

Android Developers

向另一个应用发送用户

本课程将向您展示如何

- 构建隐含 Intent

- 验证是否存在接收 Intent 的应用

- 启动具有 Intent 的 Activity

- 显示应用选择器

您也应该阅读

- 共享简单数据

Android 最重要的功能之一是应用能够基于它要执行的“操作”向另一个应用发送用户。例如，如果您的应用有您要在地图上显示的公司地址，您无需在显示地图的应用中构建 Activity。而是可以创建使用 [Intent](https://developer.android.google.cn/reference/android/content/Intent.html) (<https://developer.android.google.cn/reference/android/content/Intent.html>) 查看地址的请求。Android 系统之后启动可以在地图上显示该地址的应用。

正如第一堂课构建您的第一个应用 (<https://developer.android.google.cn/training/basics/firstapp/index.html>)中所讲述的，您必须使用 Intent 在自己应用中的 Activity 之间进行导航。您通常使用显式 Intent 执行此操作，该 Intent 定义您希望启动的组件的确切类名称。但是，当您希望另一应用执行操作时，比如“查看地图”，您必须使用隐含 Intent。

本课程向您展示如何针对特定操作创建隐含 Intent，以及如何使用该 Intent 启动在另一个应用中执行操作的 Activity。

构建隐含 Intent

隐含 Intent 不声明要启动的组件的类名称，而是声明要执行的操作。该操作指定您要执行的操作，比如查看、编辑、发送或 获取 某项。Intent 通常还包含与操作关联的数据，比如您要查看的地址或您要发送的电子邮件消息。根据要创建的 Intent，数据可能是 [Uri](https://developer.android.google.cn/reference/android/net/Uri.html) (<https://developer.android.google.cn/reference/android/net/Uri.html>)、多种其他数据类型之一，或 Intent 可能根本就不需要数据。

如果您的数据是 [Uri](https://developer.android.google.cn/reference/android/net/Uri.html) (<https://developer.android.google.cn/reference/android/net/Uri.html>)，您可以使用一个简单的 `Intent()`

This site uses cookies to store your preferences for site-specific language and display options.

OK

例如，此处显示如何使用指定电话号码的 `Uri`

(<https://developer.android.google.cn/reference/android/net/Uri.html>) 数据创建发起电话呼叫的 `Intent`：

```
Uri number = Uri.parse("tel:5551234");
Intent callIntent = new Intent(Intent.ACTION_DIAL, number);
```

当您的应用通过调用 `startActivity()`

([https://developer.android.google.cn/reference/android/app/Activity.html#startActivity\(android.content.Intent\)](https://developer.android.google.cn/reference/android/app/Activity.html#startActivity(android.content.Intent))) 调用此 `Intent` 时，“电话”应用会发起向指定电话号码的呼叫。

这里有一些其他 `Intent` 及其操作和 `Uri`

(<https://developer.android.google.cn/reference/android/net/Uri.html>) 数据对：

- 查看地图：

```
// Map point based on address
Uri location = Uri.parse("geo:0,0?q=1600+Amphitheatre+Parkway,+Mountain+View,+California");
// Or map point based on latitude/longitude
// Uri location = Uri.parse("geo:37.422219,-122.08364?z=14"); // z param is zoom level
Intent mapIntent = new Intent(Intent.ACTION_VIEW, location);
```

- 查看网页：

```
Uri webpage = Uri.parse("http://www.android.com");
Intent webIntent = new Intent(Intent.ACTION_VIEW, webpage);
```

其他类型的隐含 `Intent` 需要提供不同数据类型（比如，字符串）的“额外”数据。您可以使用各种 `putExtra()` ([https://developer.android.google.cn/reference/android/content/Intent.html#putExtra\(java.lang.String, java.lang.String\)](https://developer.android.google.cn/reference/android/content/Intent.html#putExtra(java.lang.String, java.lang.String))) 方法添加一条或多条 `extra` 数据。

默认情况下，系统基于所包含的 `Uri`

(<https://developer.android.google.cn/reference/android/net/Uri.html>) 数据确定 `Intent` 需要的相应 MIME 类型。如果您未在 `Intent` 中包含 `Uri`

(<https://developer.android.google.cn/reference/android/net/Uri.html>)，您通常应使用 `setType()` ([https://developer.android.google.cn/reference/android/content/Intent.html#setType\(java.lang.String\)](https://developer.android.google.cn/reference/android/content/Intent.html#setType(java.lang.String))) 指定与 `Intent` 关联的数据的类型。设置 MIME 类型可进一步指定哪些类型的 `Activity` 应接收 `Intent`。

