Архитектура приложения, основные компоненты

Архитектура Android приложений основана на идее многократного использования компонентов, которые являются основными строительными блоками. Каждый компонент является отдельной сущностью и помогает определить общее поведение приложения.

Система Android выстроена таким образом, что любое приложение может запускать необходимый компонент другого приложения. Например, если приложение предполагает использование камеры для создания фотографий, совершенно необязательно создавать в этом приложении активность для работы с камерой. Наверняка на устройстве уже есть приложение для получения фотографий с камеры, достаточно запустить соответствующую активность, сделать фотографию и вернуть ее в приложение, так что пользователь будет считать, что камера часть приложения, с которым он работает.

Когда система запускает компонент, она запускает процесс приложения, которому принадлежит компонент, если он еще не запущен, и создает экземпляры классов, необходимых компоненту. Поэтому в отличие от большинства других систем, в системе Android приложения не имеют единой точки входа (нет метода main(), например). В силу запуска каждого приложения в отдельном процессе и ограничений на доступ к файлам, приложение не может напрямую активировать компонент другого приложения. Таким образом для активации компонента другого приложения необходимо послать системе сообщение о намерении запустить определенный компонент, система активирует его.

Можно выделить четыре различных типа компонентов, каждый тип служит для достижения определенной цели и имеет свой особый жизненный цикл, который определяет способы создания и разрушения соответствующего компонент:

* Активности (Activities)
* Сервисы (Services)
* Контент-провайдеры (Content providers)
* Приемники широковещательных сообщений (Broadcast Receivers)

Активности (Activities)

Активность - это видимая часть приложения (экран, окно, форма), отвечает за отображение графического интерфейса пользователя. При этом приложение может иметь несколько активностей, например, в приложении, предназначенном для работы с электронной почтой, одна активность может использоваться для отображения списка новых писем, другая активность - для написания, и еще одна - для чтения писем. Несмотря на то, что для пользователя приложение представляется единым целым, все активности приложения не зависят друг от друга. В связи с этим любая из этих активностей может быть запущена из другого приложения, имеющего доступ к активностям данного приложения. Например, приложение камеры может запустить активность, создающую новые письма, чтобы отправить только что сделанную фотографию адресату, указанному пользователем.

Сервисы (Services)

Сервис - компонент, который работает в фоновом режиме, выполняет длительные по времени операции или работу для удаленных процессов. Сервис не предоставляет пользовательского интерфейса. Например, сервис может проигрывать музыку в фоновом режиме, пока пользователь использует другое приложение, может загружать данные из сети, не блокируя взаимодействие пользователя с активностью. Сервис может быть запущен другим компонентом и после этого работать самостоятельно, а может остаться связанным с этим компонентом и взаимодействовать с ним.

Контент-провайдеры (Content providers)

Контент-провайдер управляет распределенным множеством данных приложения.

Данные могут храниться:

* в файловой системе;
* в базе данных SQLite;
* в сети;
* в любом другом доступном для приложения месте.

Контент-провайдер позволяет другим приложениям при наличии у них соответствующих прав делать запросы или даже менять данные.

Например, в системе Android есть контент-провайдер, который управляет информацией о контактах пользователя. В связи с этим, любое приложение с соответствующими правами может сделать запрос на чтение и запись информации какого-либо контакта. Контент-провайдер может быть также полезен для чтения и записи приватных данных приложения, не предназначенных для доступа извне.

Приемники широковещательных сообщений (Broadcast Receivers)

Приемник - компонент, который реагирует на широковещательные извещения.

Большинство таких извещений порождаются системой, например, извещение о том, что экран отключился или низкий заряд батареи. Приложения также могут инициировать широковещание, например, разослать другим приложениям сообщение о том, что некоторые данные загружены и доступны для использования. Хотя приемники не отображают пользовательского интерфейса, они могут создавать уведомление на панели состояний, чтобы предупредить пользователя о появлении сообщения. Такой приемник служит проводником к другим компонентам и предназначен для выполнения небольшого объема работ, например, он может запустить соответствующий событию сервис.