задавать свойства при входе или выходе из структуры, и эта специальная структура анимации будет интерполироваться между этими значениями при задании <99>QPropertyAnimation</99>. |
| 395af53704-01c9-4492-96fa-5543aa6943f8 | Translated (0%) | We will look more closely at this later. | Мы рассмотрим это более подробно позже. |
| 404d494dce-a92a-4b2d-8f5e-536e1f6915a2 | Translated (0%) | Behind the scenes, the animations are controlled by a global timer, which sends <102>updates</102> to all animations that are playing. | За кулисами анимация управляется глобальным таймером, который отправляет <102>обновления</102> всем анимациям, которые исполняются. |
| 4189897fa8-31c8-49f8-bbf3-2cfb387d6ae9 | Translated (0%) | For detailed descriptions of the classes' function and roles in the framework, please look up their class descriptions. | Подробные описания функций и ролей классов во фреймворке см. в описаниях классов. |
| 427c880af4-9c6a-4d69-ac6e-57ddc4fb285b | Translated (100%) | Classes in the Animation Framework | Классы в Animation Framework |
| 4308e93d9b-7f40-4343-9e0c-9228bfdd51fb | Translated (0%) | These classes provide a framework for creating both simple and complex animations. | Эти классы обеспечивают основу для создания простых и сложных анимаций. |
| 447b3407cb-2282-484b-938d-02f397669982 | Translated (100%) | QAbstractAnimation | QAbstractAnimation |
| 45755ea72f-d299-43ff-95eb-65d82c6175d6 | Translated (100%) | The base of all animations | База всех анимаций |
| 46a47dde1e-e025-4565-8950-b61aca4ea6be | Translated (100%) | QAnimationGroup | QAnimationGroup |
| 475c34c1a2-3d14-489d-8836-f63f8fe288ea | Translated (100%) | Abstract base class for groups of animations | Абстрактный базовый класс для групп анимаций |
| 480161641c-53d0-4387-8fd6-5423a1043655 | Translated (100%) | QParallelAnimationGroup | QParallelAnimationGroup |
| 49261b3fd3-d92e-4793-beab-eead1f4b6c1b | Translated (CM) | Parallel group of animations | Параллельная группа анимаций |
| 5050be4e91-9158-44f0-a9d5-1f25297a5d82 | Translated (100%) | QPauseAnimation | QPauseAnimation |
| 51b28e5f8b-266a-424a-b954-a7e7ec69a0f3 | Translated (CM) | Pause for QSequentialAnimationGroup | Пауза для QSequentialAnimationGroup |
| 527f5dd570-9a6b-40fa-b76b-d6a939bbc0da | Translated (100%) | QPropertyAnimation | QPropertyAnimation |
| 53a914aa1c-81da-4ac1-b192-cbeab8f6399d | Translated (100%) | Animates Qt properties | Анимирует свойства в Qt |
| 547d4013e9-2a8d-448a-9e29-bdce57f88037 | Translated (100%) | QSequentialAnimationGroup | QSequentialAnimationGroup |
| 5567b151f3-9235-4e9b-8d17-7a1502391a8f | Translated (100%) | Sequential group of animations | Последовательная группа анимаций |
| 56678f919a-6c68-4dfd-a1bd-0349c133b577 | Translated (100%) | QVariantAnimation | QVariantAnimation |
| 5785f7e26c-533e-4061-9ce0-cb274c18a98f | Translated (100%) | Base class for animations | Базовый класс для анимации |
| 586e579a81-e1e0-4008-9993-e97a969a1f0b | Translated (100%) | QEasingCurve | QEasingCurve |
| 593911524c-8f5b-4680-a87e-dad61994d284 | Translated (100%) | Easing curves for controlling animation | Переходные характеристики для управления анимацией |
| 60c413f504-f720-49cf-8ee1-e32ae2caf98a | Translated (100%) | QTimeLine | QTimeLine |
| 61f8934f2f-0fb0-4474-a94e-06d2079862ae | Translated (100%) | Timeline for controlling animations | Временная шкала (timeline) для управления анимацией |
| 62d4cb5dd2-f4b2-4100-87c4-d2bf3a41491f | Translated (100%) | Animating Qt Properties | Анимация Свойств Qt |
| 63e85a361a-9093-40d3-bb6b-7f62630dd5a3 | Translated (0%) | As mentioned in the previous section, the <138>QPropertyAnimation</138> class can interpolate over Qt properties. | Как упоминалось в предыдущем разделе, класс <138>QPropertyAnimation</138> может интерполироваться по свойствам Qt. |
