

Android - 2.0

On this page:

- [Extension ID](#)
- [What's new?](#)
- [Description](#)
 - [In what situation should you install this extension?](#)
- [Features](#)
 - [AndroidManifest file](#)
 - [Android Event handling support from XML file](#)
 - [Android Event handling support from `setEventListener` and `addEventListener` in widgets](#)
 - [Android Type XML support](#)
 - [WebServices support](#)
 - [End to End transactions](#)
- [Supported Android versions](#)
- [Supported Technologies](#)
- [Function Point, Quality and Sizing support](#)
- [CAST AIP compatibility](#)
- [Supported DBMS servers](#)
- [Prerequisites](#)
- [Dependencies with other extensions](#)
- [Download and installation instructions](#)
- [Packaging, delivering and analyzing your source code](#)
 - [Packaging/delivering with the Delivery Manager Tool](#)
 - [Analysis configuration in the CAST Management Studio](#)
- [What results can you expect?](#)
 - [Android application](#)
 - [Transactions](#)
 - [Web services linker](#)
 - [Function points \(when using CAIP 8.3.0\)](#)
 - [Supported libraries](#)
 - [Objects](#)
 - [Structural Rules](#)
- [Known limitations](#)

Target audience:

Users of the extension providing **Android technologies** support for the [JEE Analyzer](#).



Summary: This document provides information about the extension providing **Android technologies** support.

Extension ID

`com.castsoftware.android`

What's new?

Please see [Android 2.0 - Release Notes](#) for more information.

Description

This extension provides support for **Android technologies**. This extension works together with the [JEE Analyzer](#) and the [Kotlin](#) extension.

In what situation should you install this extension?

If your JEE or Kotlin application source code uses **Android technologies** you should install this extension.

Features

AndroidManifest file

This file will be analyzed to understand whether the application is indeed an Android application. Some objects (exposed service/activity/receiver) are created under the manifest to represent exposed services. Then links are created from these services to the methods which are listeners of these services.

Android Event handling support from XML file

- onClickListener()
- onLongClickListener()
- onFocusChangeListener()
- onKeyDownListener()
- onTouchListener()
- onCreateContextMenu()

Android Event handling support from setEventListener and addEventListener in widgets

- The lambda function can be used to define the event handler. Sample: Button.setOnClickListener(new View.OnClickListener {...}), with lambda function: Button.setOnClickListener((v) {})

Android Type XML support

- Widget Layout
- AndroidManifest file

WebServices support

Support for third-party WebServices is provided:

- httpClient library with namespace cz.msebera.android.httpclient.
- httpClient library with namespace org.apache.http.client.HttpClient.
- HttpURLConnection from java.net.URL and java.net.HttpURLConnection.
- Create webservice via AsyncTask class of Android.
- Transaction between client and server on fullstack project.
- Transaction between event handler widget and http method call.

End to End transactions

- End to End transactions are resolved:
 - Entry point: layout ui xml file, exposed services
 - End point: web service, persisting data, intent resource calls

Supported Android versions

Version	Supported
4.4 - 10.0	✓

Supported Technologies

Technology	Supported
Java	✓
Kotlin	✓

Function Point, Quality and Sizing support

This extension provides the following support:

- **Function Points (transactions):** a green tick indicates that OMG Function Point counting and Transaction Risk Index are supported
- **Quality and Sizing:** a green tick indicates that CAST can measure size and that a minimum set of Quality Rules exist

Function Points (transactions)	✓
--------------------------------	---

Quality and Sizing



CAST AIP compatibility

This extension is compatible with:

CAST AIP release	Supported
8.3.x	

Supported DBMS servers

This extension is compatible with the following DBMS servers:

DBMS	Supported
CSS / PostgreSQL	

Prerequisites



An installation of any compatible release of CAST AIP (see table above)

Dependencies with other extensions

Some CAST extensions require the presence of other CAST extensions in order to function correctly. The **Android** extension requires that the following other CAST extensions are also installed:

- [JEE Analyzer](#)
- [Kotlin extension](#)
- **com.castsoftware.internal.platform** (internal extension) 0.8.0
- **Web services linker service** (internal technical extension) 1.6.1 funcrel



Note that:

- when using the **CAST Extension Downloader** to download the extension and the **Manage Extensions** interface in **CAST Server Manager** to install the extension, any dependent extensions are **automatically** downloaded and installed for you. You do not need to do anything.
- the JEE Analyzer is not a dependency, but since this extension is always installed with AIP, you do not need to do anything.
- the Kotlin extension is a dependency.

