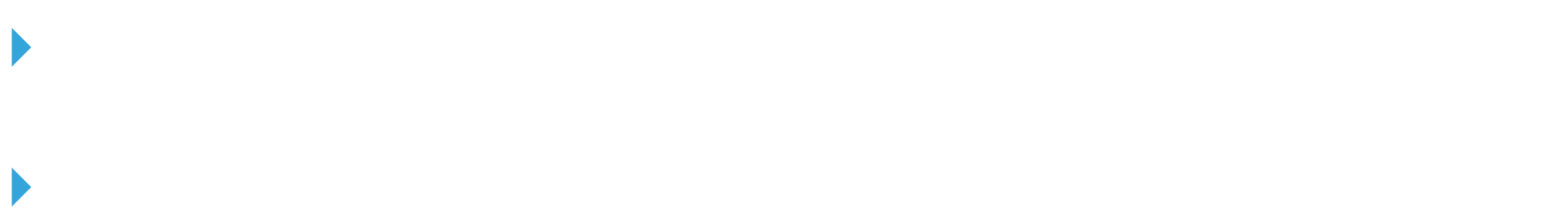








Environment Agnostic by Configurable



3

7

```
use_debug_keystore() {  
    export KEY_ALIAS="$DEBUG_KEY_ALIAS"  
    export KEY_PASSWORD="$DEBUG_KEY_PASSWORD"  
    export KEY_STORE_FILE="$DEBUG_KEY_STORE_FILE"  
    export KEY_STORE_PASSWORD="$DEBUG_KEY_STORE_PASSWORD"  
}  
  
use_release_keystore() {  
    set +x  
    export KEY_ALIAS="$RELEASE_KEY_ALIAS"  
    export KEY_PASSWORD="$RELEASE_KEY_PASSWORD"  
    export KEY_STORE_FILE="$RELEASE_KEY_STORE_FILE"  
    export KEY_STORE_PASSWORD="$RELEASE_KEY_STORE_PASSWORD"  
}
```

```
java -jar bundletool.jar \
  build-apks \
  --bundle="$BUNDLE_FILE" \
  --output=temp.apks \
  --overwrite \
  --mode=universal \
  --ks="$KEY_STORE_FILE" \
  --ks-key-alias="$KEY_ALIAS" \
  --ks-pass="pass:$KEY_STORE_PASSWORD" \
  --key-pass="pass:$KEY_PASSWORD"
```



## Objectives/Solutions

- ▶ Be stable
- ▶ Be environment agnostic
- ▶ Provide semi-interactive flow
- ▶ Give useful deployment pipeline experiences.

## Environment Agnostic by Configurable

- ▶ Switch config set (OO language is easier of course)
- ▶ We don't need to modify the core logic

```
use_debug_keystore() {  
    export KEY_ALIAS="$DEBUG_KEY_ALIAS"  
    export KEY_PASSWORD="$DEBUG_KEY_PASSWORD"  
    export KEY_STORE_FILE="$DEBUG_KEY_STORE_FILE"  
    export KEY_STORE_PASSWORD="$DEBUG_KEY_STORE_PASSWORD"  
}  
  
use_release_keystore() {  
    set +x  
    export KEY_ALIAS="$RELEASE_KEY_ALIAS"  
    export KEY_PASSWORD="$RELEASE_KEY_PASSWORD"  
    export KEY_STORE_FILE="$RELEASE_KEY_STORE_FILE"  
    export KEY_STORE_PASSWORD="$RELEASE_KEY_STORE_PASSWORD"  
}
```

```
java -jar bundletool.jar \  
    build-apks \  
    --bundle="$BUNDLE_FILE" \  
    --output=temp.apks \  
    --overwrite \  
    --mode=universal \  
    --ks="$KEY_STORE_FILE" \  
    --ks-key-alias="$KEY_ALIAS" \  
    --ks-pass="pass:$KEY_STORE_PASSWORD" \  
    --key-pass="pass:$KEY_PASSWORD"
```