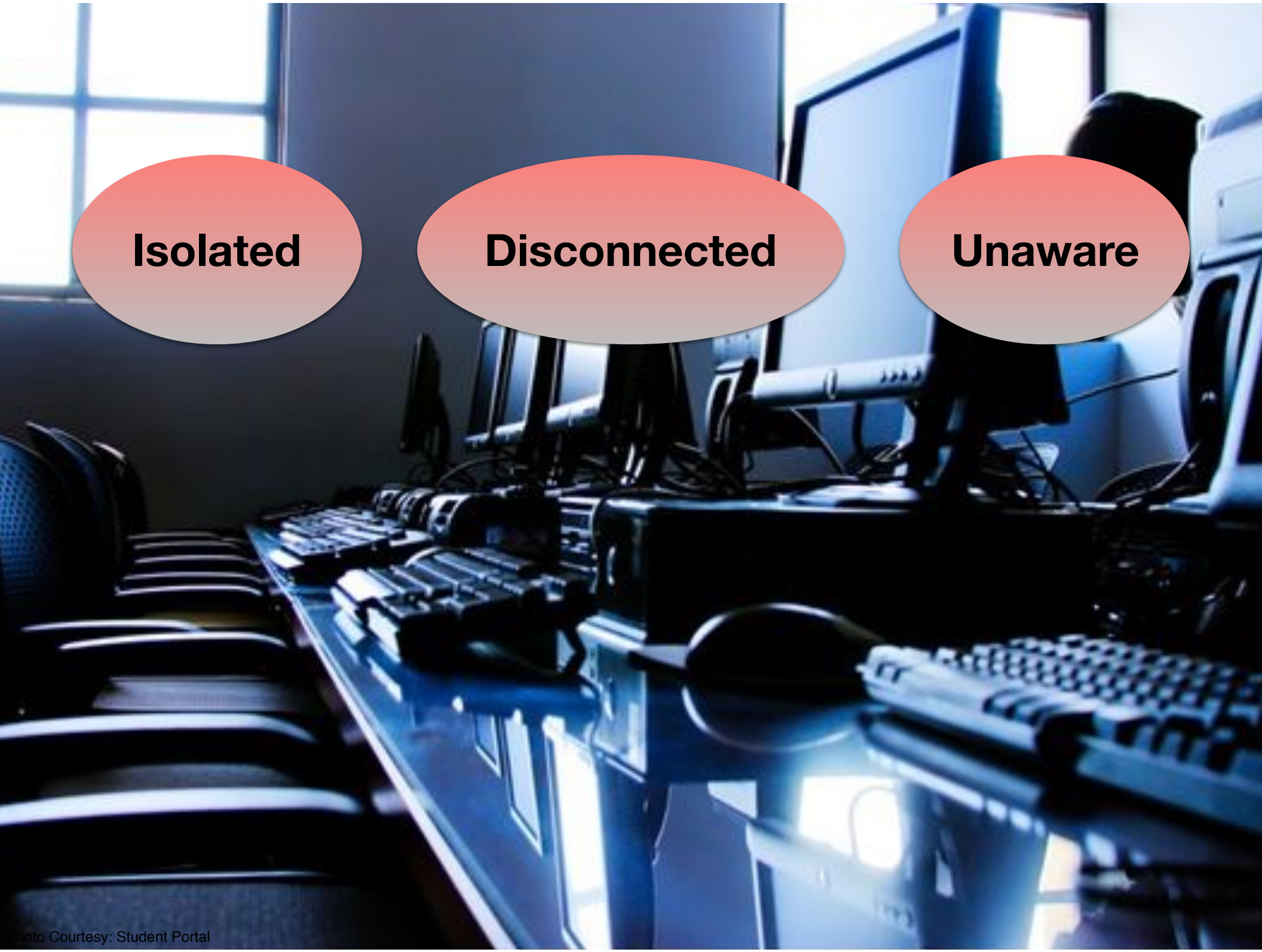
CAreDroid: Adaptation Framework for Android Context-Aware Applications

