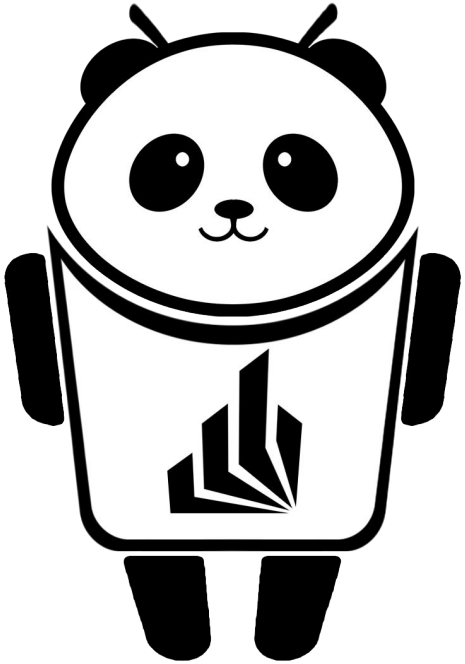


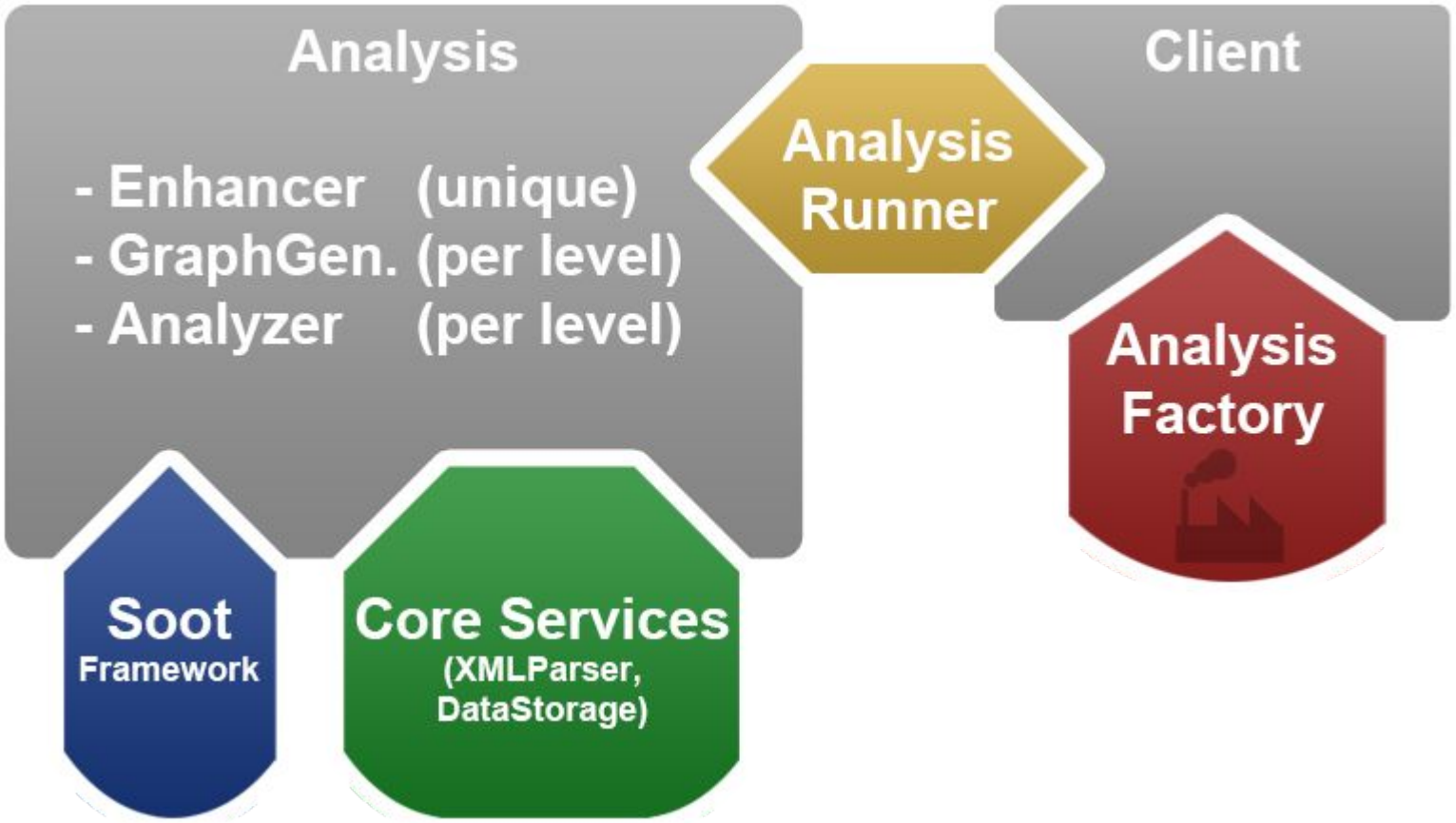
PAndA²

Paderborn **And**roid **A**pp **A**nalysis

Outline

- PAndA² Recap
- Motivation & Demo
  - Examples:
    - Intra-App Resource Usage (Level 1)
    - Inter-App Resource Usage (Level 2b)
    - Intra-App Information Flow Control (Level 2a)
- Quality Assurance
  - Tools
    - JUnit
    - Automated systemtests
- The Project (plan & status)

PAndA² Recap



Motivation

1997 LUCASFILM LDT. - Star Wars IV - A new Hope

- Is R2D2 trustworthy?

Motivation

1997 LUCASFILM LDT. - Star Wars IV - A new Hope

- Is R2D2 trustworthy? **YES, OF COURSE**

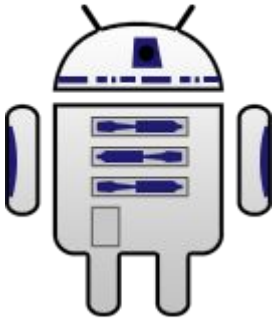
Motivation

1997 LUCASFILM LDT. - Star Wars IV - A new Hope

- Is R2D2 trustworthy? **YES, OF COURSE**
- **But:**
  - What about his Software?
  - Are there any possibly malicious Apps?

