Android Application Sandbox

Thomas Bläsing
DAI-Labor
TU Berlin

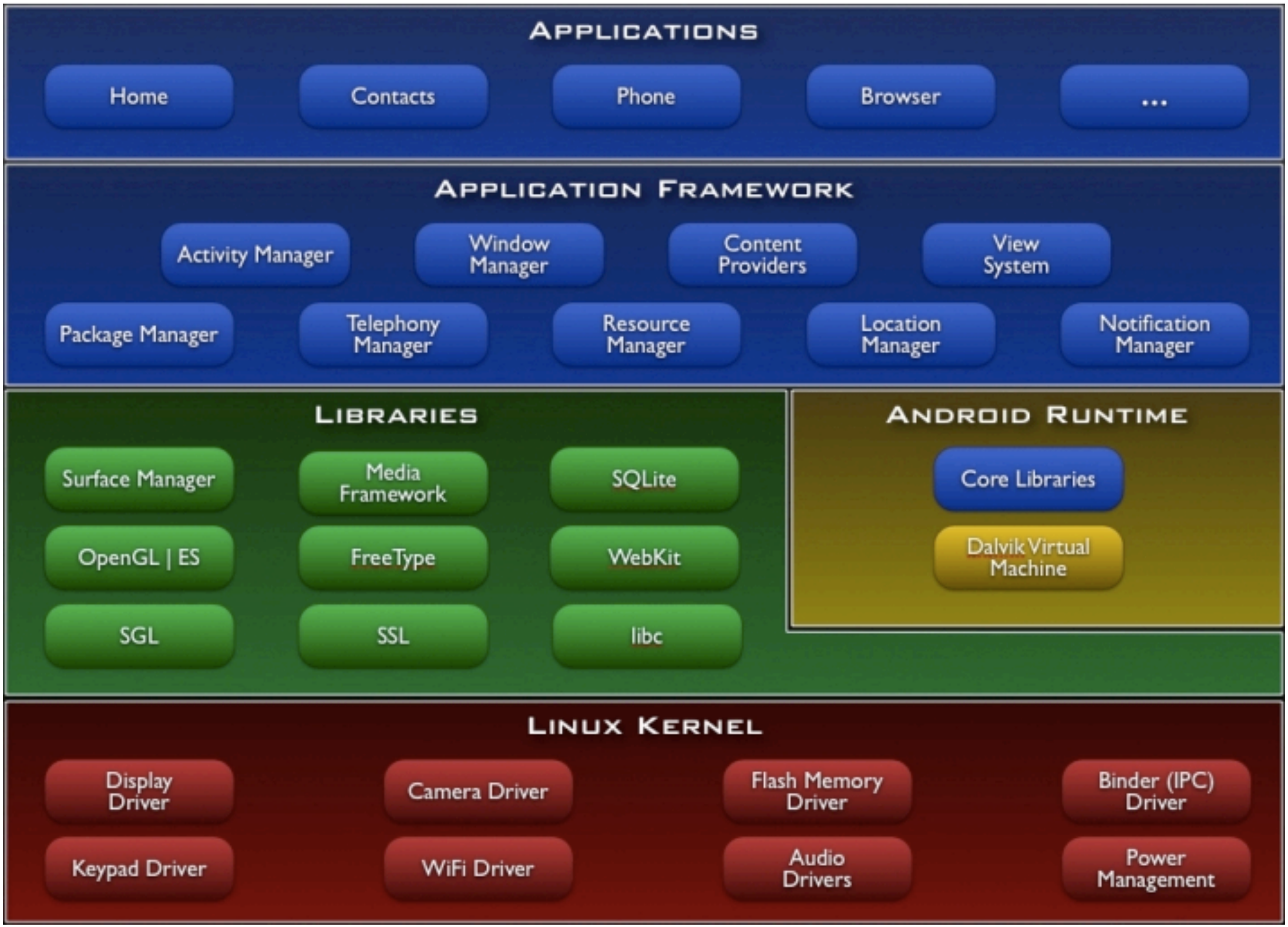
Agenda

- Introduction
  - What is Android ?
  - Malware on smartphones
  - Common countermeasures on the Android platform
- Android Application Sandbox
  - Use-Cases
  - Design
  - Conclusion
- Summary
  - Future work / Bibliography

What is Android?

- mobile OS based on a Linux 2.6 kernel
- initially developed by Google, later by the Open Handset Alliance (OHA)
- open source (Apache license) since oct 2008
  - some drivers for special mobile hardware are still not free
- official supported platform is ARM
  - but there is a port for x86 platforms
- Dalvik VM
  - register-based VM for Java
- still growing (online-)community

What is Android? (technical view)



source: http://www.techflare.com.au/media/102-android%20-%20system-architecture.jpg

Malware on smartphones

- interesting topic
- paper „Android: Next Target?“
  - Schmidt et al
- smartphones getting more and more popular
- 2 main categories of attacks:
  - direct attacks, e.g. bypassing permissions system
  - indirect attacks, e.g. information leakage
- users doesn‘t realize that most smartphones are like normal desktop PCs

Common counteractive measures on Android

- each application process is separated in an own sandbox-like environment
- strict permissions system
  - you have to explicitly grant permissions before installing the application
- all applications have to be signed
  - but signatures do not need to be certified by a trustworthy organization
- there are some Anti-Virus applications on the market
  - but just signature based detection

Problems on Android Security

- ability of using the Java Native Interface (JNI) to speed-up applications
  - bypass SDK restrictions and normal application lifecyle
  - Google provides NDK (Native Development Kit)
- rising amount of Applications for „rooted“ Android phones
  - this applications could have **COMPLETE** control over the mobile device (root access)
- ability to access internal Android packages via reflection
  - access on not officially supported API content
- Security on Android is an issue!