Salma Elmalaki
Lucas Wanner
Mani Srivastava

Networked & Embedded Systems Laboratory

NESL Networked & Embedded Systems Laboratory

UCLA

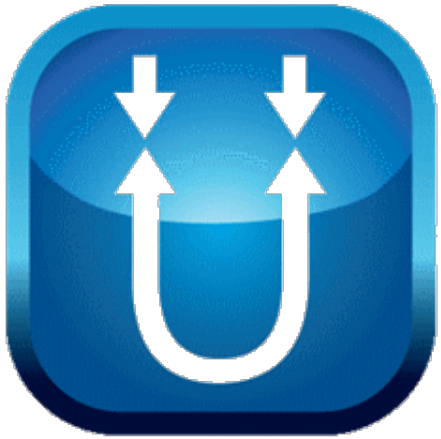

Isolated
Disconnected
Unaware
Photo Courtesy: Student Portal

Computing … From Isolation to Awareness

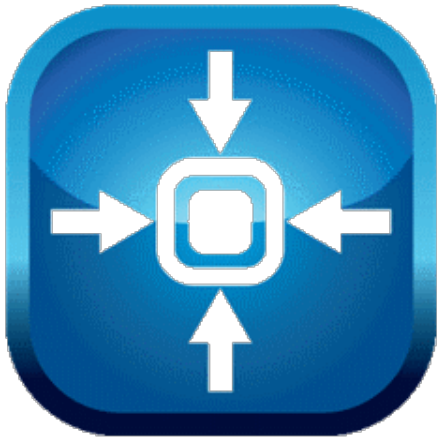
Photo Courtesy: Mobile innovation lab

eCompass

Magnetometers

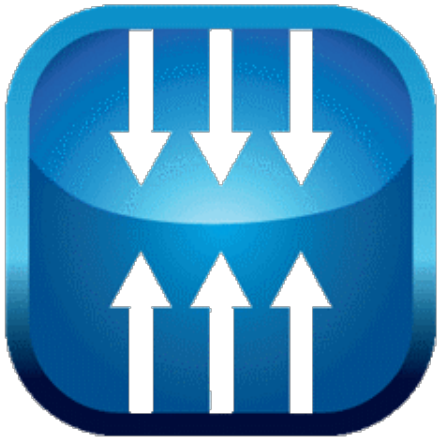


Accelerometers

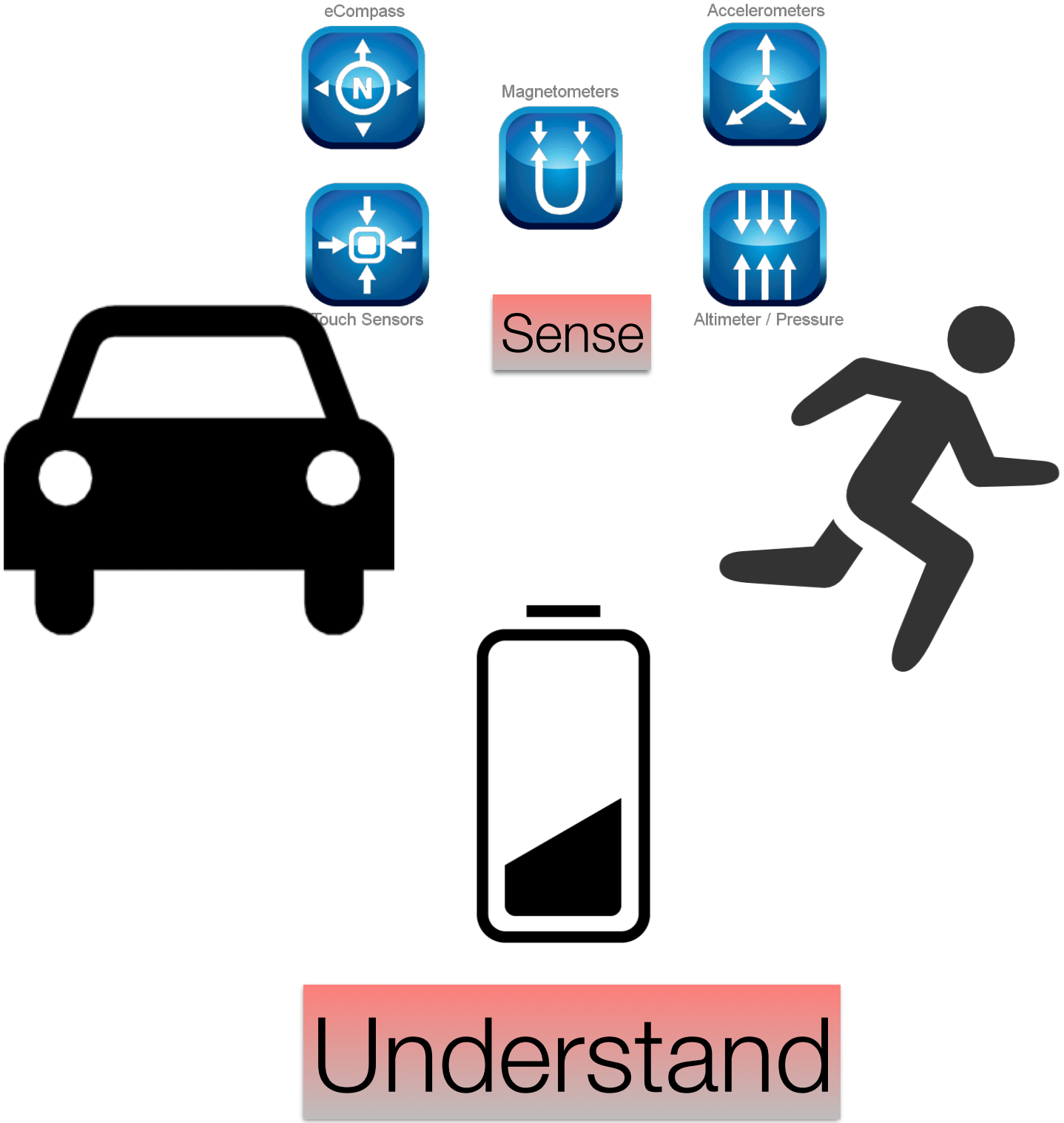


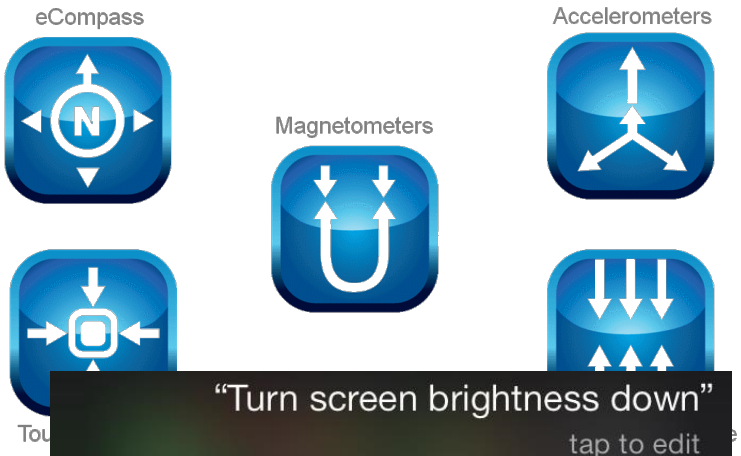
Touch Sensors

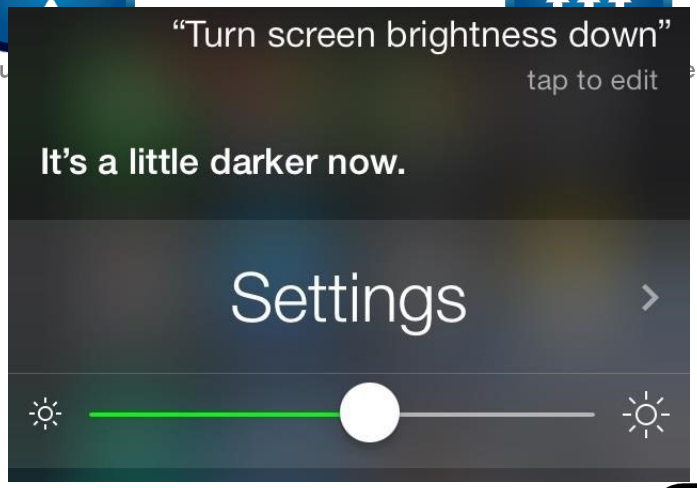
Sense



Altimeter / Pressure

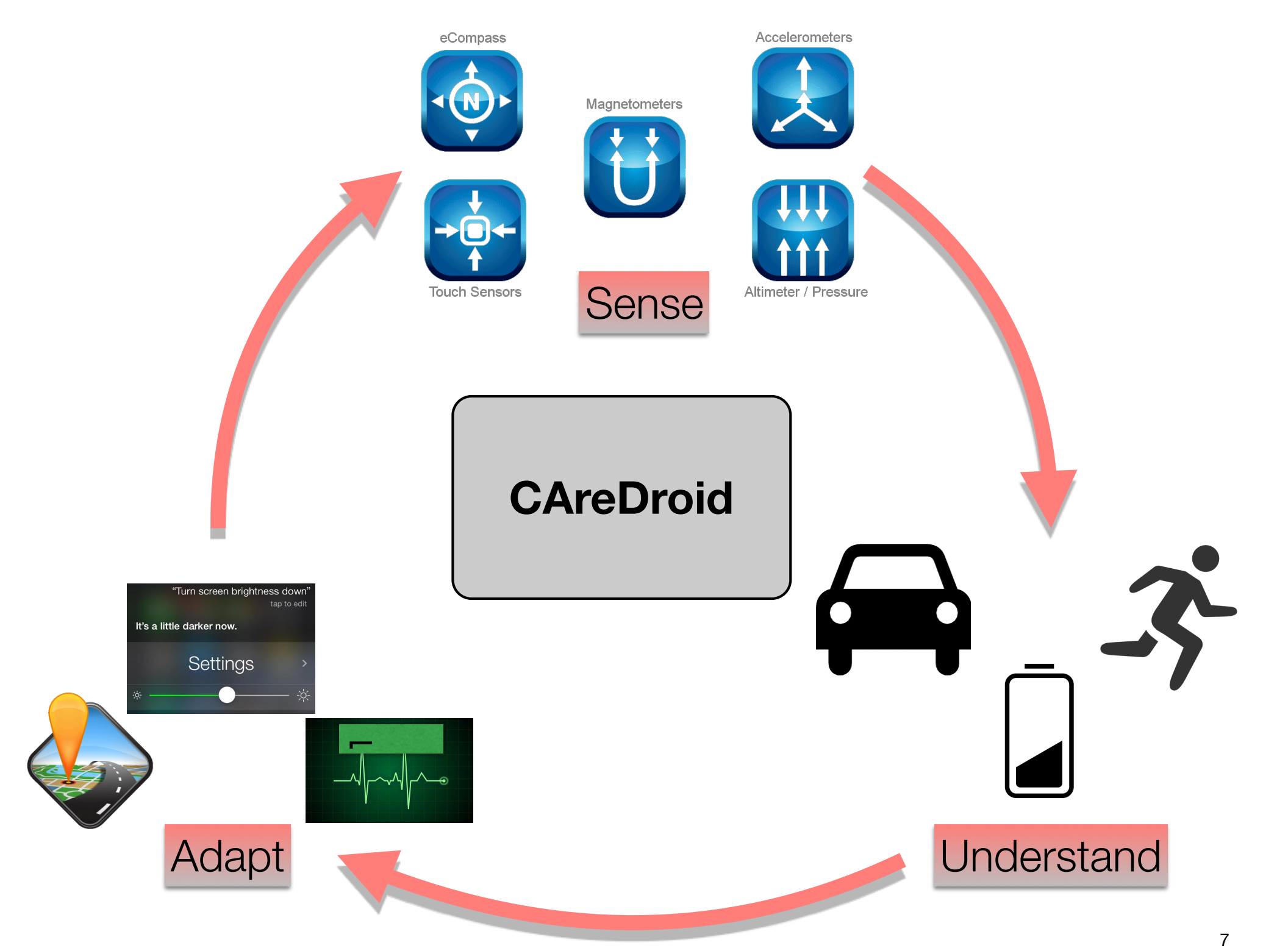

eCompass
Accelerometers
Magnetometers
Touch Sensors
Altimeter / Pressure
Sense
Understand





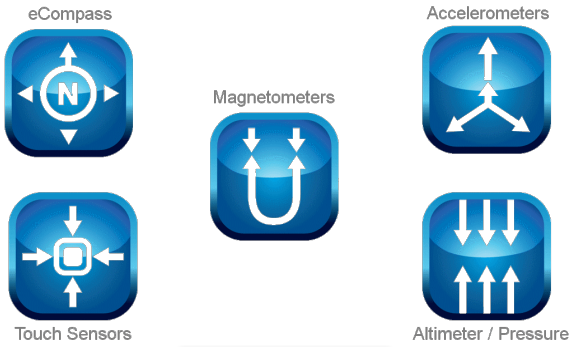
Adapt

Understand

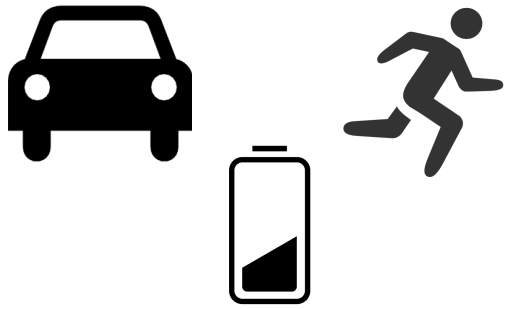

eCompass
Accelerometers
Magnetometers
Touch Sensors
Altimeter / Pressure
Sense
CAreDroid
"Turn screen brightness down"
tap to edit
It's a little darker now.
Settings
Adapt
Understand

Challenges

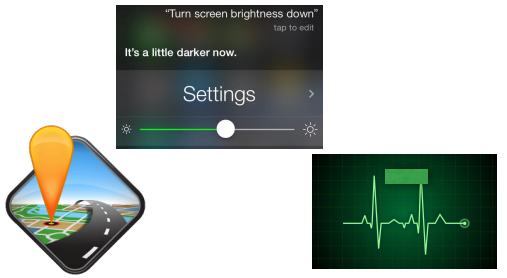
1. **Automatically adapt to the environmental change at runtime.**
2. **Adaptation should be burden-free for the user.**
3. **Overhead of reading the environmental context should be small.**