Download and installation instructions

Please see:

- [Download an extension](#)
- [Install an extension](#)



The latest [release status](#) of this extension can be seen when downloading it from the CAST Extend server.

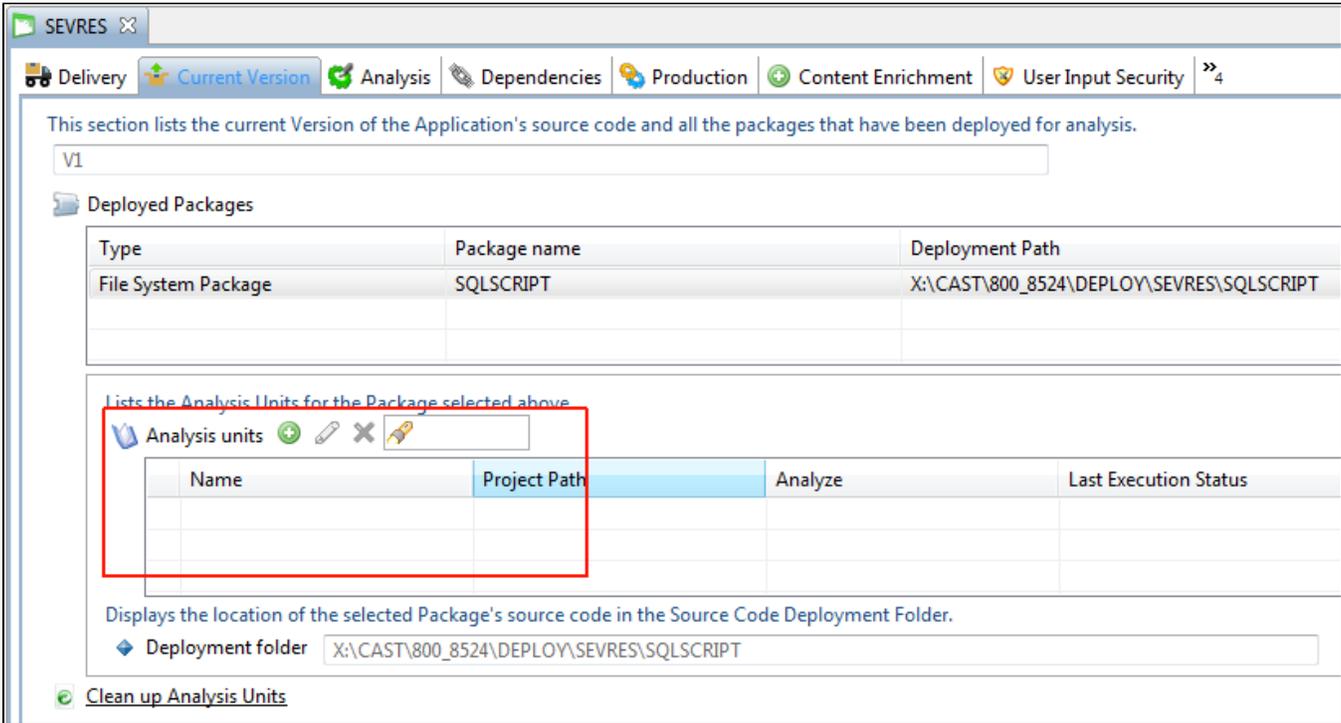
Packaging, delivering and analyzing your source code

Packaging/delivering with the Delivery Manager Tool

When the **Package action** is run: the CAST Delivery Manager Tool will **not** find any "projects" related to the Android application - this is the **expected behaviour**. However, if your source code is part of a larger application (for example a JEE application), then other projects may be found during the package action.

