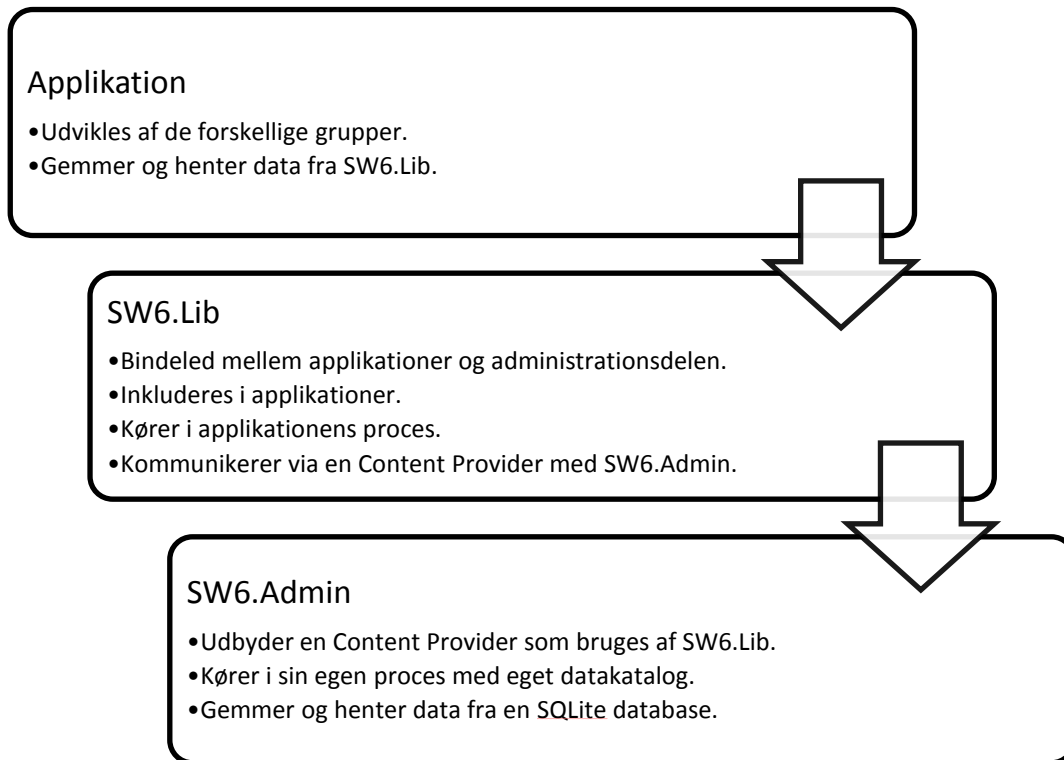


## Introduktion til SW6.Lib version 0.1

Sw6.lib er et programbibliotek udviklet af SW6b med det formål at danne et bindeled mellem administrationsdelen og de enkelte applikationer. For nemmere overskue hvor SW6.Lib skal placeres i det samlede system ses nedenstående figur.



Planen er at SW6.Admin altid er installeret på systemet således det altid er muligt at hente og gemme indstillinger i databasen. Det vil altså ikke være muligt at køre de forskellige applikationer uden SW6.Admin eftersom SW6.Lib vil forsøge at forbinde til den Content Provider der er inkluderet i SW6.Admin. Det er ligeledes planen at SW6.Lib tilbyder et stabilt og gennemttestet interface der gør det nemt at benytte den database der er placeret i SW6.Admin. Udover at tilbyde at gemme og hente data fra databasen vil der også være andre nyttige funktioner. En liste over de ting der i øjeblikket er implementeret findes i næste afsnit.

## Features i SW6.Lib version 0.1

- SW6.Admin
  - Opretter administrationsdatabase (til håndtering af fælles indstillinger) hvis der ikke er nogen i forvejen (typisk lige efter SW6.Admin er installeret).
  - Tilbyder en Content Provider.
  - Implementerer muligheder for at get/update/insert/ variabler af typen Boolean/Integer/String/Object.
- SW6.Lib
  - Tilbyder indtil videre følgende funktioner, som kan bruges af udviklere (sw6.lib.Settings):
    - getBoolean, getInteger, getString, getObject.

- setBoolean, setInteger, setString, setObject.
- addBooleanObserver, addIntegerObserver, addStringObserver, addObjectObserver, removeObserver.
- Inkluderer et eksempel på en klasse (sw6.lib.types.Interval) med mere avanceret egenskaber som benyttes via \*Object metoderne. Det er naturligvis muligt at gemme sine egne klasser og Interval-klassen skal kun ses som et eksempel.
  - Bemærk venligst at hvis der indsættes objekter i administrationsdatabasen for de fælles indstillinger, så er det påkrævet at der skrives et plugin til vores PC-interface, da plug-in'et så vil blive brugt til at repræsentere en sub-GUI til redigering af indstillinger i objektet. Det er ikke sw6b der skal skrive dette plugin. Det er op til de enkelte grupper. Men bare rolig: det er ikke super vigtigt lige nu, men nu er I advaret ;)

### *Ekstra kommentar til \*Observer metoderne*

Disse metoder gør det muligt at "abonnere" på en given variabel og få besked hvis denne variabel ændrer værdi. Dette gøres ved at sende et objekt med metoden der nedarver fra ContentObserver og som så overskriver onChange metoden. Vi har ikke noget konkrete bud på hvornår det kunne være praktisk men et eksempel kunne fx være at abonnere på hvornår mobilen ændrer position og derved sørge for at de forskellige applikationer reagerer ud fra dette.