| 64e85a361a-9093-40d3-bb6b-7f62630dd5a3 | Translated (0%) | It is often this class that should be used for animation of values; in fact, its superclass, <141>QVariantAnimation</141>, has an empty implementation of updateCurrentValue(), and does not change any value unless we change it ourselves on the <144>valueChanged signal</144>. | Зачатую этот класс используется для анимации значений; фактически, его суперкласс, <141>QVariantAnimation</141>, имеет пустую реализацию updateCurrentValue() и не изменяет никакого значения, если мы не изменим его сами по <144>сигналу valueChanged</144>. |
| 65b9468ca6-7637-4f87-aef0-f3305982086d | Translated (0%) | A major reason we chose to animate Qt properties is that it presents us with freedom to animate already existing classes in the Qt API. | Основная причина, по которой мы решили анимировать свойства Qt, заключается в том, что это дает нам свободу анимировать уже существующие классы в Qt API. |
| 66b9468ca6-7637-4f87-aef0-f3305982086d | Translated (0%) | Notably, the <147>QWidget</147> class (which we can also embed in a <150>QGraphicsView</150>) has properties for its bounds, colors, etc. Let's look at a small example: | Примечательно, что класс <147>QWidget</147> (который мы также можем встроить в <150>QGraphicsView</150>) имеет свойства для своих границ, цветов и т. д. Давайте рассмотрим небольшой пример: |
| 67dd91aef9-7291-48b4-a1a3-32dd2ff8d9be | Translated (0%) | <153>QPushButton</153> button(<156>"Animated Button"</156>); button<159>.</159>show(); <162>QPropertyAnimation</162> animation(<165>&</165>button<168>,</168> <171>"geometry"</171>); animation<174>.</174>setDuration(<177>10000</177>); animation<180>.</180>setStartValue(<183>QRect</183>(<186>0</186><189>,</189> <192>0</192><195>,</195> <198>100</198><201>,</201> <204>30</204>)); animation<207>.</207>setEndValue(<210>QRect</210>(<213>250</213><216>,</216> <219>250</219><222>,</222> <225>100</225><228>,</228> <231>30</231>)); animation<234>.</234>start(); | <153>QPushButton</153> button(<156>"Animated Button"</156>); button<159>.</159>show(); <162>QPropertyAnimation</162> animation(<165>&</165>button<168>,</168> <171>"geometry"</171>); animation<174>.</174>setDuration(<177>10000</177>); animation<180>.</180>setStartValue(<183>QRect</183>(<186>0</186><189>,</189> <192>0</192><195>,</195> <198>100</198><201>,</201> <204>30</204>)); animation<207>.</207>setEndValue(<210>QRect</210>(<213>250</213><216>,</216> <219>250</219><222>,</222> <225>100</225><228>,</228> <231>30</231>)); animation<234>.</234>start(); |
| 68fc58a3bd-63c4-4636-a7f3-2e653e5fc8d0 | Translated (0%) | This code will move <button/> from the top left corner of the screen to the position (250, 250) in 10 seconds (10000 milliseconds). | Этот код переместит <button/> из левого верхнего угла экрана в положение (250, 250) за 10 секунд (10000 миллисекунд). |
| 697862fcee-08fa-4242-824e-6bf5a151b8ea | Translated (0%) | The example above will do a linear interpolation between the start and end value. | В приведенном выше примере будет выполнена линейная интерполяция между начальным и конечным значениями. |
| 707862fcee-08fa-4242-824e-6bf5a151b8ea | Translated (0%) | It is also possible to set values situated between the start and end value. | Также можно установить значения, расположенные между начальным и конечным значением. |
| 717862fcee-08fa-4242-824e-6bf5a151b8ea | Translated (0%) | The interpolation will then go by these points. | Интерполяция будет проходить по этим точкам. |
| 7233238c50-7556-4c96-b3cc-97f414269e64 | Translated (0%) | <240>QPushButton</240> button(<243>"Animated Button"</243>); button<246>.</246>show(); <249>QPropertyAnimation</249> animation(<252>&</252>button<255>,</255> <258>"geometry"</258>); animation<261>.</261>setDuration(<264>10000</264>); animation<267>.</267>setKeyValueAt(<270>0</270><273>,</273> <276>QRect</276>(<279>0</279><282>,</282> <285>0</285><288>,</288> <291>100</291><294>,</294> <297>30</297>)); animation<300>.</300>setKeyValueAt(<303>0.8</303><306>,</306> <309>QRect</309>(<312>250</312><315>,</315> <318>250</318><321>,</321> <324>100</324><327>,</327> <330>30</330>)); animation<333>.</333>setKeyValueAt(<336>1</336><339>,</339> <342>QRect</342>(<345>0</345><348>,</348> <351>0</351><354>,</354> <357>100</357><360>,</360> <363>30</363>)); animation<366>.