- Android Application Sandbox
  - Use-Cases
  - Design
  - Conclusion

Use-Cases for AAS

(i) Application Provider (e.g. Android Market)

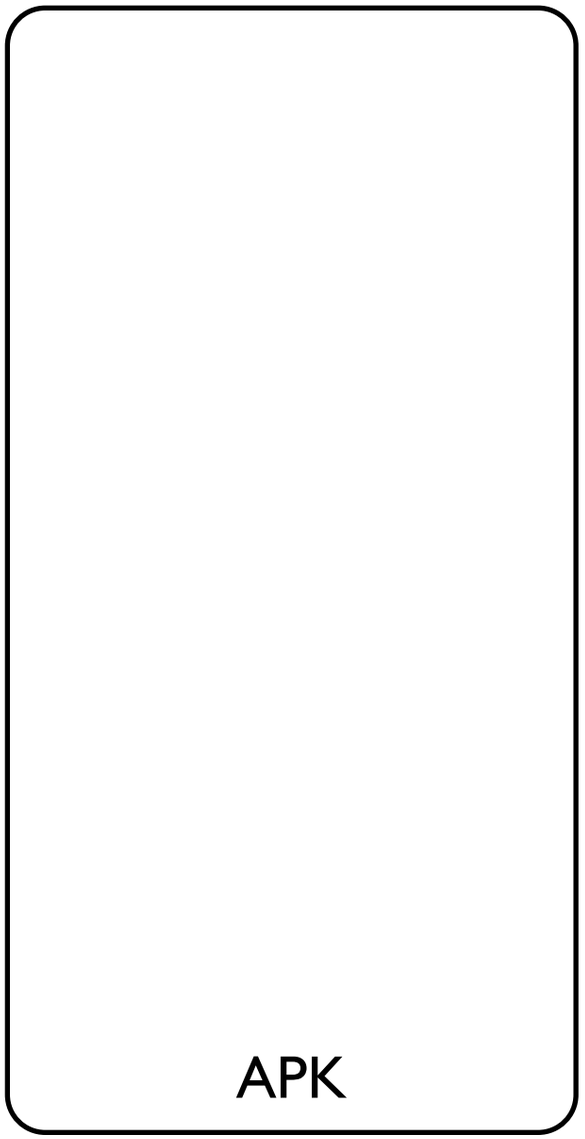
- detect malware to prevent submission to the market
- improve techniques for Anti-Virus Software

(ii) User

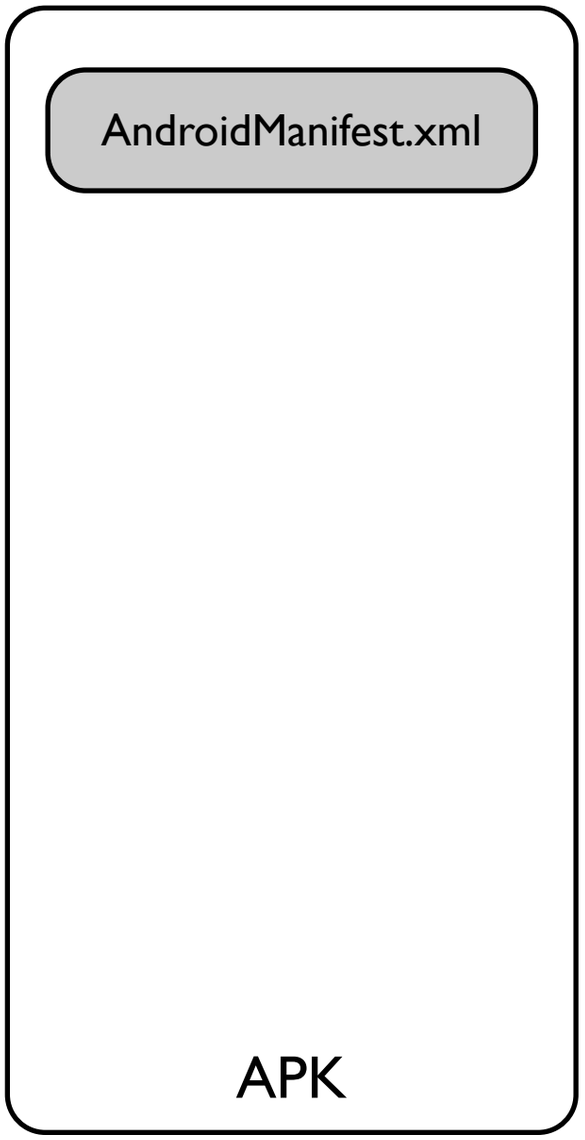
- wants to know what the Application is exactly doing on the phone
- e.g. access personal data although App didn‘t have the permission

