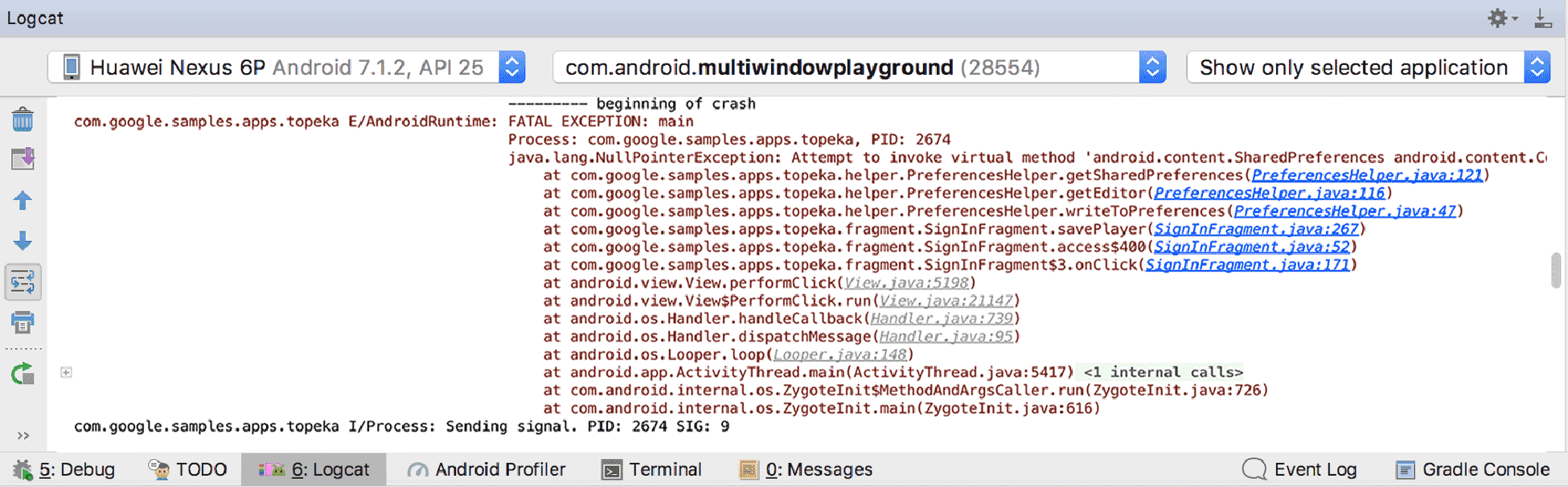
Анализировать трассировку стека

Отладка приложения часто требует работы со следами стека. Трассировка стека генерируется всякий раз, когда ваше приложение падает из-за ошибки или исключения. Вы также можете распечатать трассировку стека в любой точке кода приложения, используя такие методы, как [Thread.dumpStack()](https://developer.android.com/reference/java/lang/Thread.html" \l "dumpStack()).

Пока ваше приложение работает в режиме отладки на подключенном устройстве, Android Studio печатает и выделяет следы стека в представлении **logcat** , как показано на рисунке 1.



**Рисунок 1.** Трассировка стека в logcat.

Трассировка стека показывает список вызовов методов, которые приводят к возникновению исключения, вместе с именами файлов и номерами строк, где происходили вызовы. Вы можете нажать на выделенные имена файлов, чтобы открыть файлы и изучить источник вызова метода. Нажмите **Вверх по трассировке стека** https://developer.android.com/studio/images/buttons/logcat-arrow-up.png и **Вниз по трассировке стека,** https://developer.android.com/studio/images/buttons/logcat-arrow-down.png чтобы быстро перемещаться между строками трассировки стека, отображаемыми в окне logcat.