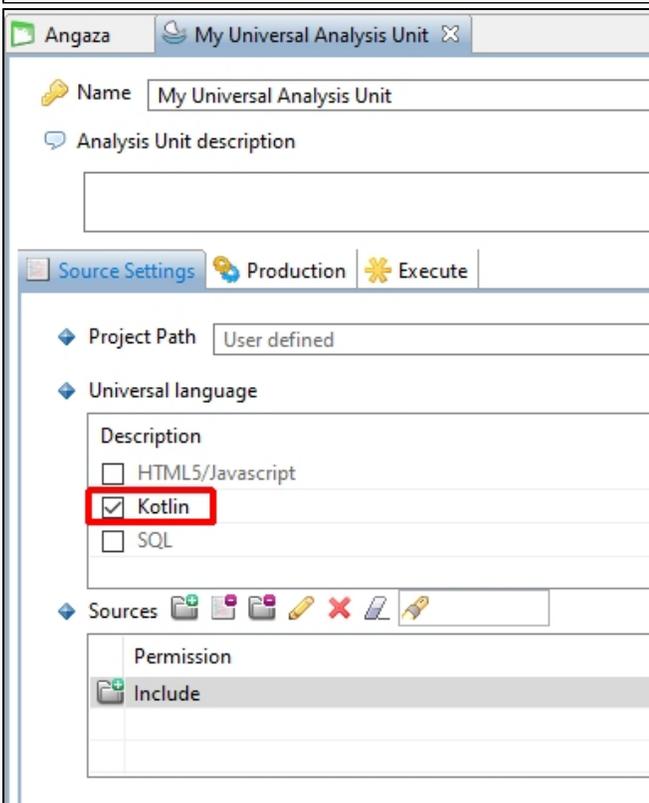
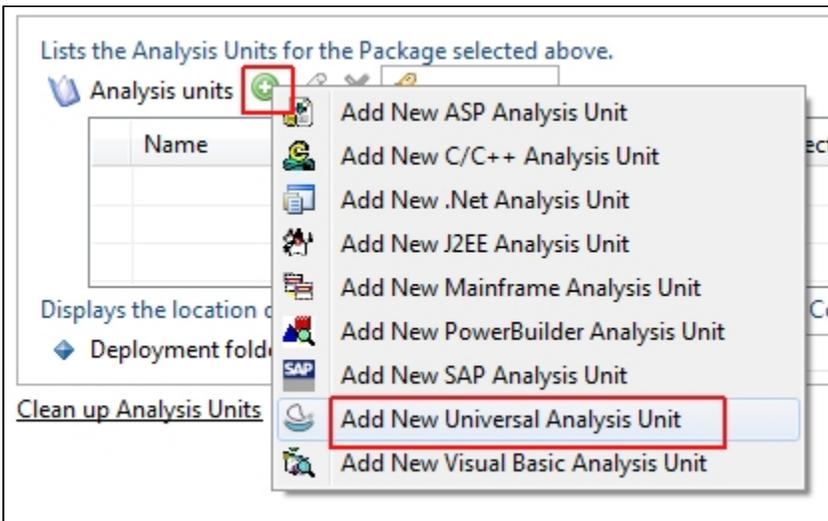
Analysis configuration in the CAST Management Studio

Accept and deploy the **Version** in the CAST Management Studio. No **Analysis Units** will be created automatically relating to the Android application - this is the **expected behavior**. However, if your Android application is part of a larger application (for example a JEE Application), then other Analysis Units may be created automatically:

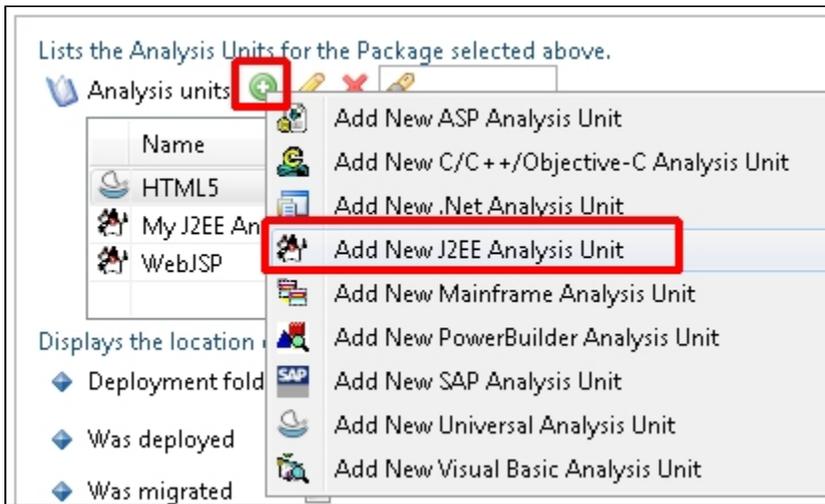


In the **Current Version** tab, add a new Analysis Unit specifically for your Android application:

- If you have **Kotlin** related files, select the **Add new Universal Analysis Unit** option and ensure you tick the **Kotlin** option:



- If you have Java related files, select the **Add new J2EE Analysis Unit** option:

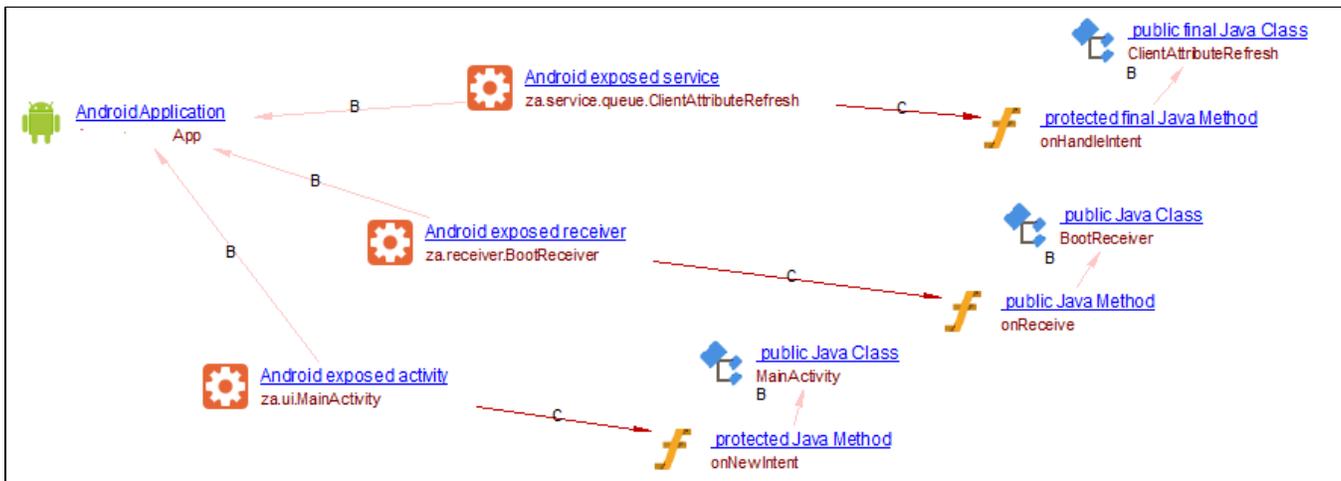


What results can you expect?

Once the analysis/snapshot generation has completed, you can view the results in the normal manner (for example via CAST Enlighten):