Sense

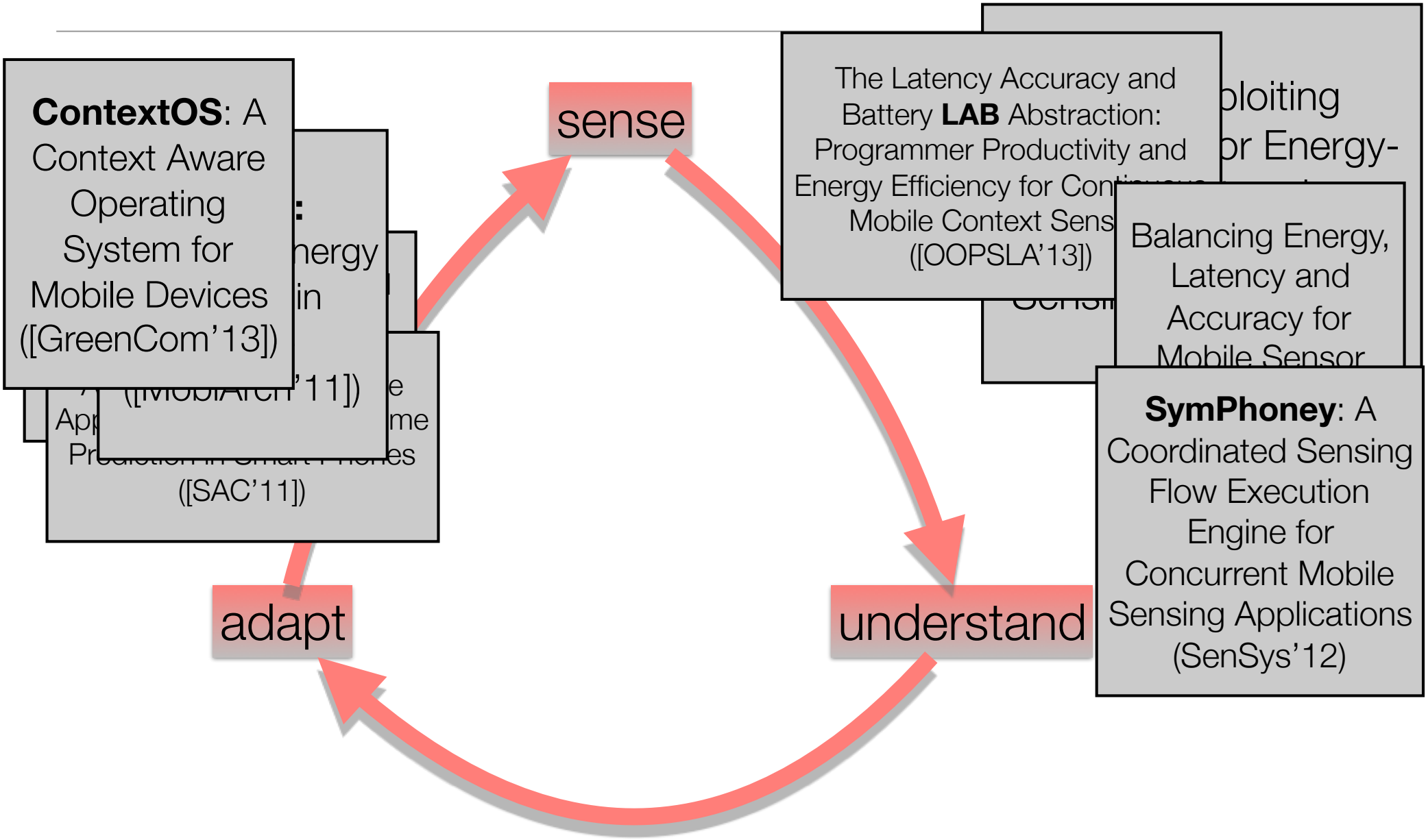


Understand

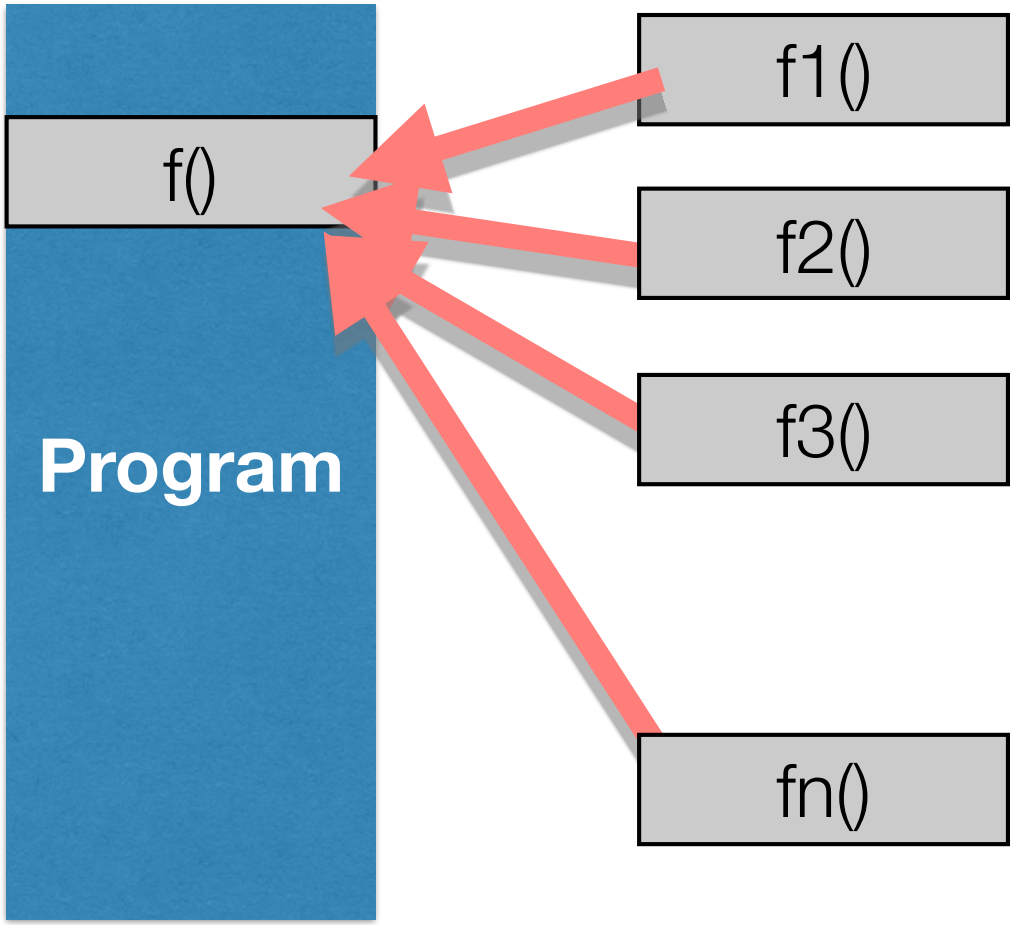


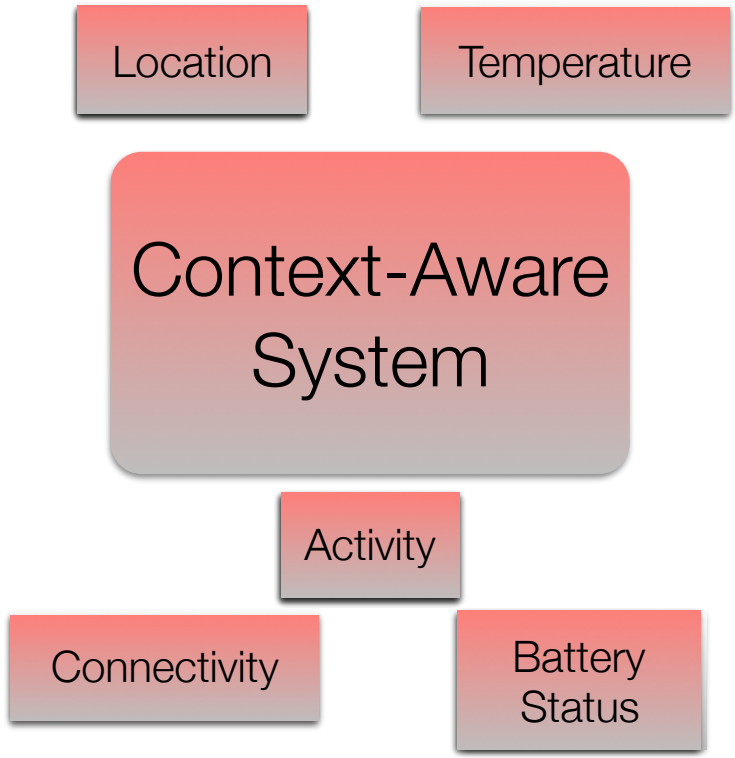
Adapt

What is Context-awareness?