Design of the AAS

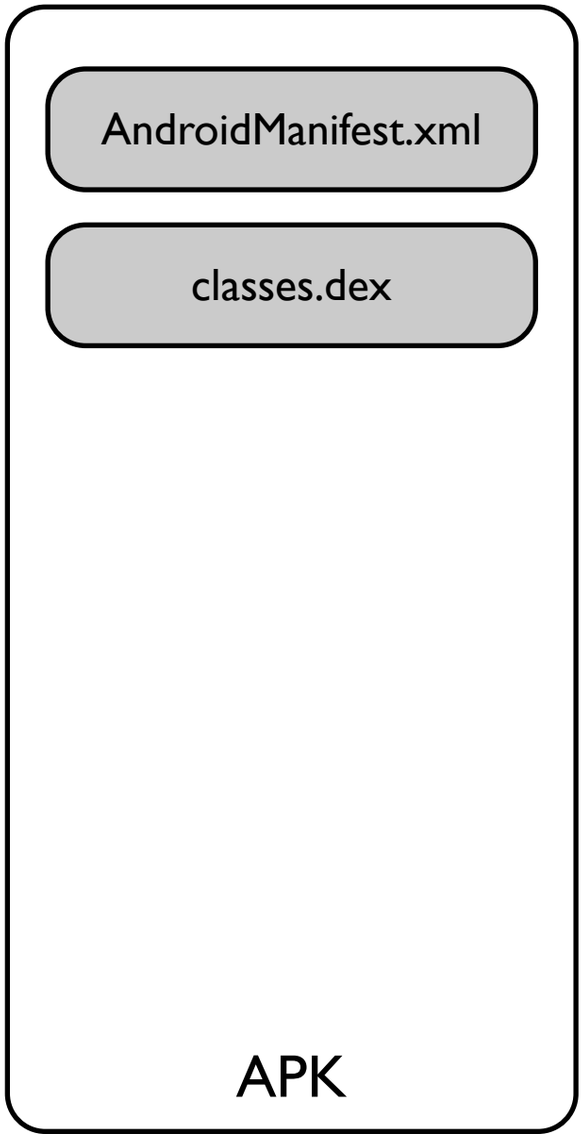
Design of the AAS



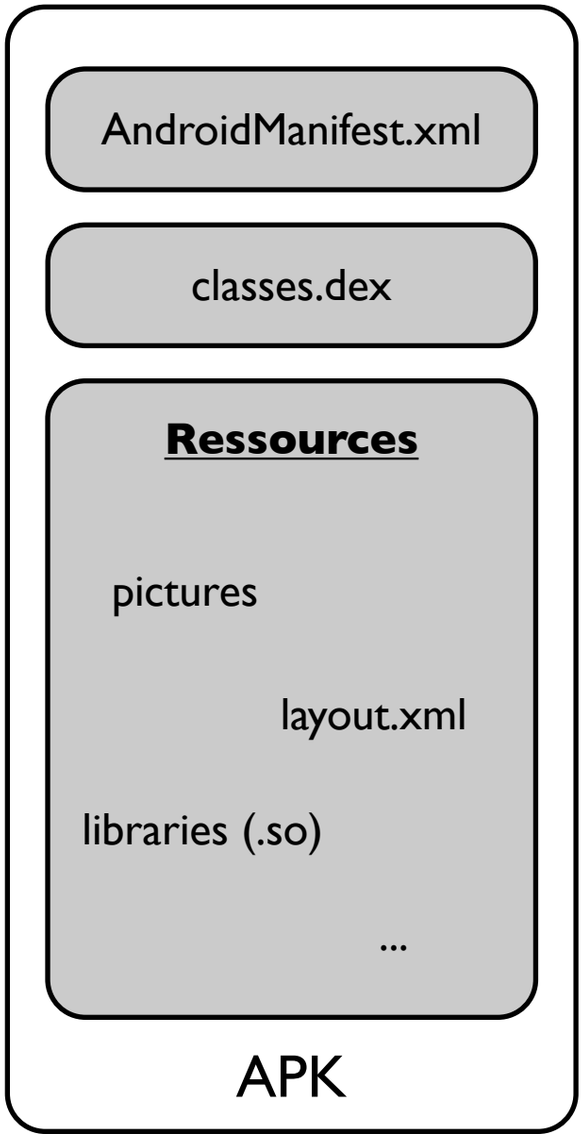
Design of the AAS



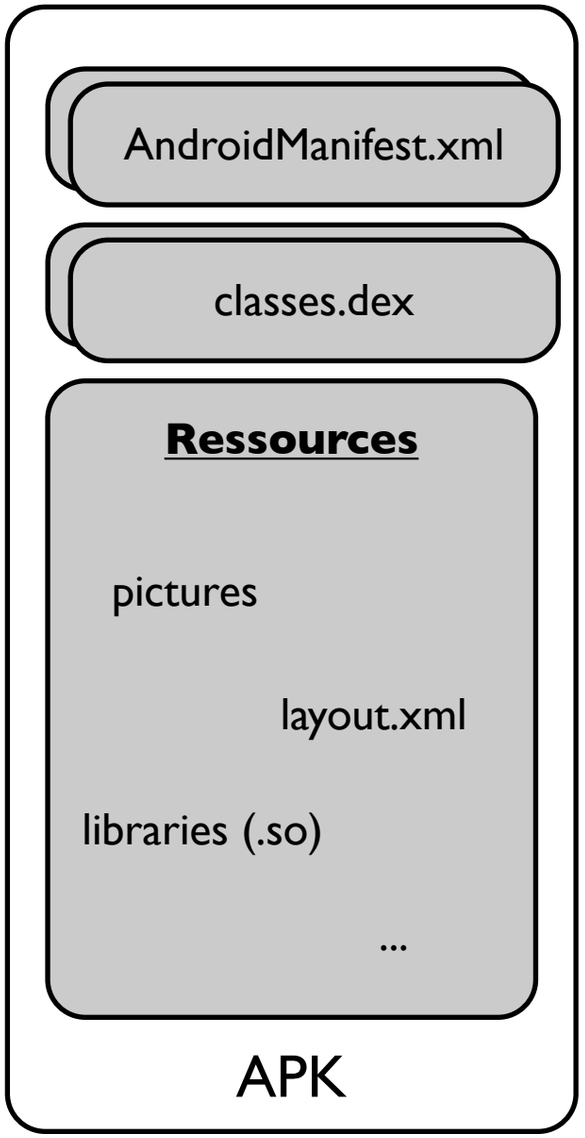
Design of the AAS



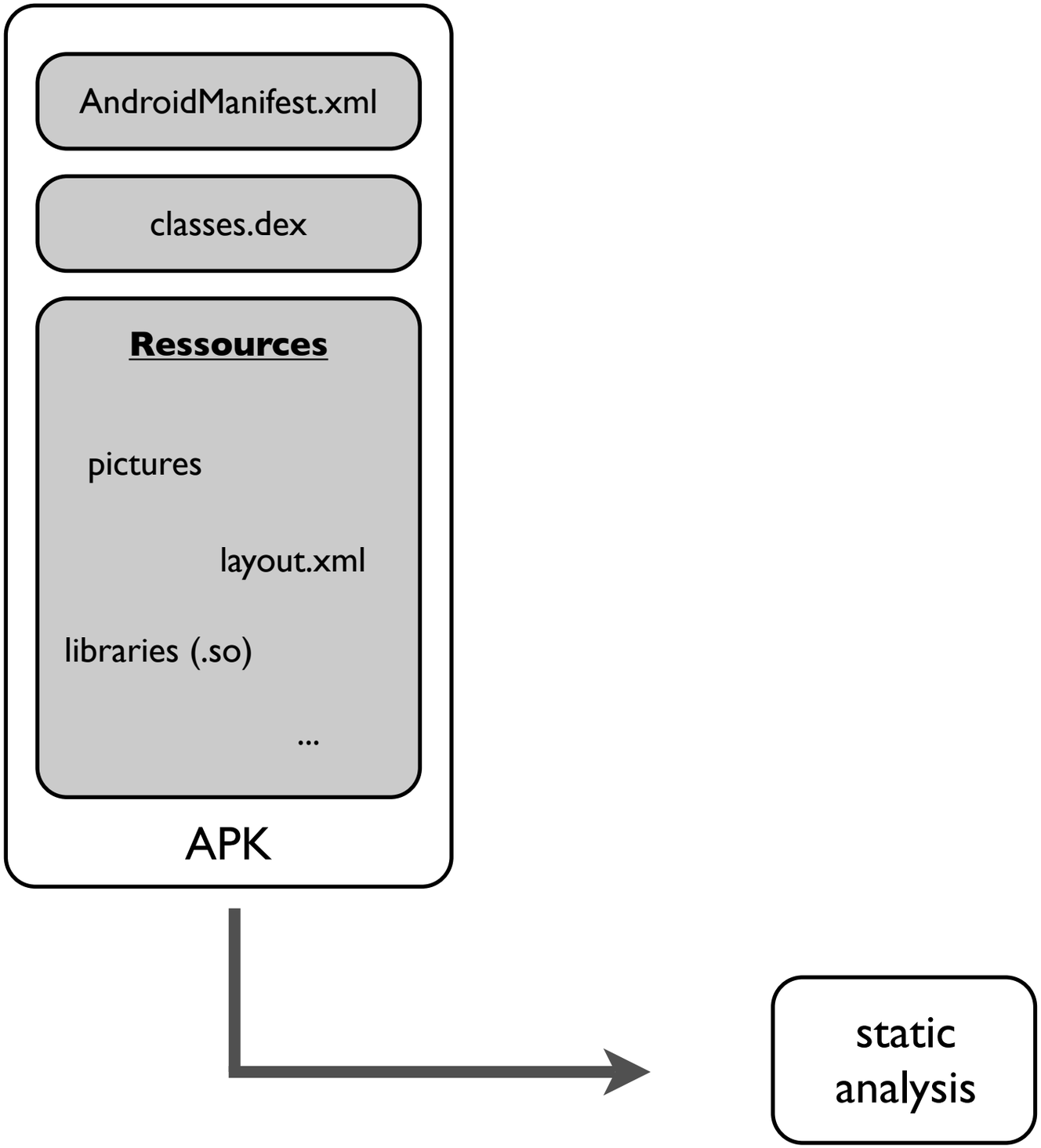
Design of the AAS



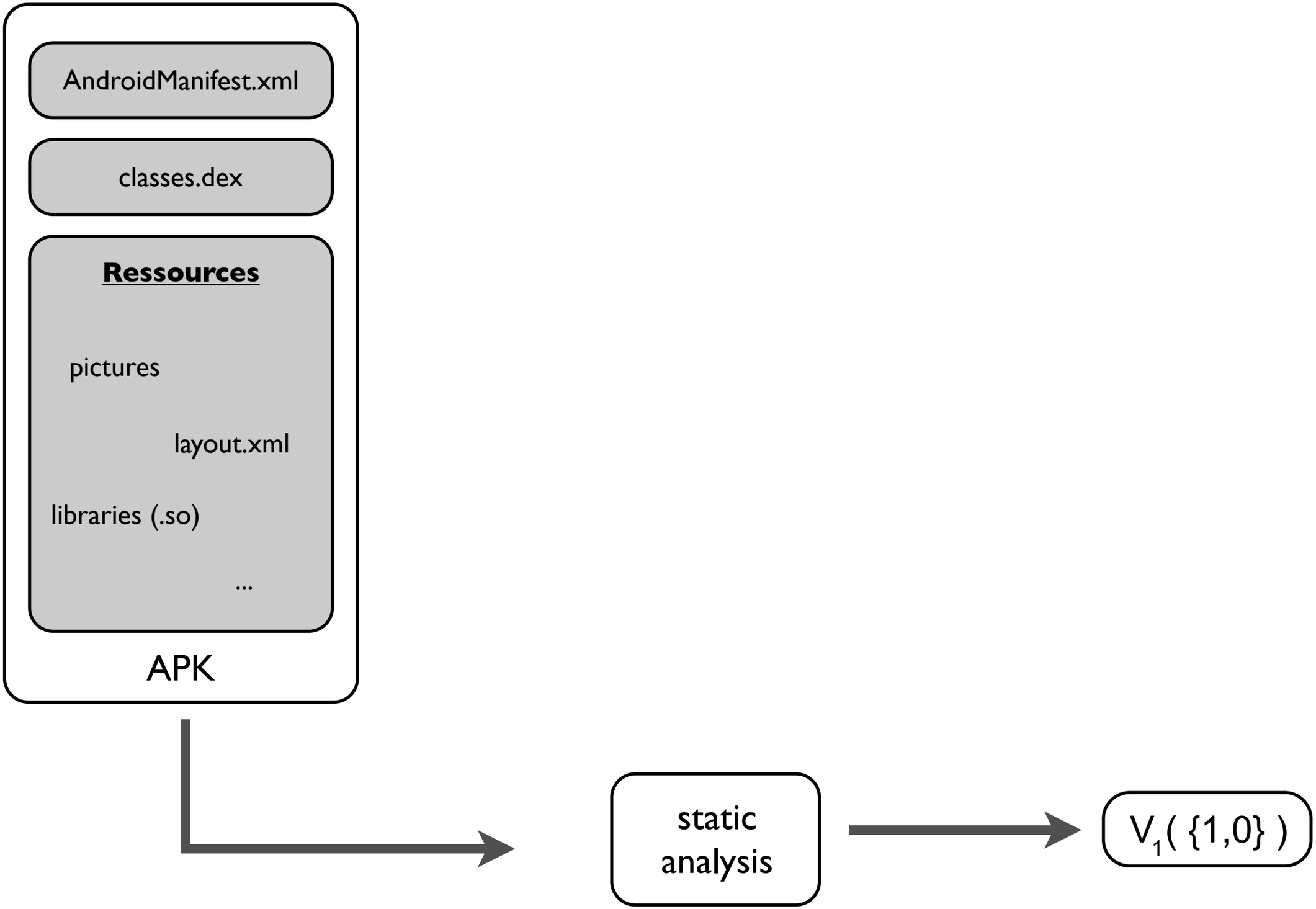
Design of the AAS



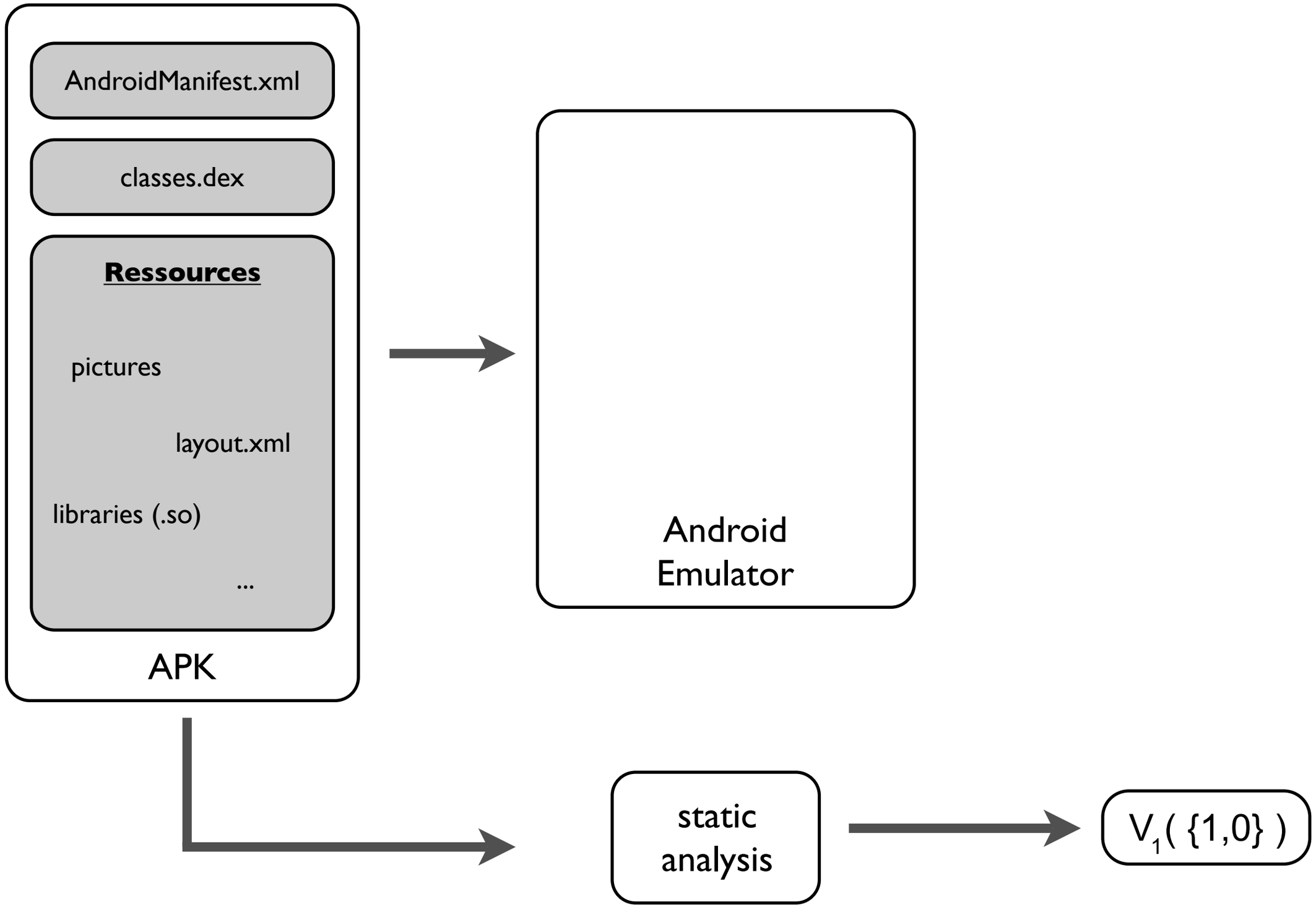