此处有更多添加额外数据以指定所需操作的 `Intent`：

- 发送带附件的电子邮件：

```
Intent emailIntent = new Intent(Intent.ACTION_SEND);
// The intent does not have a URI, so declare the "text/plain" MIME type
emailIntent.setType(HTTP.PLAIN_TEXT_TYPE);
emailIntent.putExtra(Intent.EXTRA_EMAIL, new String[] {"jon@example.com"}); // recipient
emailIntent.putExtra(Intent.EXTRA_SUBJECT, "Email subject");
```

This site uses cookies to store your preferences for site-specific language and display options.

OK

- 创建日历事件：

```
Intent calendarIntent = new Intent(Intent.ACTION_INSERT, Events.CONTENT_URI);
Calendar beginTime = Calendar.getInstance().set(2012, 0, 19, 7, 30);
Calendar endTime = Calendar.getInstance().set(2012, 0, 19, 10, 30);
calendarIntent.putExtra(CalendarContract.EXTRA_EVENT_BEGIN_TIME, beginTime.getTimeInMillis());
calendarIntent.putExtra(CalendarContract.EXTRA_EVENT_END_TIME, endTime.getTimeInMillis());
calendarIntent.putExtra(Events.TITLE, "Ninja class");
calendarIntent.putExtra(Events.EVENT_LOCATION, "Secret dojo");
```

注：只有 API 级别 14 或更高级别支持此日历事件 Intent。

注：尽可能具体地定义您的 Intent

(<https://developer.android.google.cn/reference/android/content/Intent.html>) 非常重要。例如，如果您想要使用 ACTION_VIEW

(https://developer.android.google.cn/reference/android/content/Intent.html#ACTION_VIEW) Intent 显示图像，您应指定 MIME 类型 image/*。这可防止可“查看”数据的其他类型的应用（比如地图应用）被 Intent 触发。

验证是否存在接收 Intent 的应用

尽管 Android 平台保证某些 Intent 可以分解为内置应用之一（比如，“电话”、“电子邮件”或“日历”应用），您应在调用 Intent 之前始终包含确认步骤。

注意：如果您调用了 Intent，但设备上没有可用于处理 Intent 的应用，您的应用将崩溃。

要确认是否存在可响应 Intent 的可用 Activity，请调用 `queryIntentActivities()`

([https://developer.android.google.cn/reference/android/content/pm/PackageManager.html#queryIntentActivities\(android.content.Intent,int\)](https://developer.android.google.cn/reference/android/content/pm/PackageManager.html#queryIntentActivities(android.content.Intent,int))) 来获取能够处理您的 Intent

(<https://developer.android.google.cn/reference/android/content/Intent.html>) 的 Activity 列表。如果返回的 List (<https://developer.android.google.cn/reference/java/util/List.html>) 不为空，您可以安全地使用该 Intent。例如：

```
PackageManager packageManager = getPackageManager() (https://developer.android.google.cn/reference/android/content/pm/PackageManager.html)
List activities = packageManager.queryIntentActivities(intent,
    PackageManager.MATCH_DEFAULT_ONLY);
boolean isIntentSafe = activities.size() > 0;
```

如果 `isIntentSafe` 是 `true`，则至少有一个应用将响应该 Intent。如果它是 `false`，则没有任何应用处理该 Intent。

This site uses cookies to store your preferences for site-specific language and display options.

OK

的产品 (<https://developer.android.google.cn/distribute/tools/promote/linking.html>) 。

启动具有 Intent 的 Activity

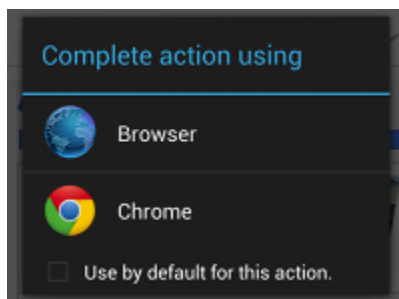


图 1. 当多个应用可处理 Intent 时显示的选择对话框示例。

一旦您已创建您的 Intent (<https://developer.android.google.cn/reference/android/content/Intent.html>) 并设置 extra 信息，调用 `startActivity()` ([https://developer.android.google.cn/reference/android/app/Activity.html#startActivity\(android.content.Intent\)](https://developer.android.google.cn/reference/android/app/Activity.html#startActivity(android.content.Intent))) 将其发送给系统。如果系统识别可处理 Intent 的多个 Activity，它会为用户显示对话框供其选择要使用的应用，如图 1 所示。如果只有一个 Activity 处理 Intent，系统会立即将其启动。

```
startActivity(intent);
```