</366>start(); | <240>QPushButton</240> button(<243>"Animated Button"</243>); button<246>.</246>show(); <249>QPropertyAnimation</249> animation(<252>&</252>button<255>,</255> <258>"geometry"</258>); animation<261>.</261>setDuration(<264>10000</264>); animation<267>.</267>setKeyValueAt(<270>0</270><273>,</273> <276>QRect</276>(<279>0</279><282>,</282> <285>0</285><288>,</288> <291>100</291><294>,</294> <297>30</297>)); animation<300>.</300>setKeyValueAt(<303>0.8</303><306>,</306> <309>QRect</309>(<312>250</312><315>,</315> <318>250</318><321>,</321> <324>100</324><327>,</327> <330>30</330>)); animation<333>.</333>setKeyValueAt(<336>1</336><339>,</339> <342>QRect</342>(<345>0</345><348>,</348> <351>0</351><354>,</354> <357>100</357><360>,</360> <363>30</363>)); animation<366>.</366>start(); |
| 7321c19551-8aab-4011-b7d5-258d6f1ad7ce | Translated (0%) | In this example, the animation will take the button to (250, 250) in 8 seconds, and then move it back to its original position in the remaining 2 seconds. | В этом примере анимация переместит кнопку на (250, 250) за 8 секунд, а затем вернет его обратно в исходное положение в оставшиеся 2 секунды. |
| 7421c19551-8aab-4011-b7d5-258d6f1ad7ce | Translated (0%) | The movement will be linearly interpolated between these points. | Движение будет линейно интерполироваться между этими точками. |
| 7508beabce-cb2d-415f-bd5c-4dcd983ced5c | Translated (0%) | You also have the possibility to animate values of a <369>QObject</369> that is not declared as a Qt property. | У вас также есть возможность анимировать значения <369>QObject</369>, который не объявлен как Qt property. |
| 7608beabce-cb2d-415f-bd5c-4dcd983ced5c | Translated (0%) | The only requirement is that this value has a setter. | Единственное требование заключается в том, что бы это значение имело сеттер. |
| 7708beabce-cb2d-415f-bd5c-4dcd983ced5c | Translated (0%) | You can then subclass the class containing the value and declare a property that uses this setter. | Затем вы можете создать подкласс класса, содержащий это значение, и объявить свойство, которое использует этот сеттер. |
| 7808beabce-cb2d-415f-bd5c-4dcd983ced5c | Translated (0%) | Note that each Qt property requires a getter, so you will need to provide a getter yourself if this is not defined. | Обратите внимание, что каждое Qt property требует геттера, поэтому вам нужно будет предоставить геттер самостоятельно, если он не определен. |
| 798e92ff82-5571-4da6-8ec5-724b5bc6c3a4 | Translated (0%) | <372>class</372> MyGraphicsRectItem : <375>public</375> <378>QObject</378><381>,</381> <384>public</384> <387>QGraphicsRectItem</387> { Q\_OBJECT Q\_PROPERTY(<390>QRectF</390> geometry READ geometry WRITE setGeometry) }; | <372>class</372> MyGraphicsRectItem : <375>public</375> <378>QObject</378><381>,</381> <384>public</384> <387>QGraphicsRectItem</387> { Q\_OBJECT Q\_PROPERTY(<390>QRectF</390> geometry READ geometry WRITE setGeometry) }; |
| 80e61777ab-ed5d-43d7-b137-dbe4692cccb7 | Translated (0%) | In the above code example, we subclass <393>QGraphicsRectItem</393> and define a geometry property. | В приведенном выше примере кода мы создаем подкласс <393>QGraphicsRectItem</393> и определяем свойство геометрии. |
| 81e61777ab-ed5d-43d7-b137-dbe4692cccb7 | Translated (0%) | We can now animate the widgets geometry even if <396>QGraphicsRectItem</396> does not provide the geometry property. | Теперь мы можем анимировать геометрию виджетов, даже если <396>QGraphicsRectItem</396> не предоставляет свойство геометрии. |
| 820954b508-8964-4110-a0ab-6fb3a52191f7 | Translated (0%) | For a general introduction to the Qt property system, see its <399>overview</399>. | Для общего ознакомления с системой свойств Qt см. ее <399>обзор</399>. |
| 837ef42047-8bc4-4bc6-8966-bc2eb38a0a0c | Translated (100%) | Animations and the Graphics View Framework | Анимация и Graphics View Framework |
| 84cfbf21b3-26c1-4623-a544-943d874a76c1 | Translated (0%) | When you want to animate <405>QGraphicsItem</405>s, you also use <408>QPropertyAnimation</408>. | Если вы хотите анимировать объекты <405>QGraphicsItem</405>, вам также нужно использовать <408>QPropertyAnimation</408>. |