Intra-App Resource Usage

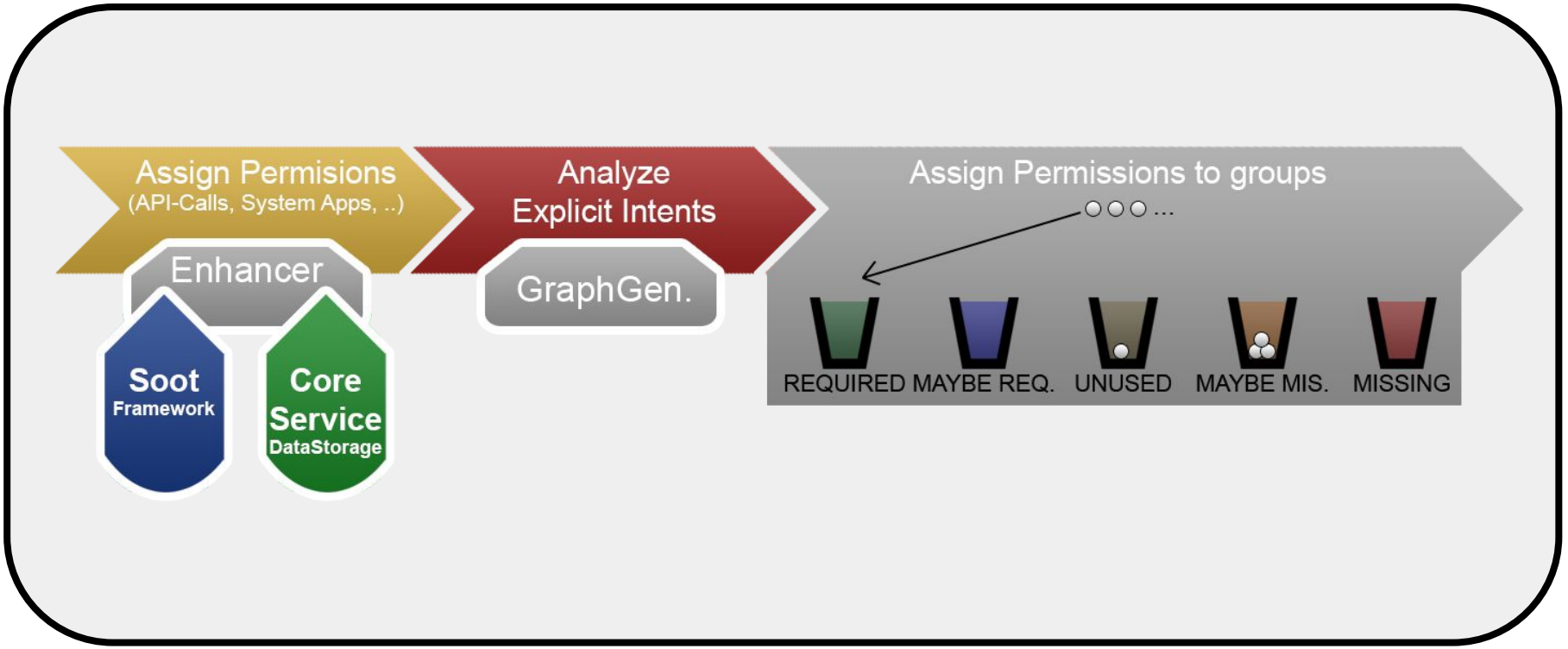
- App 1 - Uses ONLY the Camera



App 1

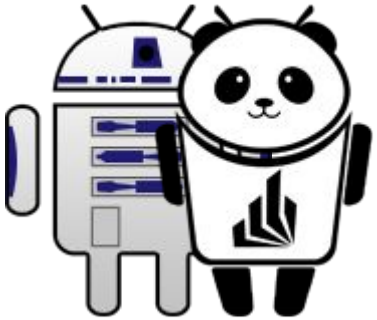
Intra-App Resource Usage

- App 1 - Uses ONLY the Camera



Intra-App Resource Usage

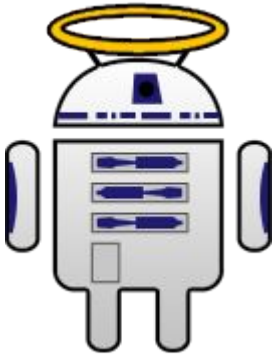
- App 1 - Uses ONLY the Camera



App 1

Intra-App Resource Usage

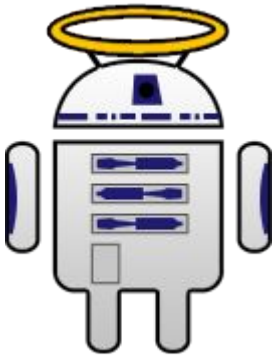
- App 1 - Uses ONLY the Camera



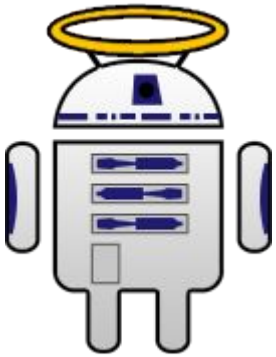
App 1
CAMERA

Intra-App Resource Usage

- App 3 - Uses SMS



