

Android Services & Security: Programming Started Services with Intents & Messengers (Part 3)

Douglas C. Schmidt

d.schmidt@vanderbilt.edu

www.dre.vanderbilt.edu/~schmidt



Professor of Computer Science

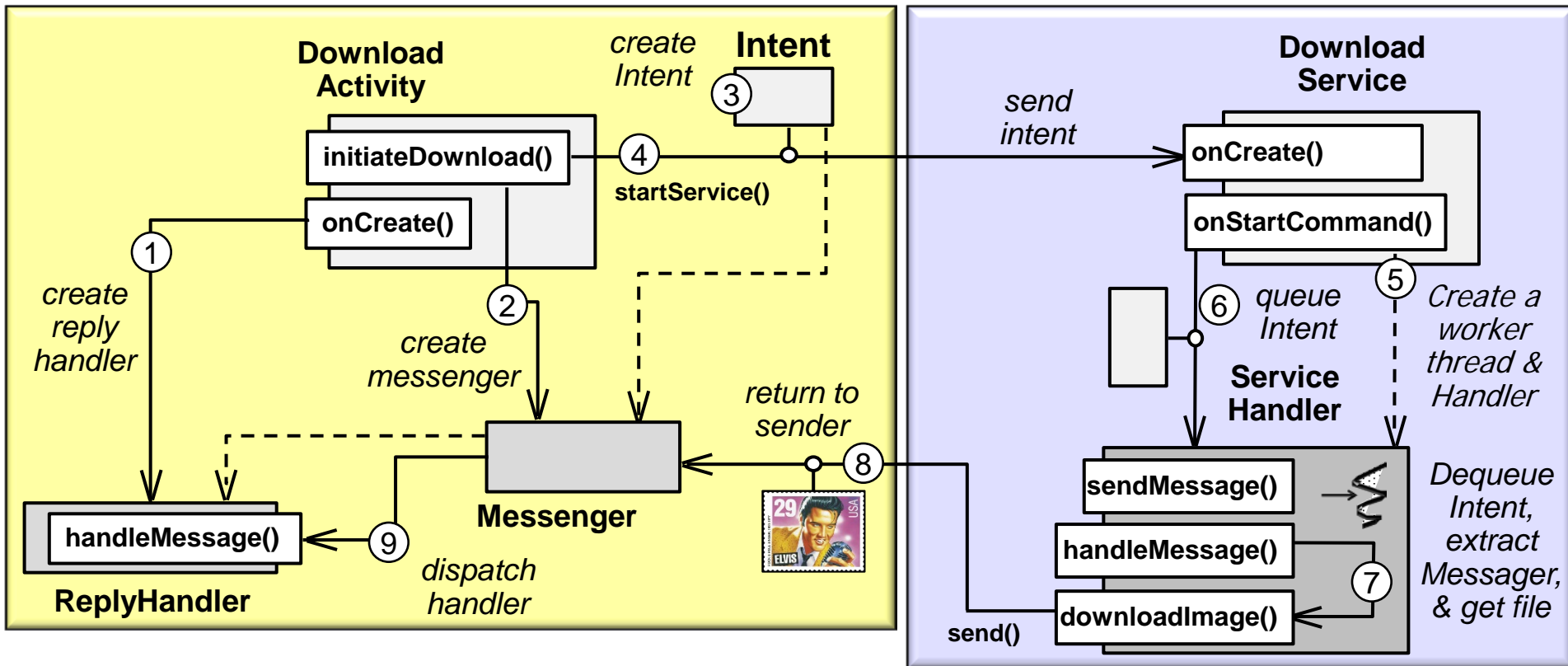
Institute for Software
Integrated Systems

Vanderbilt University
Nashville, Tennessee, USA



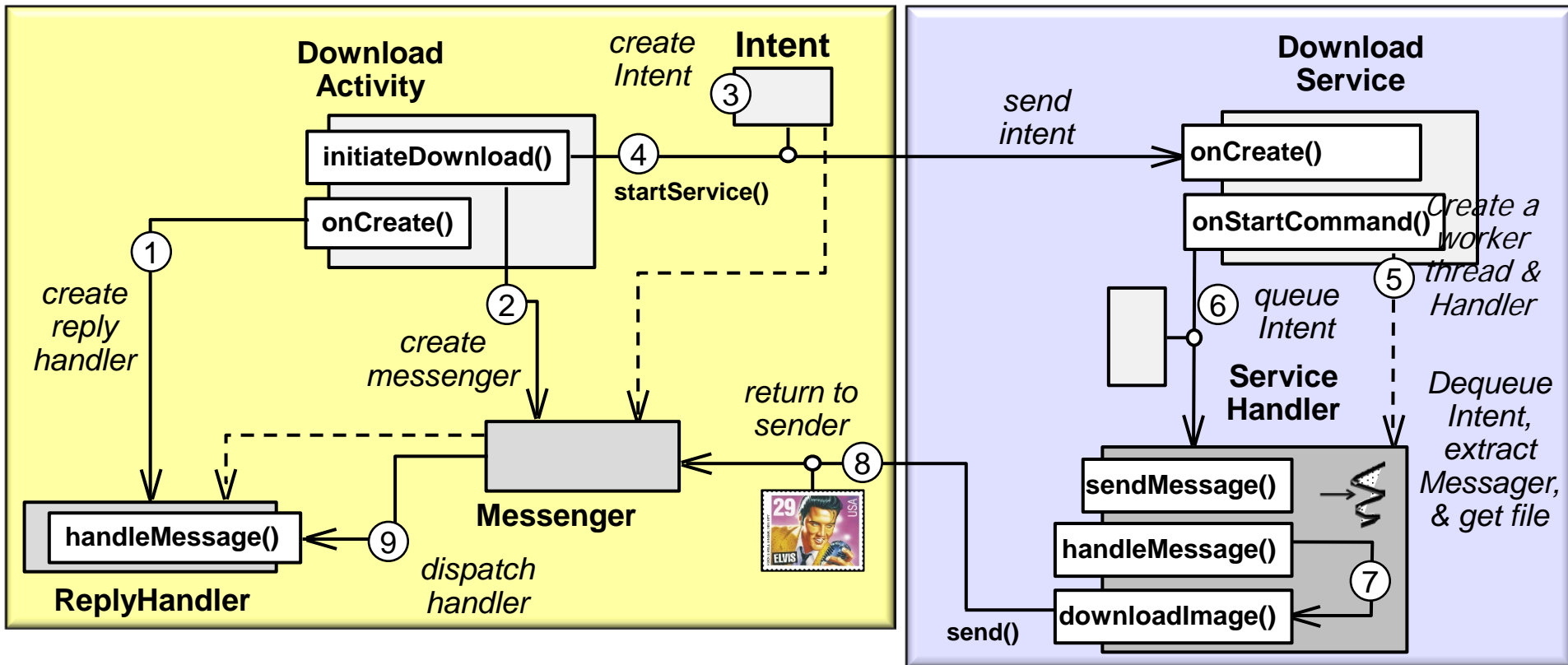
Learning Objectives in this Part of the Module

- Understand how to use a Messenger to communicate from the DownloadService back to the DownloadActivity



Learning Objectives in this Part of the Module

- Understand how to use a Messenger to communicate from the DownloadService back to the DownloadActivity
- Messengers provide a flexible mechanism for intra- & inter-process communication between Activities & Services in Android

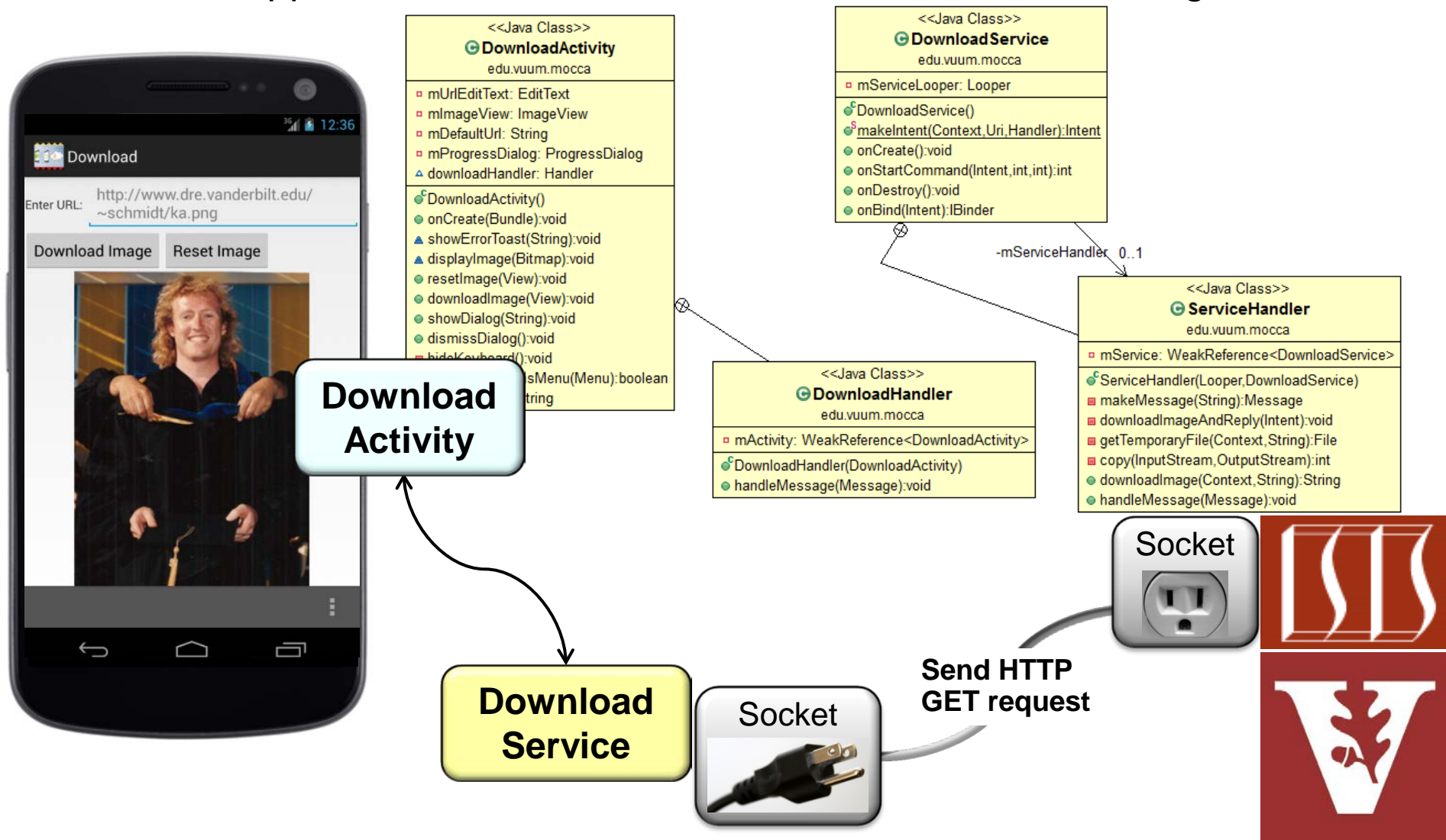


We'll focus on using Messengers to send Messages from Started Services to Activities

Using Messenger in the Download Application

Using Messenger in the Download Application

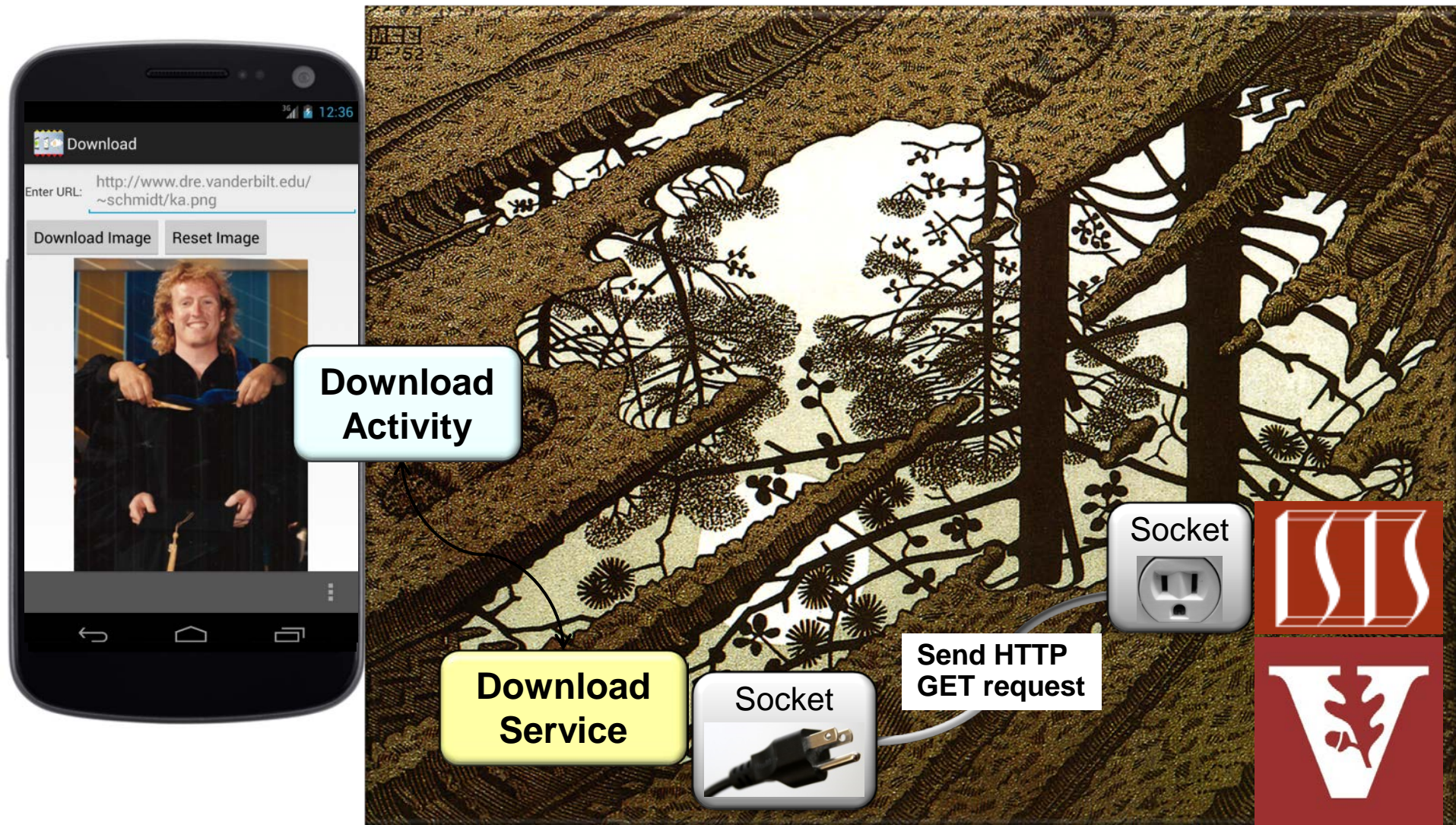
- Download Application uses a Started Service, Intent, & Messenger



See earlier parts on "Programming Started Services with Intents & Messengers"

Using Messenger in the Download Application

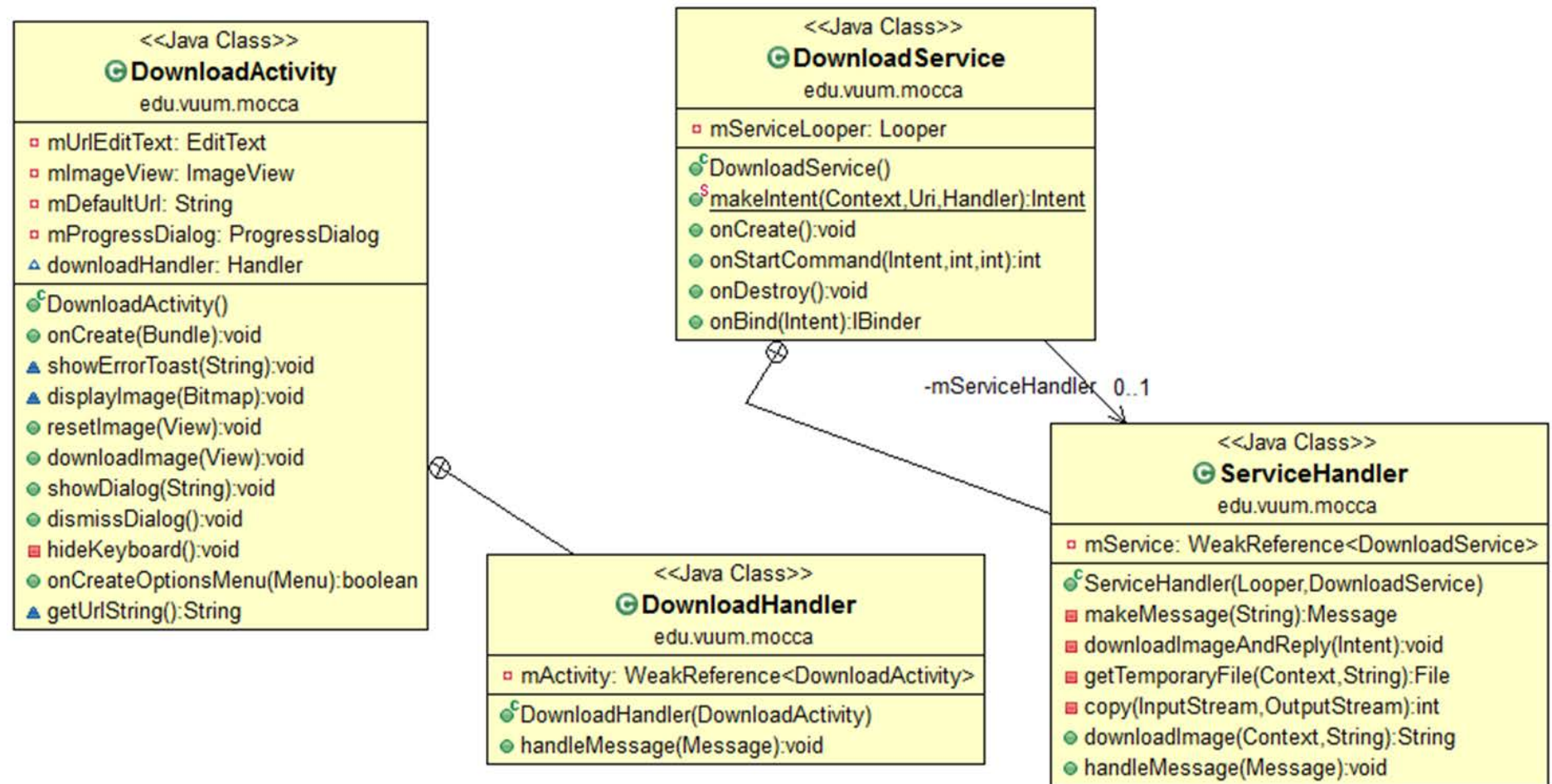
- Download Application uses a Started Service, Intent, & Messenger