| 85cfbf21b3-26c1-4623-a544-943d874a76c1 | Translated (0%) | However, <411>QGraphicsItem</411> does not inherit <414>QObject</414>. | Однако <411>QGraphicsItem</411> не наследует <414>QObject</414>. |
| 86cfbf21b3-26c1-4623-a544-943d874a76c1 | Translated (0%) | A good solution is to subclass the graphics item you wish to animate. | Хорошим решением будет создание подкласса QGraphicsItem, который вы хотите анимировать. |
| 87cfbf21b3-26c1-4623-a544-943d874a76c1 | Translated (0%) | This class will then also inherit <417>QObject</417>. | Также этот класс наследует <417>QObject</417>. |
| 88cfbf21b3-26c1-4623-a544-943d874a76c1 | Translated (0%) | This way, <420>QPropertyAnimation</420> can be used for <423>QGraphicsItem</423>s. | Таким образом, <420>QPropertyAnimation</420> можно использовать с объектами <423>QGraphicsItem</423>. |
| 89cfbf21b3-26c1-4623-a544-943d874a76c1 | Translated (0%) | The example below shows how this is done. | В приведенном ниже примере показано, как это делается. |
| 90cfbf21b3-26c1-4623-a544-943d874a76c1 | Translated (0%) | Another possibility is to inherit <426>QGraphicsWidget</426>, which already is a <429>QObject</429>. | Другой возможностью будет наследование объекта <426>QGraphicsWidget</426>, который уже является <429>QObject</429>. |
| 91816e4c0a-b238-4ef2-8aaf-28fd51bd7342 | Translated (0%) | <432>class</432> Pixmap : <435>public</435> <438>QObject</438><441>,</441> <444>public</444> <447>QGraphicsPixmapItem</447> { Q\_OBJECT Q\_PROPERTY(<450>QPointF</450> pos READ pos WRITE setPos) <453>.</453><456>.</456><459>.</459> | <432>class</432> Pixmap : <435>public</435> <438>QObject</438><441>,</441> <444>public</444> <447>QGraphicsPixmapItem</447> { Q\_OBJECT Q\_PROPERTY(<450>QPointF</450> pos READ pos WRITE setPos) <453>.</453><456>.</456><459>.</459> |
| 92d21a9a7e-c2df-4590-ab06-d135f89e5ca6 | Translated (0%) | As described in the previous section, we need to define properties that we wish to animate. | Как описано в предыдущем разделе, нам нужно определить свойства, которые мы хотим анимировать. |
| 93eaadea54-9983-44cf-9311-a997eb7e0e3a | Translated (0%) | Note that <462>QObject</462> must be the first class inherited as the meta-object system demands this. | Обратите внимание, что класс <462>QObject</462> должен быть унаследован первым, поскольку этого требует метаобъектная система. |
| 94b93dc42c-479c-4567-8014-f828bc57f783 | Translated (100%) | Easing Curves | Переходные характеристики |
| 9511524fca-4b6b-43da-812b-30a92ffed479 | Translated (0%) | As mentioned, <468>QPropertyAnimation</468> performs an interpolation between the start and end property value. | Как уже упоминалось, <468>QPropertyAnimation</468> выполняет интерполяцию между значением свойства start и end. |
| 9611524fca-4b6b-43da-812b-30a92ffed479 | Translated (0%) | In addition to adding more key values to the animation, you can also use an easing curve. | Помимо добавления дополнительных ключевых значений в анимацию, можно также использовать переходную характеристику. |
| 9711524fca-4b6b-43da-812b-30a92ffed479 | Translated (0%) | Easing curves describe a function that controls how the speed of the interpolation between 0 and 1 should be, and are useful if you want to control the speed of an animation without changing the path of the interpolation. | Переходная характеристика описывает функцию, которая управляет скоростью интерполяции между 0 и 1 и полезна, если вы хотите управлять скоростью анимации без изменения пути интерполяции. |