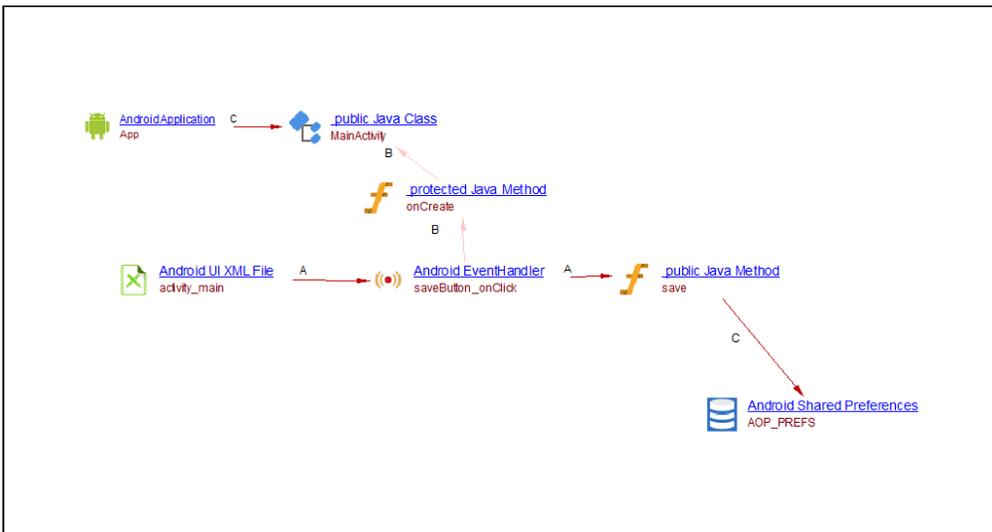
Android application

Click to enlarge



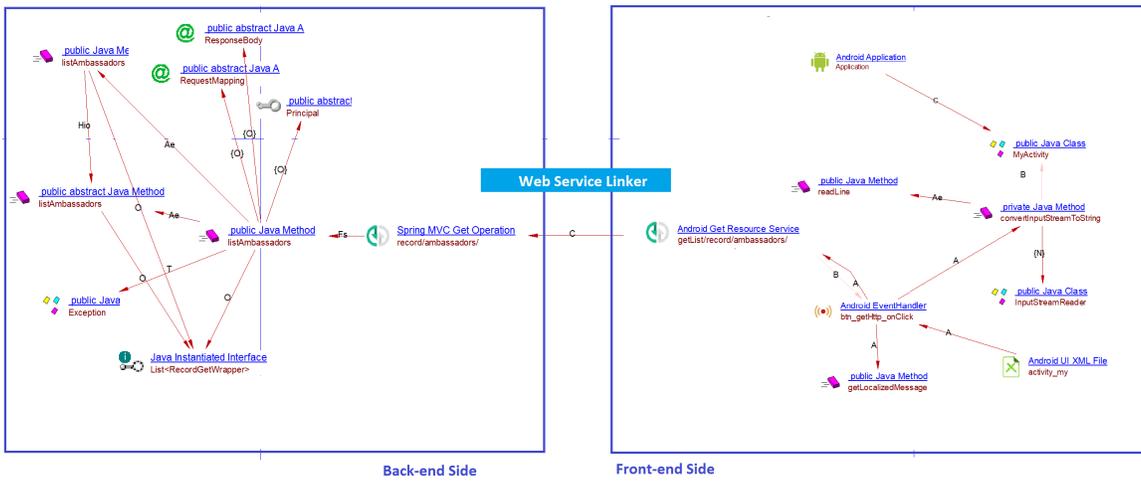
Transactions

Click to enlarge



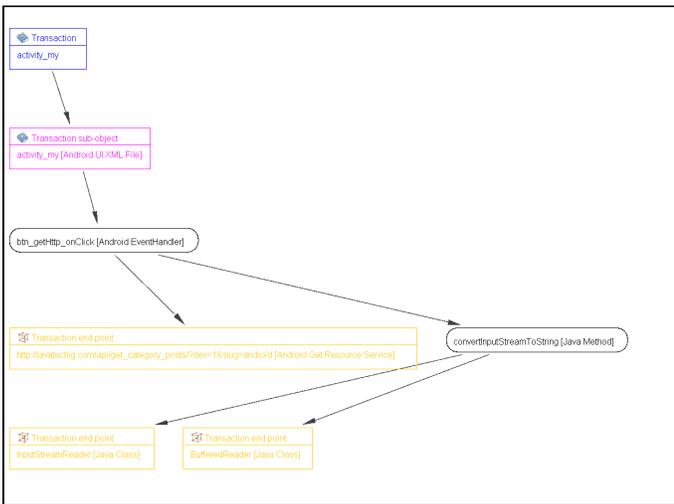
Web services linker

Click to enlarge



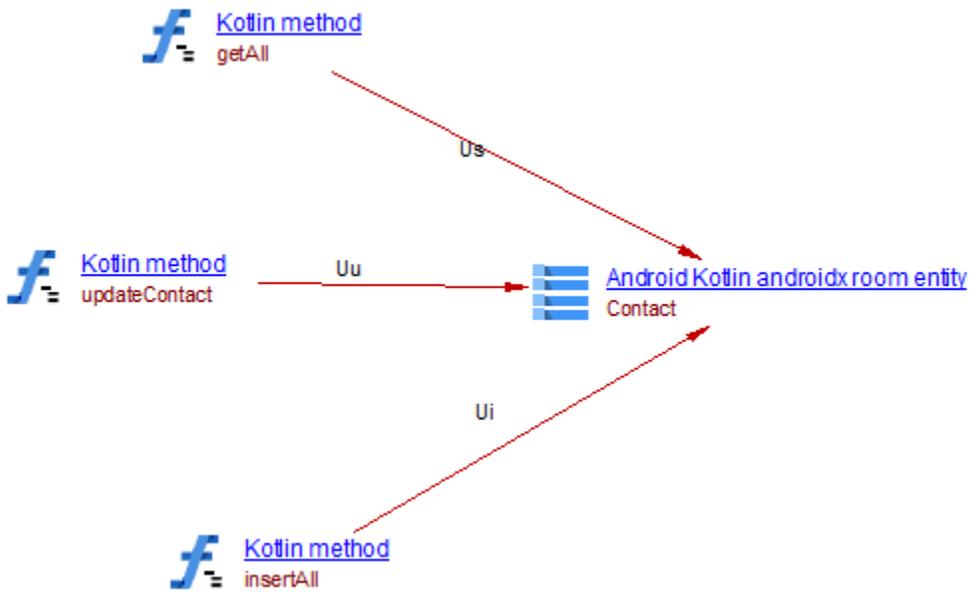
Function points (when using CAIP 8.3.0)

Click to enlarge



Supported libraries

androidx.room for kotlin (since 2.0.1-funcrel)



```
import androidx.room.Entity

@Entity(tableName = "contact", primaryKeys = ["contactId", "contactUuid"])
data class Contact(
    var contactId: String = "",
    var contactUuid: String = "",
    var displayName: String = "",
    var pictureUri: String? = null,
    var contactInfo: List<ContactAddressBook> = listOf()
) : Parcelable
```

```

import androidx.room.*

@Dao
abstract class ContactDao {
    @Query("SELECT * FROM contact")
    abstract fun getAll(): LiveData<List<Contact>>

    @Update
    abstract fun updateContact(contact: Contact)

    @Insert(onConflict = OnConflictStrategy.REPLACE)
    abstract fun insertAll(contact: List<Contact>): List<Long>
}

```

Objects

The following objects are displayed in CAST Enlighten:

Icon	Description
	Android EventHandler
	Android Get Resource Service
	Android Put Resource Service
	Android Post Resource Service
	Android Delete Resource Service
	Android Shared preferences, Android internal storage, Android external storage
	Android Application
	Android UI XML File
	Android exposed service, Android exposed activity, Android exposed receiver
	Android intent resource
	Android androidx room entity

Structural Rules

The following structural rules are provided:

2.0.1-funcrel	https://technologies.castsoftware.com/rules?sec=srs_android&ref= 2.0.1-funcrel
2.0.0-funcrel	https://technologies.castsoftware.com/rules?sec=srs_android&ref= 2.0.0-funcrel

Known limitations

- CAST recommends using the most recent stable release of CAST AIP with the extension for best results.
- Some links won't be resolved because of missing jar file.
- Lambda expression is supported from CAST AIP 8.2.11. There are still some limitations for transactions.
- Web Service URI name isn't stable for more complex situations.
- Currently there is no discoverer available, so the extension will not recognize Android projects. Users need to create **JEE or Kotlin Analysis Units manually** in order to analyze the Android application source code.