Once again, we explore the Download application from multiple perspectives

Using Messenger in the Download Application

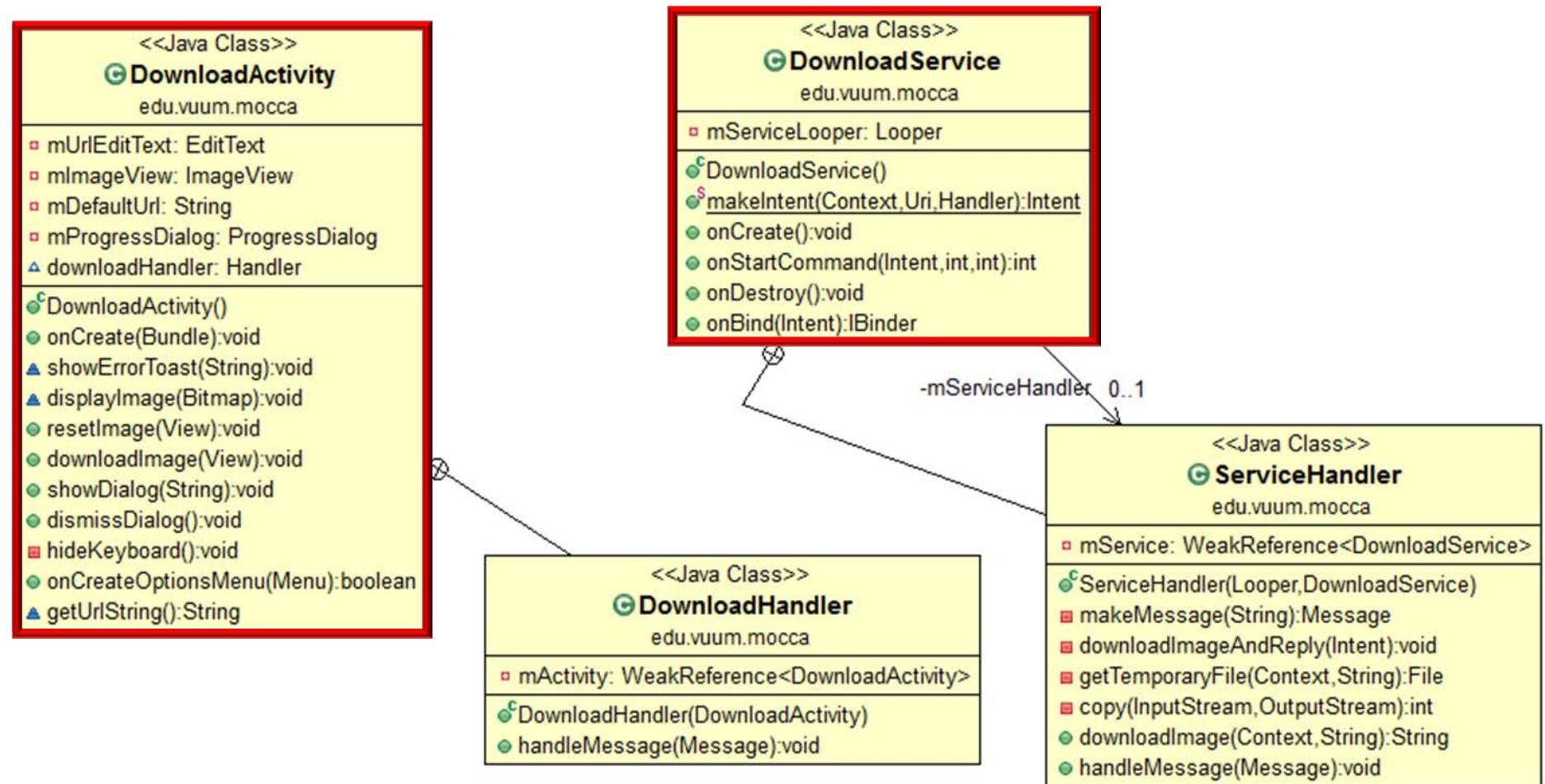
- Download Application uses a Started Service, Intent, & Messenger



github.com/douglasraigschmidt/POSA-14/tree/master/ex/DownloadApplication

Using Messenger in the Download Application

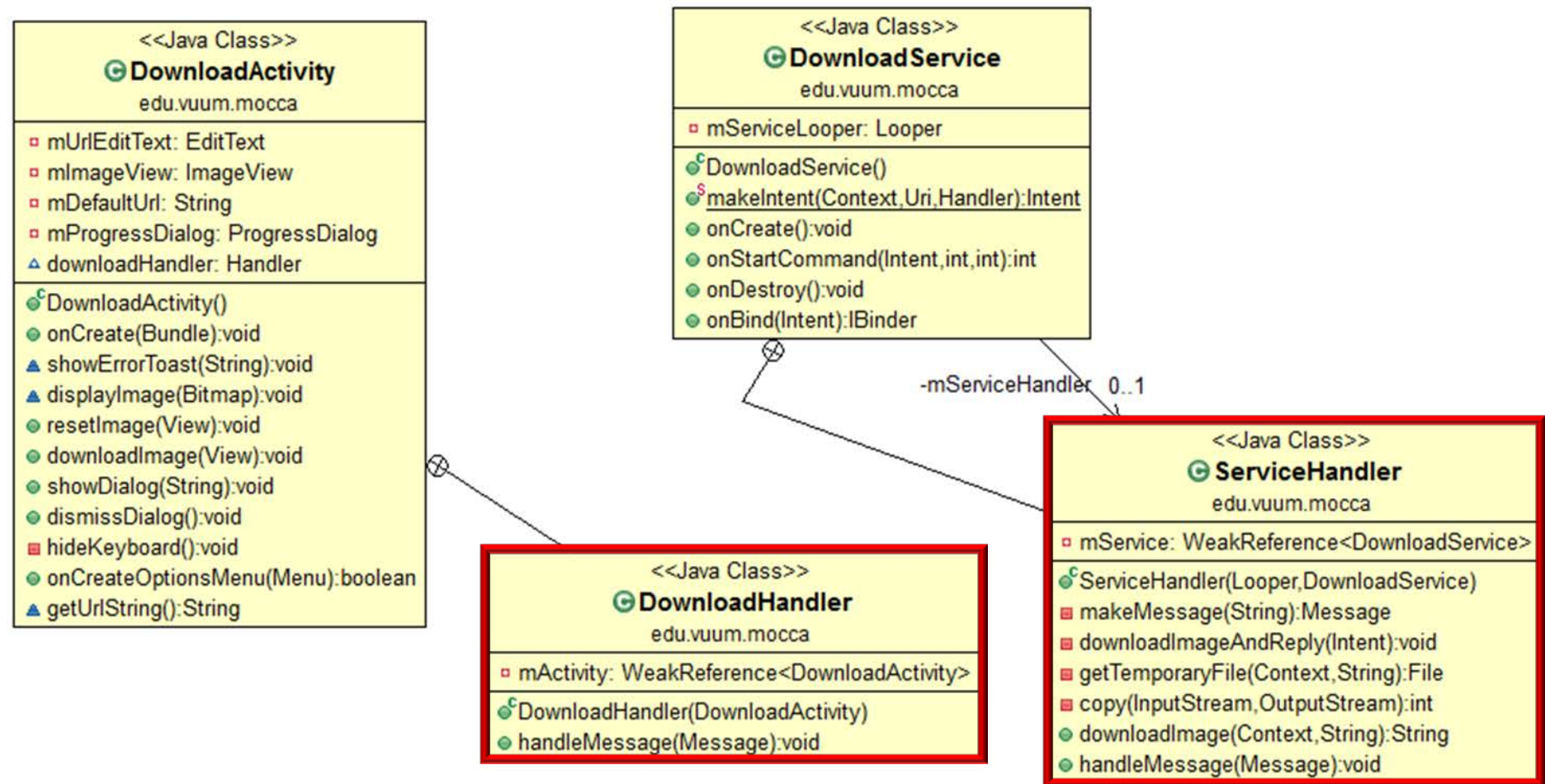
- Download Application uses a Started Service, Intent, & Messenger



DownloadActivity & DownloadService
are key architectural components

Using Messenger in the Download Application

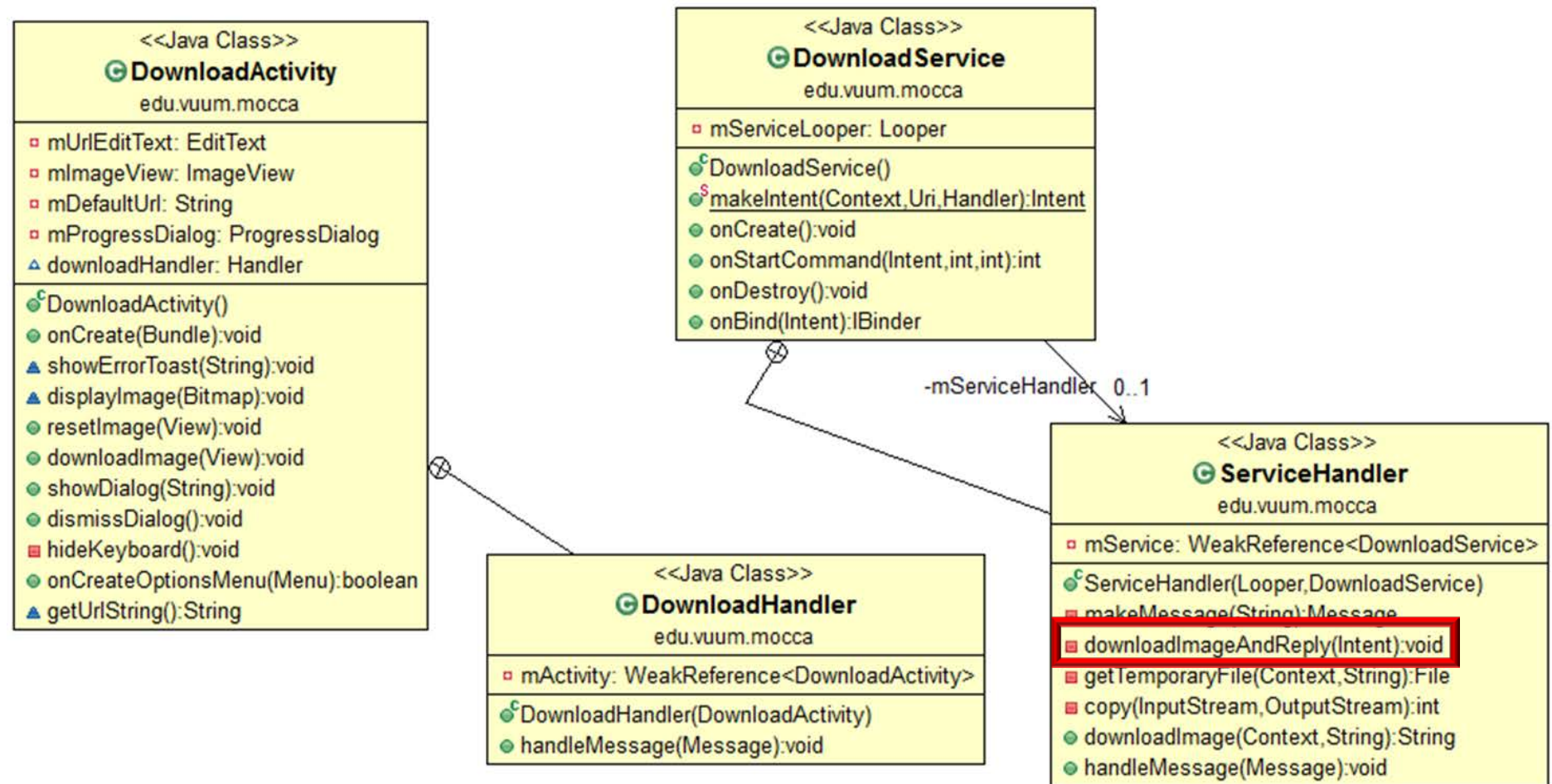
- Download Application uses a Started Service, Intent, & Messenger



DownloadHandler & ServiceHandler
implement various patterns & idioms

Using Messenger in the Download Application

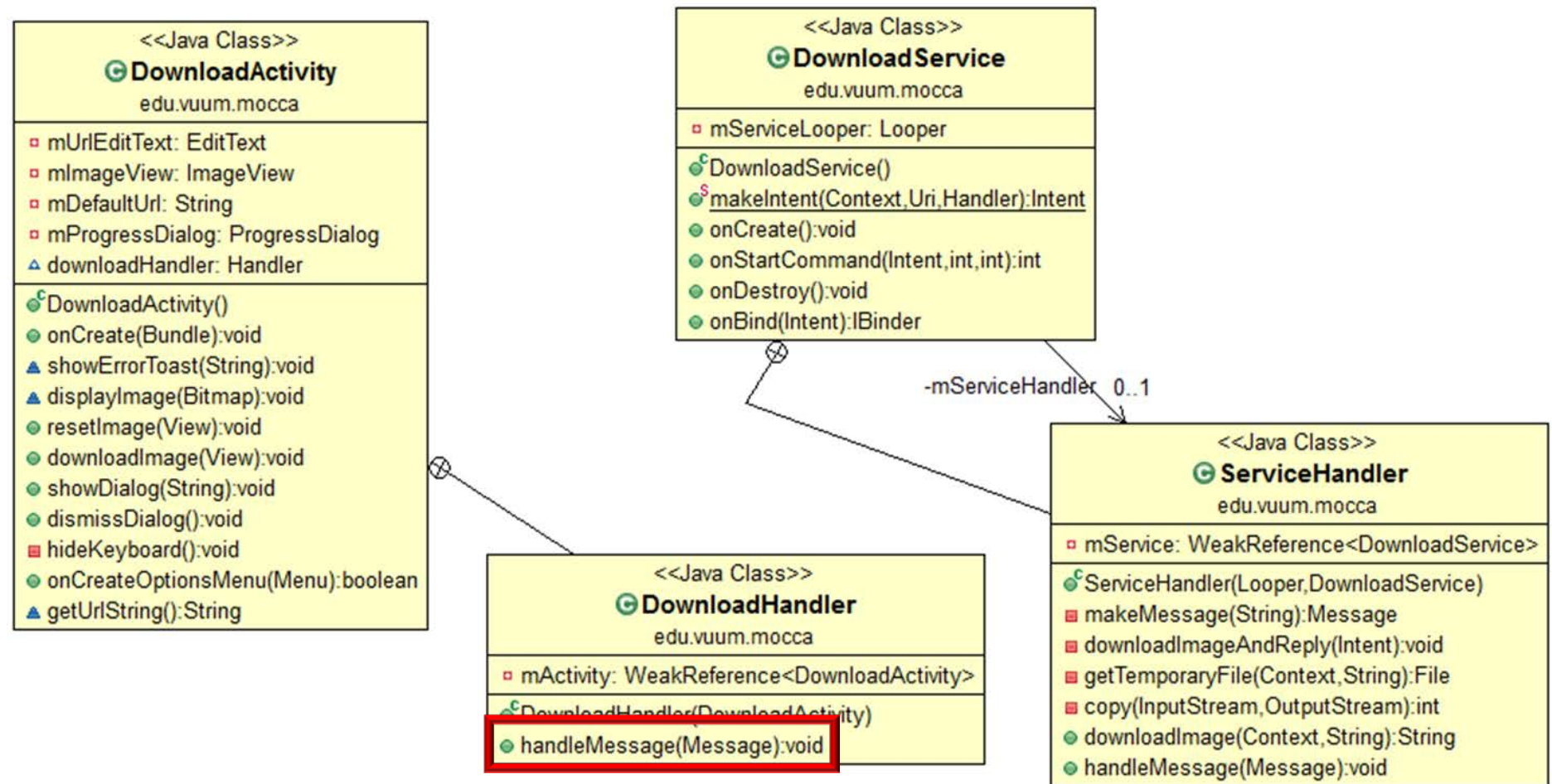
- Download Application uses a Started Service, Intent, & Messenger



Downloads the image from remote server & sends pathname to Activity

Using Messenger in the Download Application

- Download Application uses a Started Service, Intent, & Messenger



Receives the image pathname from Service & displays it to the user

Using Messenger in DownloadActivity

Using Messenger in the DownloadActivity

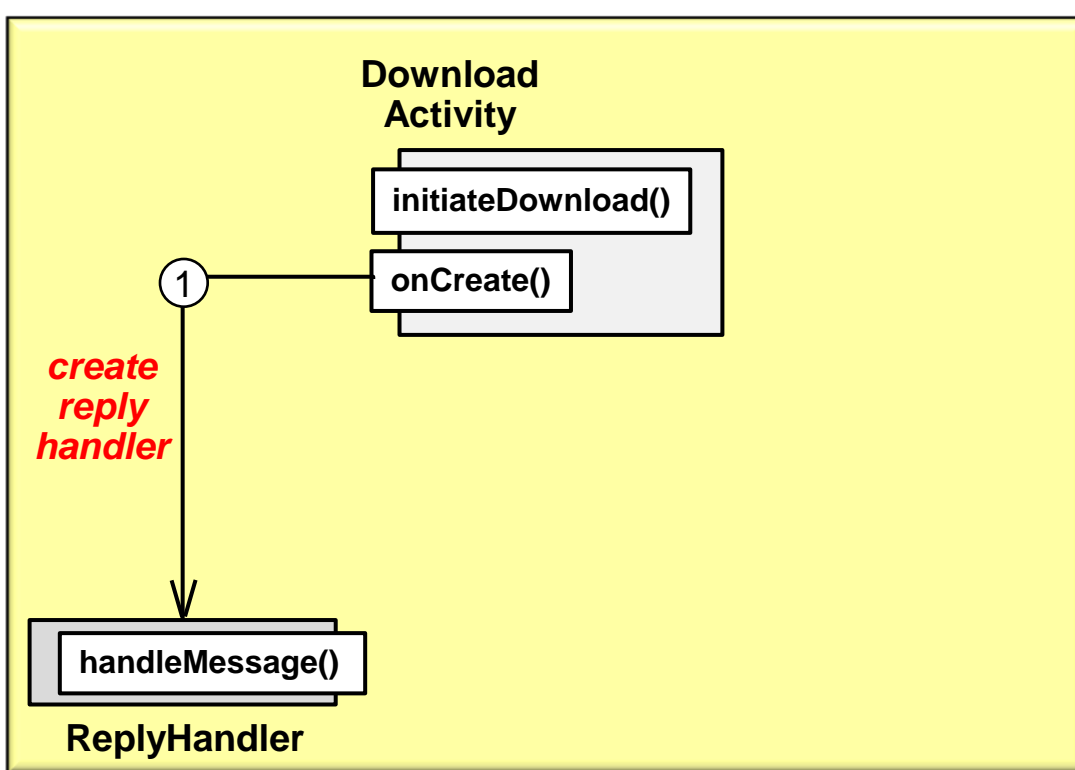
- DownloadActivity performs several steps