| 985d7f2625-138e-499a-95dc-331d2e898d37 | Translated (0%) | <471>QPushButton</471> button(<474>"Animated Button"</474>); button<477>.</477>show(); <480>QPropertyAnimation</480> animation(<483>&</483>button<486>,</486> <489>"geometry"</489>); animation<492>.</492>setDuration(<495>3000</495>); animation<498>.</498>setStartValue(<501>QRect</501>(<504>0</504><507>,</507> <510>0</510><513>,</513> <516>100</516><519>,</519> <522>30</522>)); animation<525>.</525>setEndValue(<528>QRect</528>(<531>250</531><534>,</534> <537>250</537><540>,</540> <543>100</543><546>,</546> <549>30</549>)); animation<552>.</552>setEasingCurve(<555>QEasingCurve</555><558>::</558>OutBounce); animation<561>.</561>start(); | <471>QPushButton</471> button(<474>"Animated Button"</474>); button<477>.</477>show(); <480>QPropertyAnimation</480> animation(<483>&</483>button<486>,</486> <489>"geometry"</489>); animation<492>.</492>setDuration(<495>3000</495>); animation<498>.</498>setStartValue(<501>QRect</501>(<504>0</504><507>,</507> <510>0</510><513>,</513> <516>100</516><519>,</519> <522>30</522>)); animation<525>.</525>setEndValue(<528>QRect</528>(<531>250</531><534>,</534> <537>250</537><540>,</540> <543>100</543><546>,</546> <549>30</549>)); animation<552>.</552>setEasingCurve(<555>QEasingCurve</555><558>::</558>OutBounce); animation<561>.</561>start(); |
| 99b8f8f5f6-2b89-457b-81bc-8a709ea769c2 | Translated (0%) | Here the animation will follow a curve that makes it bounce like a ball as if it was dropped from the start to the end position. | Здесь анимация будет следовать кривой, заставляющей ее отскакивать подобно мячу, как будто она движется от начала до конца. |
| 100b8f8f5f6-2b89-457b-81bc-8a709ea769c2 | Translated (0%) | <564>QEasingCurve</564> has a large collection of curves for you to choose from. | <564>QEasingCurve</564> имеет большое разнообразие кривых на ваш выбор. |
| 101b8f8f5f6-2b89-457b-81bc-8a709ea769c2 | Translated (0%) | These are defined by the <567>QEasingCurve::Type</567> enum. | Они определяются - <567>QEasingCurve::Type</567> enum. |
| 102b8f8f5f6-2b89-457b-81bc-8a709ea769c2 | Translated (0%) | If you are in need of another curve, you can also implement one yourself, and register it with <570>QEasingCurve</570>. | Если вам нужна другая кривая, вы также можете реализовать ее самостоятельно и зарегистрировать ее в <570>QEasingCurve</570>. |
| 1032d786c14-6e14-44c7-84a9-895f617348a8 | Translated (100%) | Putting Animations Together | Объединяем анимацию |
| 104f075f0dc-bb21-4db5-8be2-172e80f39e22 | Translated (0%) | An application will often contain more than one animation. | Приложение часто содержит более одной анимации. |
| 105f075f0dc-bb21-4db5-8be2-172e80f39e22 | Translated (0%) | For instance, you might want to move more than one graphics item simultaneously or move them in sequence after each other. | Например, может потребоваться переместить несколько графических элементов одновременно или последовательно друг за другом. |
| 1061c0842de-6ca6-48a0-bfae-901edc202017 | Translated (0%) | The subclasses of <576>QAnimationGroup</576> (<579>QSequentialAnimationGroup</579> and <582>QParallelAnimationGroup</582>) are containers for other animations so that these animations can be animated either in sequence or parallel. | Подклассы <576>QAnimationGroup</576>(<579>QSequentialAnimationGroup</579> и <582>QParallelAnimationGroup</582>) являются контейнерами для других анимаций, так что эти анимации могут быть анимированы либо последовательно, либо одновременно. |
| 1071c0842de-6ca6-48a0-bfae-901edc202017 | Translated (0%) | The <585>QAnimationGroup</585> is an example of an animation that does not animate properties, but it gets notified of time changes periodically. | <585>QAnimationGroup</585> является примером анимации, которая не анимирует свойства, но периодически получает уведомления об изменениях времени. |
| 1081c0842de-6ca6-48a0-bfae-901edc202017 | Translated (0%) | This enables it to forward those time changes to its contained animations, and thereby controlling when its animations are played. | Это позволяет ему пересылать эти изменения времени в содержащиеся анимации и тем самым контролировать, когда воспроизводятся анимации. |
| 109835ea4c2-241c-42f2-a3e6-2ce57b4400fb | Translated (0%) | Let's look at code examples that use both <588>QSequentialAnimationGroup</588> and <591>QParallelAnimationGroup</591>, starting off with the latter. | Давайте рассмотрим примеры кода, которые используют как <588>QSequentialAnimationGroup</588>, так и <591>QParallelAnimationGroup</591>, начиная с последнего. |