Следы открытого стека из внешних источников

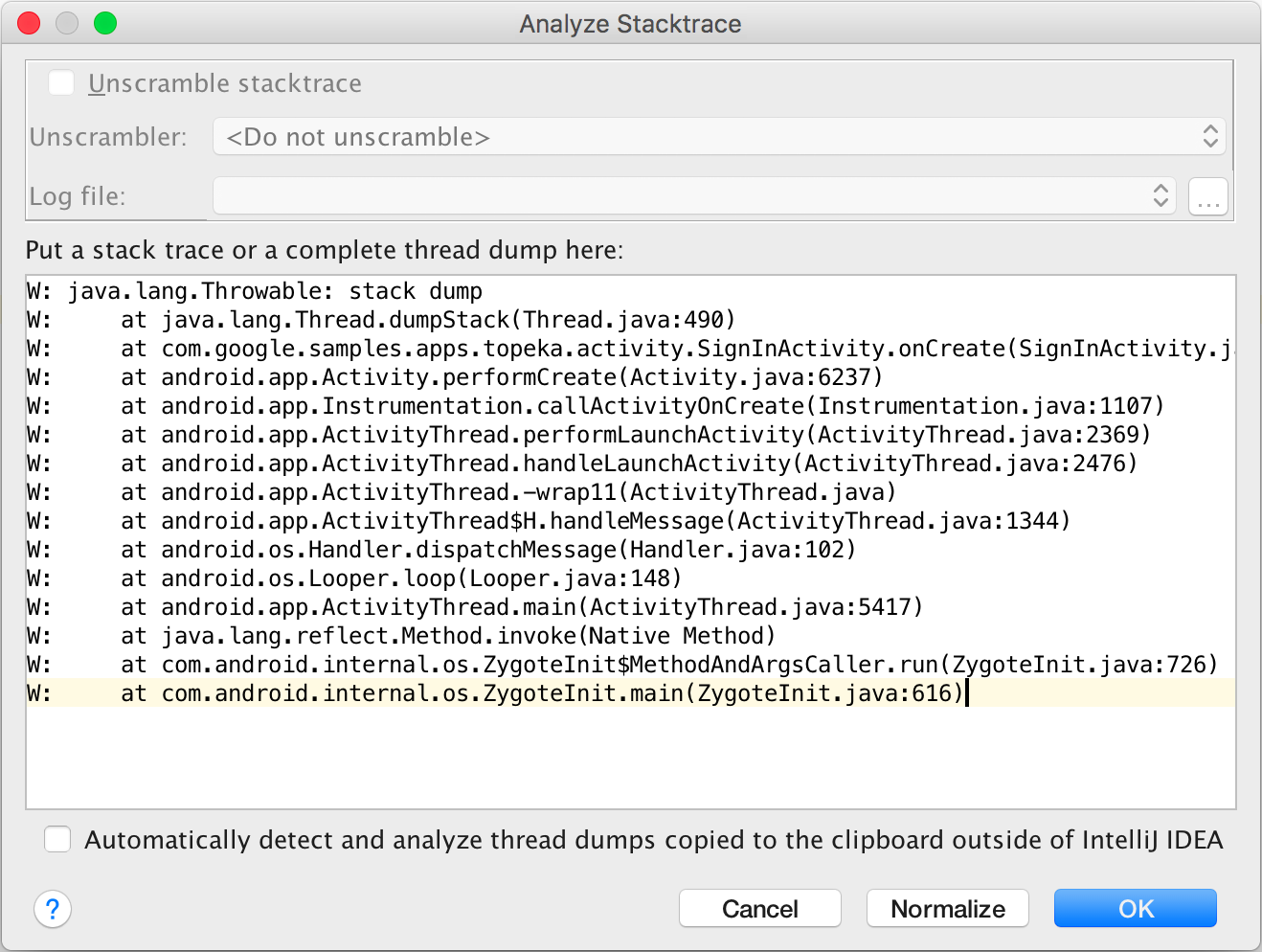
Иногда вы хотите проанализировать трассировки стека, которые были переданы вам в отчете об ошибке, а не те, которые вы нашли во время отладки. Например, вы можете собирать трассировки стеков, сгенерированные на устройствах ваших пользователей, с помощью консоли Google Play или с помощью другого инструмента, такого как [Firebase Crash Reporting](https://firebase.google.com/docs/crash/) .