**Download
Activity**

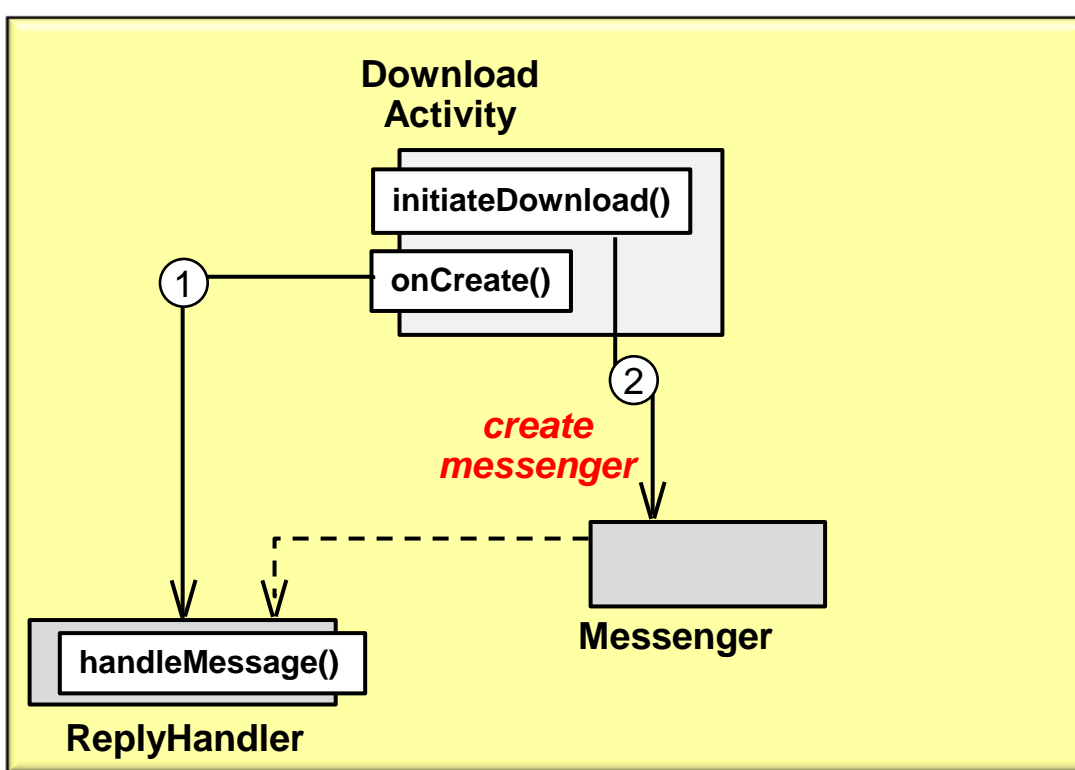
Using Messenger in the DownloadActivity

- DownloadActivity performs several steps
 - Creates a DownloadHandler & a Messenger that encapsulates the Handler



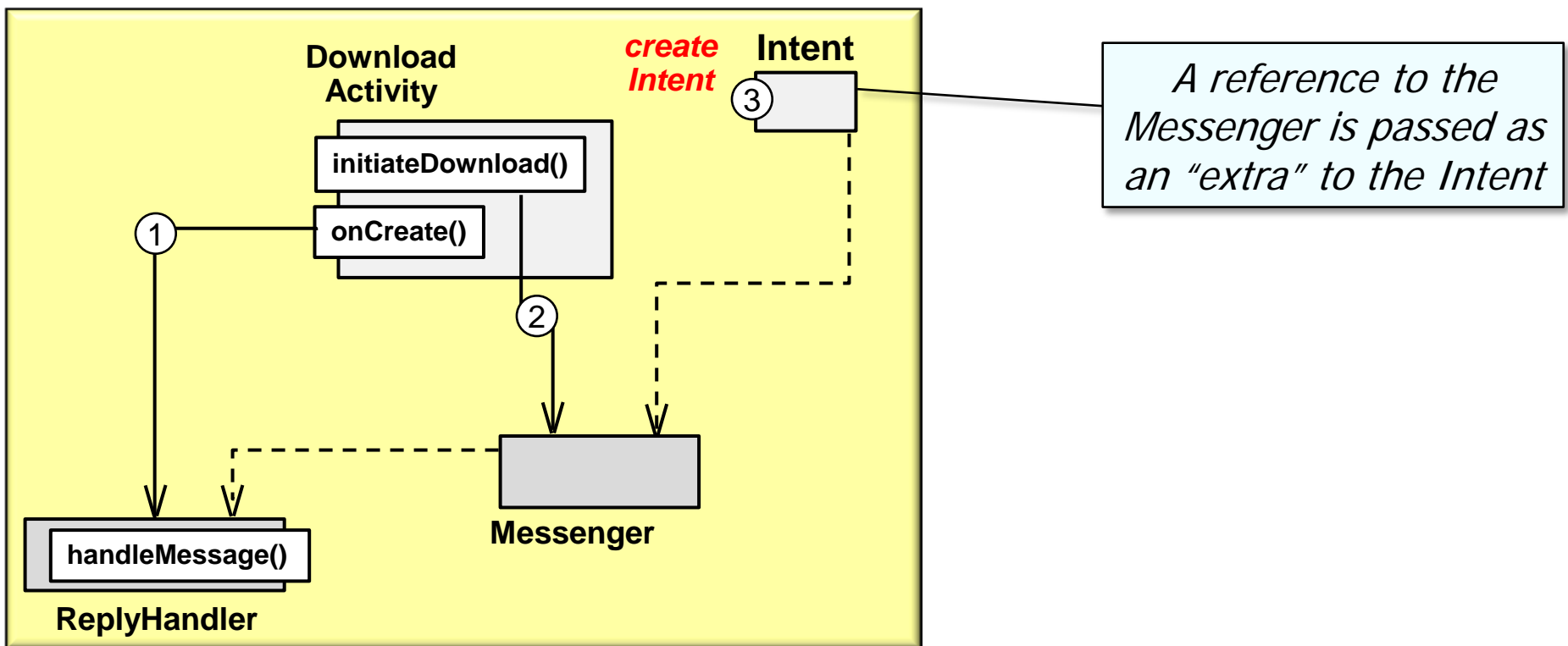
Using Messenger in the DownloadActivity

- DownloadActivity performs several steps
 - Creates a DownloadHandler & a Messenger that encapsulates the Handler



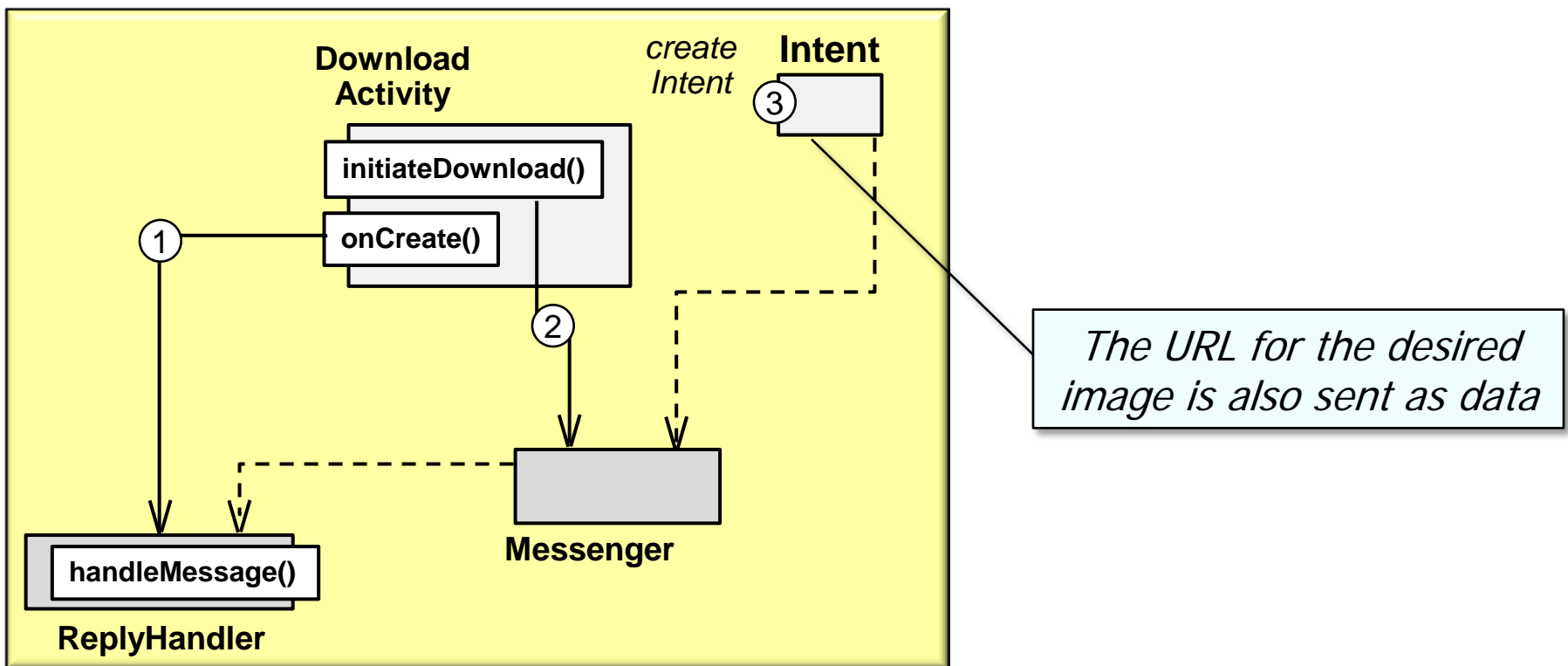
Using Messenger in the DownloadActivity

- DownloadActivity performs several steps
 - Creates a DownloadHandler & a Messenger that encapsulates the Handler
 - Creates an Intent used to start DownloadService



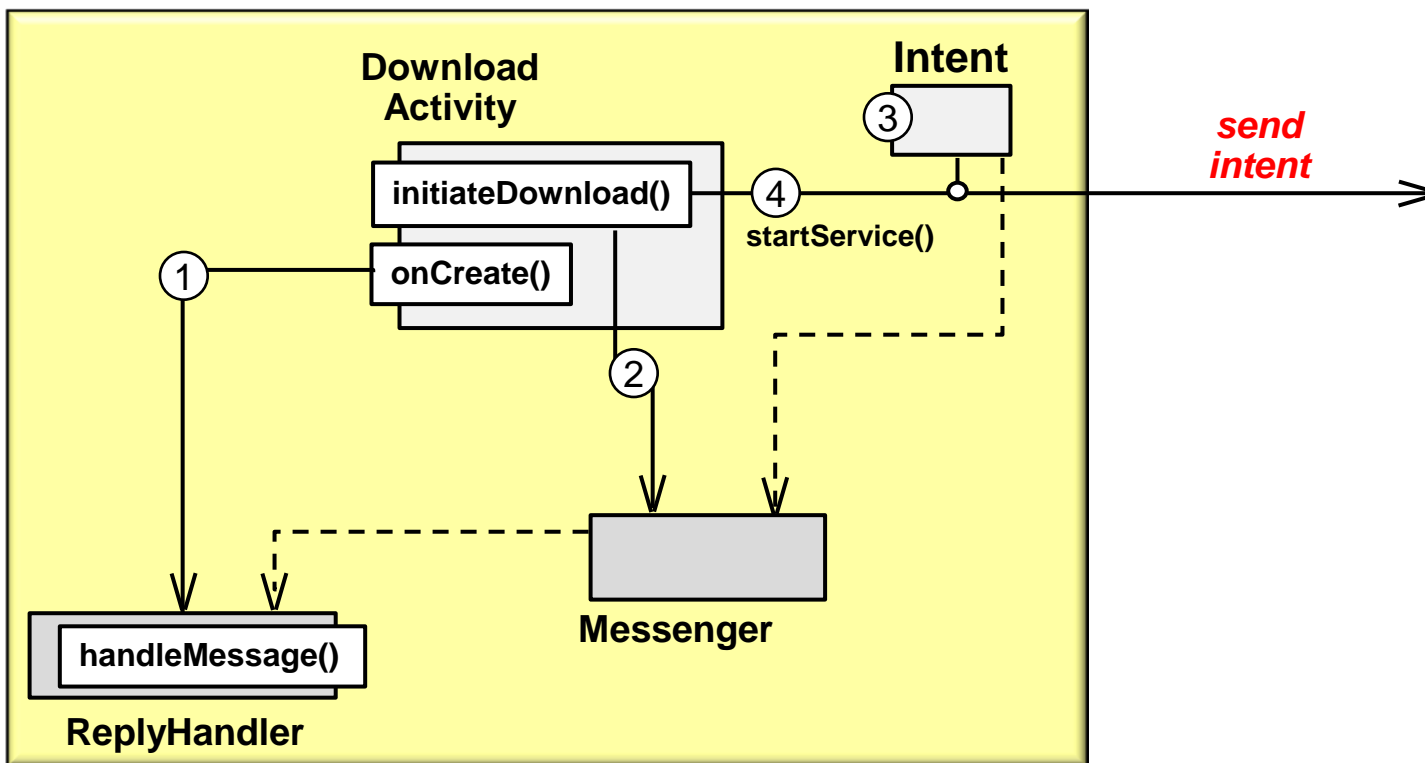
Using Messenger in the DownloadActivity

- DownloadActivity performs several steps
 - Creates a DownloadHandler & a Messenger that encapsulates the Handler
 - Creates an Intent used to start DownloadService



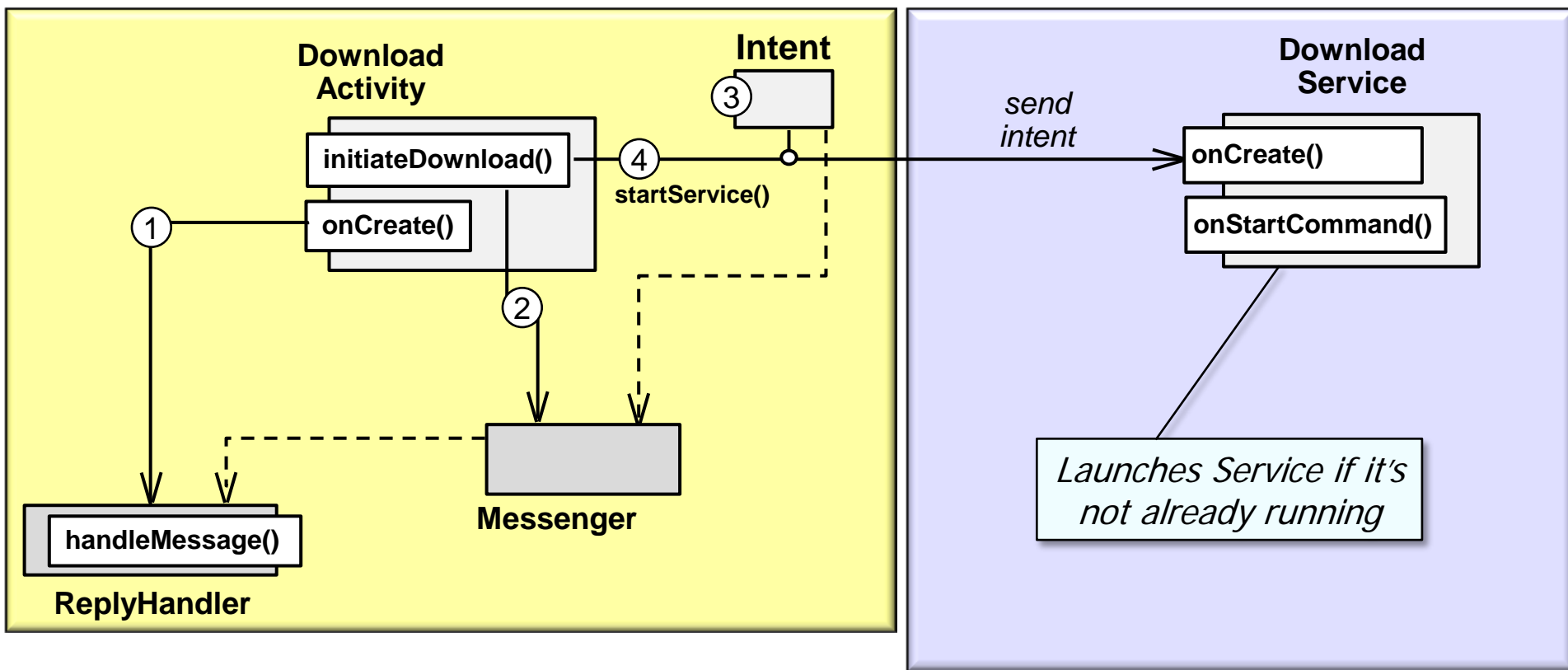
Using Messenger in the DownloadActivity

- DownloadActivity performs several steps
 - Creates a DownloadHandler & a Messenger that encapsulates the Handler
 - Creates an Intent used to start DownloadService
 - Calls startService(), passing the Intent command as a parameter