Design of the AAS



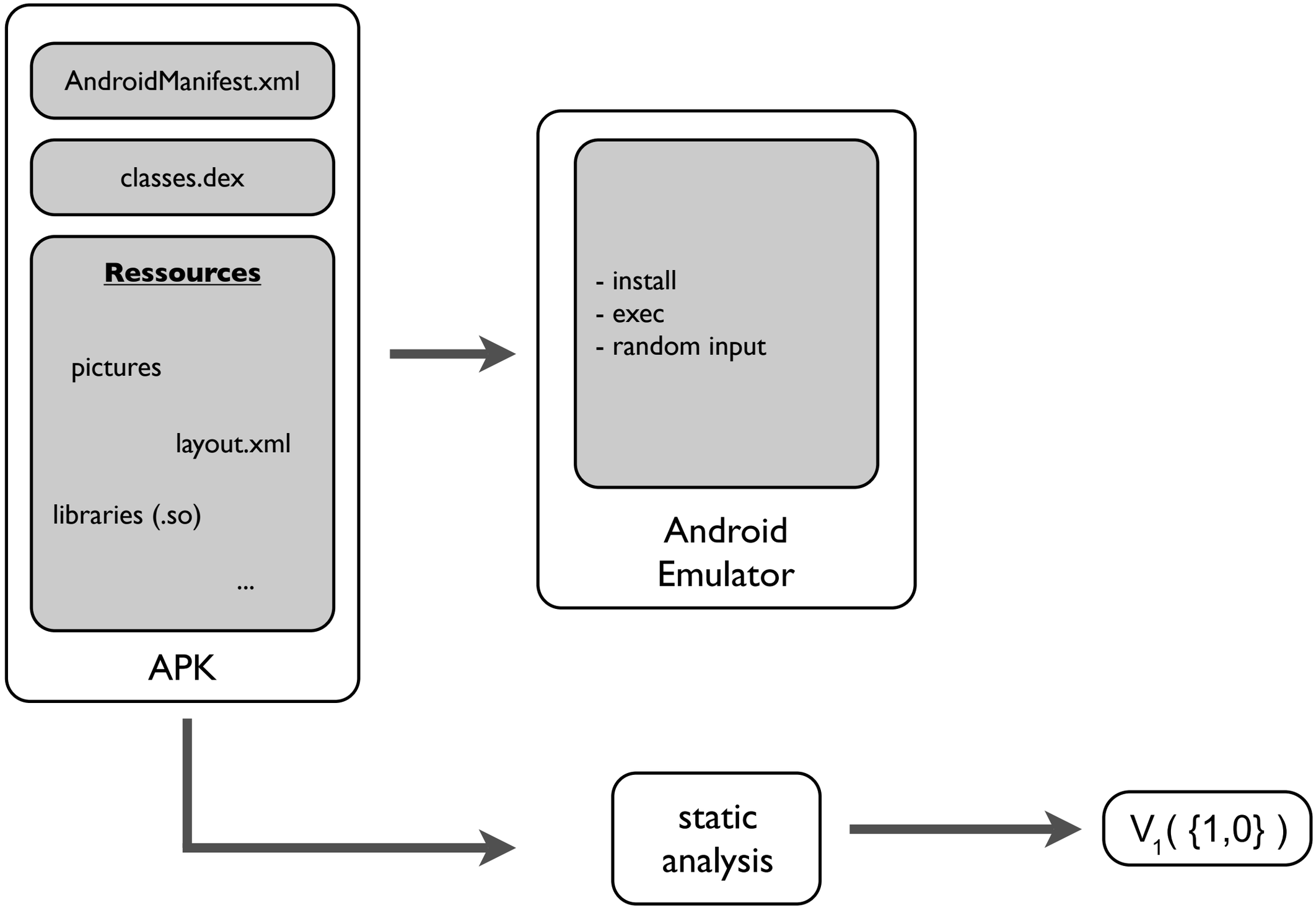
Design of the AAS



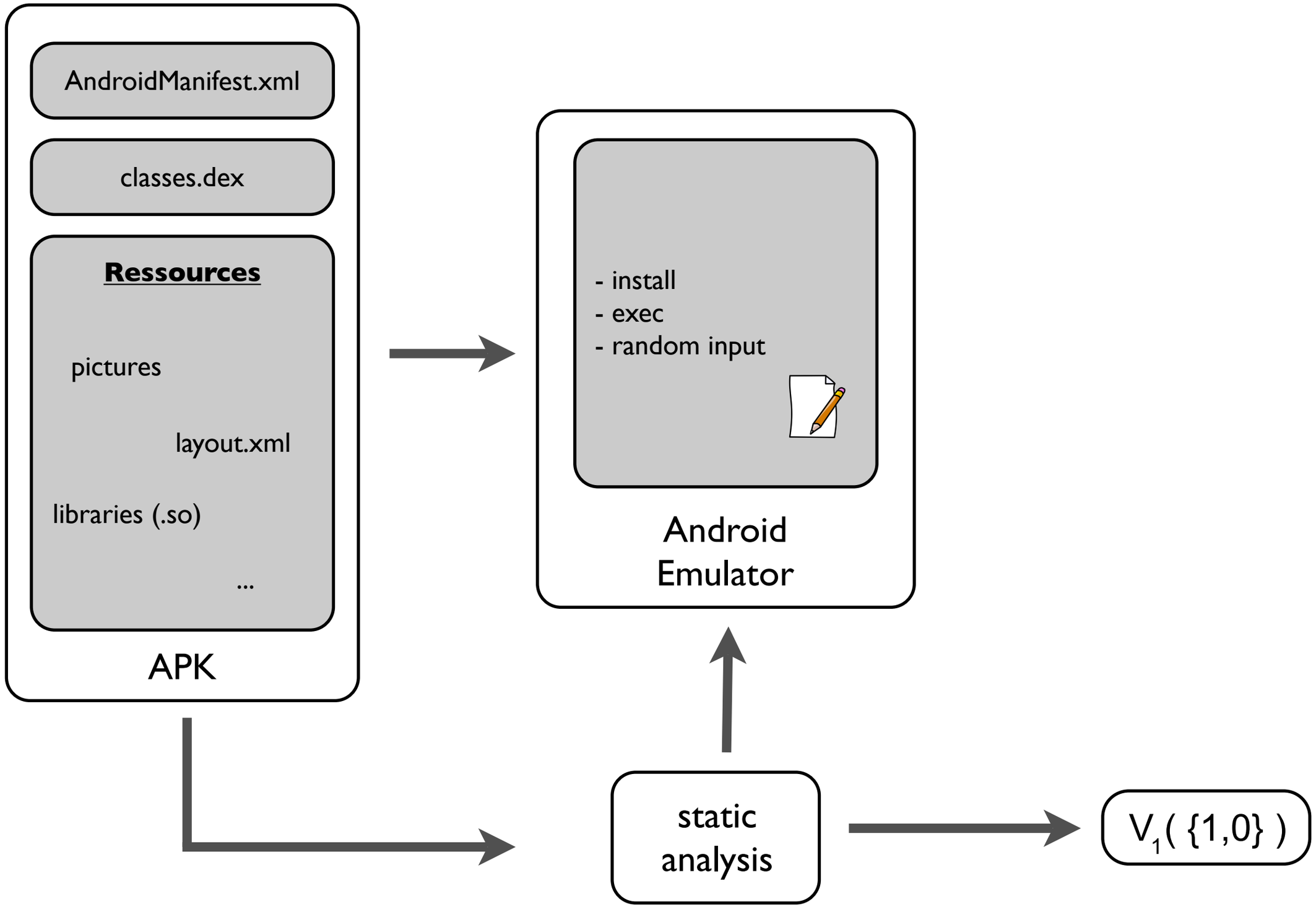
Design of the AAS



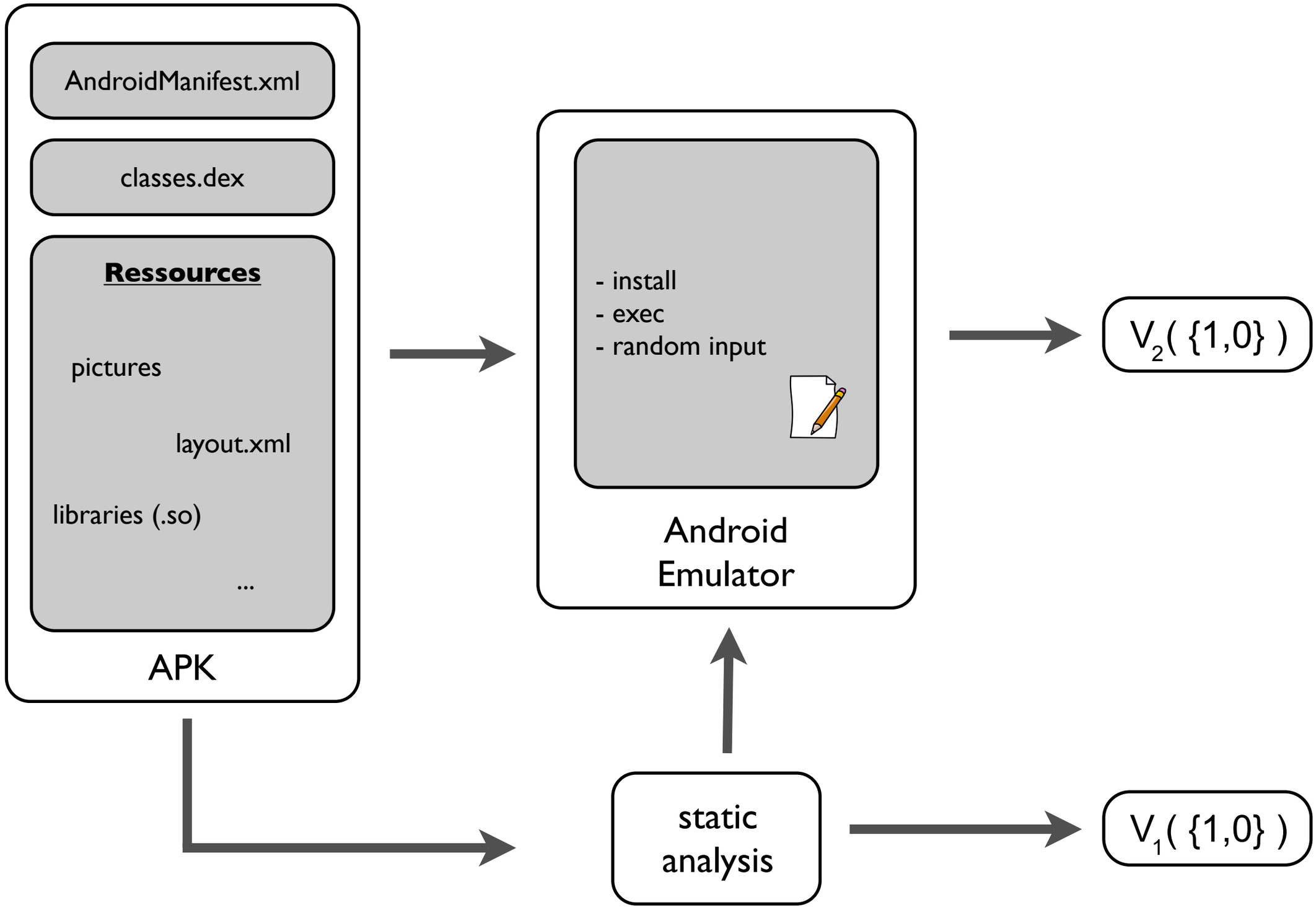
Design of the AAS



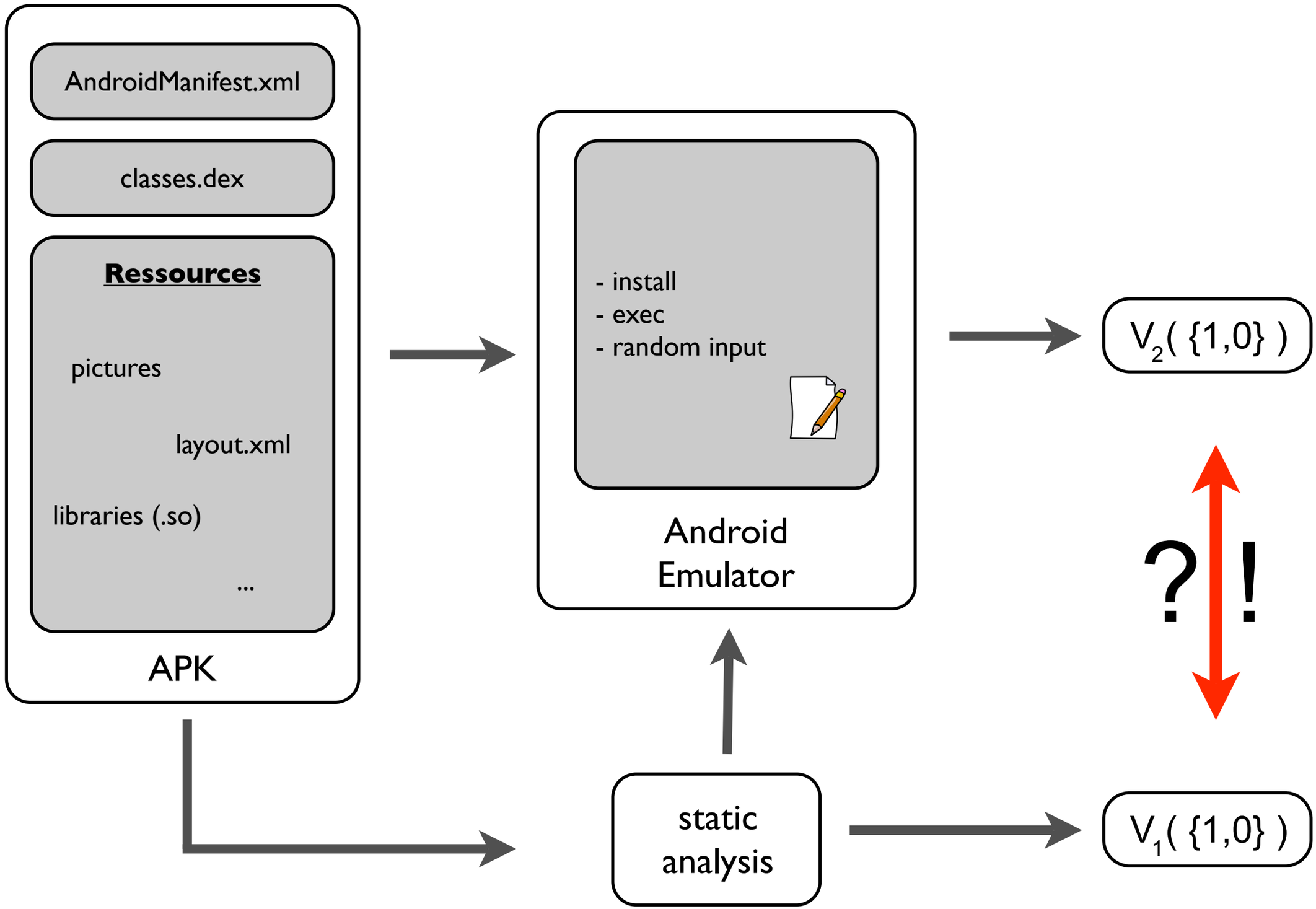
Design of the AAS



Design of the AAS



Design of the AAS



Design of the AAS (2)

- first step done via ‚disassemble' and ‚grep'