| 110be70a4d6-6ba8-4253-bdd4-321216cd1bac | Translated (0%) | <594>QPushButton</594> <597>\*</597>bonnie <600>=</600> <603>new</603> <606>QPushButton</606>(<609>"Bonnie"</609>); bonnie<612>-</612><615>></615>show(); <618>QPushButton</618> <621>\*</621>clyde <624>=</624> <627>new</627> <630>QPushButton</630>(<633>"Clyde"</633>); clyde<636>-</636><639>></639>show(); <642>QPropertyAnimation</642> <645>\*</645>anim1 <648>=</648> <651>new</651> <654>QPropertyAnimation</654>(bonnie<657>,</657> <660>"geometry"</660>); <663>// Set up anim1</663> <666>QPropertyAnimation</666> <669>\*</669>anim2 <672>=</672> <675>new</675> <678>QPropertyAnimation</678>(clyde<681>,</681> <684>"geometry"</684>); <687>// Set up anim2</687> <690>QParallelAnimationGroup</690> <693>\*</693>group <696>=</696> <699>new</699> <702>QParallelAnimationGroup</702>; group<705>-</705><708>></708>addAnimation(anim1); group<711>-</711><714>></714>addAnimation(anim2); group<717>-</717><720>></720>start(); | <594>QPushButton</594> <597>\*</597>bonnie <600>=</600> <603>new</603> <606>QPushButton</606>(<609>"Bonnie"</609>); bonnie<612>-</612><615>></615>show(); <618>QPushButton</618> <621>\*</621>clyde <624>=</624> <627>new</627> <630>QPushButton</630>(<633>"Clyde"</633>); clyde<636>-</636><639>></639>show(); <642>QPropertyAnimation</642> <645>\*</645>anim1 <648>=</648> <651>new</651> <654>QPropertyAnimation</654>(bonnie<657>,</657> <660>"geometry"</660>); <663>// Set up anim1</663> <666>QPropertyAnimation</666> <669>\*</669>anim2 <672>=</672> <675>new</675> <678>QPropertyAnimation</678>(clyde<681>,</681> <684>"geometry"</684>); <687>// Set up anim2</687> <690>QParallelAnimationGroup</690> <693>\*</693>group <696>=</696> <699>new</699> <702>QParallelAnimationGroup</702>; group<705>-</705><708>></708>addAnimation(anim1); group<711>-</711><714>></714>addAnimation(anim2); group<717>-</717><720>></720>start(); |
| 1118d48c5be-ff6a-405f-a854-898e2c4272c0 | Translated (0%) | A parallel group plays more than one animation at the same time. | Объект group воспроизводит более одной анимации одновременно. |
| 1128d48c5be-ff6a-405f-a854-898e2c4272c0 | Translated (0%) | Calling its <723>start()</723> function will start all animations it governs. | Вызов его функции <723>start()</723> запустит все анимации, которыми он управляет. |
| 1131c9c1f36-1829-44a5-8f83-f7a0f1fd4431 | Translated (0%) | <726>QPushButton</726> button(<729>"Animated Button"</729>); button<732>.</732>show(); <735>QPropertyAnimation</735> anim1(<738>&</738>button<741>,</741> <744>"geometry"</744>); anim1<747>.</747>setDuration(<750>3000</750>); anim1<753>.</753>setStartValue(<756>QRect</756>(<759>0</759><762>,</762> <765>0</765><768>,</768> <771>100</771><774>,</774> <777>30</777>)); anim1<780>.</780>setEndValue(<783>QRect</783>(<786>500</786><789>,</789> <792>500</792><795>,</795> <798>100</798><801>,</801> <804>30</804>)); <807>QPropertyAnimation</807> anim2(<810>&</810>button<813>,</813> <816>"geometry"</816>); anim2<819>.</819>setDuration(<822>3000</822>); anim2<825>.</825>setStartValue(<828>QRect</828>(<831>500</831><834>,</834> <837>500</837><840>,</840> <843>100</843><846>,</846> <849>30</849>)); anim2<852>.</852>setEndValue(<855>QRect</855>(<858>1000</858><861>,</861> <864>500</864><867>,</867> <870>100</870><873>,</873> <876>30</876>)); <879>QSequentialAnimationGroup</879> group; group<882>.</882>addAnimation(<885>&</885>anim1); group<888>.</888>addAnimation(<891>&</891>anim2); group<894>.</894>start(); | <726>QPushButton</726> button(<729>"Animated Button"</729>); button<732>.</732>show(); <735>QPropertyAnimation</735> anim1(<738>&</738>button<741>,</741> <744>"geometry"</744>); anim1<747>.</747>setDuration(<750>3000</750>); anim1<753>.</753>setStartValue(<756>QRect</756>(<759>0</759><762>,</762> <765>0</765><768>,</768> <771>100</771><774>,</774> <777>30</777>)); anim1<780>.</780>setEndValue(<783>QRect</783>(<786>500</786><789>,</789> <792>500</792><795>,</795> <798>100</798><801>,</801> <804>30</804>)); <807>QPropertyAnimation</807> anim2(<810>&</810>button<813>,</813> <816>"geometry"</816>); anim2<819>.</819>setDuration(<822>3000</822>); anim2<825>.</825>setStartValue(<828>QRect</828>(<831>500</831><834>,</834> <837>500</837><840>,</840> <843>100</843><846>,</846> <849>30</849>)); anim2<852>.</852>setEndValue(<855>QRect</855>(<858>1000</858><861>,</861> <864>500</864><867>,</867> <870>100</870><873>,</873> <876>30</876>)); <879>QSequentialAnimationGroup</879> group; group<882>.</882>addAnimation(<885>&</885>anim1); group<888>.</888>addAnimation(<891>&</891>anim2); group<894>.</894>start(); |