Using Messenger in the DownloadActivity

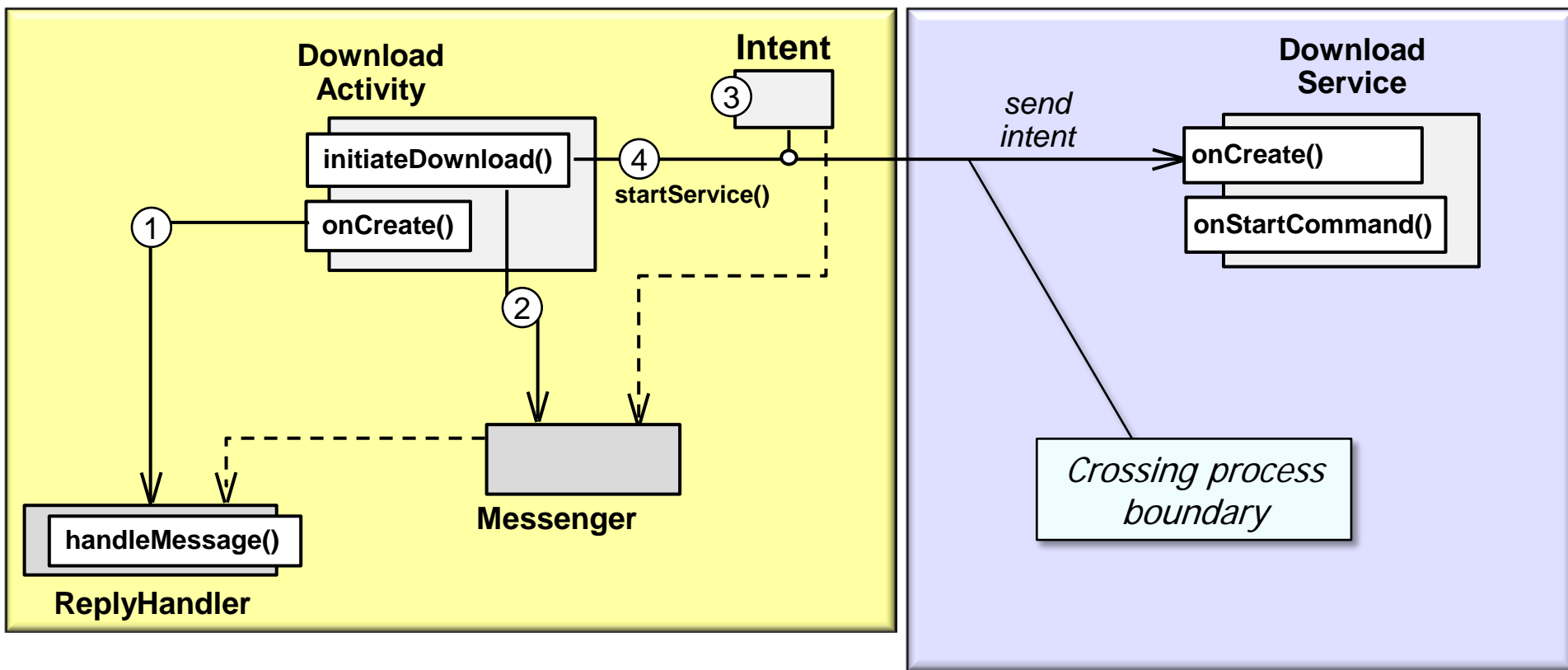
- DownloadActivity performs several steps
 - Creates a DownloadHandler & a Messenger that encapsulates the Handler
 - Creates an Intent used to start DownloadService
 - Calls startService(), passing the Intent command as a parameter



Using Messenger in DownloadService

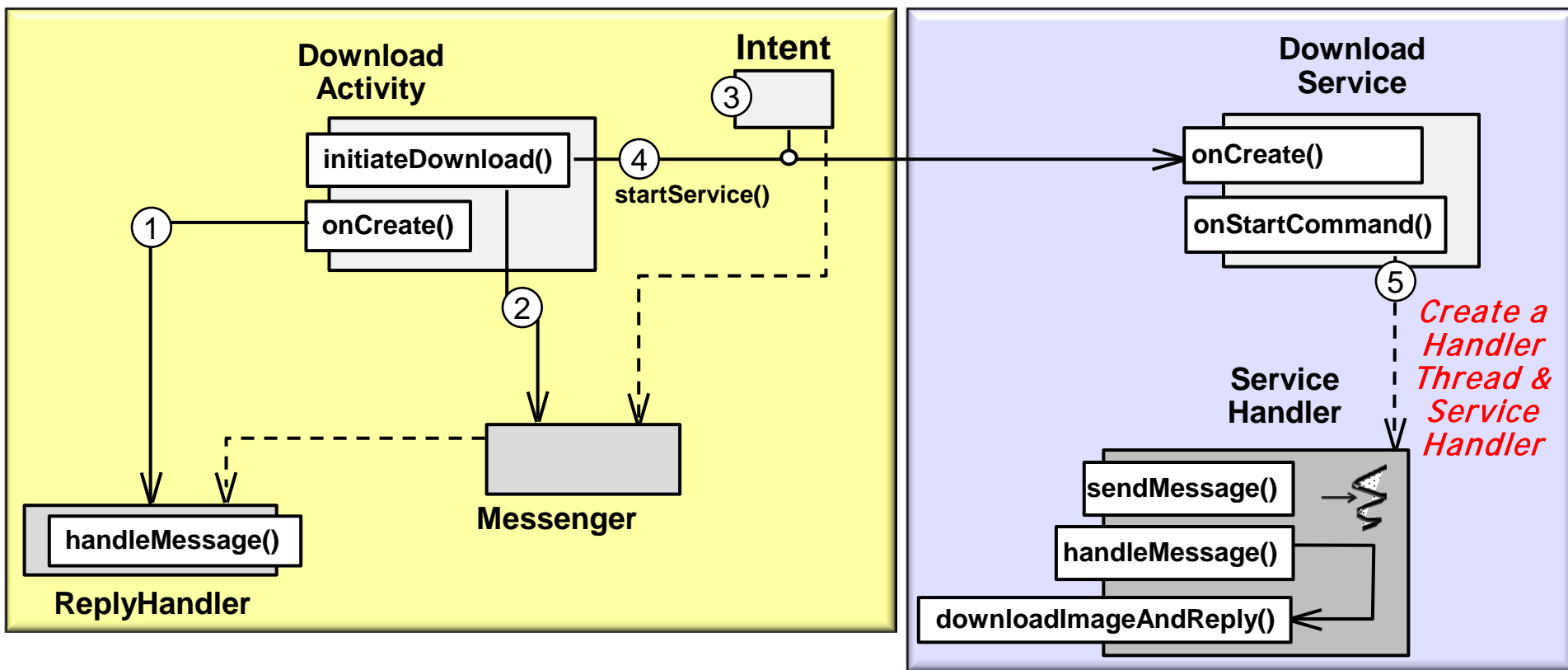
Using Messenger in the DownloadService

- Download Service also performs several steps when it receives an Intent



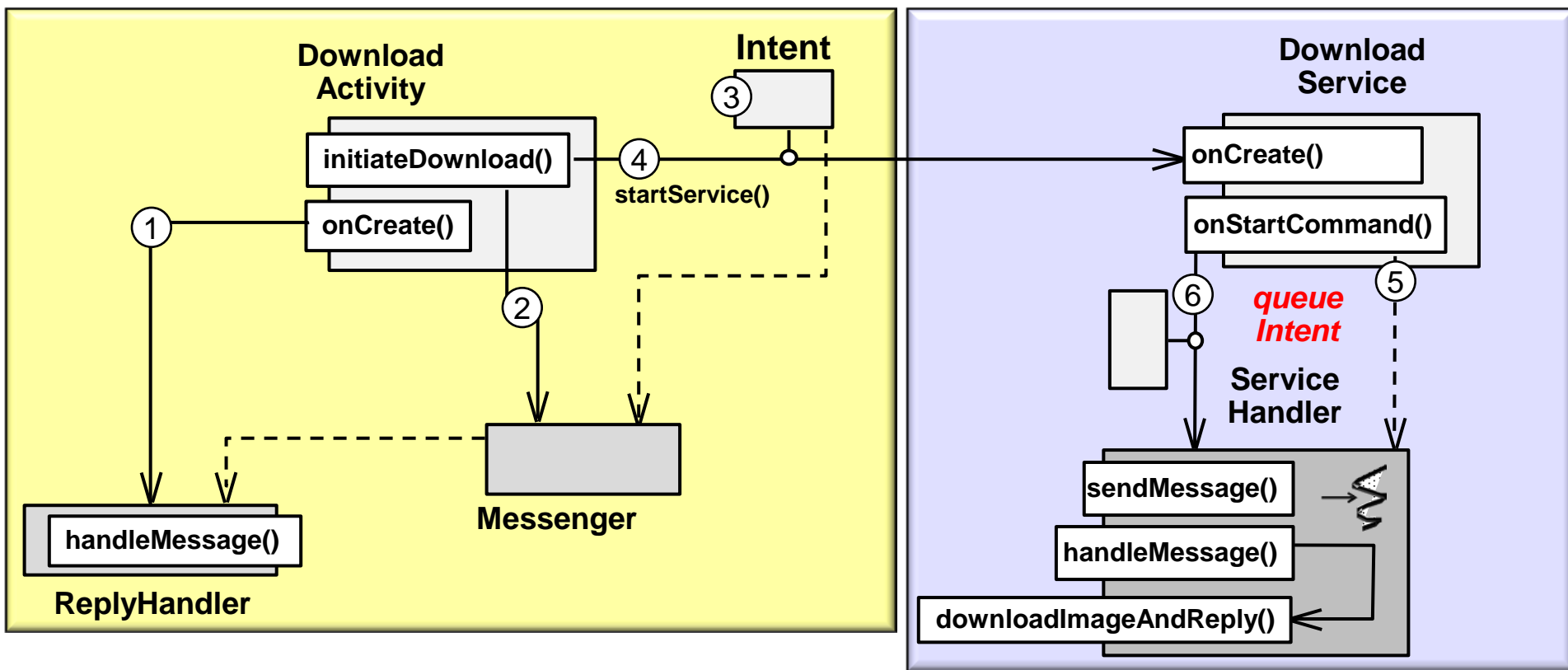
Using Messenger in the DownloadService

- Download Service also performs several steps when it receives an Intent
 - Creates a Handler Thread & Service Handler



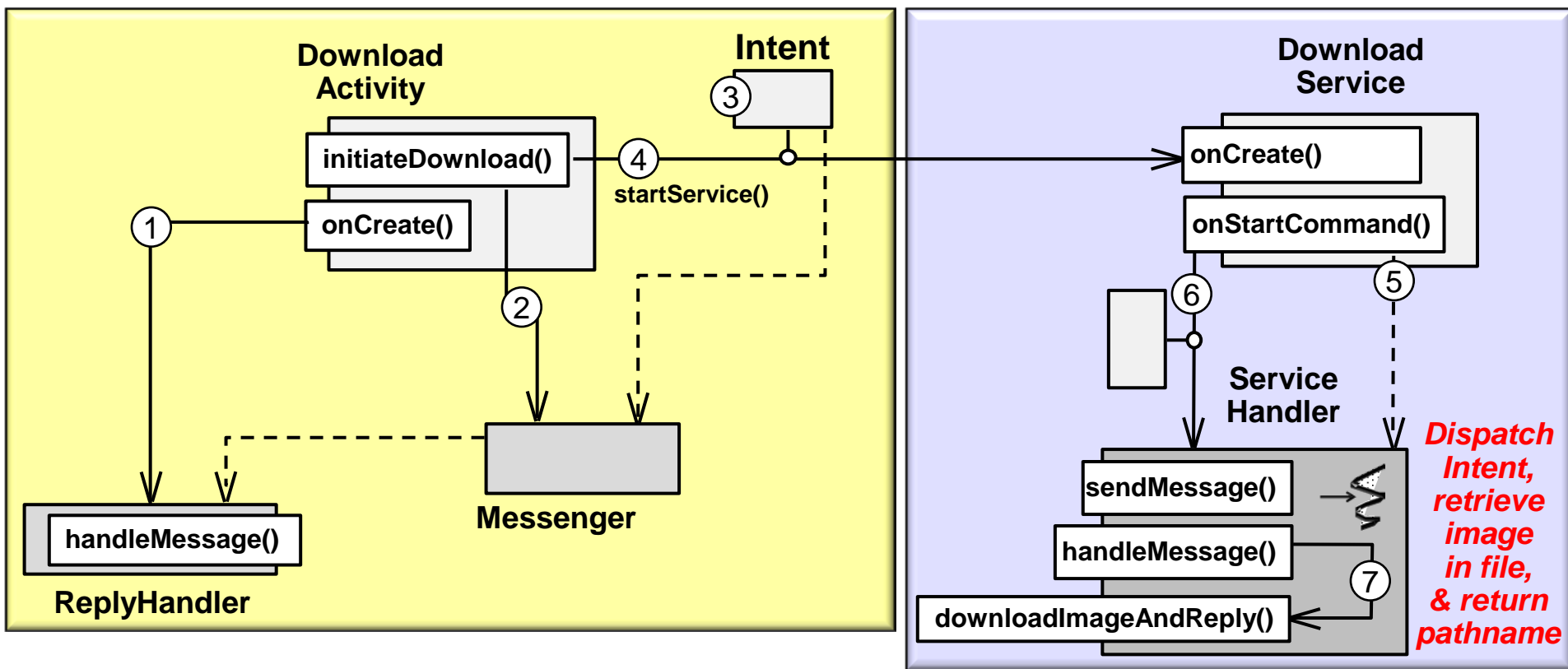
Using Messenger in the DownloadService

- Download Service also performs several steps when it receives an Intent
 - Creates a Handler Thread & Service Handler
 - Queues Intent in the Service Handler's MessageQueue



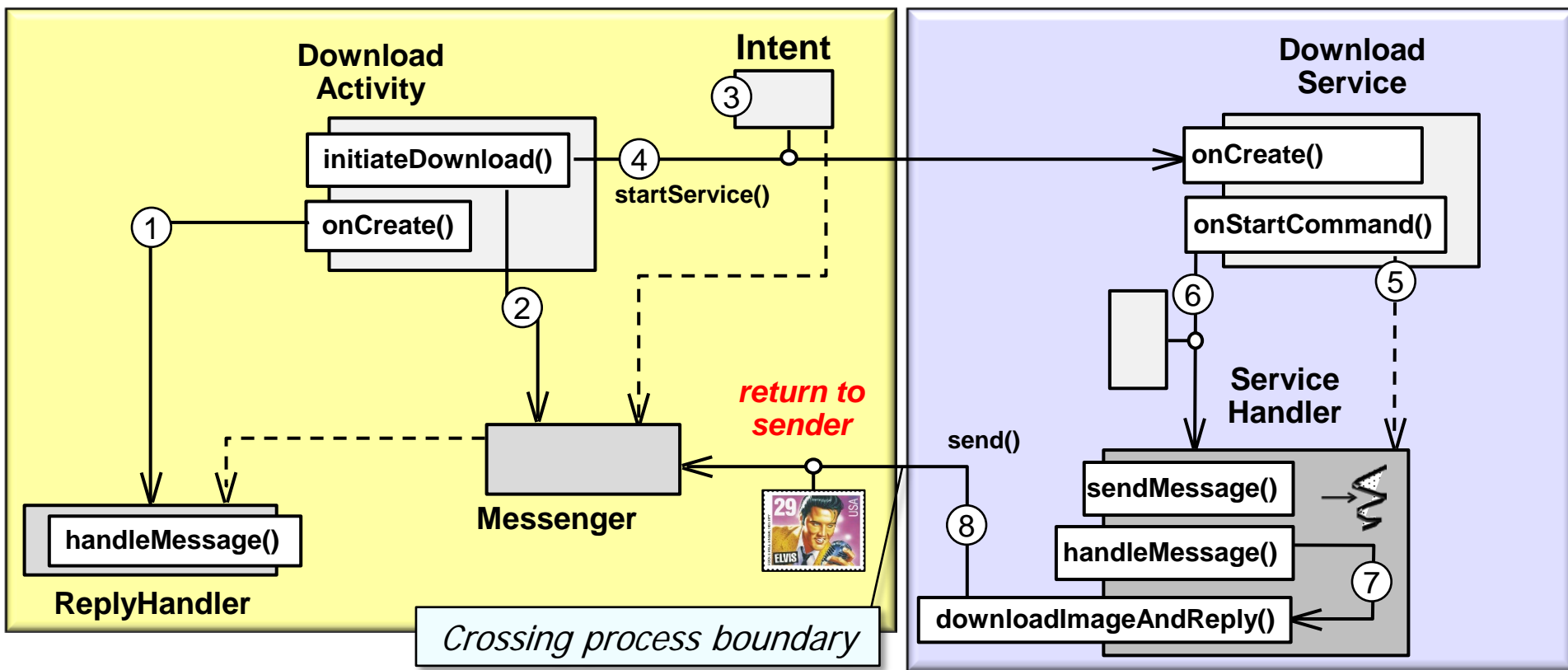
Using Messenger in the DownloadService

- Download Service also performs several steps when it receives an Intent
 - Creates a Handler Thread & Service Handler
 - Queues Intent in the Service Handler's MessageQueue
 - Dispatch Intent to handleMessage(), which calls downloadImageAndReply()