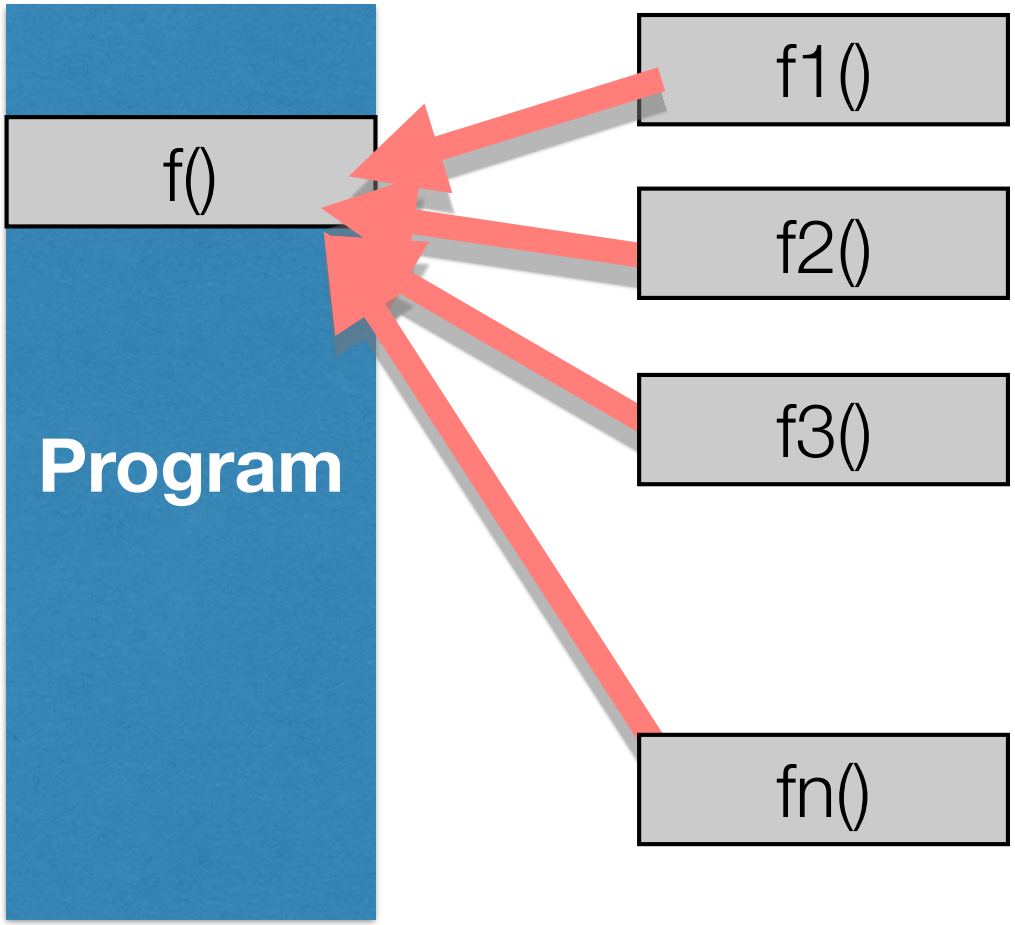
Adaptation Realization

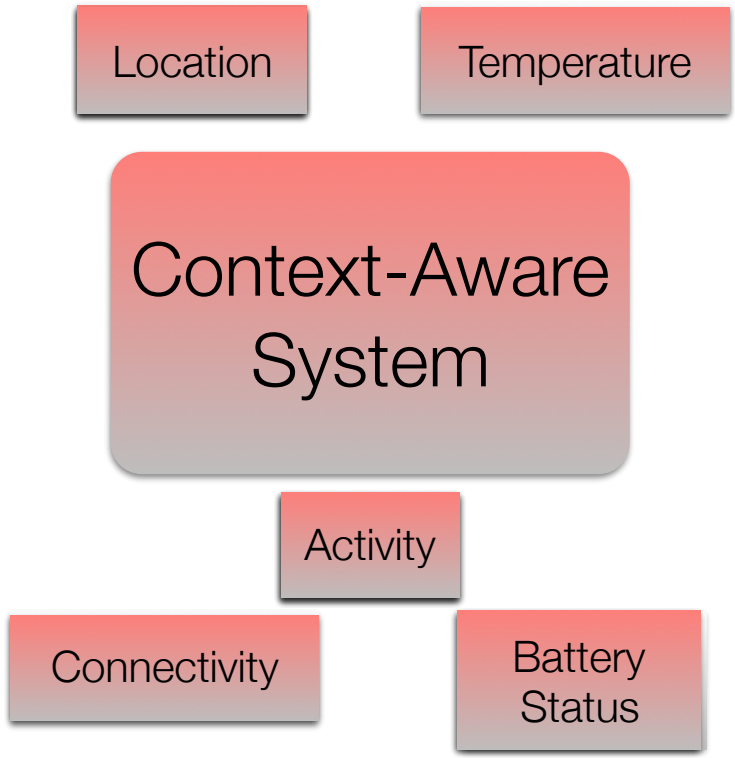




Function Adaptation

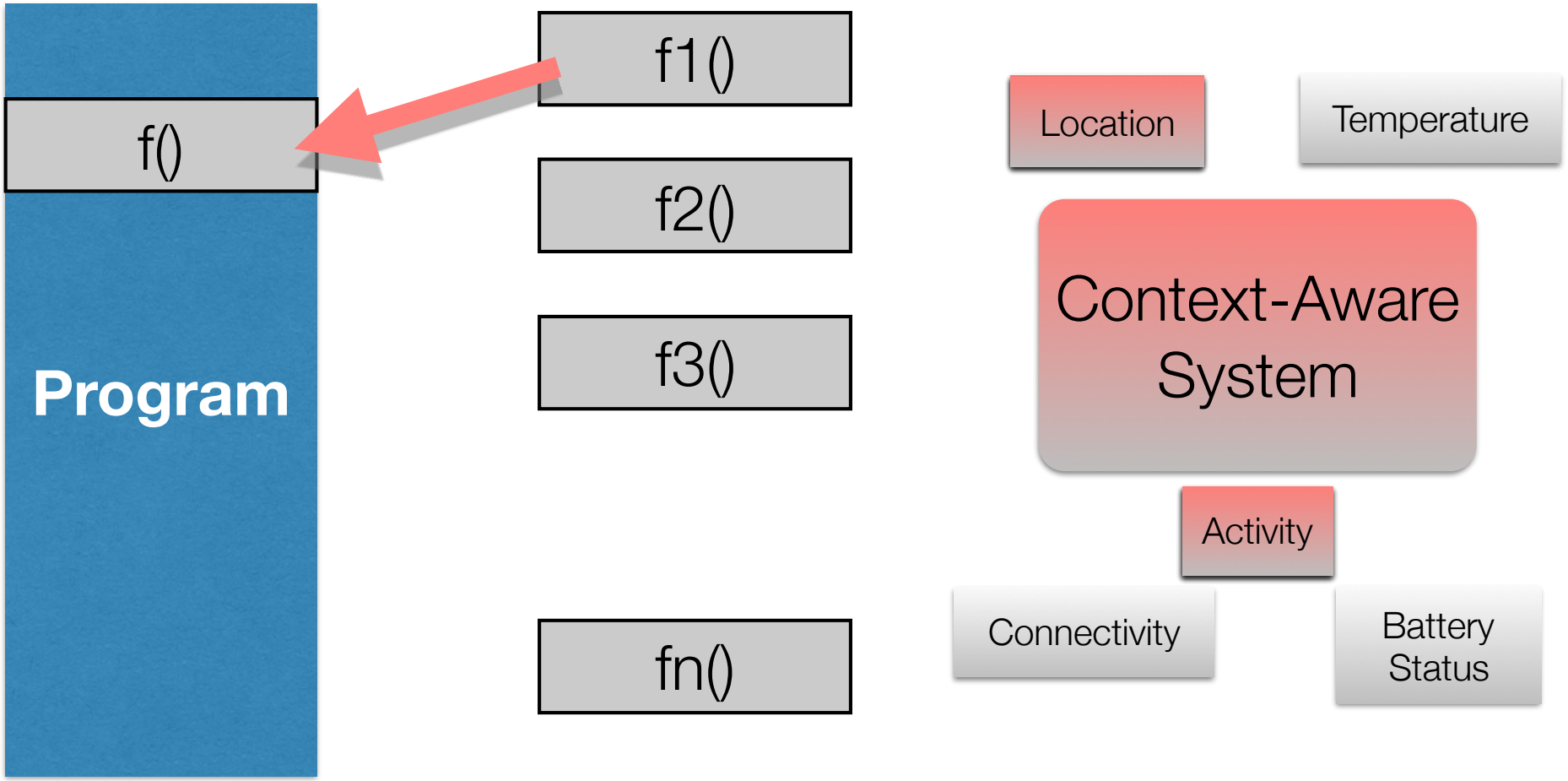
Adaptation Realization





Function Adaptation

Adaptation Realization



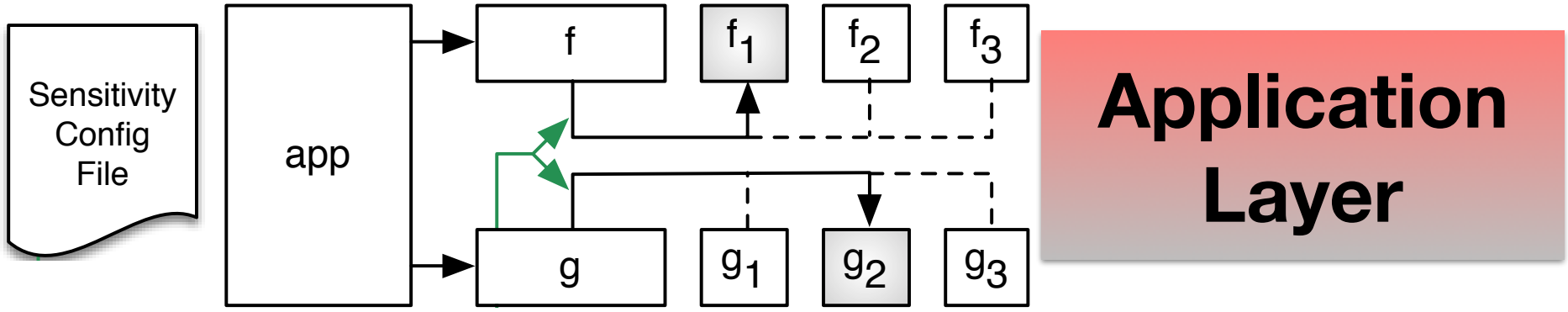
Function Adaptation

CAreDroid

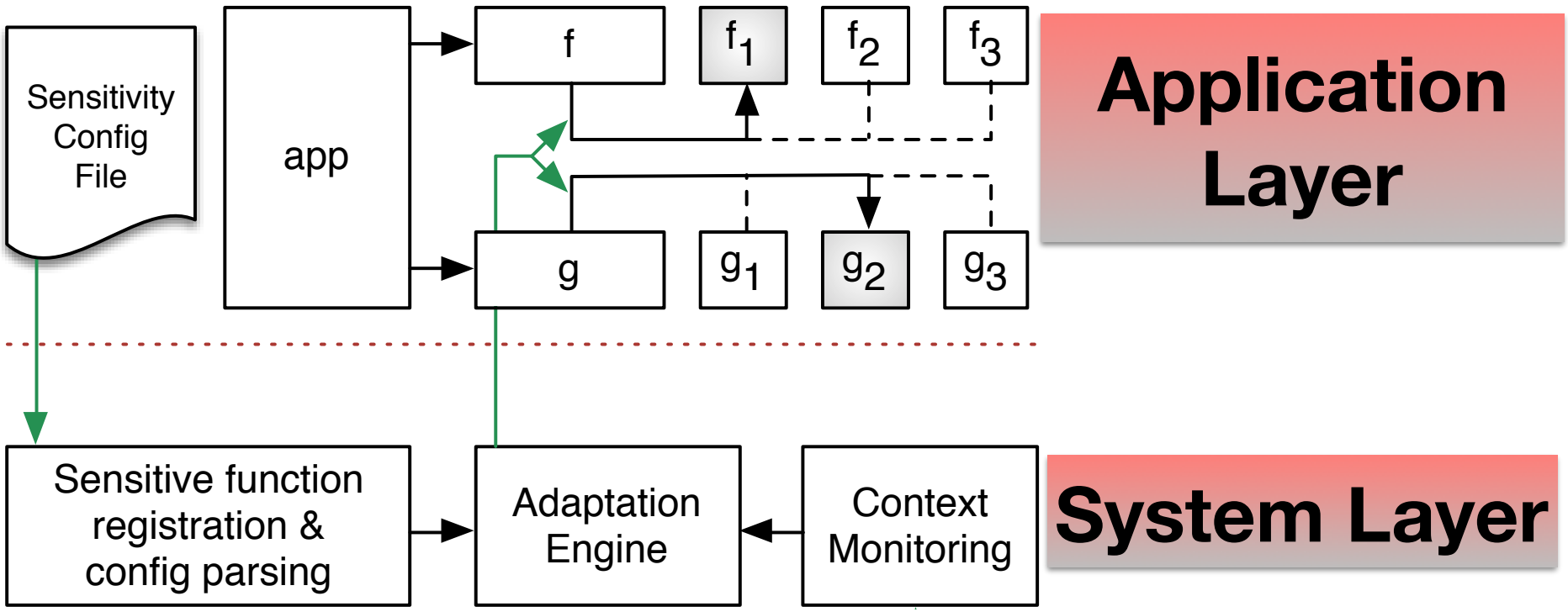
Architecture

Usability **Performance**

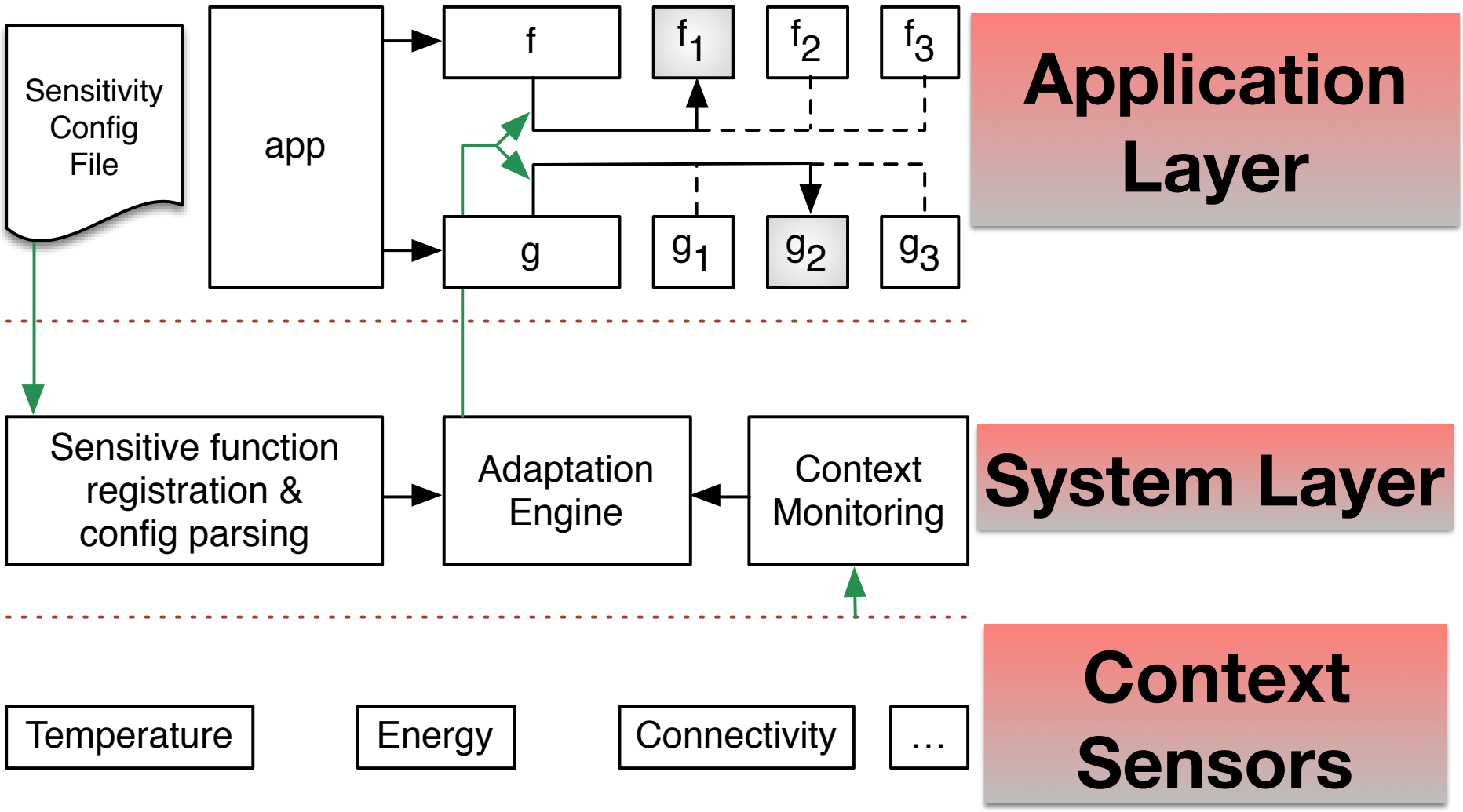
CAreDroid Architecture



CAreDroid Architecture

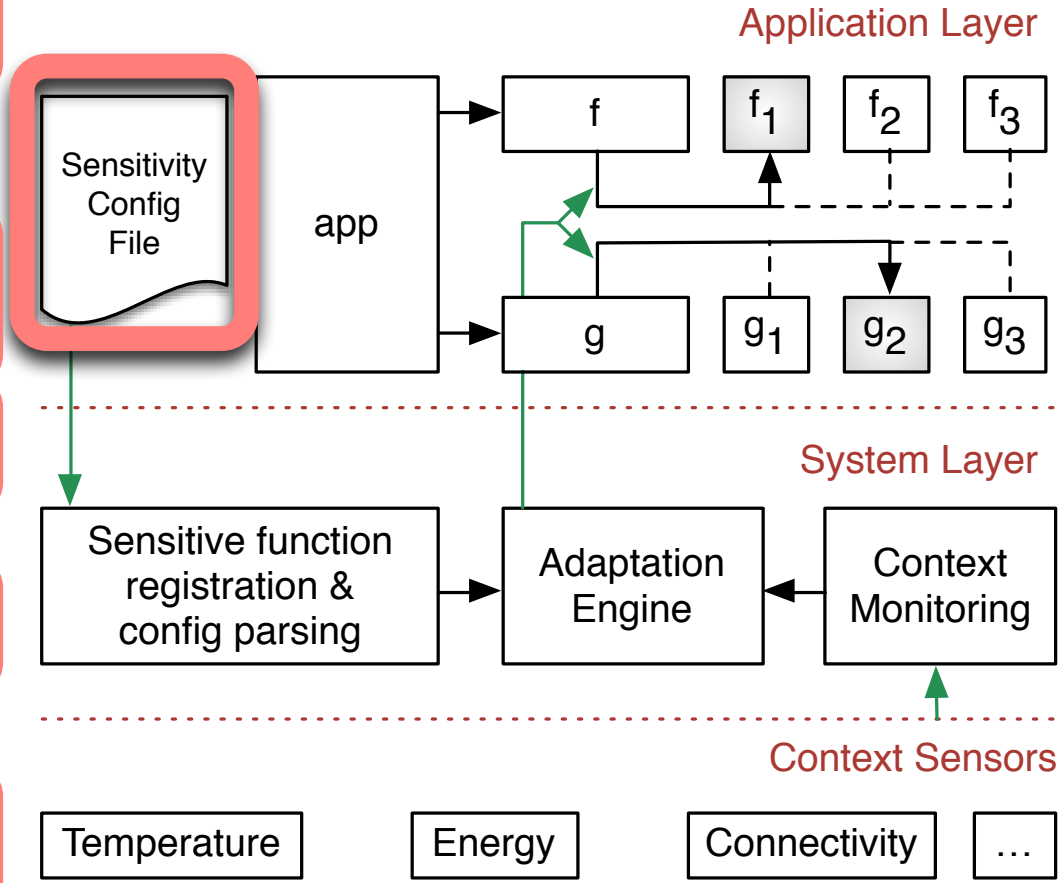


CAreDroid Architecture



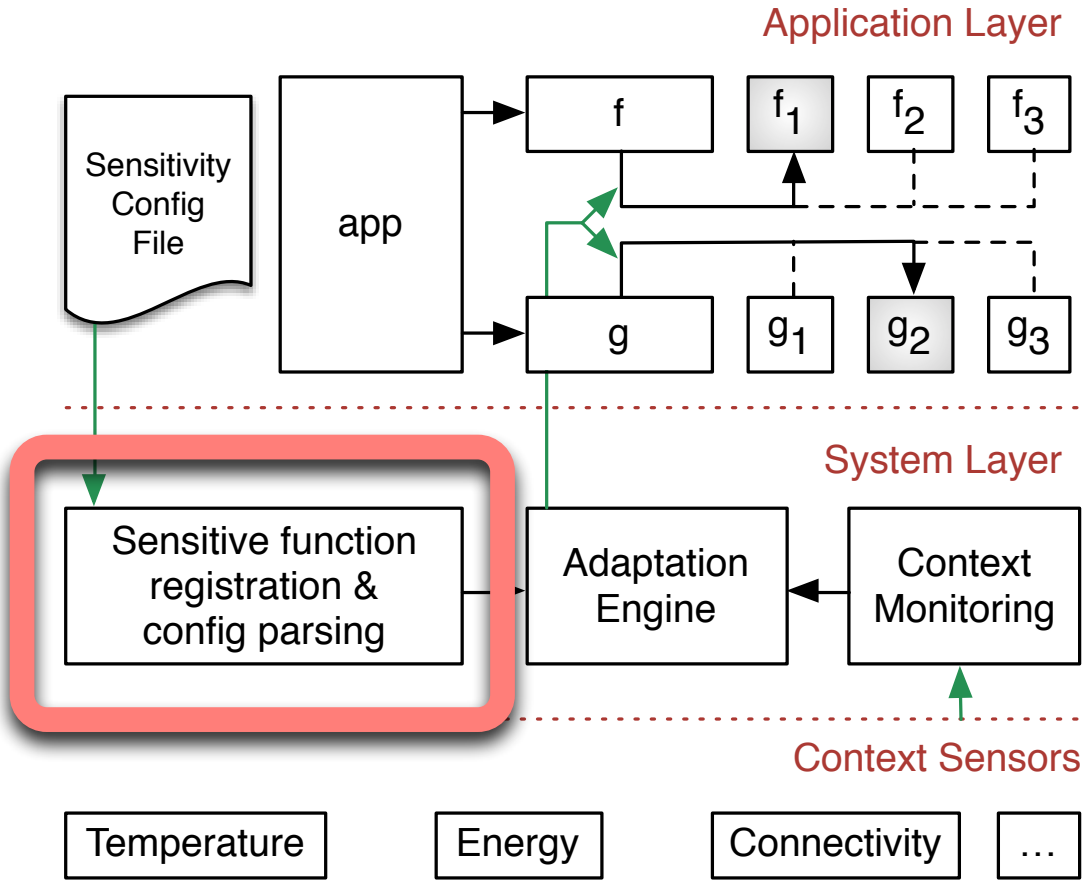
Sensitivity Configuration File

```
Method>
  <MethodName>AdjustCameraPowerAware
  <priority>1</priority>
    <itemName>BatteryCapacity</itemName>
    <vstart>0</vstart>
    <vend>25</vend>
  <mobility>run,walk,still</mobility>
  <wifi>on, off </wifi>
  <MethodName>AdjustCameraWhileRunning
    <itemName>BatteryCapacity</itemName>
    <vstart>20</vstart>
    <vend>100</vend>
  <mobility>run</mobility>
  <wifi>on, off </wifi>
```

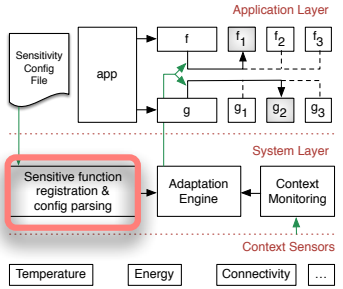


Function Registration

- The sensitive function registration part is responsible for:
  - understanding the ranges of operations of each method
  - integrate it in the Android run-time system