## Vejledning i brugen af SW6.Lib

Denne vejledning forudsætter at brugen har:

- Installeret og opsat Eclipse eller MOTODEV til Android.
- Installeret og opsat SVN understøttelse til Google Code.

Guiden vil tage udgangspunkt i MOTODEV.

1. Importer projektet sw6.lib/trunk.

Projektet er nu importeret. Der er sat en indstilling således at det kun bliver betragtet som et programbibliotek og vil derfor ikke kunne køres for sig. Det betyder også at de projekter der vælger at inkludere projektet, automatisk vil få source koden med i sit projekt. Det er muligt at programmere alene ud fra SW6.Lib men hvis udvikleren ønsker at teste sin applikation er det nødvendigt at importere SW6.Admin projektet som skal lægges ned på telefonen.

2. Importer projektet sw6.admin/trunk.

Herefter er det på tide at tilføje SW6.Lib som et programbibliotek til dit eget projekt. Dette gøres ved følgende vejledning:

3. Højreklik på dit eget projekt og vælg *Properties*.
4. Vælg noden med navnet *Android*.
5. I *Library* området vælg *Add...*

6. Vælg sw6.lib i oversigten og tryk derefter OK.
7. Nu skulle der gerne være et grønt flueben ved siden af "../sw6.lib". Tryk herefter OK.

Nu er det muligt at programmere op imod de metoder der er implementeret i SW6.Lib. De metoder der bør benyttes af udviklerne er dem i roden af sw6.lib pakken (fx klassen sw6.lib.Settings). Der findes et eksempel på et dataobjekt i pakken sw6.lib.types (sw6.lib.types.Interval) men planen er at der senere kommer flere klasser herinde der kan benyttes. Klasserne der er defineret her vil være "officielt" understøttet hvilket i sidste ende vil betyde at vi (admin-gruppen) laver GUI elementer til indstillinger for disse klasser.

Som tidligere beskrevet så er SW6.Lib blot et bindeled mellem applikationen og SW6.Admin. Den nemmeste måde at sørge for at SW6.Admin ligger på telefonen er at gøre din egen applikation afhængig af SW6.Admin eftersom Android SDK selv herefter vil sørge for at SW6.Admin bliver uploaded og evt. opdateret på telefonen. Fremgangsmåden er:

8. Højre klik på dit eget projekt og vælg *Properties*.
9. Vælg noden med navnet *Java Build Path*.
10. Vælg den tab ved navn *Projects*.
11. Tryk *Add...*
12. Marker *sw6.admin* og tryk herefter OK.
13. Tryk OK.

Ved næste build&run af dit projekt skulle der gerne komme følgende i Console:

```
...
[2011-03-07 15:49:14 - sw6.schedule] Project dependency found, installing:
sw6.admin
[2011-03-07 15:49:14 - sw6.admin] Uploading sw6.admin.apk onto device 'emulator-
5554'
[2011-03-07 15:49:14 - sw6.admin] Installing sw6.admin.apk...
[2011-03-07 15:49:19 - sw6.admin] Success!
```

Ligeledes vil sw6.admin.apk blive installeret hvis der sker opdateringer med SW6.Admin projektet (som fx ved SVN update). **Det er dog vigtigt at deaktivere afhængigheden med SW6.Admin projektet når en endelig APK fil skal laves.**

## Dokumentation

Vi dokumenterer løbende funktionerne i lib'et (primært Settings.java er dokumenteret nu) og kører Doxygen på det hele. Link til dokumentationen er her:

<http://dropbox.lcdev.dk/doc/index.html>

## Hvad så nu...

Det er klart at der stadig mangler en hel del før vi kan kalde vores SW6.Lib færdigudviklet. Vores fokus er i øjeblikket at implementere de underliggende systemer i SW6.Admin som fx brugen af XML filer der specificerer hvilke variabler de enkelte applikationer ønsker databasen skal tilbyde.

Formålet med denne version 0.1 udgivelse af SW6.Lib (og tilhørende SW6.Admin) er primært at få feedback på det interface der er lavet i klassen sw6.lib.Settings og se efter om der kan opstå nogle uforudsete fejl. Det er også samtidig en mulighed for at modtage forslag til andre metoder som vi herefter vil vurdere om er relevante for administrationsdelen.

Det er **VIGTIGT** at være opmærksom på at det interface der i øjeblikket er til stede ikke skal ses som stabilt. Vi har forsøgt at teste metoderne og lavet dem der virker mest oplagte således der er mindst mulig risiko for at de metoder der er i version 0.1 bliver ændret med tiden. Med andre ord anser vi den nuværende funktionalitet som kerne-funktioner i vores programbibliotek.

Vi har dog i øjeblikket ændret funktionaliteten for Set\* metoderne. I version 0.1 vil metoderne være i stand til at tilføje til databasen (normalt vil metoderne blot opdatere eksisterende variabler) således det er nemt for jer udviklere at teste vores Lib. Dette vil dog blive ændret i kommende versioner således at de variabler der ønskes benyttet skal beskrives i en XML fil. Da der stadig arbejdes på denne feature kan vi ikke komme med yderligere detaljer omkring dette endnu.

## Kort sagt!

- Hent SW6.Lib.
- Hent SW6.Admin.
- Lav således jeres applikation(er) bruger SW6.Lib og se om der mangler noget funktionalitet.
- Hvis der bliver fundet fejl eller mangler så kom med dem på:  
<http://code.google.com/p/sw6android/issues/list>
- Evt. spørgsmål kan sendes til gruppe SW6b MEN LÆS DOKUMENTATIONEN FØRST (RTFM) PÅ:  
<http://dropbox.lcdev.dk/doc/index.html>