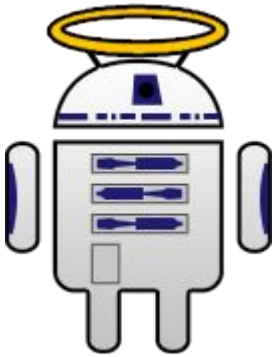
App 1
CAMERA



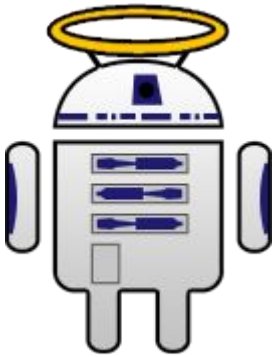
App 2
GPS

App 3
SMS
GPS

Inter-App Resource Usage



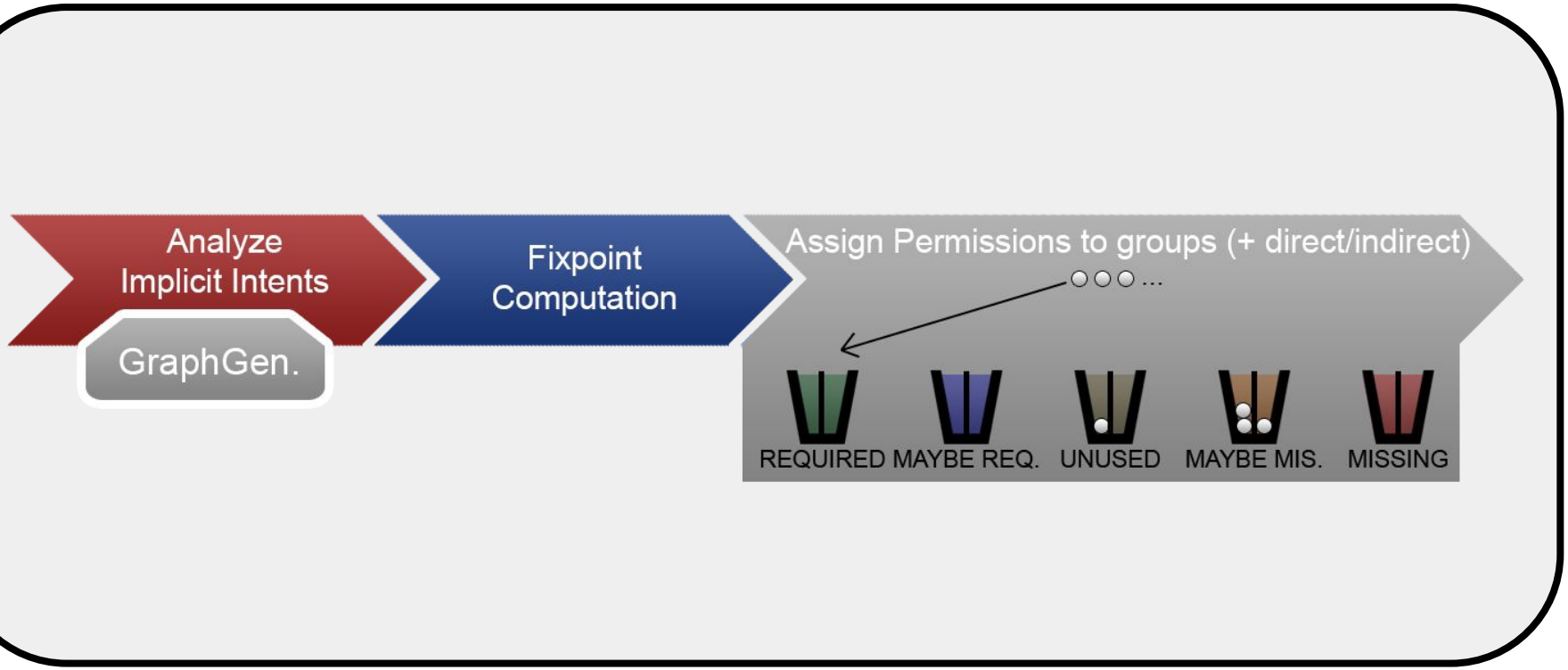
App 1
CAMERA



App 2
GPS

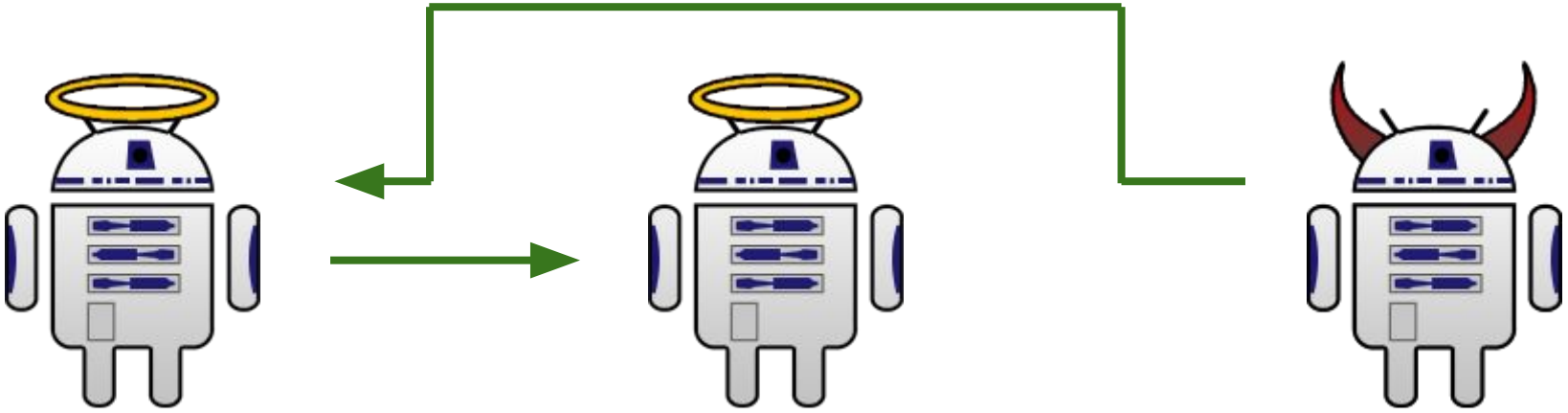
App 3
SMS
CAMERA

Inter-App Resource Usage



Inter-App Resource Usage

- Intent: →



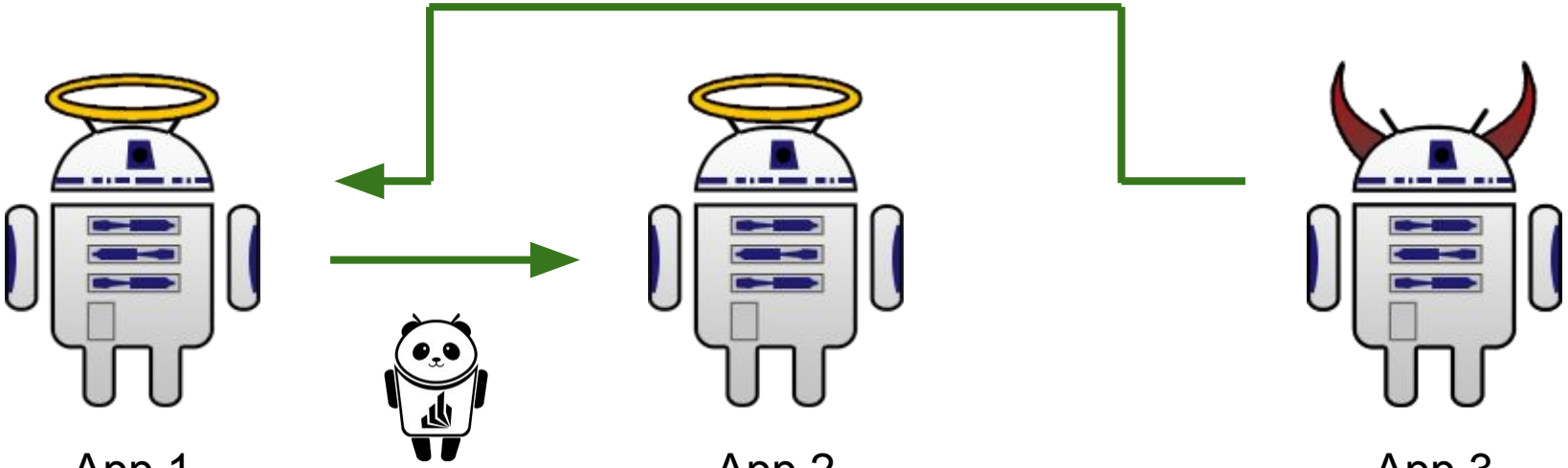
App 1
CAMERA

App 2
GPS

App 3
SMS
CAMERA

Inter-App Resource Usage

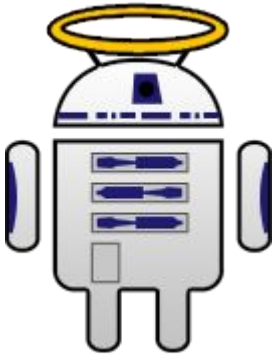
- Intent: →



Inter-App Resource Usage

- Intent:

App 1
CAMERA
GPS
SMS

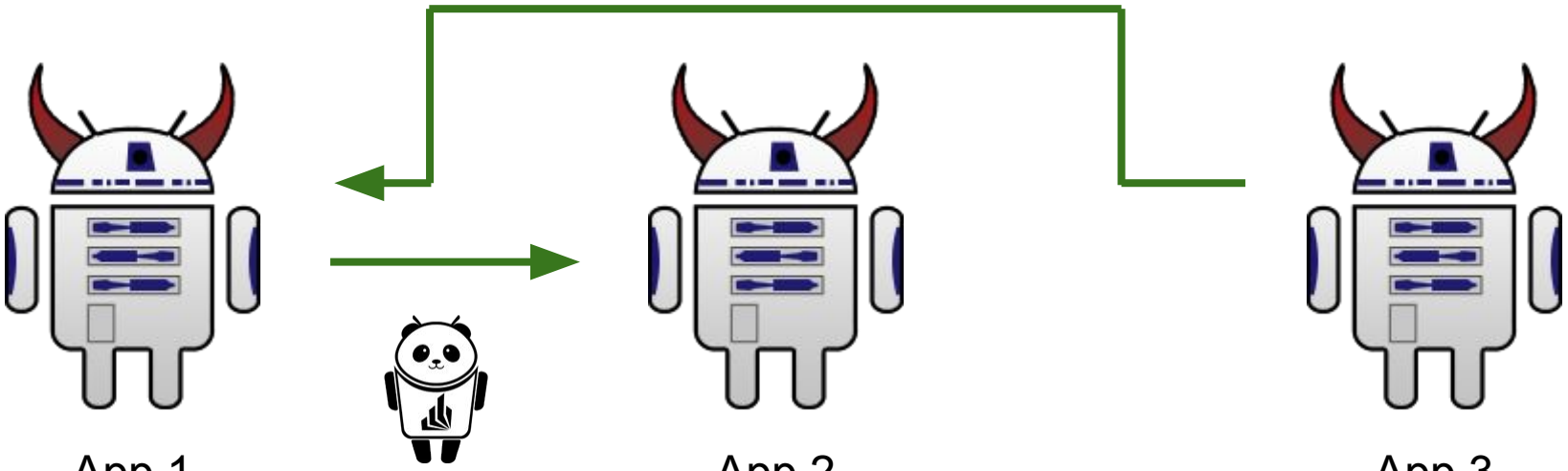


App 2
GPS
CAMERA

App 3
SMS
CAMERA
GPS

Inter-App Resource Usage

- Intent: →



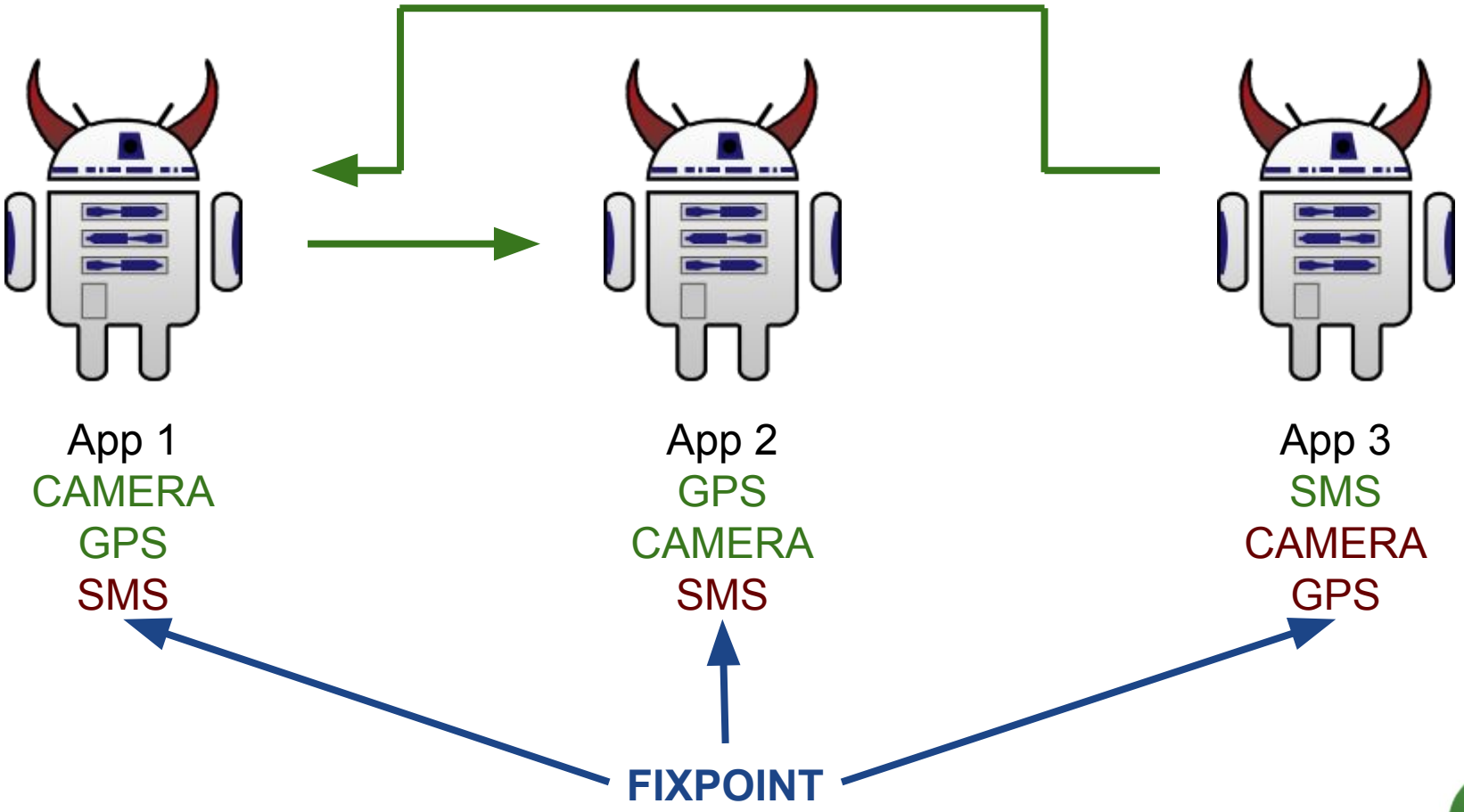
App 1
CAMERA
GPS
SMS

App 2
GPS
CAMERA
SMS

App 3
SMS
CAMERA
GPS

Inter-App Resource Usage

- Intent: →



Inter-App Resource Usage

- Is there any path from source to sink?
  - CAMERA → SMS
  - GPS → SMS

App 3
SMS
CAMERA
GPS

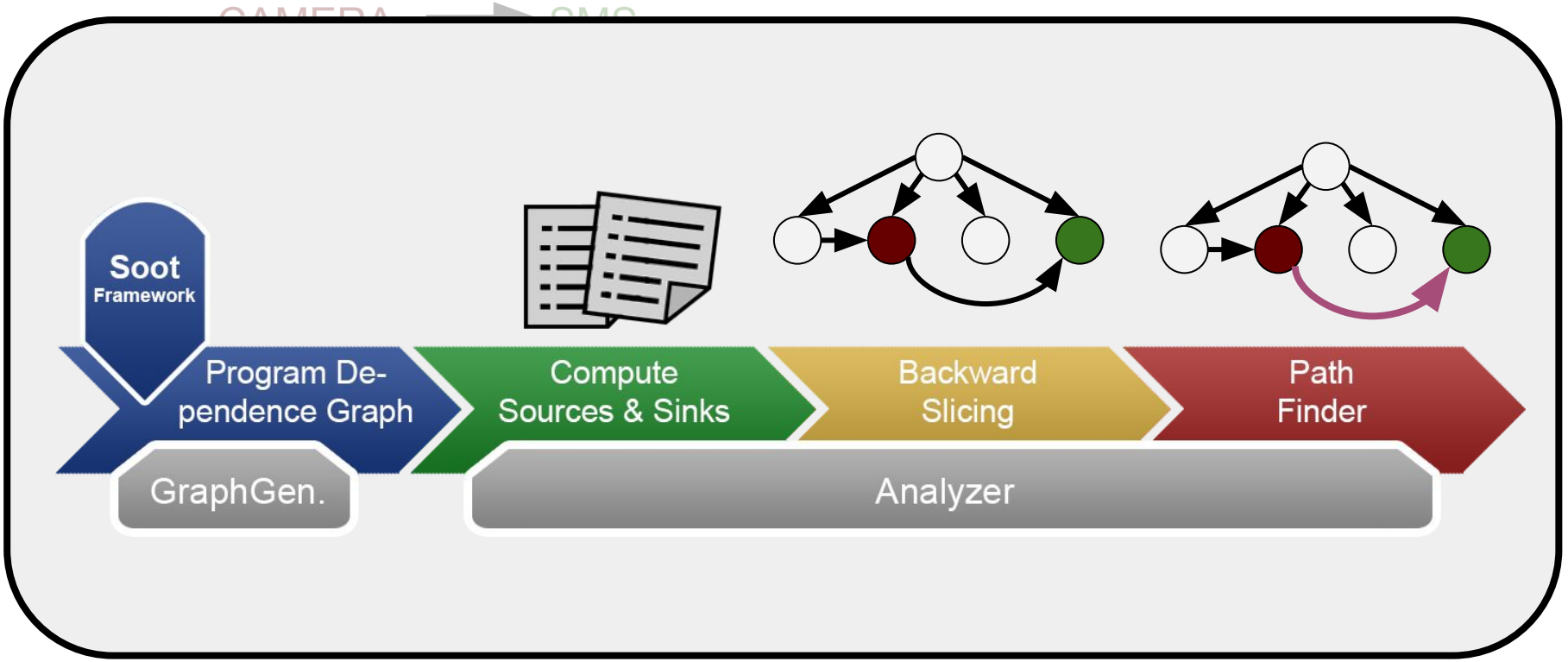
Intra-App Information Flow Control

- Is there any path from source to sink?