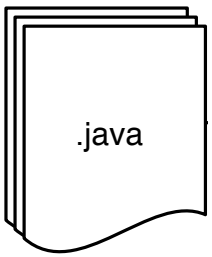


Function Registration & Config. Parsing

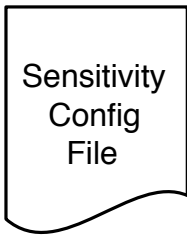


Development time

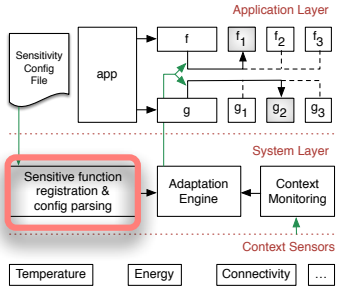
Normal Android flow

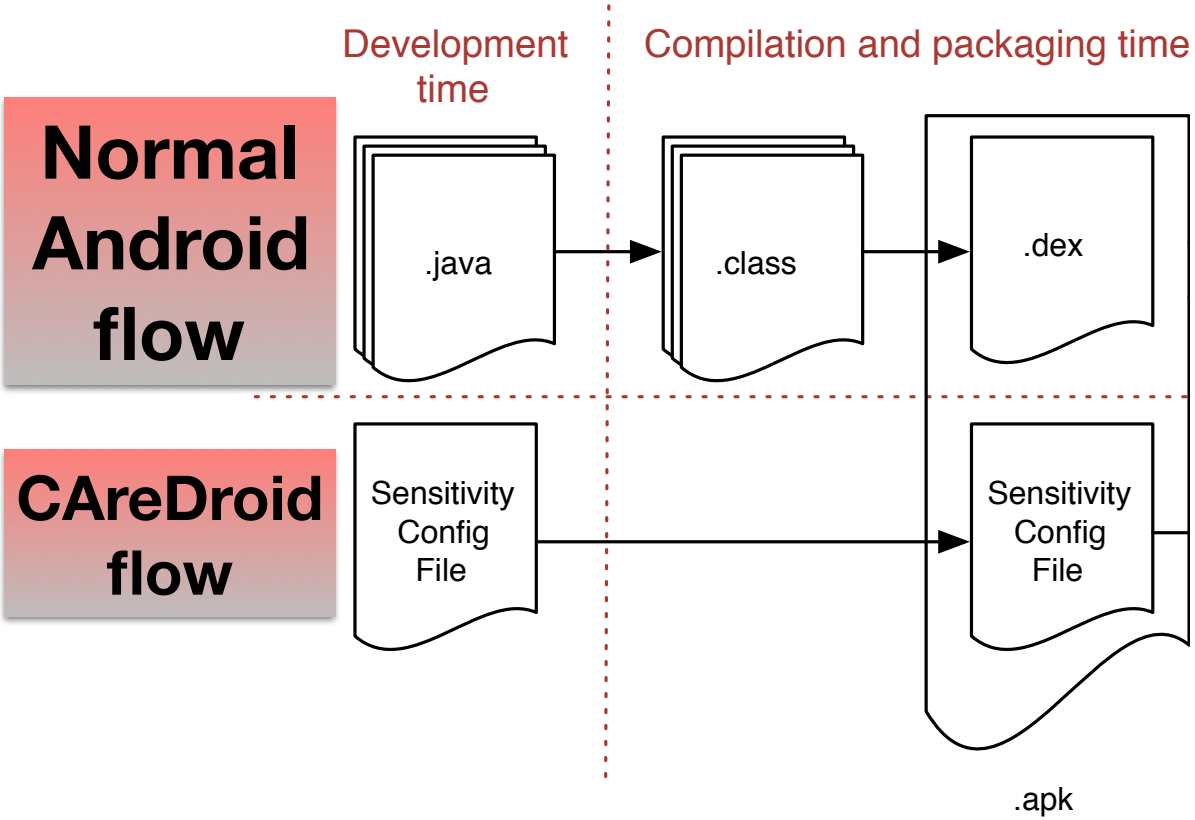


CAreDroid flow

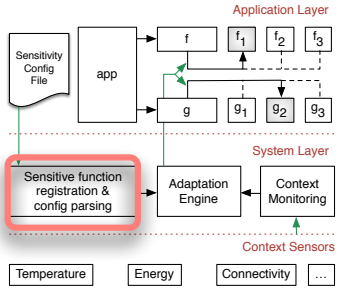


Function Registration & Config. Parsing





Function Registration & Config. Parsing



Normal Android flow

CAreDroid flow

Development time

Compilation and packaging time

Installation time

.java

.class

.dex

intercept

.odex

Sensitivity Config File

Sensitivity Config File

Sensitive Method Discovery

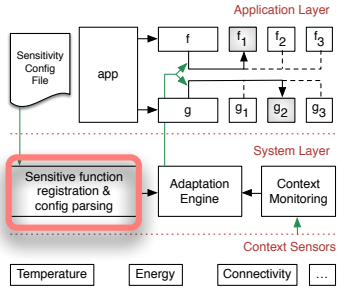
Replacement Map

Table of Range Identifiers

.apk

extended .odex

Function Registration & Config. Parsing



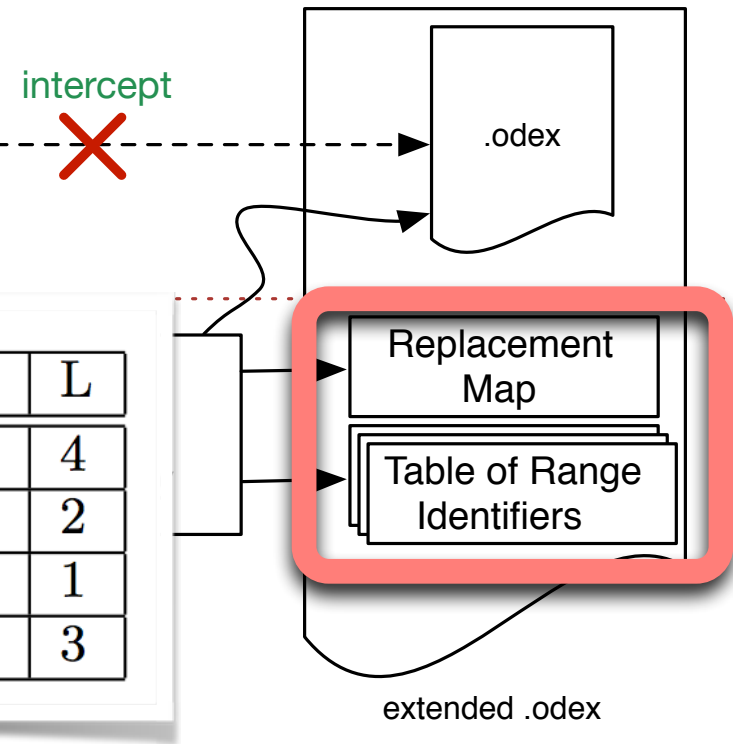
Nor
Anc
flo

B-Range	ORI
0→100	1
20→30	2
10→20	3
30→100	4

S-Range	ORI
0→2	1
1→4	2
2→3	3
0→3	4

C

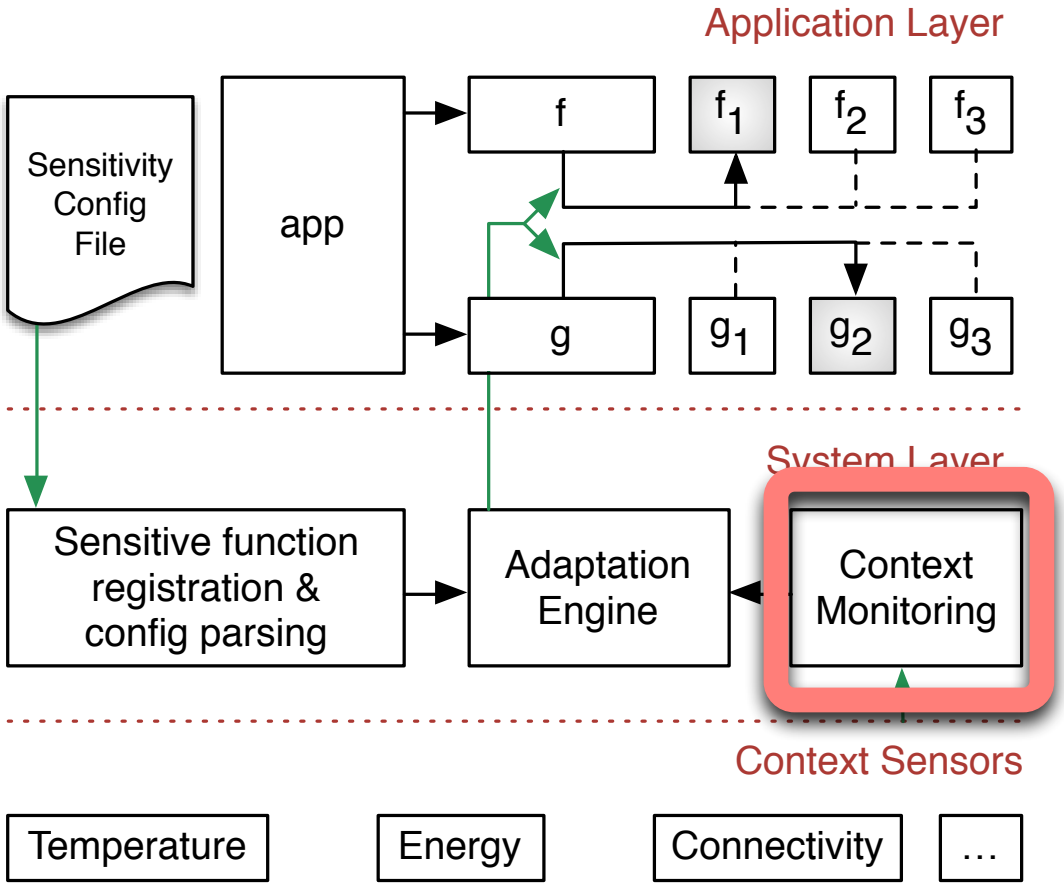
Class ID	Method ID	B	T	V	W	Q	S	M	L
0x01	0x00F	1	2	1	2	4	2	1	4
0x01	0x01E	2	3	4	2	3	3	2	2
0x02	0x02A	2	2	1	0	0	0	2	1
0x02	0x01F	2	2	1	0	0	0	8	3



Context Monitoring

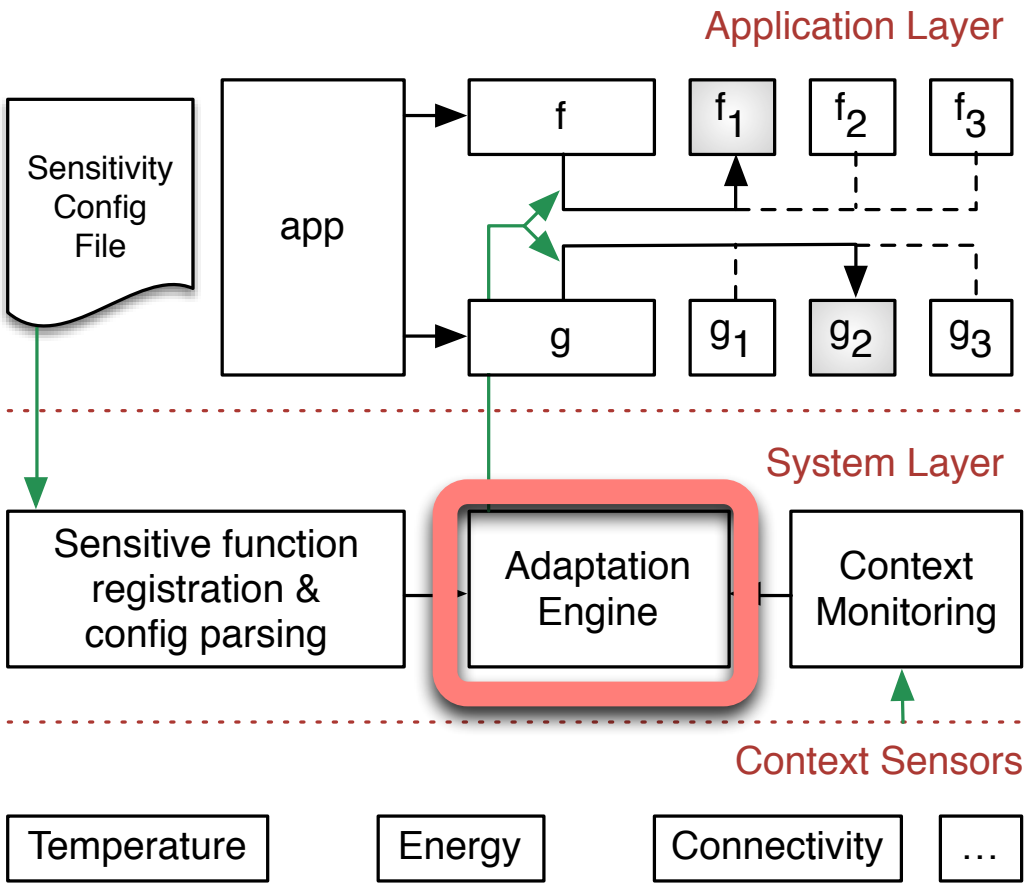
1. **Battery state:** capacity, temperature, voltage
2. **Connectivity state:** WiFi connection status, WiFi link quality, RSSI received signal strength indication
3. **Location:** GPS location
4. **Mobility:** run, walk, still

- Bypassing the **HAL layer** and the associated **sensor managers**.
- Snooping on the interface between the HAL and the low-level device drivers through the sysfs virtual file system.

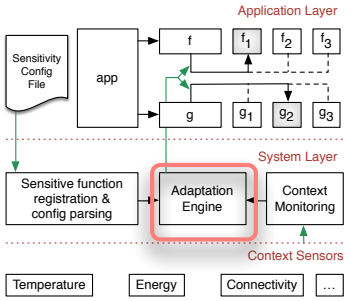


Adaptation Engine

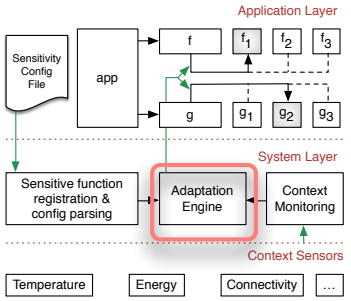
It is where the **method replacement** happens at runtime.