| 11439b31337-a318-4e66-a399-7de4309c45f1 | Translated (0%) | As you no doubt have guessed, <897>QSequentialAnimationGroup</897> plays its animations in sequence. | Как вы, без сомнения, догадались, <897>QSequentialAnimationGroup</897> воспроизводит свои анимации по очереди. |
| 11539b31337-a318-4e66-a399-7de4309c45f1 | Translated (0%) | It starts the next animation in the list after the previous is finished. | Он запускает следующую анимацию в списке после завершения предыдущей. |
| 116fcb6dc12-444e-4691-8041-ce32753d06a8 | Translated (0%) | Since an animation group is an animation itself, you can add it to another group. | Поскольку группа анимации - это тоже анимация, ее можно добавить в другую группу. |
| 117fcb6dc12-444e-4691-8041-ce32753d06a8 | Translated (0%) | This way, you can build a tree structure of animations which specifies when the animations are played in relation to each other. | Таким образом, можно построить древовидную структуру анимаций, которая определяет, когда анимации воспроизводятся относительно друг друга. |
| 1183496c932-8292-44d7-8bfe-7a6858b6b462 | Translated (100%) | Animations and States | Анимации и состояния |
| 119f1aad432-8dcb-4016-85d2-973d31a86203 | Translated (0%) | When using a <903>state machine</903>, we can associate one or more animations to a transition between states using a <906>QSignalTransition</906> or <909>QEventTransition</909> class. | При использовании <903>state machine</903> можно связать одну или несколько анимаций с переходом между состояниями с помощью класса <906>QSignalTransition</906> или <909>QEventTransition</909>. |
| 120f1aad432-8dcb-4016-85d2-973d31a86203 | Translated (0%) | These classes are both derived from <912>QAbstractTransition</912>, which defines the convenience function <915>addAnimation()</915> that enables the appending of one or more animations triggered when the transition occurs. | Оба эти класса являются производными от <912>QAbstractTransition</912>, который определяет вспомогательную функцию <915>addAnimation()</915>, которая позволяет добавлять одну или несколько анимаций, запускаемых при переходе. |
| 12193a43d15-12b7-45f3-abb9-466a819182e4 | Translated (0%) | We also have the possibility to associate properties with the states rather than setting the start and end values ourselves. | У нас также есть возможность связать свойства с состояниями, а не устанавливать начальные и конечные значения самостоятельно. |
| 12293a43d15-12b7-45f3-abb9-466a819182e4 | Translated (0%) | Below is a complete code example that animates the geometry of a <918>QPushButton</918>. | Ниже приведен полный пример кода, который анимирует геометрию <918>QPushButton</918>. |
| 1234ec9d8ac-df4b-4d46-85b6-d7e105d5db96 | Translated (0%) | <921>QPushButton</921> <924>\*</924>button <927>=</927> <930>new</930> <933>QPushButton</933>(<936>"Animated Button"</936>); button<939>-</939><942>></942>show(); <945>QStateMachine</945> <948>\*</948>machine <951>=</951> <954>new</954> <957>QStateMachine</957>; <960>QState</960> <963>\*</963>state1 <966>=</966> <969>new</969> <972>QState</972>(machine); state1<975>-</975><978>></978>assignProperty(button<981>,</981> <984>"geometry"</984><987>,</987> <990>QRect</990>(<993>0</993><996>,</996> <999>0</999><1002>,</1002> <1005>100</1005><1008>,</1008> <1011>30</1011>)); machine<1014>-</1014><1017>></1017>setInitialState(state1); <1020>QState</1020> <1023>\*</1023>state2 <1026>=</1026> <1029>new</1029> <1032>QState</1032>(machine); state2<1035>-</1035><1038>></1038>assignProperty(button<1041>,</1041> <1044>"geometry"</1044><1047>,</1047> <1050>QRect</1050>(<1053>250</1053><1056>,</1056> <1059>250</1059><1062>,</1062> <1065>100</1065><1068>,</1068> <1071>30</1071>)); <1074>QSignalTransition</1074> <1077>\*</1077>transition1 <1080>=</1080> state1<1083>-</1083><1086>></1086>addTransition(button<1089>,</1089> SIGNAL(clicked())<1092>,</1092> state2); transition1<1095>-</1095><1098>></1098>addAnimation(<1101>new</1101> <1104>QPropertyAnimation</1104>(button<1107>,</1107> <1110>"geometry"</1110>)); <1113>QSignalTransition</1113> <1116>\*</1116>transition2 <1119>=</1119> state2<1122>-</1122><1125>></1125>addTransition(button<1128>,</1128> SIGNAL(clicked())<1131>,</1131> state1); transition2<1134>-</1134><1137>></1137>addAnimation(<1140>new</1140> <1143>QPropertyAnimation</1143>(button<1146>,</1146> <1149>"geometry"</1149>)); machine<1152>-</1152><1155>></1155>start(); | <921>QPushButton</921> <924>\*</924>button <927>=</927> <930>new</930> <933>QPushButton</933>(<936>"Animated Button"</936>); button<939>-</939><942>></942>show(); <945>QStateMachine</945> <948>\*</948>machine <951>=</951> <954>new</954> <957>QStateMachine</957>; <960>QState</960> <963>\*</963>state1 <966>=</966> <969>new</969> <972>QState</972>(machine); state1<975>-</975><978>></978>assignProperty(button<981>,</981> <984>"geometry"</984><987>,</987> <990>QRect</990>(<993>0</993><996>,</996> <999>0</999><1002>,</1002> <1005>100</1005><1008>,</1008> <1011>30</1011>)); machine<1014>-</1014><1017>></1017>setInitialState(state1); <1020>QState</1020> <1023>\*</1023>state2 <1026>=</1026> <1029>new</1029> <1032>QState</1032>(machine); state2<1035>-</1035><1038>></1038>assignProperty(button<1041>,</1041> <1044>"geometry"</1044><1047>,</1047> <1050>QRect</1050>(<1053>250</1053><1056>,</1056> <1059>250</1059><1062>,</1062> <1065>100</1065><1068>,</1068> <1071>30</1071>)); <1074>QSignalTransition</1074> <1077>\*</1077>transition1 <1080>=</1080> state1<1083>-</1083><1086>></1086>addTransition(button<1089>,</1089> SIGNAL(clicked())<1092>,</1092> state2); transition1<1095>-</1095><1098>></1098>addAnimation(<1101>new</1101> <1104>QPropertyAnimation</1104>(button<1107>,</1107> <1110>"geometry"</1110>)); <1113>QSignalTransition</1113> <1116>\*</1116>transition2 <1119>=</1119> state2<1122>-</1122><1125>></1125>addTransition(button<1128>,</1128> SIGNAL(clicked())<1131>,</1131> state1); transition2<1134>-</1134><1137>></1137>addAnimation(<1140>new</1140> <1143>QPropertyAnimation</1143>(button<1146>,</1146> <1149>"geometry"</1149>)); machine<1152>-</1152><1155>></1155>start(); |
| 12449c364bc-595c-41b4-9e78-3f8a4d318c0d | Translated (0%) | For a more comprehensive example of how to use the state machine framework for animations, see the states example (it lives in the <examples/animation/states/> directory). | Более полный пример использования фреймворка state machine для анимации см. В примере states example (он находится в каталоге <examples/animation/states/>). |
| 12519b1cadc-07a6-4008-8f10-c63b6b5efec7 | Translated (100%) | 2016 The Qt Company Ltd. Documentation contributions included herein are the copyrights of their respective owners. | 2016 The Qt Company Ltd. Documentation contributions included herein are the copyrights of their respective owners. |
| 12619b1cadc-07a6-4008-8f10-c63b6b5efec7 | Translated (100%) | The documentation provided herein is licensed under the terms of the <1164>GNU Free Documentation License version 1.3</1164> as published by the Free Software Foundation. | The documentation provided herein is licensed under the terms of the <1164>GNU Free Documentation License version 1.3</1164> as published by the Free Software Foundation. |
| 12719b1cadc-07a6-4008-8f10-c63b6b5efec7 | Translated (100%) | Qt and respective logos are trademarks of The Qt Company Ltd. in Finland and/or other countries worldwide. | Qt and respective logos are trademarks of The Qt Company Ltd. in Finland and/or other countries worldwide. |
| 12819b1cadc-07a6-4008-8f10-c63b6b5efec7 | Translated (100%) | All other trademarks are property of their respective owners. | All other trademarks are property of their respective owners. |