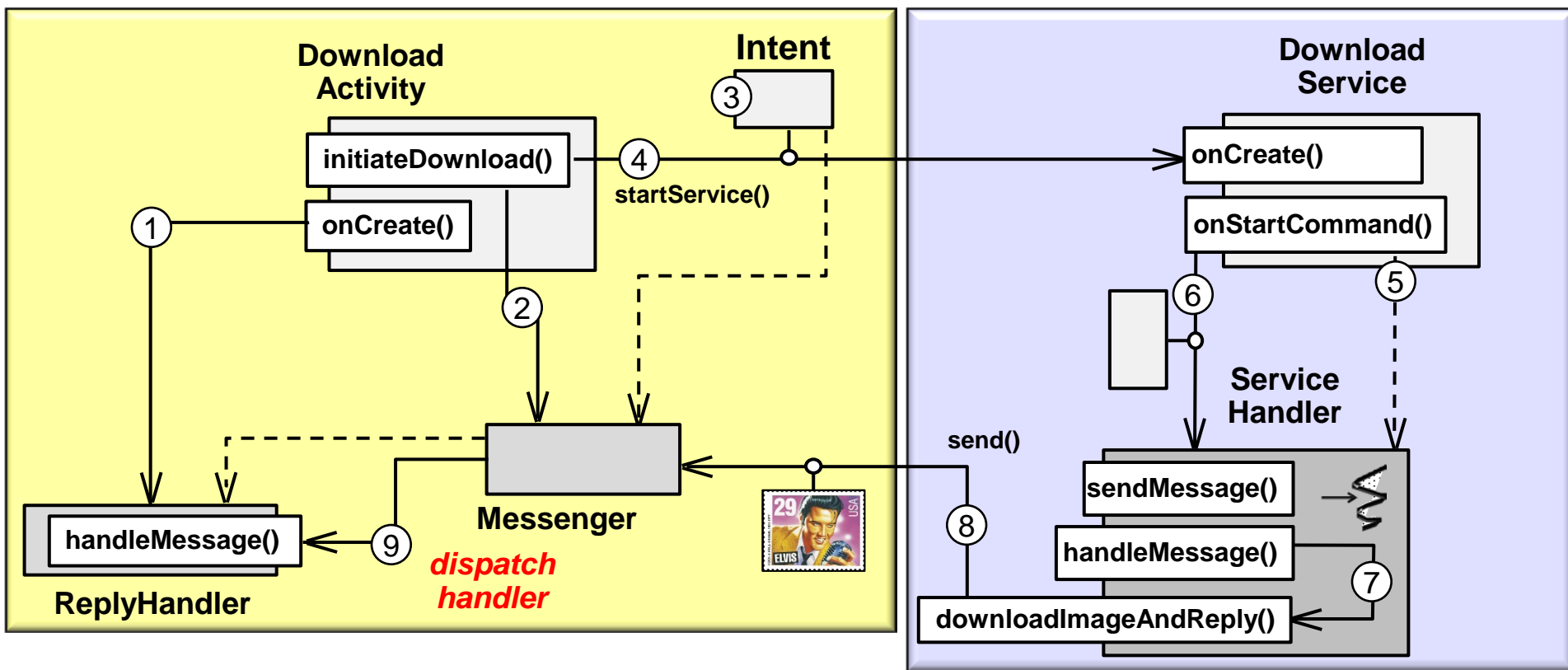
Using Messenger in the DownloadService

- Download Service also performs several steps when it receives an Intent
 - Creates a Handler Thread & Service Handler
 - Queues Intent in the Service Handler's MessageQueue
 - Dispatch Intent to handleMessage(), which calls downloadImageAndReply()



Using Messenger in the DownloadService

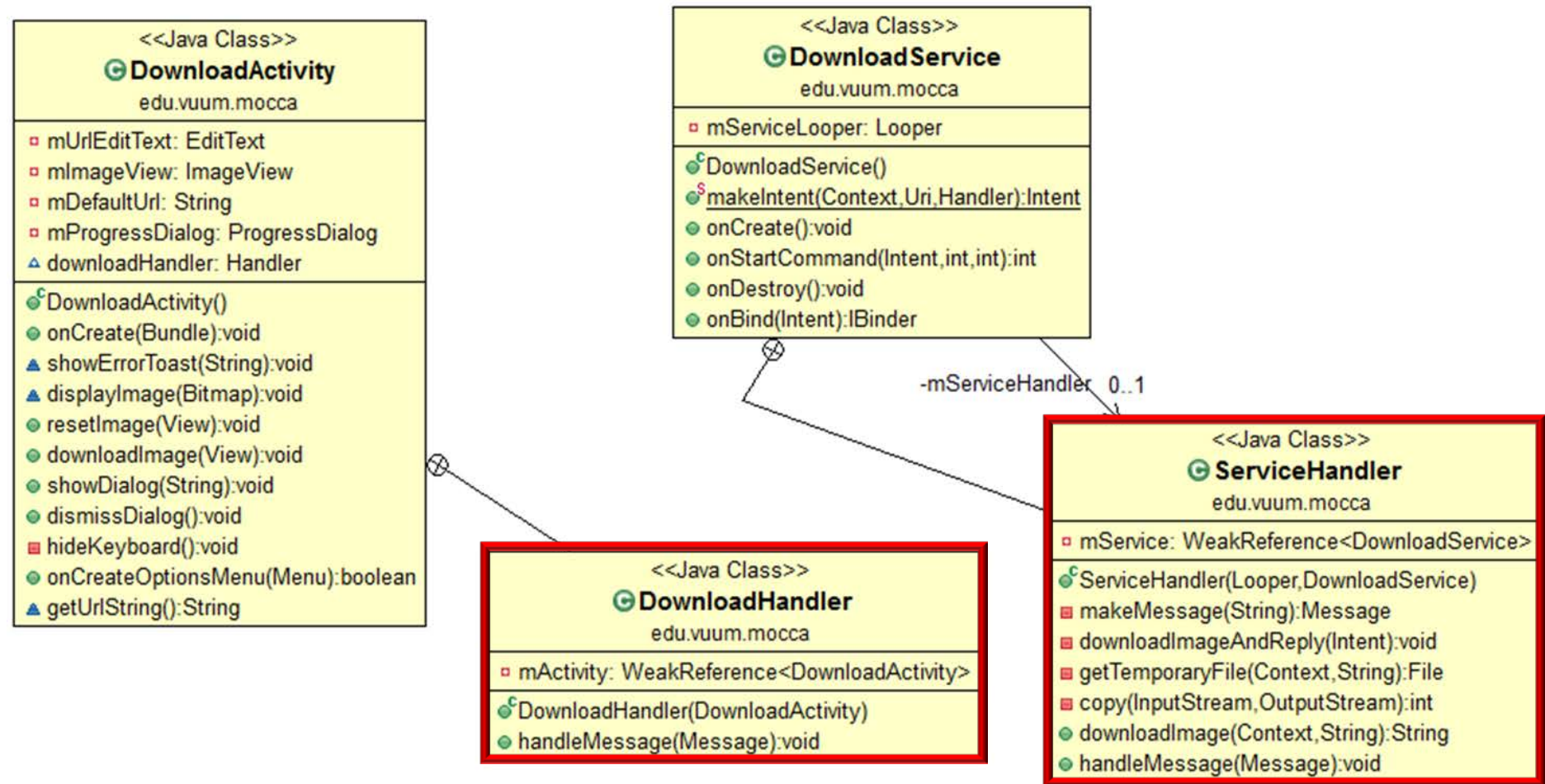
- Download Service also performs several steps when it receives an Intent
 - Creates a Handler Thread & Service Handler
 - Queues Intent in the Service Handler's MessageQueue
 - Dispatch Intent to handleMessage(), which calls downloadImageAndReply()



Patterns & Idioms in the Download Application

Patterns & Idioms in the Download Application

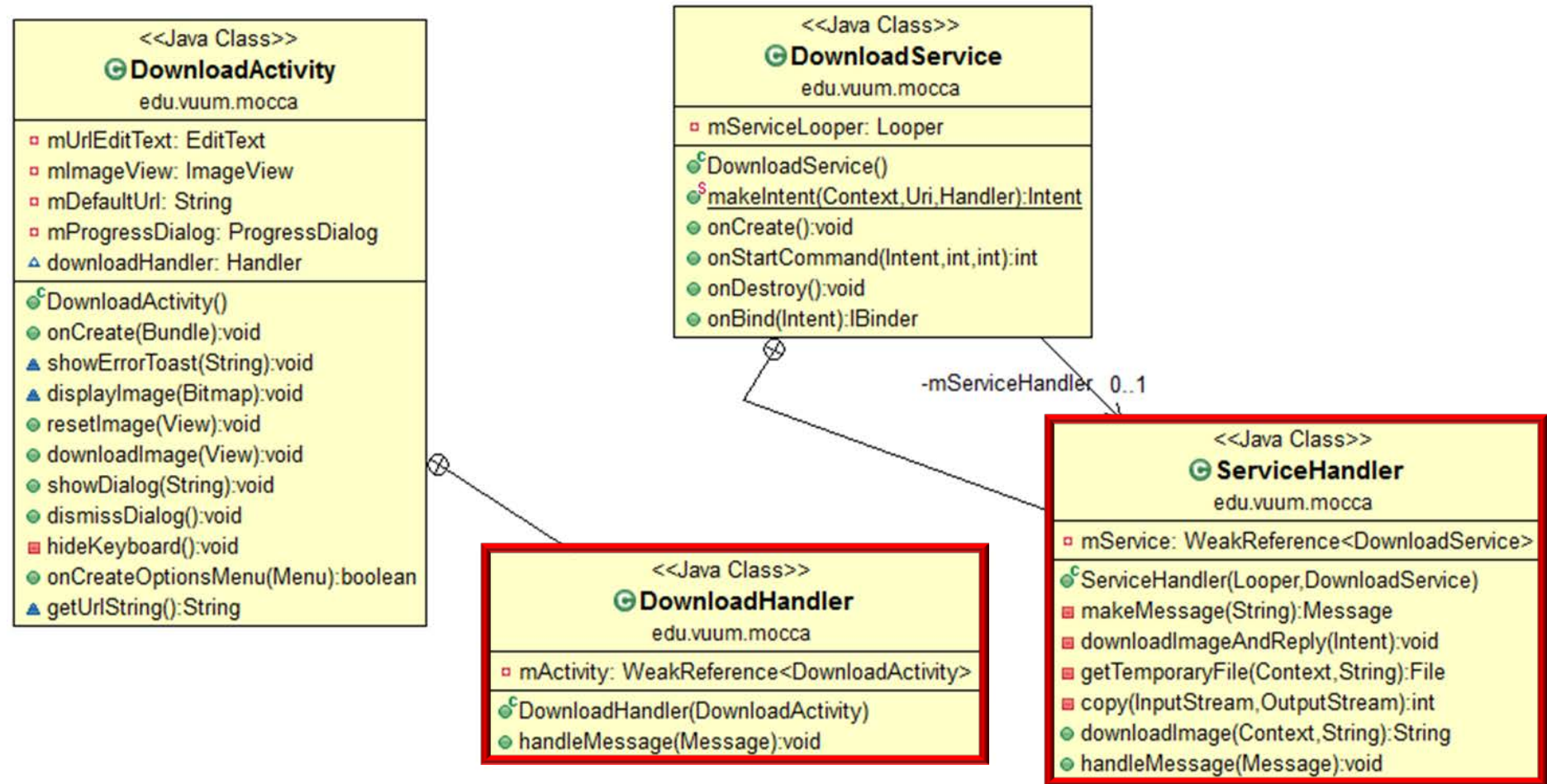
- DownloadHandler & ServiceHandler implement various patterns & idioms



See earlier parts on "Programming Started Services with Intents & Messengers"

Patterns & Idioms in the Download Application

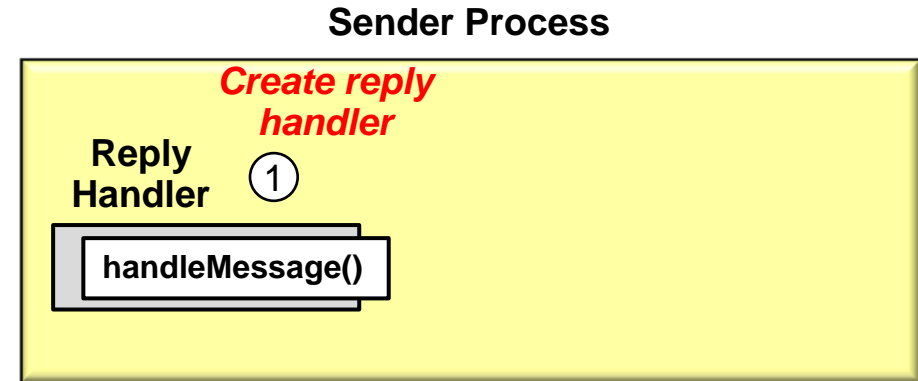
- DownloadHandler & ServiceHandler implement various patterns & idioms
 - e.g., *CommandProcessor* pattern & *Concurrent Service Stopping* idiom



See earlier parts on
"Programming Started Services"

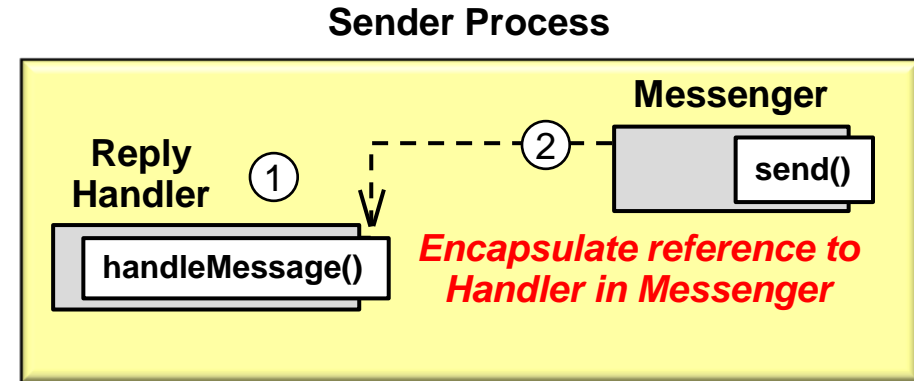
Patterns & Idioms in the Download Application

- DownloadHandler & ServiceHandler implement various patterns & idioms
- The structure & flow of control also applies a common Android Service concurrency idiom



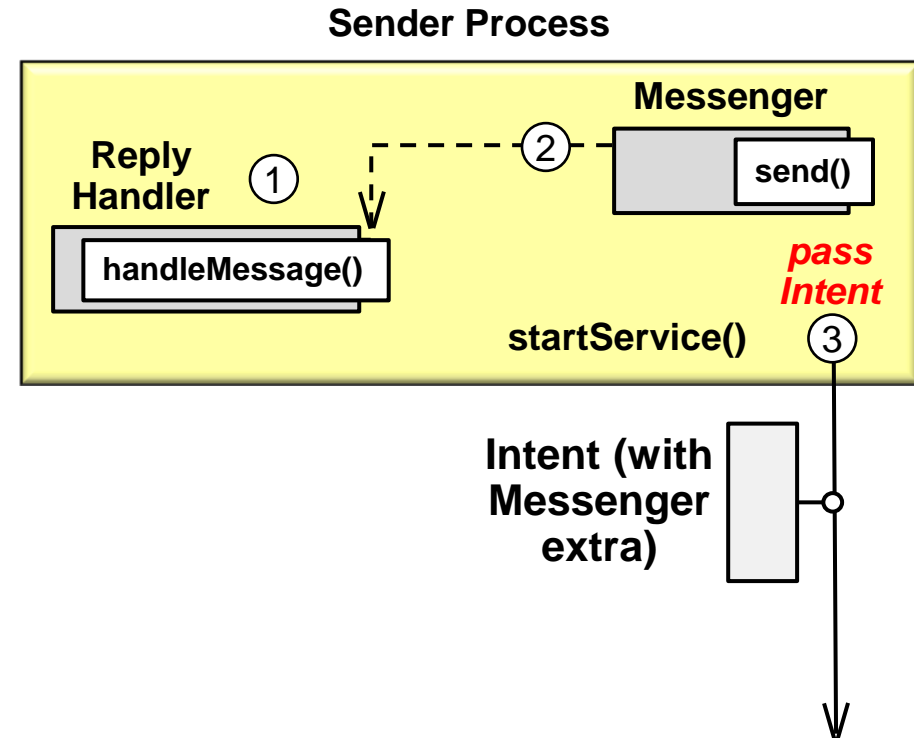
Patterns & Idioms in the Download Application

- DownloadHandler & ServiceHandler implement various patterns & idioms
- The structure & flow of control also applies a common Android Service concurrency idiom



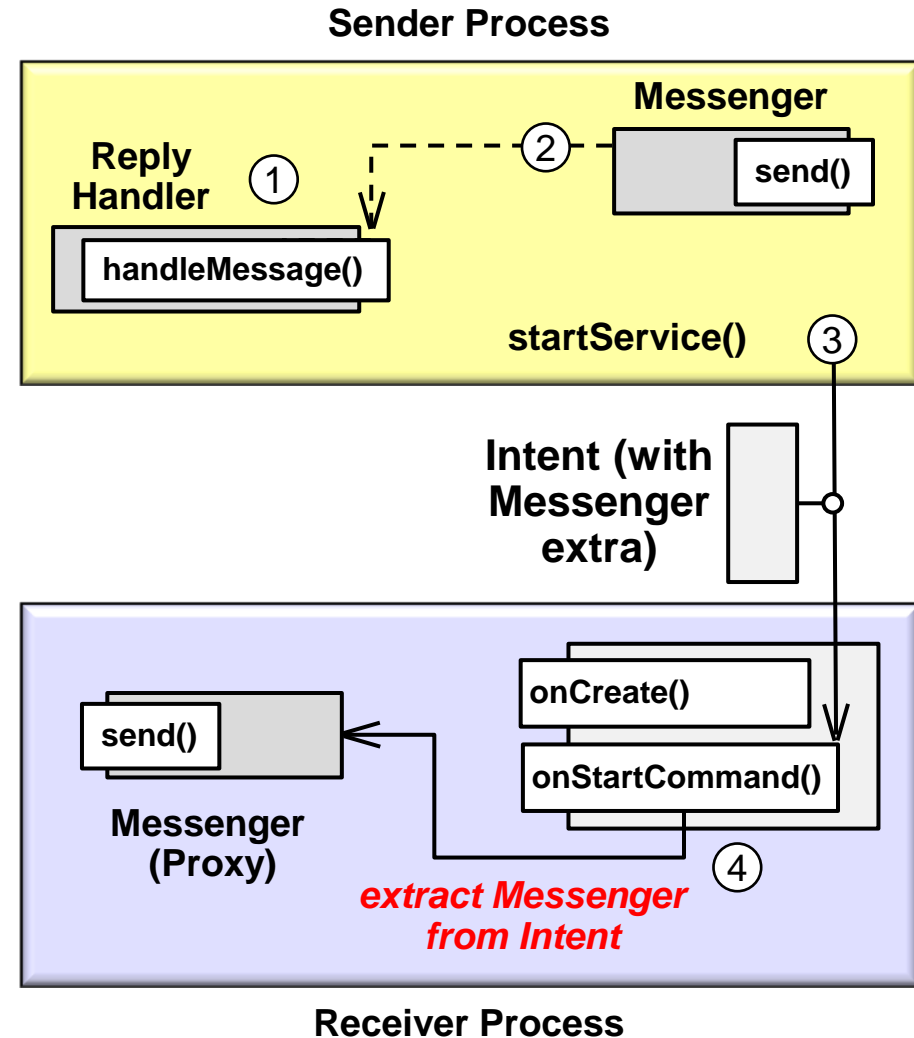
Patterns & Idioms in the Download Application