Adaptation Engine



Adaptation Engine



invoke-virtual-quick

Normal Android flow

Dalvik Interpreter

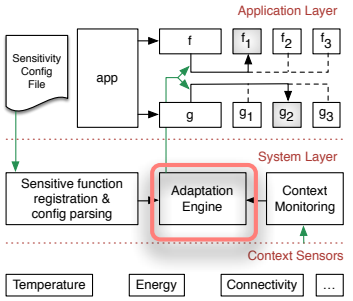
Class ID, Method ID

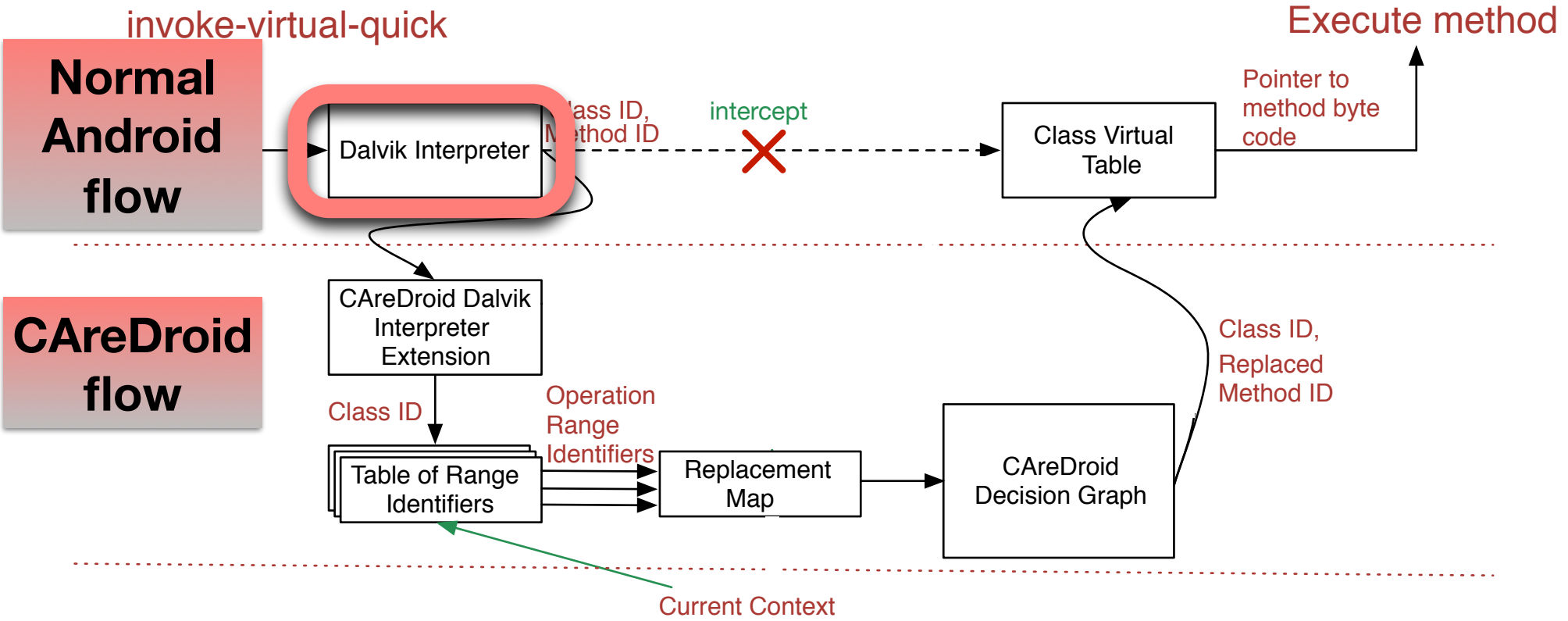
Class Virtual Table

Pointer to method byte code

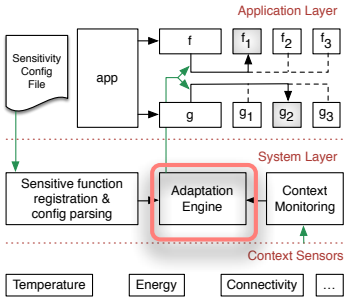
Execute method

Adaptation Engine





Adaptation Engine



invoke-virtual-quick

Execute method

Normal Android flow

Dalvik Interpreter

Class ID, Method ID

intercept

Class Virtual Table

Pointer to method byte code

CAreDroid flow

CAreDroid Dalvik Interpreter Extension

Class ID

Operation Range Identifiers

Table of Range Identifiers

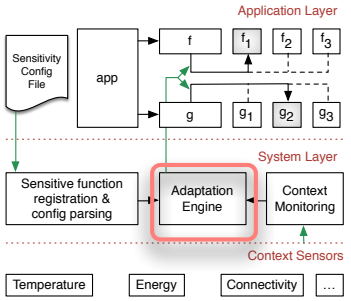
Replacement Map

CAreDroid Decision Graph

Class ID, Replaced Method ID

Current Context

Adaptation Engine



invoke-virtual-quick

Execute method

Normal Android flow

Dalvik Interpreter

Class ID, Method ID

intercept

Class Virtual Table

Pointer to method byte code

CAreDroid flow

CAreDroid Dalvik Interpreter Extension

Class ID

Operation

Table of Range Identifiers

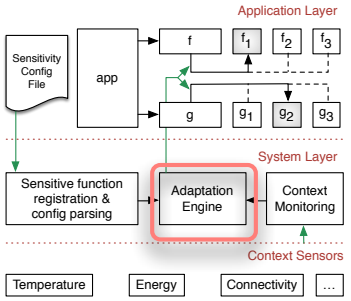
Replacement Map

CAreDroid Decision Graph

Class ID, Replaced Method ID

Current Context

Adaptation Engine



invoke-virtual-quick

Execute method

Normal Android flow

Dalvik Interpreter

Class ID, Method ID

intercept

Class Virtual Table

Pointer to method byte code

CAreDroid flow

CAreDroid Dalvik Interpreter Extension

Class ID

Operation Range Identifiers

Table of Range Identifiers

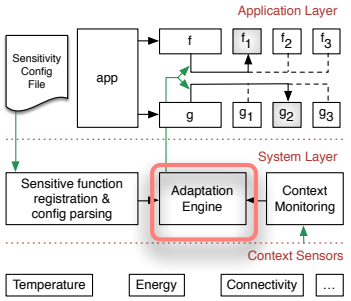
Replacement Map

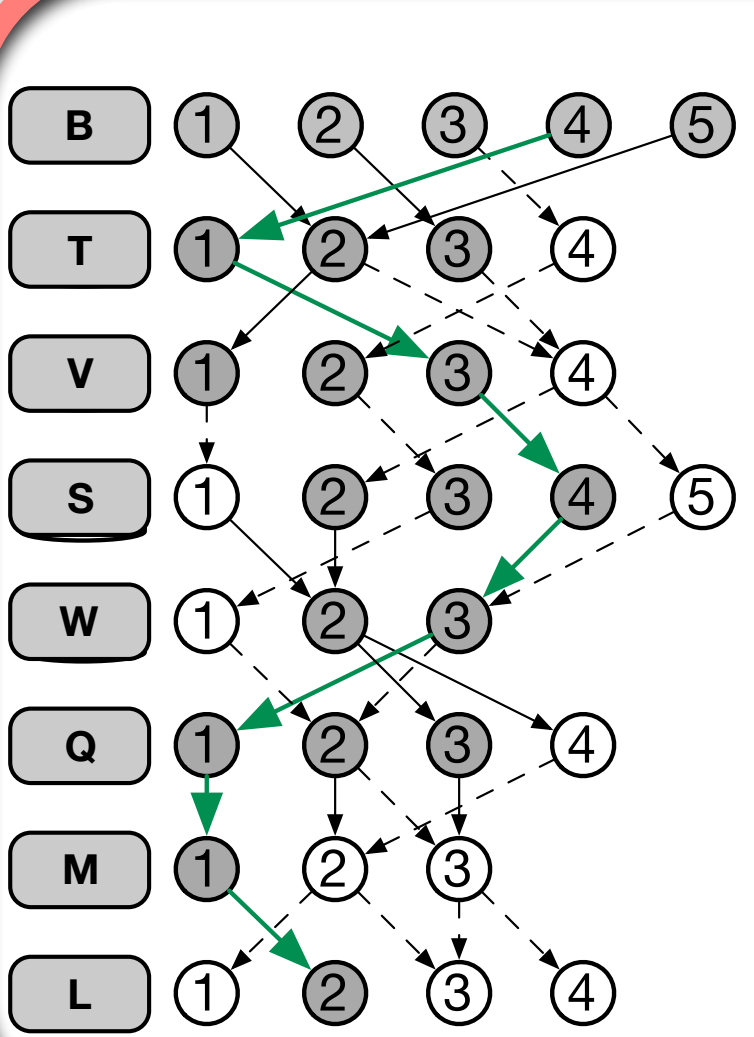
CAreDroid Decision Graph

Class ID, Replaced Method ID

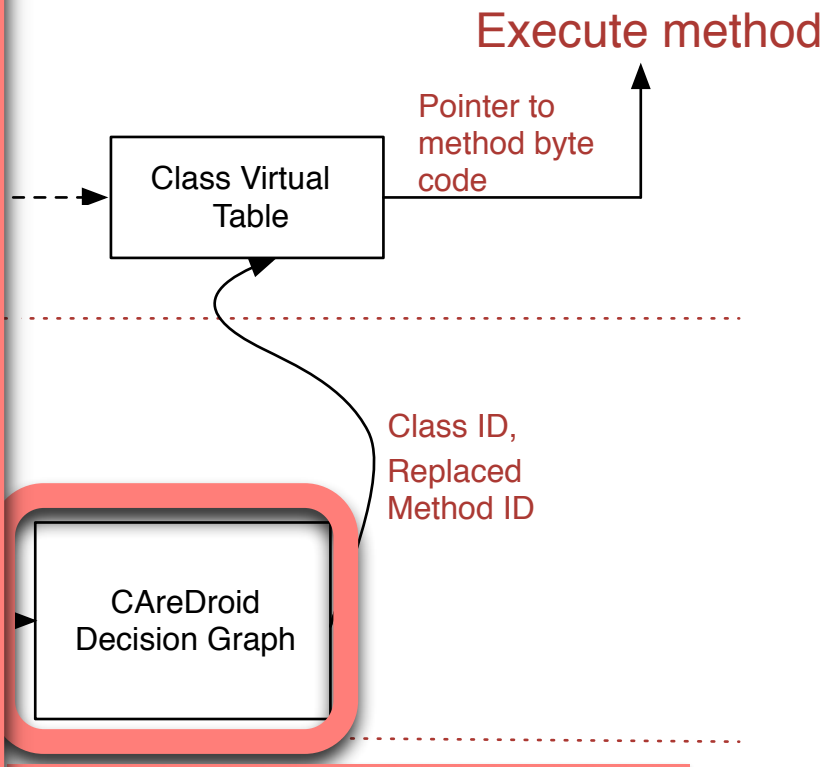
Current Context

Adaptation Engine





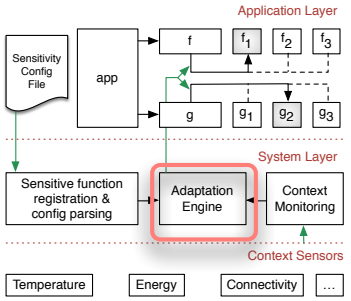
M1	M2	M3	M4	M5
1	2	3	4	5
2	3	4	1	2
1	4	2	3	4
1	2	3	4	5
2	2	1	3	3
4	3	2	1	2
2	3	3	1	2
1	4	3	2	3