此处显示完整的示例：如何创建查看地图的 Intent，验证是否存在处理 Intent 的应用，然后启动它：

```
// Build the intent
Uri location = Uri.parse("geo:0,0?q=1600+Amphitheatre+Parkway,+Mountain+View,+California");
Intent mapIntent = new Intent(Intent.ACTION_VIEW, location);

// Verify it resolves
PackageManager packageManager = getPackageManager() (https://developer.android.google.cn/refer
List<ResolveInfo> activities = packageManager.queryIntentActivities(mapIntent, 0);
boolean isIntentSafe = activities.size() > 0;

// Start an activity if it's safe
if (isIntentSafe) {
    startActivity(mapIntent);
}
```

显示应用选择器

This site uses cookies to store your preferences for site-specific language and display options.

OK

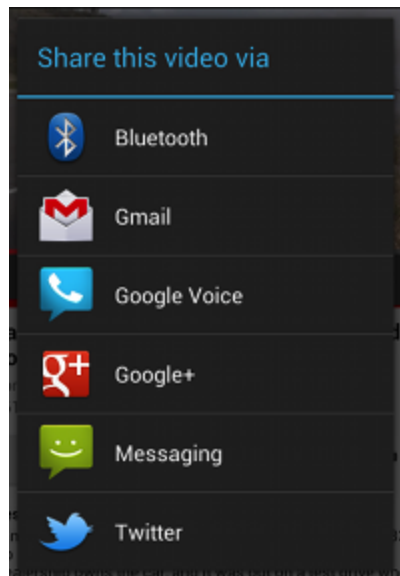


图 2. 选择器对话框。

注意，当您通过将您的 `Intent`

(<https://developer.android.google.cn/reference/android/content/Intent.html>) 传递至 `startActivity()` ([https://developer.android.google.cn/reference/android/app/Activity.html#startActivity\(android.content.Intent\)](https://developer.android.google.cn/reference/android/app/Activity.html#startActivity(android.content.Intent))) 而启动 Activity 时，有多个应用响应 `Intent`，用户可以选择默认使用哪个应用（通过选中对话框底部的复选框；见图 1）。当执行用户通常希望每次使用相同应用进行的操作时，比如当打开网页（用户可能只使用一个网络浏览器）或拍照（用户可能习惯使用一个相机）时，这非常有用。

但是，如果要执行的操作可由多个应用处理并且用户可能习惯于每次选择不同的应用——比如“共享”操作，用户有多个应用分享项目——您应明确显示选择器对话框，如图 2 所示。选择器对话框强制用户选择用于每次操作的应用（用户不能对此操作选择默认的应用）。

要显示选择器，请使用 `createChooser()`

([https://developer.android.google.cn/reference/android/content/Intent.html#createChooser\(android.content.Intent, java.lang.CharSequence\)](https://developer.android.google.cn/reference/android/content/Intent.html#createChooser(android.content.Intent, java.lang.CharSequence))) 创建 `Intent`

(<https://developer.android.google.cn/reference/android/content/Intent.html>) 并将其传递给 `startActivity()`

([https://developer.android.google.cn/reference/android/app/Activity.html#startActivity\(android.content.Intent\)](https://developer.android.google.cn/reference/android/app/Activity.html#startActivity(android.content.Intent)))。例如：

```
Intent intent = new Intent(Intent.ACTION_SEND);
...

// Always use string resources for UI text.
// This says something like "Share this photo with"
String title = getResources().getString(R.string.chooser_title);
// Create intent to show chooser
Intent chooser = Intent.createChooser(intent, title);

// Verify the intent will resolve to at least one activity
if (intent.resolveActivity(getPackageManager()) != null) {
    startActivity(chooser);
}
```

This site uses cookies to store your preferences for site-specific language and display options.

OK

这将显示一个对话框，其中包含响应传递给 `createChooser()`

([`https://developer.android.google.cn/reference/android/content/Intent.html#createChooser\(android.content.Intent, java.lang.CharSequence\)`](https://developer.android.google.cn/reference/android/content/Intent.html#createChooser(android.content.Intent, java.lang.CharSequence))) 方法的 `Intent` 的应用列表，并且将提供的文本用作对话框标题。

This site uses cookies to store your preferences for site-specific language and display options.

OK