  - CAMERA → SMS
  - GPS → SMS

App 3
SMS
CAMERA
GPS

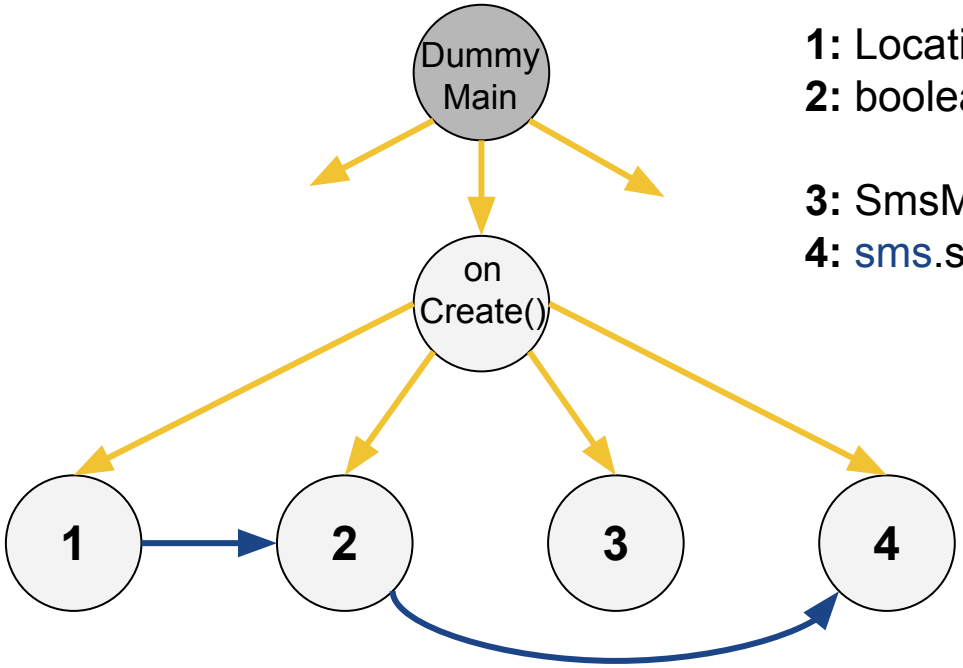
Intra-App Information Flow Control

- Is there any path from source to sink?
  - CAMERA ⟶ SMS



- Is there any path from source to sink?
  - CAMERA → SMS
  - GPS → SMS

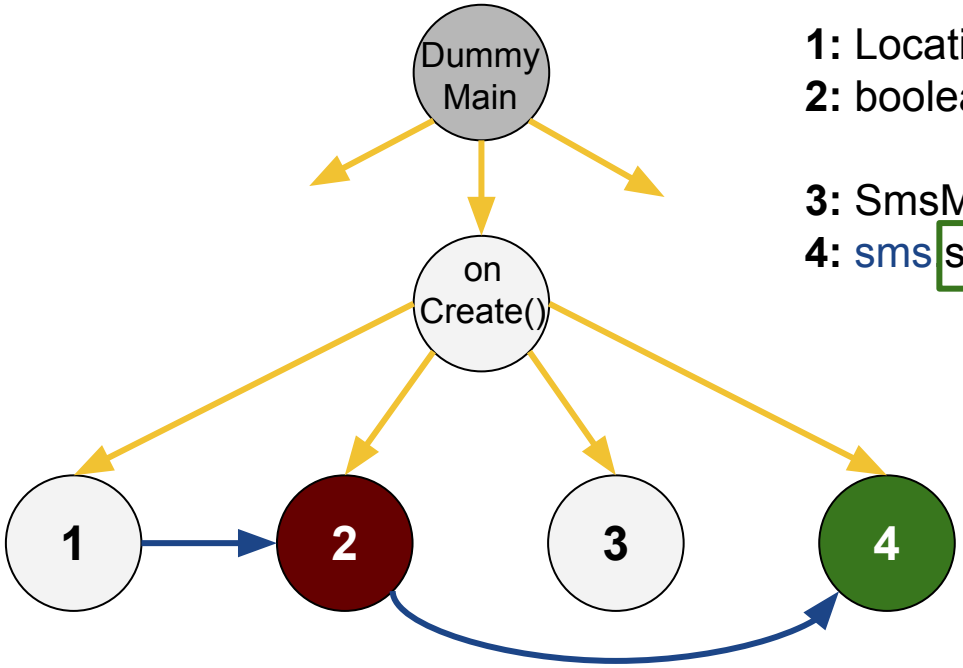
Step 1 / 4: Compute Program Dependence Graph




```
1: LocationManager st = getSystemService(...);
2: boolean secretData = st.sendExtraCommand(...);

3: SmsManager sms = SmsManager.getDefault();
4: sms.sendTextMessage(secretData);
```