  - disassemble of .dex file with baksmali (http://code.google.com/p/smali/)
- second step, LKM which hijacks some syscalls
  - LKM acts like a rootkit
  - logging of syscall usage into logfile which will be analysed after execution time

Conclusion / Thesis

- 2 step analysis
  - Does the result of the static analysis imply the result of the dynamic analysis ?
- first step **really** fast
- second step very **expensive**
- consider the 2 Use-Cases! (User/AppMarket)
  - Do you always need the 2 steps or just one of them, and if so: which one ?
- to prove thesis I need a **lot** of applications
  - Problem: Google AppMarket closed source, no direct access, only via mobile phone
  - I wrote a robot which has downloaded ~150 APKs for researching :)

- Summary
  - Future work / Bibliography

Future Work

- better reverse engineering of mobile applications
- real-time malware detection on smartphones
  - anomaly detection possible with AAS, but actually not in real-time!
- Android Honeypot
  - Honeynet

Bibliography

- MOBILE APPLICATION SECURITY ON ANDROID: Context on Android security (Burns, Black Hat 2009)
- Smartphone Malware Evolution Revisited: Android Next Target? (Aubrey-Derrick Schmidt et al., 4th International Conference on Malicious and Unwanted Software (Malware 2009), Montreal, Quebec, Canada - to appear)
- Developing and benchmarking native linux applications on android. (Leonid Batyuk et al., In mobile Wireless-middleware, Operating Systems, and Applications, 2009)

End

thanks
for your
attention