- DownloadHandler & ServiceHandler implement various patterns & idioms
- The structure & flow of control also applies a common Android Service concurrency idiom



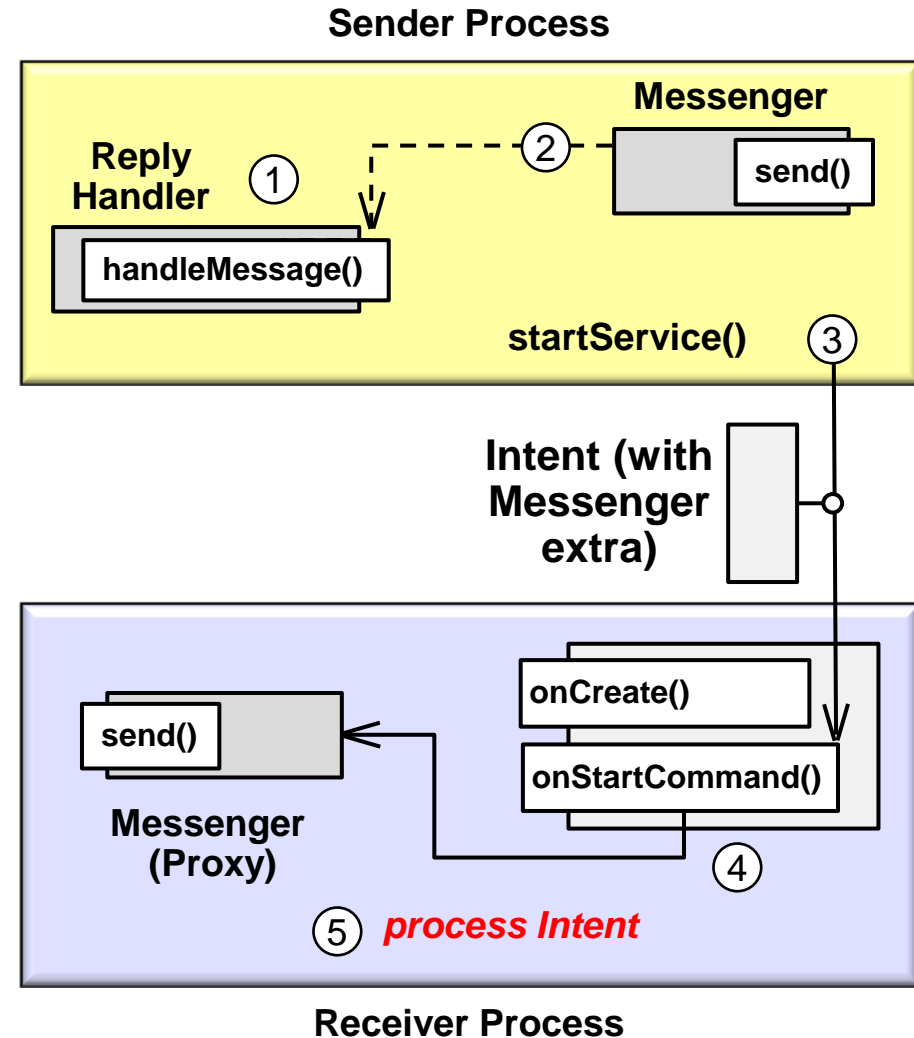
Patterns & Idioms in the Download Application

- DownloadHandler & ServiceHandler implement various patterns & idioms
- The structure & flow of control also applies a common Android Service concurrency idiom



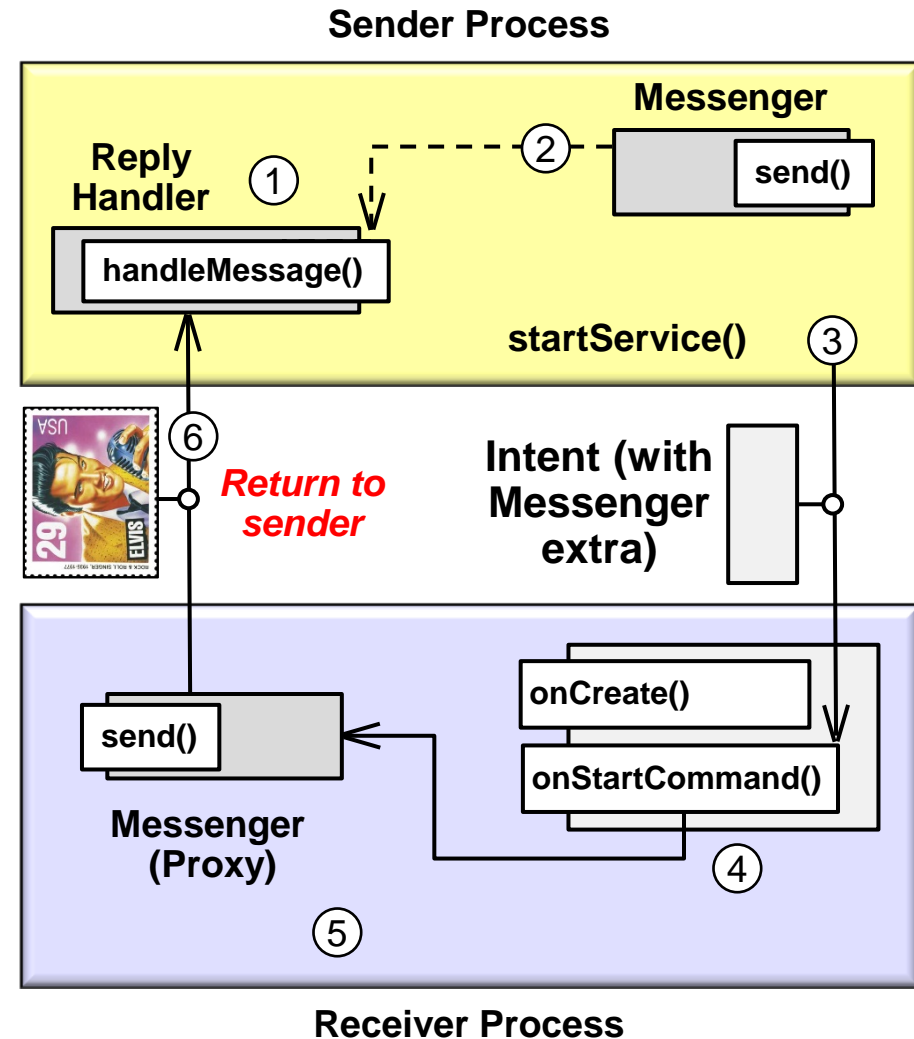
Patterns & Idioms in the Download Application

- DownloadHandler & ServiceHandler implement various patterns & idioms
- The structure & flow of control also applies a common Android Service concurrency idiom



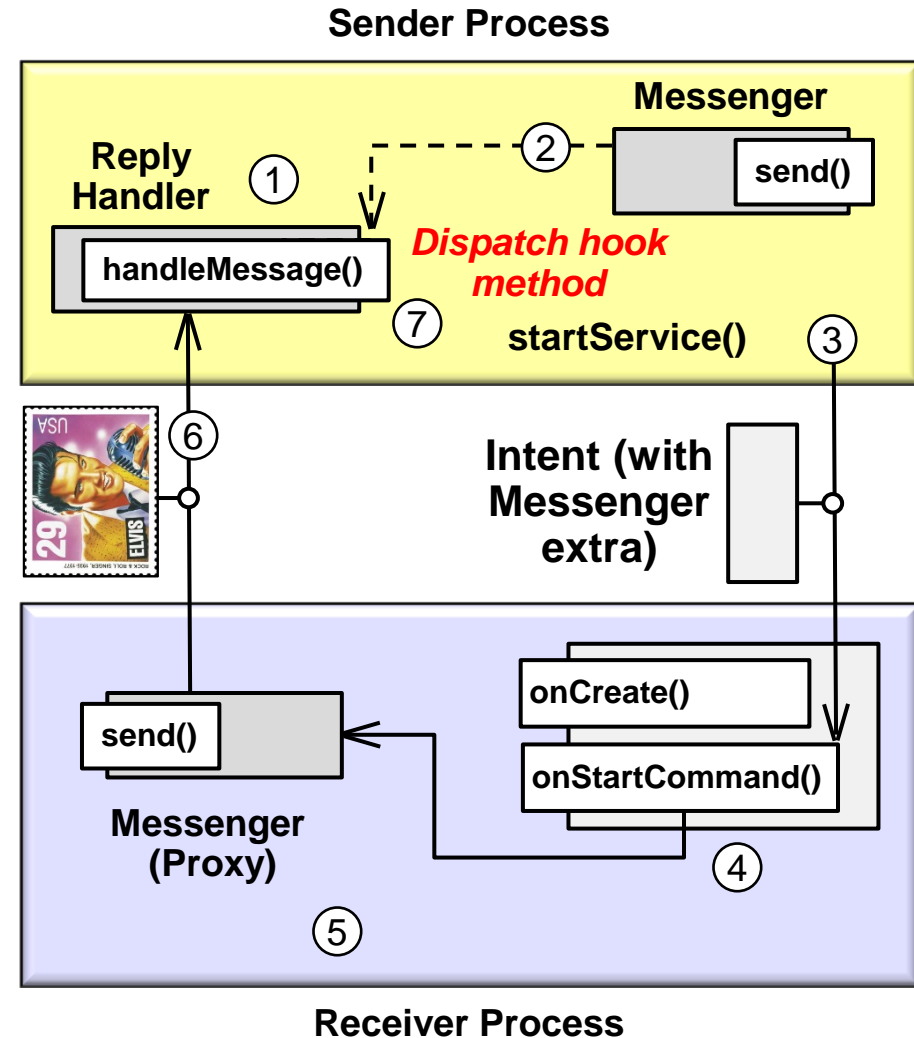
Patterns & Idioms in the Download Application

- DownloadHandler & ServiceHandler implement various patterns & idioms
- The structure & flow of control also applies a common Android Service concurrency idiom



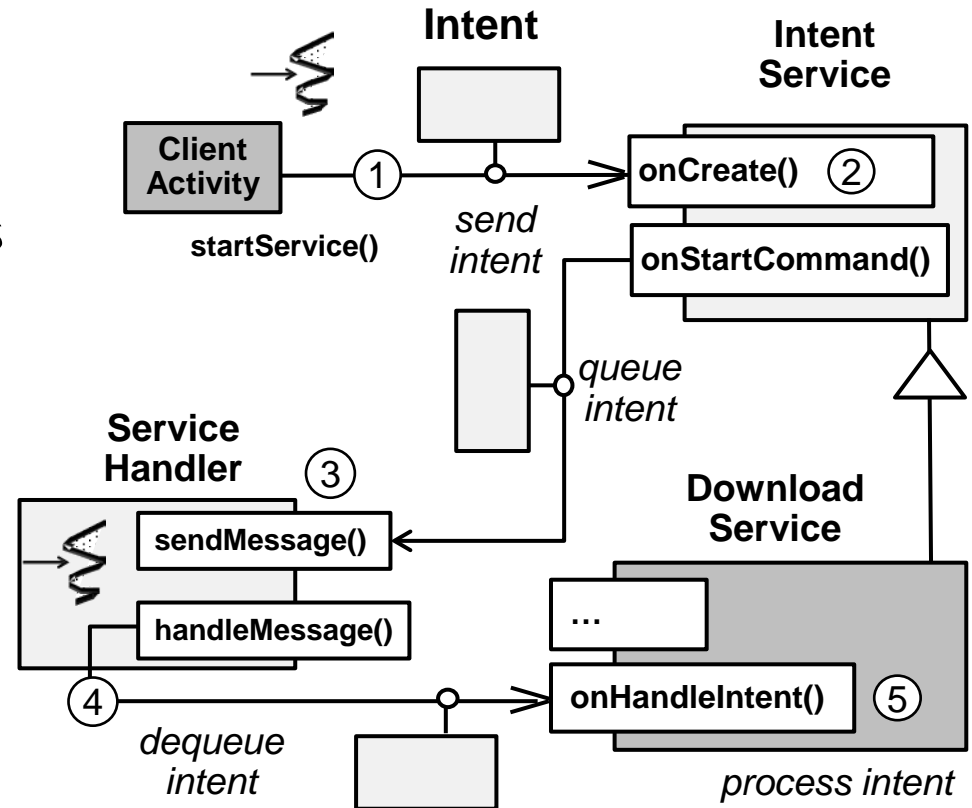
Patterns & Idioms in the Download Application

- DownloadHandler & ServiceHandler implement various patterns & idioms
- The structure & flow of control also applies a common Android Service concurrency idiom



Patterns & Idioms in the Download Application

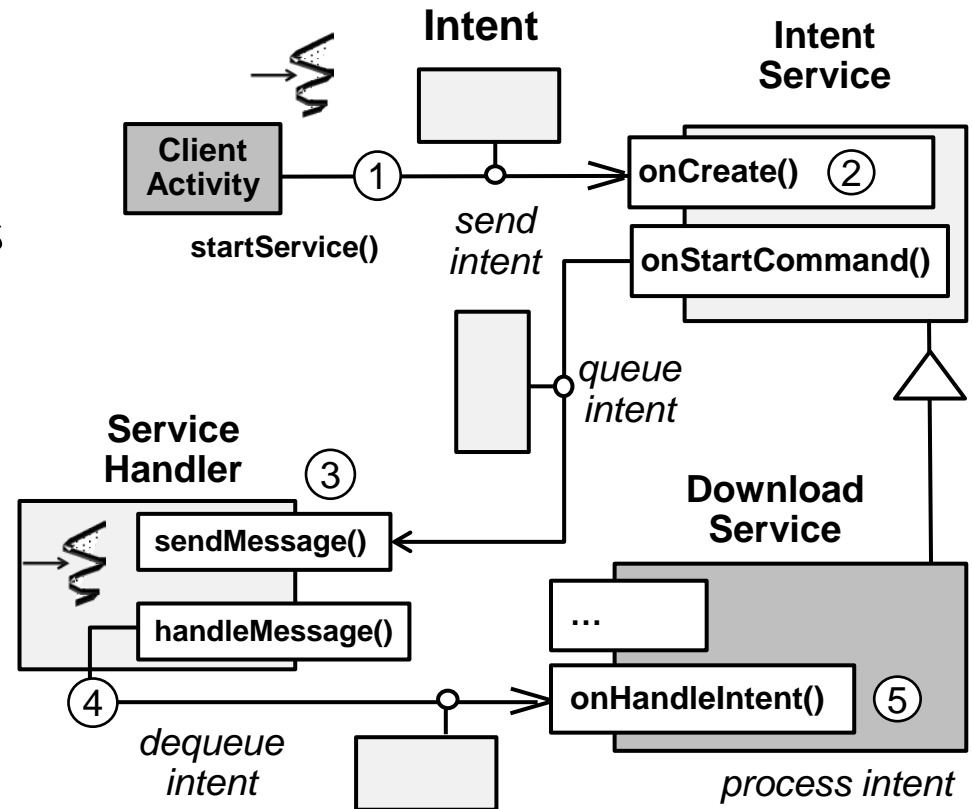
- DownloadHandler & ServiceHandler implement various patterns & idioms
- The structure & flow of control also applies a common Android Service concurrency idiom
- Android's IntentService codifies this idiom in a reusable framework



See upcoming part on
"Android IntentService"

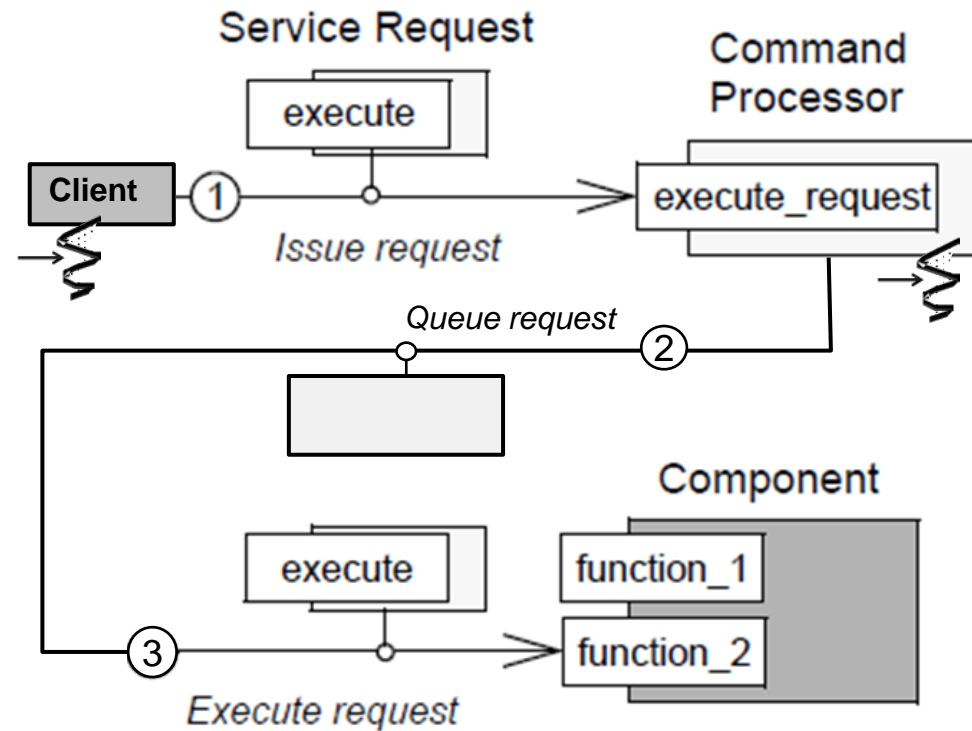
Patterns & Idioms in the Download Application

- DownloadHandler & ServiceHandler implement various patterns & idioms
- The structure & flow of control also applies a common Android Service concurrency idiom
- Android's IntentService codifies this idiom in a reusable framework
 - Handles async commands expressed as Intents in a background thread



Patterns & Idioms in the Download Application

- DownloadHandler & ServiceHandler implement various patterns & idioms
- The structure & flow of control also applies a common Android Service concurrency idiom
- Android's IntentService codifies this idiom in a reusable framework
 - Handles async commands expressed as Intents in a background thread
- Applies the *Command Processor* pattern