- Is there any path from source to sink?
  - CAMERA → SMS
  - GPS → SMS

Step 2 / 4: Determine Sources and Sinks

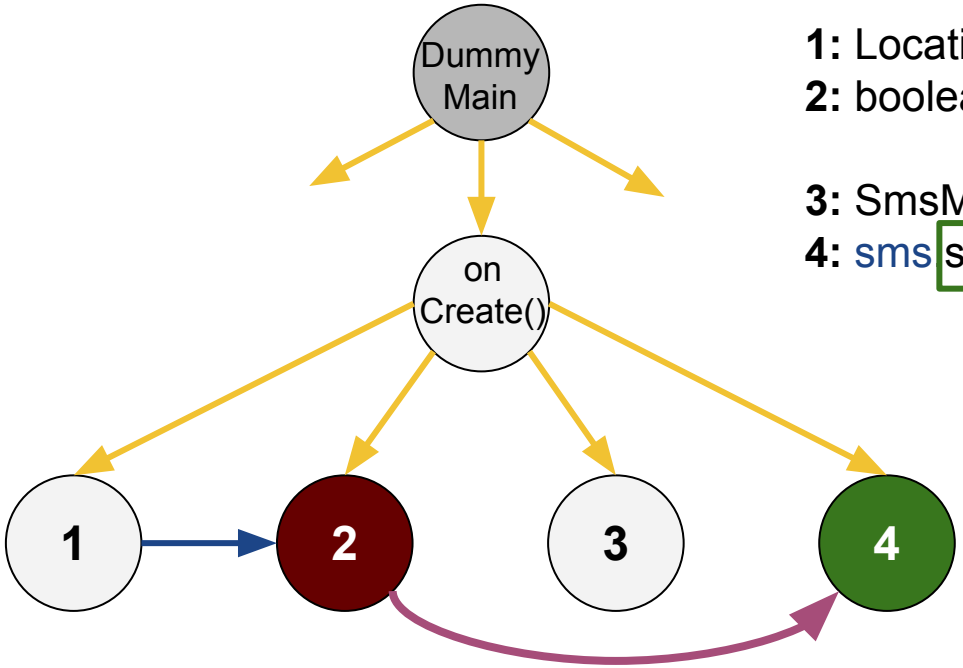



```
1: LocationManager st = getSystemService(...);
2: boolean secretData = st.sendExtraCommand(...);

3: SmsManager sms = SmsManager.getDefault();
4: sms.sendTextMessage(secretData);
```

Intra-App Information Flow Control

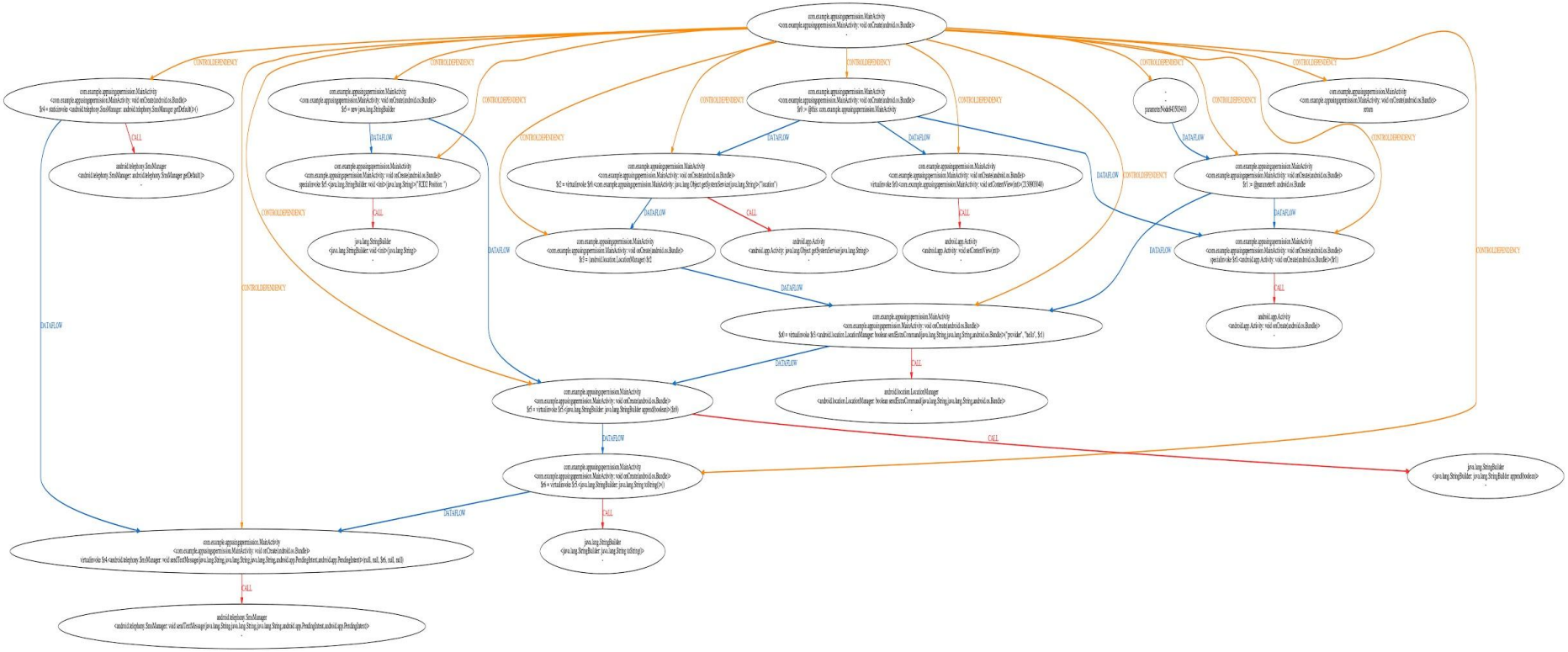
- Is there any path from source to sink?
  - CAMERA → SMS
  - GPS → SMS

Step 3,4 / 4:
- **Find information flow by backward-slicing**
- **Compute paths between sources and sinks**



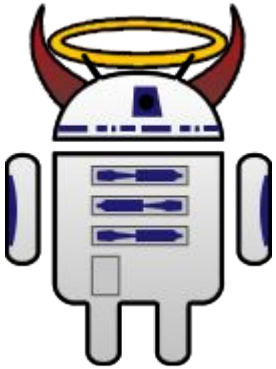

```
1: LocationManager st = getSystemService(...);
2: boolean secretData = st.sendExtraCommand(...);

3: SmsManager sms = SmsManager.getDefault();
4: sms.sendTextMessage(secretData);
```