Чтобы получить такое же выделенное и интерактивное представление трассировки внешнего стека из отчета об ошибке, выполните следующие действия:

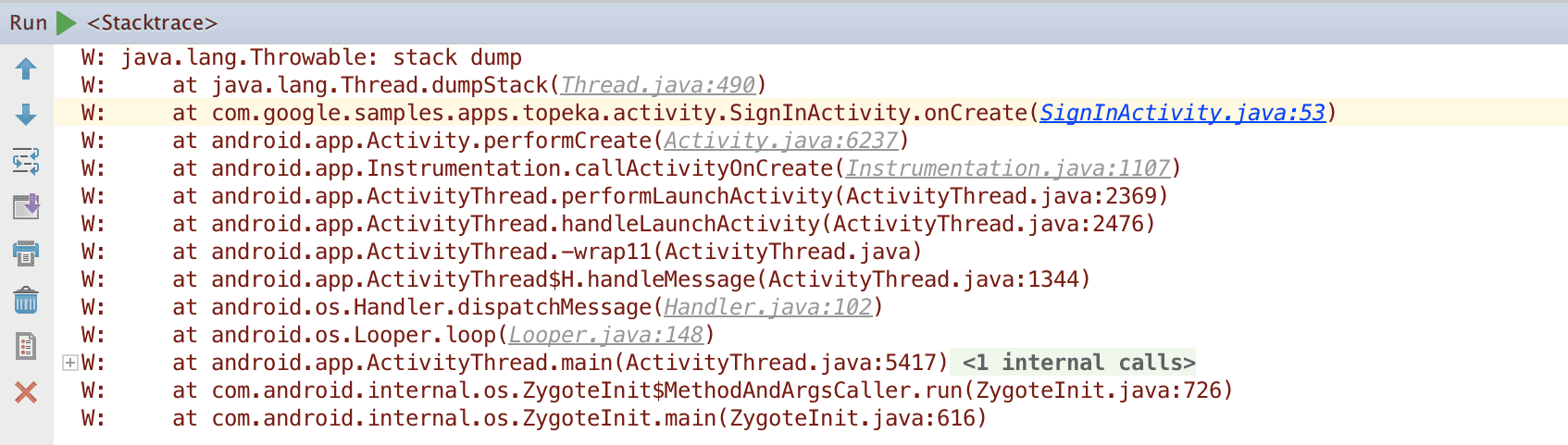
1. Откройте свой проект в Android Studio.

**Примечание.** Убедитесь, что исходный код, который вы просматриваете, относится к той же версии вашего приложения, которая сгенерировала трассировку стека. Если код отличается, вы получите несоответствия между именами файлов и номерами строк или порядком вызовов в трассировке стека и вашего проекта.

1. Из **Analyze** меню, нажмите кнопку **Анализ трассировки стека** .



1. Вставьте текст трассировки **стека в** окно « **Анализ трассировки стека»** и нажмите « **ОК»** .
2. Android Studio открывает новую вкладку **<Stacktrace>** с трассировкой стека, вставленной вами в окно « **Выполнить»** .



Мониторинг буфера обмена для новых следов стека

Если вы много работаете с внешними трассировками стека, вы можете повысить свою производительность, позволив Android Studio постоянно отслеживать системный буфер обмена на наличие новых трассировок стека:

1. Откройте инструмент **Анализ стека** .
2. Установите флажок **Автоматически обнаруживать и анализировать дампы потоков, скопированные в буфер обмена вне IntelliJ IDEA** .
3. Скопируйте трассировку стека в другом приложении (например, из веб-браузера) в системный буфер обмена.
4. Когда вы вернетесь в окно Android Studio, трассировка стека откроется автоматически под окном « **Выполнить»** , без необходимости вставлять ее в окно «Анализировать трассировку стека».