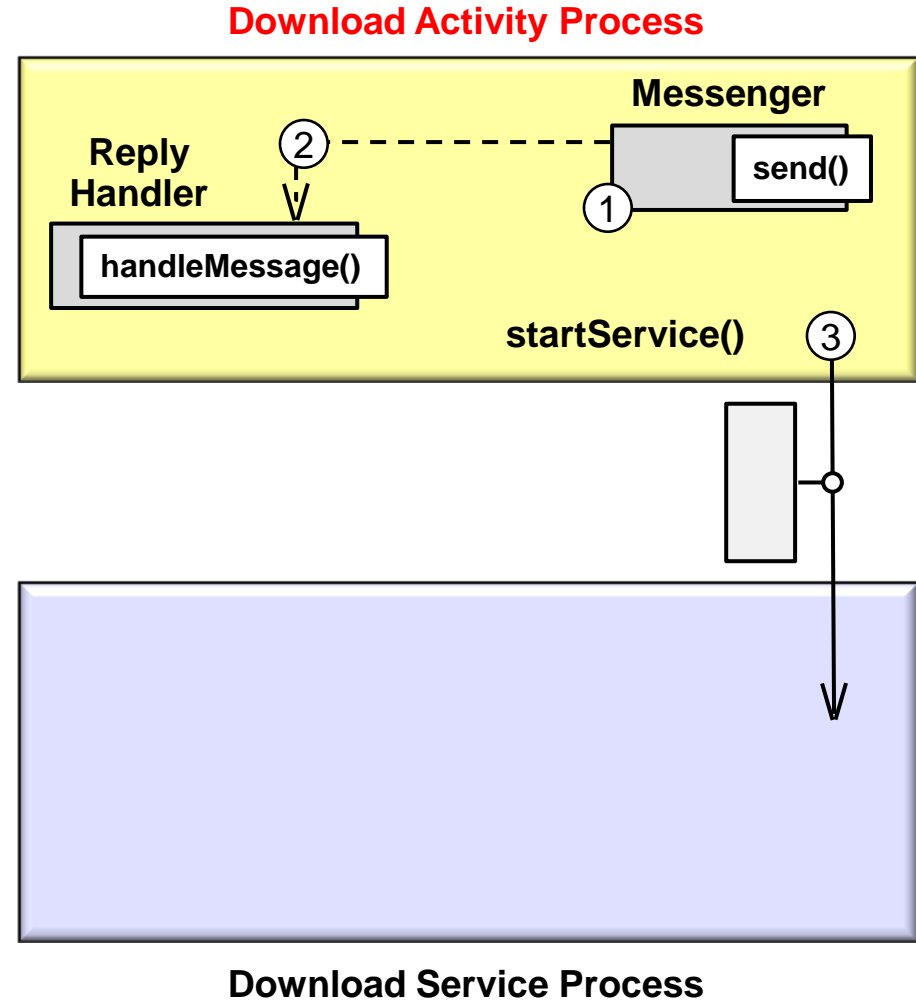


See www.dre.vanderbilt.edu/~schmidt/PDF/CommandRevisited.pdf

Programming the Messenger in the Download Application (Part 1)

Programming Messenger in Download Application

- DownloadActivity passes a Messenger to the DownloadService



Programming Messenger in Download Application

- DownloadActivity passes a Messenger to the DownloadService

```
public class DownloadActivity extends Activity {  
    ...  
    Handler mDownloadHandler = null;  
  
    public void onCreate(Bundle savedInstanceState) {  
        ...  
  
        mDownloadHandler =  
            new DownloadHandler(this);  
  
    }  
    ...  
}
```

Programming Messenger in Download Application

- DownloadActivity passes a Messenger to the DownloadService

```
public class DownloadActivity extends Activity {  
    ...  
    Handler mDownloadHandler = null;  
  
    public void onCreate(Bundle savedInstanceState) {  
        ...  
  
        mDownloadHandler =  
            new DownloadHandler(this);  
    }  
    ...  
}
```



Create a DownloadHandler to process
replies from DownloadService

Programming Messenger in Download Application

- DownloadActivity passes a Messenger to the DownloadService

```
public class DownloadActivity extends Activity {  
    ...
```



Called in response to a user button click

```
    public void downloadImage(View v) {  
        Uri url = getUrlString();
```

```
        Intent intent = DownloadService.makeIntent  
            (this, Uri.parse(url), mDownloadHandler);
```

```
        startService(intent);  
    }
```

```
    ...
```


Programming Messenger in Download Application

- DownloadActivity passes a Messenger to the DownloadService

```
public class DownloadActivity extends Activity {  
    ...
```

```
    public void downloadImage(View v) {  
        Uri url = getUrlString();
```

Factory method to make the Intent



```
        Intent intent = DownloadService.makeIntent  
                                (this, Uri.parse(url), mDownloadHandler);
```

```
        startService(intent);  
    }  
    ...
```

Programming Messenger in Download Application

- DownloadActivity passes a Messenger to the DownloadService
- DownloadService replies to Activity via Messenger's send() method

```
public class DownloadService extends Service {  
  
    public static Intent makeIntent(Context context,  
                                   Uri url,  
                                   Handler downloadHandler) {  
        Intent intent = new Intent(context,  
                                   DownloadService.class);  
  
        intent.setData(url);  
  
        intent.putExtra(MESSENGER,  
                        new Messenger(downloadHandler));  
        return intent;  
    }  
    ...  
}
```

Programming Messenger in Download Application

- DownloadActivity passes a Messenger to the DownloadService
- DownloadService replies to Activity via Messenger's send() method

```
public class DownloadService extends Service {
```

Factory method to make
the right intent



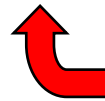
```
    public static Intent makeIntent(Context context,  
                                   Uri url,  
                                   Handler downloadHandler) {  
        Intent intent = new Intent(context,  
                                    DownloadService.class);  
  
        intent.setData(url);  
  
        intent.putExtra(MESSENGER,  
                        new Messenger(downloadHandler));  
        return intent;  
    }  
    ...
```

Programming Messenger in Download Application

- DownloadActivity passes a Messenger to the DownloadService
- DownloadService replies to Activity via Messenger's send() method

```
public class DownloadService extends Service {
```

```
    public static Intent makeIntent(Context context,
                                   Uri url,
                                   Handler downloadHandler) {
        Intent intent = new Intent(context,
                                   DownloadService.class);
        intent.setData(url);
        intent.putExtra(MESSENGER,
                        new Messenger(downloadHandler));
        return intent;
    }
    ...
}
```



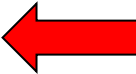
Create Intent that launches
DownloadService

Programming Messenger in Download Application

- DownloadActivity passes a Messenger to the DownloadService
- DownloadService replies to Activity via Messenger's send() method

```
public class DownloadService extends Service {
```

```
    public static Intent makeIntent(Context context,
                                    Uri url,
                                    Handler downloadHandler) {
        Intent intent = new Intent(context,
                                    DownloadService.class);

        intent.setData(url);  Add the image URL as
                               data to the Intent

        intent.putExtra(MESSENGER,
                        new Messenger(downloadHandler));
        return intent;
    }
    ...
}
```

Programming Messenger in Download Application

- DownloadActivity passes a Messenger to the DownloadService
- DownloadService replies to Activity via Messenger's send() method

```
public class DownloadService extends Service {
```

```
    public static Intent makeIntent(Context context,  
                                   Uri url,  
                                   Handler downloadHandler) {  
        Intent intent = new Intent(context,  
                                   DownloadService.class);
```

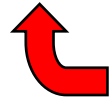
```
        intent.setData(url);
```

```
        intent.putExtra(MESSENGER,  
                        new Messenger(downloadHandler));
```

```
        return intent;
```

```
    }
```

```
    ...
```



Pass a Messenger as "extra" in Intent
used to start DownloadService

Programming Messenger in Download Application

- DownloadActivity passes a Messenger to the DownloadService
- DownloadService replies to Activity via Messenger's send() method

```
public class DownloadService extends Service {
```

```
    public static Intent makeIntent(Context context,
                                   Uri url,
                                   Handler downloadHandler) {
        Intent intent = new Intent(context,
                                   DownloadService.class);
```

```
        intent.setData(url);
```

```
        intent.putExtra(MESSENGER,
                        new Messenger(downloadHandler));
```

```
        return intent;
```

```
    }
```

```
    ...
```



Pass a Messenger as "extra" in Intent
used to start DownloadService

Programming Messenger in Download Application

- DownloadActivity passes a Messenger to the DownloadService
- DownloadService replies to Activity via Messenger's send() method

```
public class DownloadService extends Service {
```

```
    public static Intent makeIntent(Context context,
                                    Uri url,
                                    Handler downloadHandler) {
        Intent intent = new Intent(context,
                                    DownloadService.class);
```

```
        intent.setData(url);
```

```
        intent.putExtra(MESSENGER,
                        new Messenger(downloadHandler));
```

```
        return intent;
```

```
    }
```

```
    ...
```



Pass a Messenger as "extra" in Intent
used to start DownloadService

Programming Messenger in Download Application

- DownloadActivity passes a Messenger to the DownloadService
- DownloadService replies to Activity via Messenger's send() method

```
public class DownloadService extends Service {
```

```
    public static Intent makeIntent(Context context,  
                                   Uri url,  
                                   Handler downloadHandler) {  
        Intent intent = new Intent(context,  
                                   DownloadService.class);
```

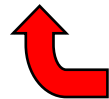
```
        intent.setData(url);
```

```
        intent.putExtra(MESSENGER,  
                        new Messenger(downloadHandler));
```

```
        return intent;
```

```
    }
```

```
    ...
```



Pass a Messenger as "extra" in Intent
used to start DownloadService

Programming Messenger in Download Application

- DownloadActivity passes a Messenger to the DownloadService

```
public class DownloadActivity extends Activity {  
    ...  
  
    public void downloadImage(View v) {  
        Uri url = getUrlString();  
  
        Intent intent = DownloadService.makeIntent  
            (this, Uri.parse(url), mDownloadHandler);  
  
        startService(intent);  
    }  
    ...  
}
```

Programming Messenger in Download Application

- DownloadActivity passes a Messenger to the DownloadService

```
public class DownloadActivity extends Activity {  
    ...
```

```
    public void downloadImage(View v) {  
        Uri url = getUrlString();
```

Factory method to make the Intent



```
        Intent intent = DownloadService.makeIntent  
            (this, Uri.parse(url), mDownloadHandler);
```

```
        startService(intent);
```


```
    }
```

```
    ...
```

Programming Messenger in Download Application

- DownloadActivity passes a Messenger to the DownloadService


```
public class DownloadActivity extends Activity {  
    ...  
  
    public void downloadImage(View v) {  
        Uri url = getUrlString();  
  
        Intent intent = DownloadService.makeIntent  
            (this, Uri.parse(url), mDownloadHandler);  
  
        startService(intent);  
    }  
    ...
```

 **Start the DownloadService**

Programming Messenger in Download Application

- DownloadActivity passes a Messenger to the DownloadService

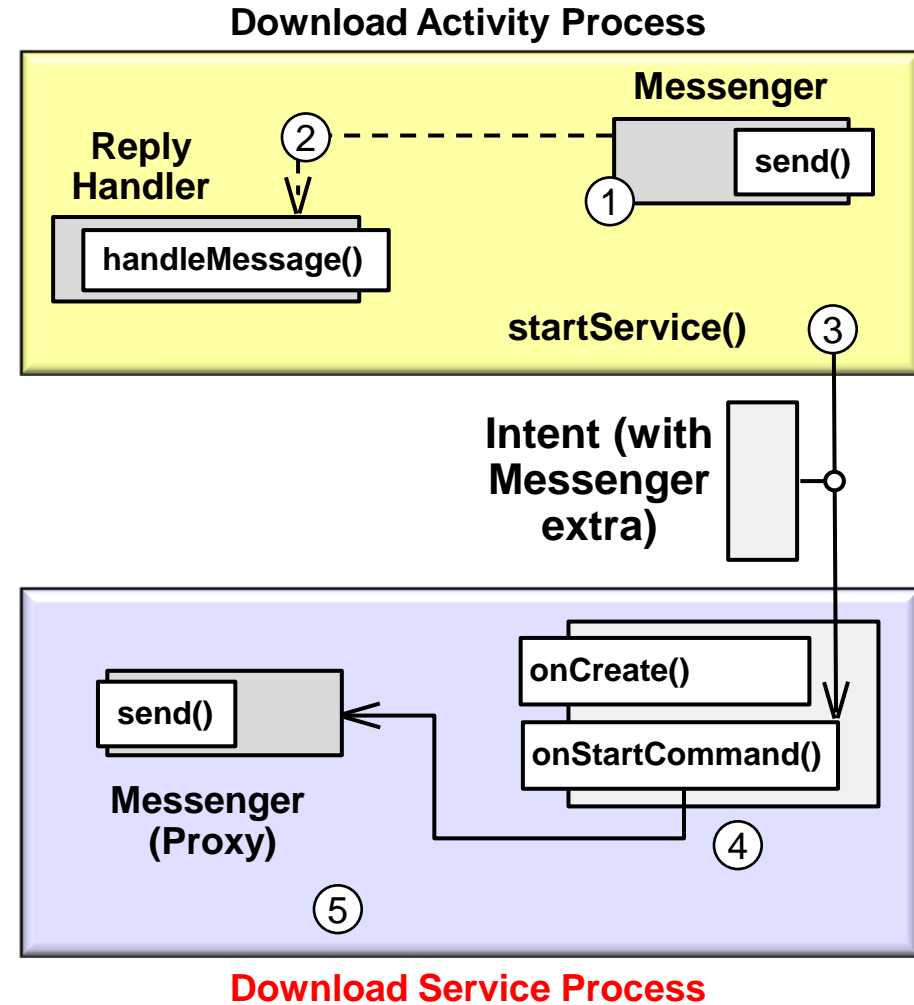
```
public class DownloadActivity extends Activity {  
    ...  
  
    public void downloadImage(View v) {  
        Uri url = getUrlString();  
  
        Intent intent = DownloadService.makeIntent  
            (this, Uri.parse(url), mDownloadHandler);  
  
        startService(intent);  
    }  
    ...  
}
```

 **Start the DownloadService**

Programming the Messenger in the Download Application (Part 2)

Programming Messenger in Download Application

- DownloadService replies to Activity via Messenger's send() method



Programming Messenger in Download Application

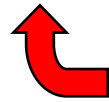
- DownloadService replies to Activity via Messenger's send() method

```
public class DownloadService extends Service {  
    ...  
    private final class ServiceHandler extends Handler {  
  
        private void downloadImageAndReply(Intent intent) {  
  
            String pathname =  
                downloadImage(DownloadService.this,  
                             intent.getData().toString());  
  
            Messenger messenger = (Messenger) intent.getExtras().  
                                   get(MESSENGER);  
            sendPath(messenger, pathname);  
        }  
        ...  
    }  
}
```

Programming Messenger in Download Application

- DownloadService replies to Activity via Messenger's send() method


```
public class DownloadService extends Service {  
    ...  
    private final class ServiceHandler extends Handler {  
  
        private void downloadImageAndReply(Intent intent) {  
  
            String pathname =  
                downloadImage(DownloadService.this,  
                             intent.getData().toString());  
  
            Messenger messenger = (Messenger) intent.getExtras().  
                                   get(MESSENGER);  
            sendPath(messenger, pathname);  
        }  
        ...  
    }  
}
```



Retrieve the designated image &
reply to the DownloadActivity

Programming Messenger in Download Application


- DownloadService replies to Activity via Messenger's send() method

```
public class DownloadService extends Service {  
    ...  
    private final class ServiceHandler extends Handler {  
  
        private void downloadImageAndReply(Intent intent) {  
  
            String pathname =  
                downloadImage(DownloadService.this,  
                               intent.getData().toString());  
  
             Retrieve the designated image  
  
            Messenger messenger = (Messenger) intent.getExtras().  
                                   get(MESSENGER);  
            sendPath(messenger, pathname);  
        }  
        ...  
    }  
}
```

Programming Messenger in Download Application

- DownloadService replies to Activity via Messenger's send() method

```
public class DownloadService extends Service {  
    ...  
    private final class ServiceHandler extends Handler {  
  
        private void downloadImageAndReply(Intent intent) {  
  
            String pathname =  
                downloadImage(DownloadService.this,  
                    intent.getData().toString());  
  
            Messenger messenger = (Messenger) intent.getExtras().  
                get(MESSENGER);  
            sendPath(messenger, pathname);  
        }  
        ...  
    }  
}
```




Extract Messenger
from the Intent

Programming Messenger in Download Application

- DownloadService replies to Activity via Messenger's send() method

```
public class DownloadService extends Service {  
    ...  
    private final class ServiceHandler extends Handler {  
  
        private void downloadImageAndReply(Intent intent) {  
  
            String pathname =  
                downloadImage(DownloadService.this,  
                             intent.getData().toString());  
  
            Messenger messenger = (Messenger) intent.getExtras().  
                                   get(MESSENGER);  
  
            sendPath(messenger, pathname);  
        }  
    }  
    ...  
}
```

 **Send the pathname via the Messenger**

Programming Messenger in Download Application

- DownloadService replies to Activity via Messenger's send() method

```
public class DownloadService extends Service {
```

```
...
```

```
private final class ServiceHandler extends Handler {
```



Send the pathname via the Messenger

```
private void sendPath(Messenger messenger,  
                      String pathname) {  
    Message message = makeReplyMessage(pathname);
```

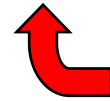
```
    try {  
        messenger.send(message);  
    }  
    ...
```

Programming Messenger in Download Application

- DownloadService replies to Activity via Messenger's send() method

```
public class DownloadService extends Service {  
    ...  
    private final class ServiceHandler extends Handler {
```

```
        private void sendPath(Messenger messenger,  
                               String pathname) {  
            Message message = makeReplyMessage(pathname);
```



Factory method that creates a
Message to return to Download
Activity

```
        try {  
            messenger.send(message);  
        }  
        ...
```