Intra-App Information Flow Control



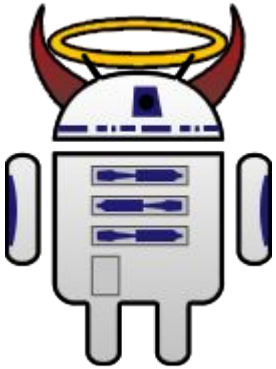
Intra-App Information Flow Control

- Is there any path from source to sink?
  - ~~CAMERA → SMS~~
  - GPS → SMS



Intra-App Information Flow Control

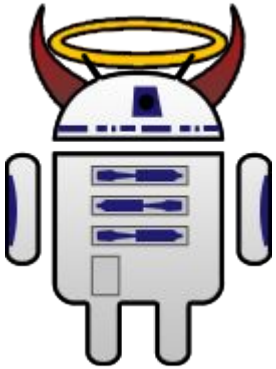
- Is there any path from source to sink?
  - ~~CAMERA → SMS~~
  - GPS → SMS



- Luckily R2D2's information are trustworthy in the end.

Intra-App Information Flow Control

- Is there any path from source to sink?
  - ~~CAMERA → SMS~~
  - GPS → SMS



- Luckily R2D2's information are trustworthy in the end.
- **But:**
  - What if PAndA² was not working correct?

- Tools for Quality Assurance

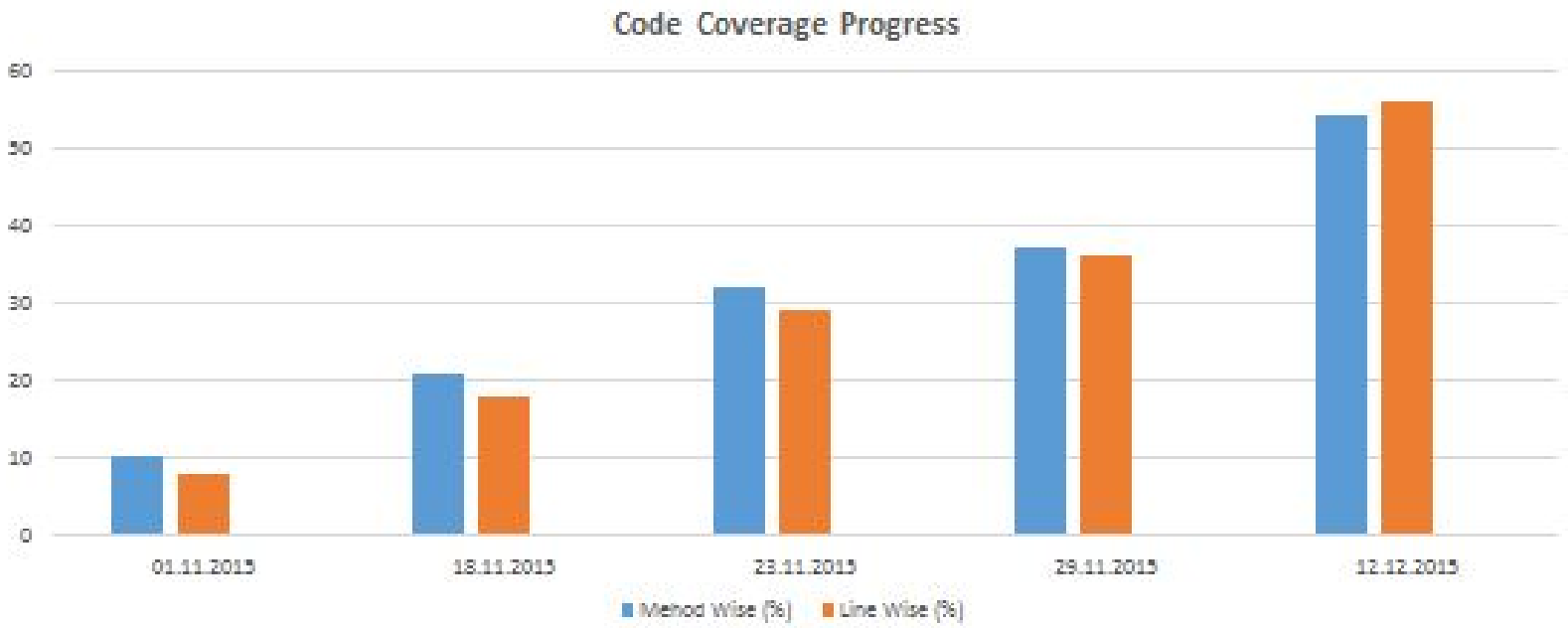
EclEmma: Code Coverage
  - Code coverage through Junit test : Methods, Lines, Branches, Instructions

- **PMD:** Code Quality
  - Code quality check based on predefined rules (> 250 rules)

- **Code Pro:** Similar Code, Dead Code
  - Checks for duplicate code and not reachable code.

Quality Assurance

- Current Code Coverage
  - JUnit test cases for all packages except GUI
  - Progress tracking in regular intervals