Best Fit policy

Must Fit policy

Adaptation Engine



invoke-virtual-quick

Execute method

Normal Android flow

Dalvik Interpreter

Class ID, Method ID

intercept

Class Virtual Table

Pointer to method byte code

CAreDroid flow

CAreDroid Dalvik Interpreter Extension

Class ID

Operation Range Identifiers

Table of Range Identifiers

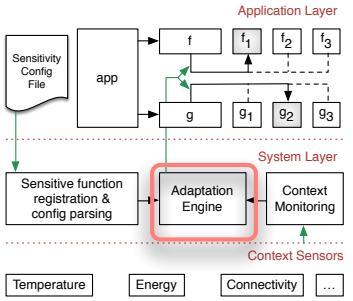
Replacement Map

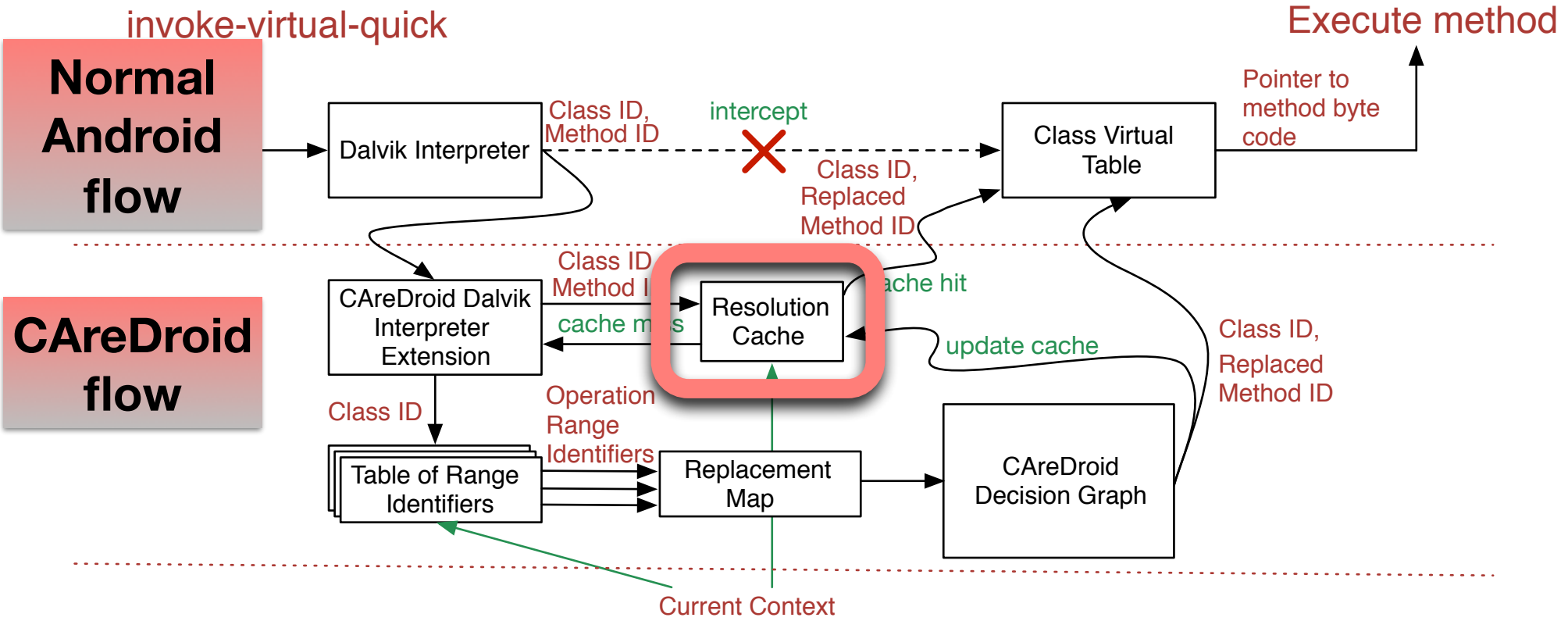
CAreDroid Decision Graph

Class ID, Replaced Method ID

Current Context

Adaptation Engine





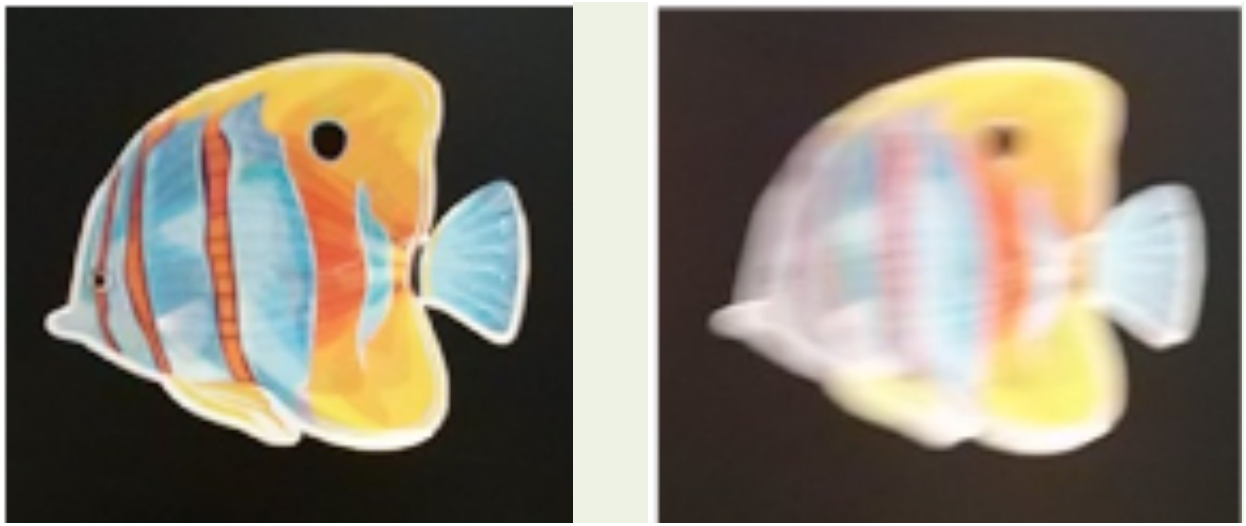
Evaluation

1. Push context monitoring and adaptation to runtime system
2. At least 2x fewer lines of code
3. At least 10× more efficient in execution time

Context-aware Camera

Still position

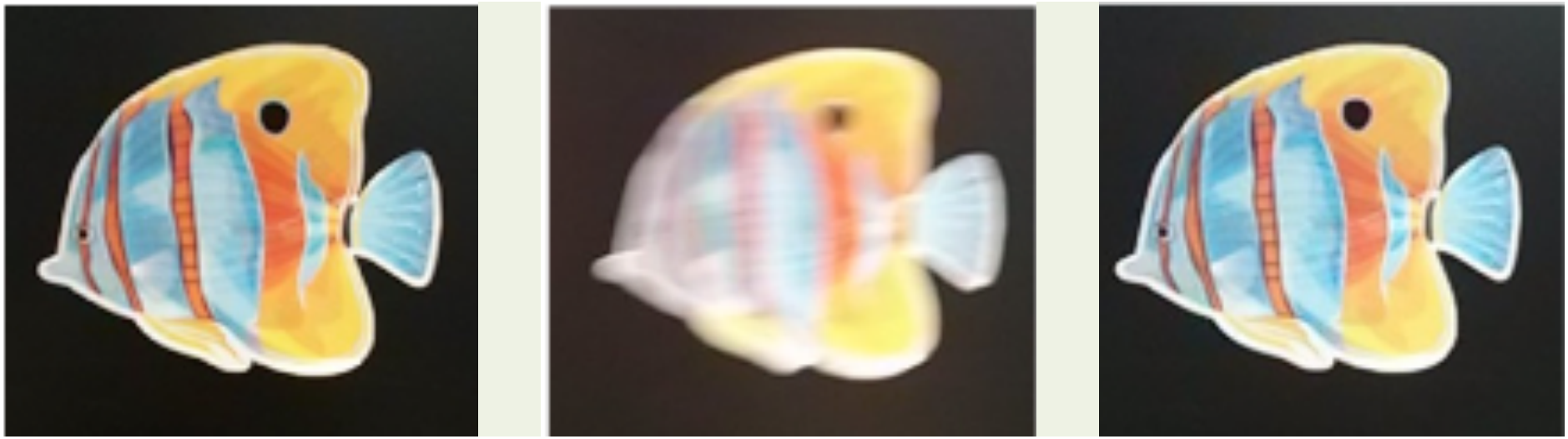
Context-aware Camera



Still position

Walking (non-context aware)

Context-aware Camera



Still position

Walking (non-context aware)

Walking (With CAreDroid)

Analysis - Significant Line of Code (SLOC)

Platform	SLOC	%Increase
Non-context aware (Base)	277	-
Context-aware (Pure Java)	782	182%
CAreDroid	277 +133[a]	48%

[a]XML Configuration file

Analysis of CAreDroid

1. Analyze the effect of cache in the adaptation engine
2. Examine the overhead of context-monitoring module
3. The difference in energy and execution time between pure Java implementation using Android APIs and CAreDroid

Testbed and testing environment

- Nexus 4
- Android 4.2 - API 17
- Overhead of system image is **4.6%**

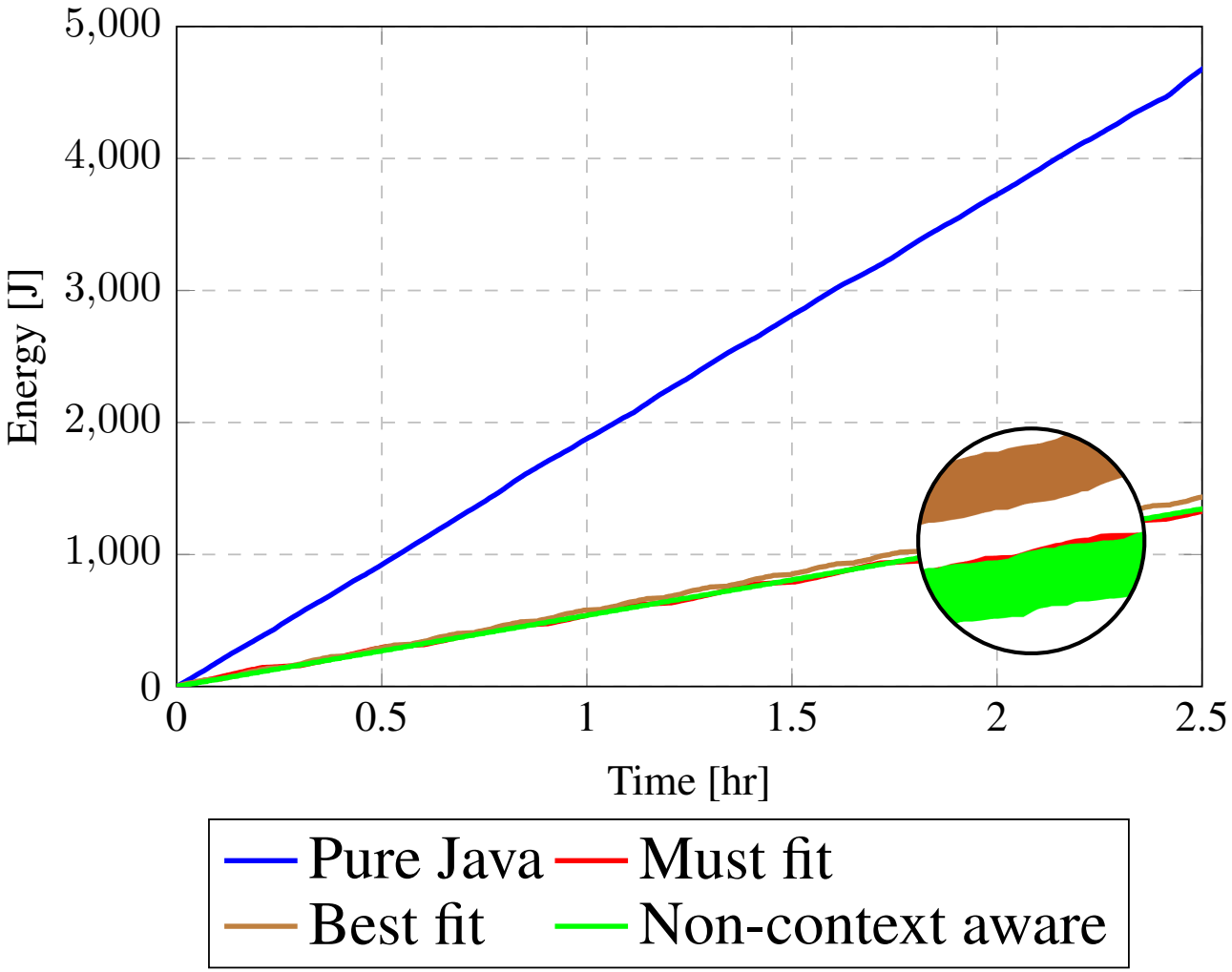
Analysis—Execution time

Platform		CPU time (ms)						Overhead	
		Method time	Decision Tree		Context Monitoring (parallel thread)	Total			
			without cache	with cache		without cache	with cache	Without cache	With cache
Non-context aware (Base)	f1	8.322	-		-	8.322		-	
	f2	16.872	-		-	16.872		-	
	f3	13.375			-	13.375		-	
Context aware (Pure Java)	f1	8.322	0.227		5.093	13.642		63.92%	
	f2	16.872	0.776		5.093	25.741		52.56%	
	f3	13.375	0.351		5.093	18.819		40.70%	
CAreDroid (Must Fit)	f1	8.322	0.183	0.030	0.336	8.841	8.688	6.23%	4.39%
	f2	16.872	0.335	0.031	0.336	17.543	17.239	3.98%	2.17%
	f3	13.375	0.198	0.030	0.336	13.909	13.741	3.99%	2.736%
CAreDroid (Best Fit)	f1	8.322	0.183	0.031	0.336	8.841	8.689	6.23%	4.41%
	f2	16.872	0.732	0.031	0.336	17.635	17.239	4.522%	2.17%
	f3	13.375	0.489	0.030	0.336	14.213	13.741	6.17%	2.73%

Average of 10 executions with variance 0.5% ~ 1%

Analysis — Energy

1. CAreDroid consumes only 6.73% more energy compared with the non-context aware implementation and
2. CAreDroid gives 69.33% energy saving compared to the pure Java implementation.
3. Measurements are taken every 5 seconds. Average is shown with variance < 1%



Future Work

1. Migrate to new versions of Android that are based on ART instead of Dalvik (Kitkat and Lollipop).

2. Add support for **more abstract contexts**, for example, stress, smoking, fitness, … etc

3. The adaptation can be **privacy-aware.**

4. **Time of the day** is another axis that can be used to enhancing the adaptation support

Thank you!

http://www.seas.ucla.edu/~salma/

Keeping the developer/user in the loop

- Make the user *not* fully obfuscated
  - Why this adaptation takes place?
  - Special CAreDroid API - “**read_operating_point()**”