Programming Messenger in Download Application

- DownloadService replies to Activity via Messenger's send() method

```
public class DownloadService extends Service {  
    ...  
    private final class ServiceHandler extends Handler {  
  
        private Message makeReplyMessage(String pathname){  
  
            Message message = Message.obtain();  
            message.arg1 = pathname == null  
                ? Activity.RESULT_CANCELED : Activity.RESULT_OK;  
  
            Bundle bundle = new Bundle();  
            bundle.putString(PATHNAME, pathname);  
            message.setData(bundle);  
            return message;  
        }  
        ...  
    }  
}
```



Factory method that creates a Message
to return to Download Activity

Programming Messenger in Download Application

- DownloadService replies to Activity via Messenger's send() method

```
public class DownloadService extends Service {  
    ...  
    private final class ServiceHandler extends Handler {
```

```
        private Message makeReplyMessage(String pathname){
```

```
            Message message = Message.obtain();
```

```
            message.arg1 = pathname == null
```

```
                ? Activity.RESULT_CANCELED : Activity.RESULT_OK;
```

```
            Bundle bundle = new Bundle();
```

```
            bundle.putString(PATHNAME, pathname);
```

```
            message.setData(bundle);
```

```
            return message;
```

```
        }
```

```
    ...  
}
```

Creates a Message to encapsulate
pathname for downloaded image



Programming Messenger in Download Application

- DownloadService replies to Activity via Messenger's send() method

```
public class DownloadService extends Service {  
    ...  
    private final class ServiceHandler extends Handler {  
        private Message makeReplyMessage(String pathname){
```



Store result indicating if download succeeded or failed

```
        Message message = Message.obtain();  
        message.arg1 = pathname == null  
            ? Activity.RESULT_CANCELED : Activity.RESULT_OK;
```

```
        Bundle bundle = new Bundle();  
        bundle.putString(PATHNAME, pathname);  
        message.setData(bundle);  
        return message;
```

```
    }  
    ...
```

Programming Messenger in Download Application

- DownloadService replies to Activity via Messenger's send() method

```
public class DownloadService extends Service {  
    ...  
    private final class ServiceHandler extends Handler {  
  
        private Message makeReplyMessage(String pathname){  
  
            Message message = Message.obtain();  
            message.arg1 = pathname == null  
                ? Activity.RESULT_CANCELED : Activity.RESULT_OK;  
  
            Bundle bundle = new Bundle();  
            bundle.putString(PATHNAME, pathname);  
            message.setData(bundle);  
            return message;  
        }  
    }  
    ...  
}
```



Create a Bundle to store the pathname
String for the downloaded image

See [developer.android.com/reference/
android/os/Bundle.html](http://developer.android.com/reference/android/os/Bundle.html)

Programming Messenger in Download Application

- DownloadService replies to Activity via Messenger's send() method

```
public class DownloadService extends Service {  
    ...  
    private final class ServiceHandler extends Handler {  
  
        private Message makeReplyMessage(String pathname){  
  
            Message message = Message.obtain();  
            message.arg1 = pathname == null  
                ? Activity.RESULT_CANCELED : Activity.RESULT_OK;  
  
            Bundle bundle = new Bundle();  
            bundle.putString(PATHNAME, pathname);  
            message.setData(bundle);  
            return message;  
        }  
        ...  
    }  
}
```




Create a Bundle to store the pathname
String for the downloaded image

Programming Messenger in Download Application

- DownloadService replies to Activity via Messenger's send() method

```
public class DownloadService extends Service {  
    ...  
    private final class ServiceHandler extends Handler {  
  
        private Message makeReplyMessage(String pathname){  
  
            Message message = Message.obtain();  
            message.arg1 = pathname == null  
                ? Activity.RESULT_CANCELED : Activity.RESULT_OK;  
  
            Bundle bundle = new Bundle();  
            bundle.putString(PATHNAME, pathname);  
            message.setData(bundle);  
            return message;  
        }  
        ...  
    }  
}
```

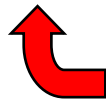


Create a Bundle to store the pathname
String for the downloaded image

Programming Messenger in Download Application

- DownloadService replies to Activity via Messenger's send() method

```
public class DownloadService extends Service {  
    ...  
    private final class ServiceHandler extends Handler {  
  
        private Message makeReplyMessage(String pathname){  
  
            Message message = Message.obtain();  
            message.arg1 = pathname == null  
                ? Activity.RESULT_CANCELED : Activity.RESULT_OK;  
  
            Bundle bundle = new Bundle();  
            bundle.putString(PATHNAME, pathname);  
            message.setData(bundle);  
            return message;  
        }  
    }  
    ...
```

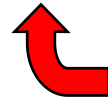


Return the reply Message

Programming Messenger in Download Application

- DownloadService replies to Activity via Messenger's send() method

```
public class DownloadService extends Service {  
    ...  
    private final class ServiceHandler extends Handler {  
  
        private void sendPath(Messenger messenger,  
                               String pathname) {  
            Message message = makeReplyMessage(pathname);  
  
            try {  
                messenger.send(message);  
            }  
            ...  
        }  
    }  
}
```



Factory method that
creates a Message to return
to the Download Activity

Programming Messenger in Download Application

- DownloadService replies to Activity via Messenger's send() method

```
public class DownloadService extends Service {  
    ...  
    private final class ServiceHandler extends Handler {  
  
        private void sendPath(Messenger messenger,  
                               String pathname) {  
            Message message = makeReplyMessage(pathname);  
  
            try {  
                messenger.send(message);  
            }  
            ...  
        }  
    }  
}
```



Pass the Message back to
the Download Activity

Programming Messenger in Download Application

- DownloadService replies to Activity via Messenger's send() method

```
public class DownloadService extends Service {  
    ...  
    private final class ServiceHandler extends Handler {  
  
        private void sendPath(Messenger messenger,  
                               String pathname) {  
            Message message = makeReplyMessage(pathname);  
  
            try {  
                messenger.send(message);  
            }  
            ...  
        }  
    }  
}
```



Pass the Message back to
the Download Activity

The Message is sent using the
Android Binder IPC framework

Programming Messenger in Download Application

- DownloadService replies to Activity via Messenger's send() method

```
public class DownloadService extends Service {  
    ...  
    private final class ServiceHandler extends Handler {  
  
        private void sendPath(Messenger messenger,  
                               String pathname) {  
            Message message = makeReplyMessage(pathname);  
  
            try {  
                messenger.send(message);  
            } catch (RemoteException e) {  
                ...  
            }  
        }  
    }  
}
```



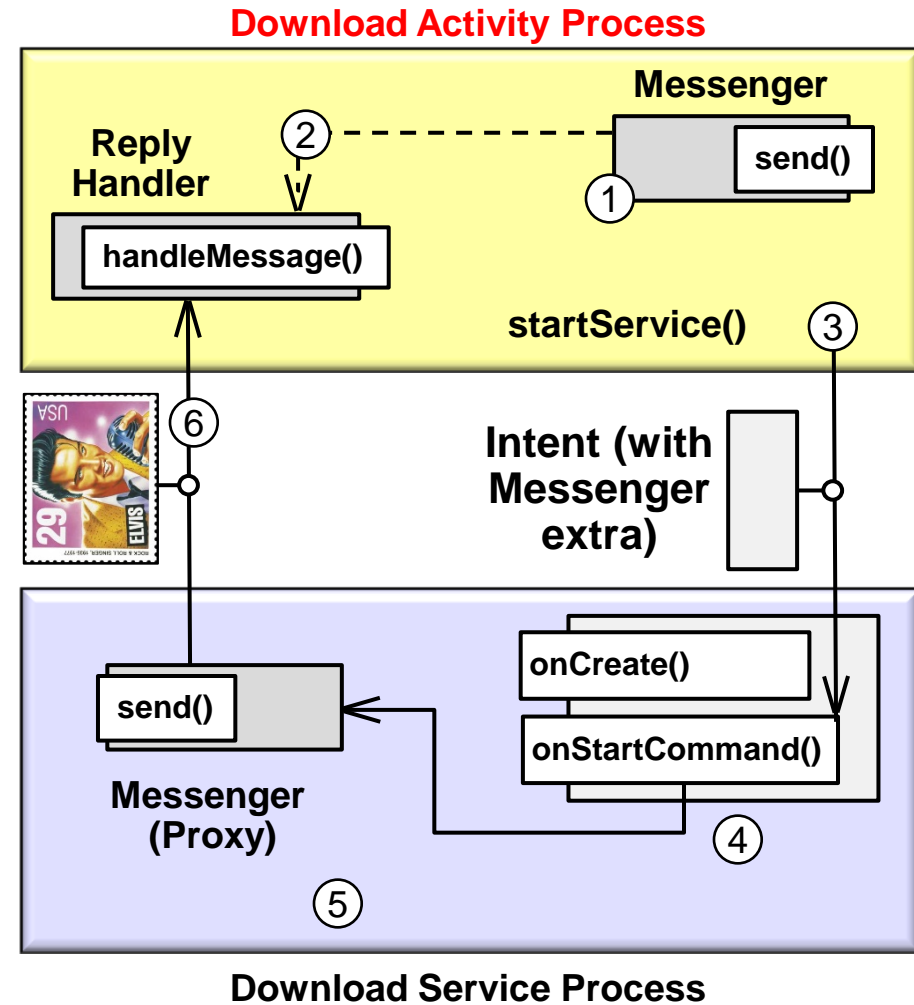
DeadObjectException is thrown if
target Handler no longer exists

See [developer.android.com/reference/
android/os/DeadObjectException.html](http://developer.android.com/reference/android/os/DeadObjectException.html)

Programming the Messenger in the Download Application (Part 3)

Programming Messenger in Download Application

- DownloadActivity receives Message via its Handler in the UI Thread



The Android Binder IPC framework collaborates with the HaMeR framework

Programming Messenger in Download Application

- DownloadActivity receives Message via its Handler in the UI Thread

```
public class DownloadActivity extends Activity {  
    ...  
    private static class DownloadHandler extends Handler {  
        public void handleMessage(Message message) {  
            ...  
  
            String pathname = DownloadService.getPathname(message);  
  
            if (pathname == null)  
                activity.showDialog("failed download");  
  
            ...  
            activity.displayImage  
                (BitmapFactory.decodeFile(pathname));  
            ...  
        }  
    }  
}
```

Programming Messenger in Download Application

- DownloadActivity receives Message via its Handler in the UI Thread

```
public class DownloadActivity extends Activity {  
    ...  
    private static class DownloadHandler extends Handler {  
        public void handleMessage(Message message) {  
            ...  
             Dispatched to process reply from DownloadService  
            String pathname = DownloadService.getPathname(message);  
  
            if (pathname == null)  
                activity.showDialog("failed download");  
  
            ...  
            activity.displayImage  
                (BitmapFactory.decodeFile(pathname));  
            ...  
        }  
    }  
}
```


This hook method runs in the
context of the UI Thread

Programming Messenger in Download Application

- DownloadActivity receives Message via its Handler in the UI Thread

```
public class DownloadActivity extends Activity {  
    ...  
    private static class DownloadHandler extends Handler {  
        public void handleMessage(Message message) {  
            ...  
            String pathname = DownloadService.getPathname(message);  
  
            if (pathname == null)  
                activity.showDialog("failed download");  
  
            ...  
            activity.displayImage  
                (BitmapFactory.decodeFile(pathname));  
            ...  
        }  
    }  
}
```

Extract the pathname
from the Message



Programming Messenger in Download Application

- DownloadActivity receives Message via its Handler in the UI Thread
- DownloadService provides a helper method for getting the pathname

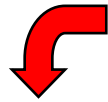
```
public class DownloadService extends Service {  
    ...  
  
    public static String getPathname(Message message) {  
  
        Bundle data = message.getData();  
  
        String pathname = data.getString(PATHNAME);  
  
        if (message.arg1 != Activity.RESULT_OK  
            || pathname == null)  
            return null;  
        return pathname;  
    }  
}
```


Programming Messenger in Download Application

- DownloadActivity receives Message via its Handler in the UI Thread
- DownloadService provides a helper method for getting the pathname

```
public class DownloadService extends Service {
```

```
...
```



Shields DownloadActivity from details of passing pathname across processes

```
public static String getPathname(Message message) {
```

```
    Bundle data = message.getData();
```

```
    String pathname = data.getString(PATHNAME);
```

```
    if (message.arg1 != Activity.RESULT_OK  
        || pathname == null)
```

```
        return null;
```

```
    return pathname;
```

```
}
```

Programming Messenger in Download Application

- DownloadActivity receives Message via its Handler in the UI Thread
- DownloadService provides a helper method for getting the pathname

```
public class DownloadService extends Service {  
    ...
```

```
    public static String getPathname(Message message) {
```

```
        Bundle data = message.getData();
```



Extract Bundle from Message

```
        String pathname = data.getString(PATHNAME);
```

```
        if (message.arg1 != Activity.RESULT_OK  
            || pathname == null)
```

```
            return null;
```

```
        return pathname;
```

```
    }
```

Programming Messenger in Download Application

- DownloadActivity receives Message via its Handler in the UI Thread
- DownloadService provides a helper method for getting the pathname

```
public class DownloadService extends Service {  
    ...
```

```
    public static String getPathname(Message message) {
```

```
        Bundle data = message.getData();
```

```
        String pathname = data.getString(PATHNAME);
```

```
        if (message.arg1 != Activity.RESULT_OK  
            || pathname == null)
```

```
            return null;
```

```
        return pathname;
```

```
    }
```

 Extract pathname
from the Bundle

Programming Messenger in Download Application

- DownloadActivity receives Message via its Handler in the UI Thread
- DownloadService provides a helper method for getting the pathname

```
public class DownloadService extends Service {
```

```
...
```

```
public static String getPathname(Message message) {
```

```
    Bundle data = message.getData();
```

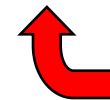
```
    String pathname = data.getString(PATHNAME);
```

```
    if (message.arg1 != Activity.RESULT_OK  
        || pathname == null)
```

```
        return null;
```

```
    return pathname;
```

```
}
```



Check the results &
return the pathname
if all went well

Programming Messenger in Download Application

- DownloadActivity receives Message via its Handler in the UI Thread
- DownloadService provides a helper method for getting the pathname

```
public class DownloadService extends Service {  
    ...
```

```
    public static String getPathname(Message message) {
```

```
        Bundle data = message.getData();
```

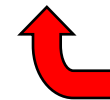
```
        String pathname = data.getString(PATHNAME);
```

```
        if (message.arg1 != Activity.RESULT_OK  
            || pathname == null)
```

```
            return null;
```

```
        return pathname;
```

```
    }
```



Check the results &
return the pathname
if all went well

Programming Messenger in Download Application

- DownloadActivity receives Message via its Handler in the UI Thread
- DownloadService provides a helper method for getting the pathname

```
public class DownloadService extends Service {  
    ...
```

```
    public static String getPathname(Message message) {
```

```
        Bundle data = message.getData();
```

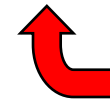
```
        String pathname = data.getString(PATHNAME);
```

```
        if (message.arg1 != Activity.RESULT_OK  
            || pathname == null)
```

```
            return null;
```

```
        return pathname;
```

```
    }
```



Check the results &
return the pathname
if all went well

Programming Messenger in Download Application

- DownloadActivity receives Message via its Handler in the UI Thread

```
public class DownloadActivity extends Activity {  
    ...  
    private static class DownloadHandler extends Handler {  
        public void handleMessage(Message message) {  
            ...  
  
            String pathname = DownloadService.getPathname(message);  
  
  
            if (pathname == null)  
                activity.showDialog("failed download");  
  
            ...  
            activity.displayImage  
                (BitmapFactory.decodeFile(pathname));  
            ...  
        }  
    }  
}
```

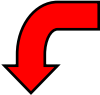
Programming Messenger in Download Application

- DownloadActivity receives Message via its Handler in the UI Thread

```
public class DownloadActivity extends Activity {  
    ...  
    private static class DownloadHandler extends Handler {  
        public void handleMessage(Message message) {  
            ...  
  
            String pathname = DownloadService.getPathname(message);  
  
             See if the download succeeded or not  
  
            if (pathname == null)  
                activity.showDialog("failed download");  
  
            ...  
            activity.displayImage  
                (BitmapFactory.decodeFile(pathname));  
            ...  
        }  
    }  
}
```


Programming Messenger in Download Application


- DownloadActivity receives Message via its Handler in the UI Thread

```
public class DownloadActivity extends Activity {  
    ...  
    private static class DownloadHandler extends Handler {  
        public void handleMessage(Message message) {  
            ...  
  
            String pathname = DownloadService.getPathname(message);  
  
             See if the download succeeded or not  
  
            if (pathname == null)  
                activity.showDialog("failed download");  
  
            ...  
            activity.displayImage  
                (BitmapFactory.decodeFile(pathname));  
            ...  
        }  
    }  
}
```

Programming Messenger in Download Application

- DownloadActivity receives Message via its Handler in the UI Thread

```
public class DownloadActivity extends Activity {  
    ...  
    private static class DownloadHandler extends Handler {  
        public void handleMessage(Message message) {  
            ...  
  
            String pathname = DownloadService.getPathname(message);  
  
            if (pathname == null)  
                activity.showDialog("failed download");  
  
            ...  
            activity.displayImage  
                (BitmapFactory.decodeFile(pathname));  
            ...  
        }  
    }  
}
```

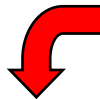
 Decode & display the
image in UI Thread

Programming Messenger in Download Application

- DownloadActivity receives Message via its Handler in the UI Thread

```
public class DownloadActivity extends Activity {  
    ...  
    private static class DownloadHandler extends Handler {  
        public void handleMessage(Message message) {  
            ...  
  
            String pathname = DownloadService.getPathname(message);  
  
            if (pathname == null)  
                activity.showDialog("failed download");  
  
            ...  
            activity.displayImage  
                (BitmapFactory.decodeFile(pathname));  
            ...  
        }  
    }  
}
```

Decode & display the
image in UI Thread



Programming Messenger in Download Application

- DownloadActivity receives Message via its Handler in the UI Thread

```
public class DownloadActivity extends Activity {  
    ...  
    private static class DownloadHandler extends Handler {  
        public void handleMessage(Message message) {  
            ...  
  
            String pathname = DownloadService.getPathname(message);  
  
  
            if (pathname == null)  
                activity.showDialog("failed download");  
  
            ...  
            activity.displayImage  
                (BitmapFactory.decodeFile(pathname));  
            ...  
        }  
    }  
}
```

DownloadActivity never blocks synchronously
during any long-duration operations