- Current Code Coverage

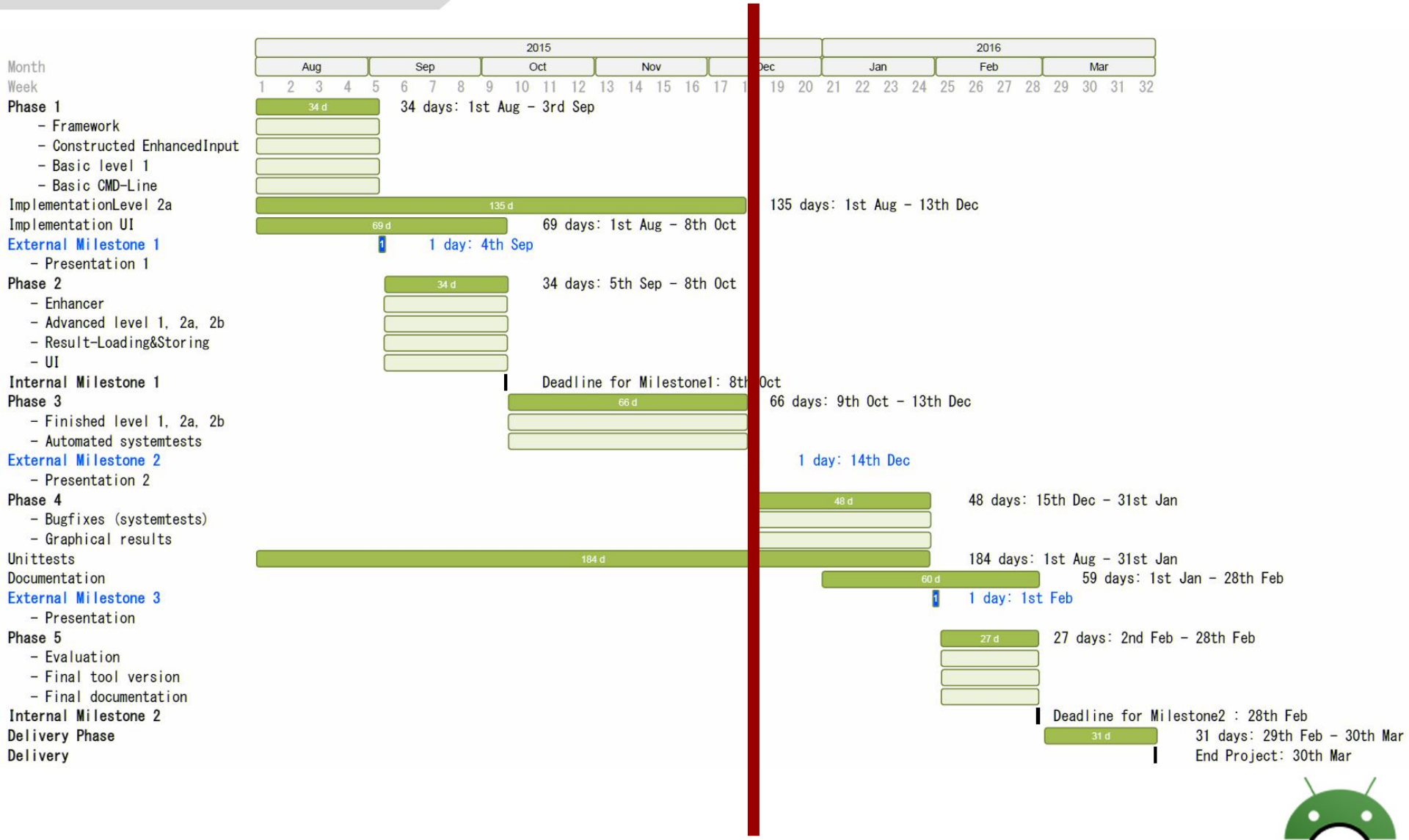
Element	Coverage	Covered Methods	Missed Methods	Total Methods
src	45.8 %	513	606	1,119
de.upb.pga3.panda2.client.gui	38.5 %	77	123	200
de.upb.pga3.panda2.client.gui2	0.0 %	0	112	112
de.upb.pga3.panda2.client.core	28.1 %	34	87	121
de.upb.pga3.panda2.extension.lvl2a	27.9 %	24	62	86
de.upb.pga3.panda2.core.services	59.1 %	78	54	132
de.upb.pga3.panda2.extension.lvl2a.graphgenerator	52.1 %	50	46	96
de.upb.pga3.panda2.core.datastructures	72.8 %	67	25	92

Method Wise Coverage (Except GUI):
Methods Covered(433)/ Total Methods excluding GUI (802)= **54%**

Line Wise Coverage (Except GUI):
Lines Covered (3274) / Total Lines (5866) = **55.81%**

- Automated Testing
  - Java program for executing (>10) test cases in one go.
  - Invokes our tool’s jar file along with input parameters
  - Informs if any test case fails

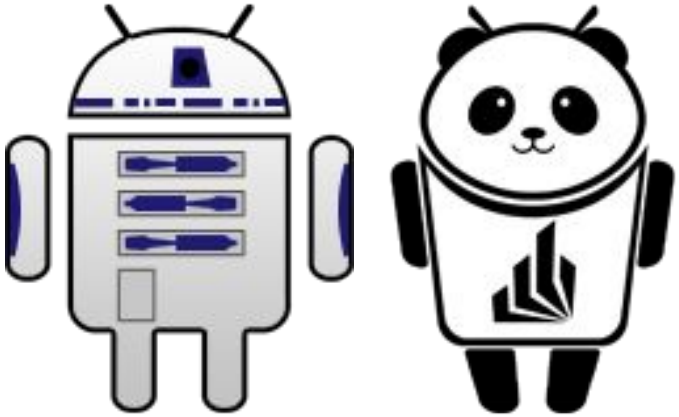
The Project


2015
2016
Month
Aug
Sep
Oct
Nov
Dec
Jan
Feb
Mar
Week
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32
Phase 1
34 d
34 days: 1st Aug - 3rd Sep
- Framework
- Constructed EnhancedInput
- Basic level 1
- Basic CMD-Line
ImplementationLevel 2a
135 d
135 days: 1st Aug - 13th Dec
Implementation UI
69 d
69 days: 1st Aug - 8th Oct
External Milestone 1
1
1 day: 4th Sep
- Presentation 1
Phase 2
34 d
34 days: 5th Sep - 8th Oct
- Enhancer
- Advanced level 1, 2a, 2b
- Result-Loading&Storing
- UI
Internal Milestone 1
Deadline for Milestone1: 8th Oct
Phase 3
66 d
66 days: 9th Oct - 13th Dec
- Finished level 1, 2a, 2b
- Automated systemtests
External Milestone 2
1 day: 14th Dec
- Presentation 2
Phase 4
48 d
48 days: 15th Dec - 31st Jan
- Bugfixes (systemtests)
- Graphical results
Unittests
184 d
184 days: 1st Aug - 31st Jan
Documentation
60 d
59 days: 1st Jan - 28th Feb
External Milestone 3
1
1 day: 1st Feb
- Presentation
Phase 5
27 d
27 days: 2nd Feb - 28th Feb
- Evaluation
- Final tool version
- Final documentation
Internal Milestone 2
Deadline for Milestone2 : 28th Feb
Delivery Phase
31 d
31 days: 29th Feb - 30th Mar
Delivery
End Project: 30th Mar

The Project

- **Previously** (Milestone 1):
  - Only basic level 1
  - No unittests
  - GUI without functionality
  - ...
- **Current** (Milestone 2):
  - Analyzes are finished
  - Unittests are ~½ done
  - UI including the GUI is done
  - Changes to the architecture are briefly documented
- **Next** (Delivery):
  - Graphical analysis Results (SVG)
  - Documentation
  - Evaluation
  - Tests ⟷ Bugfixes

Summary



- Information is trustworthy
- PAndA² works as intended
- We are well within our schedule

References

- Backward Slicing Algorithm:
  - Christian Hammer. **Information Flow Control for Java - a comprehensive approach based on Path Conditions in Dependence Graphs**. IEEE International Symposium on Secure Software Engineering (ISSSE 2006), Arlington, VA, March 2006

- List of permissions used in the Enhancer: **PScout**
  - Based on Soot
  - Computing available Permissions based on Android Source code
    - Kathy Wain Yee Au, Yi Fan Zhou, Zhen Huang and David Lie. **PScout: Analyzing the Android Permission Specification**. In the Proceedings of the 19th ACM Conference on Computer and Communications Security (CCS 2012). October 2012.

References

- List of sources and sinks: **SuSi**
  - Based on Soot
  - Directly provides a list of sources and sinks
    - Steven Arzt, Siegfried Rasthofer and Eric Bodden. **Susi: a tool for the fully automated classification and categorization of android sources and sinks**.

Summary