

**Allmänt:**

Modellen är snarare domändesign än domänmodell.

**Som utvecklare, skulle modellen vara till hjälp? Varför, varför inte?**

Om modellen ses som resultatet av en första iteration, skulle den kunna användas för vidare arbete. Modellen skulle efter vissa korrigeringar vara en bra grund för implementation.

**Tror du att en domänexpert skulle förstå modellen? Varför, varför inte?**

Till viss del. Objektnamnen är generellt sett enkla och tydliga, men begrepp (metoder och datatyper) tillhörande domändesignen hade kunnat tas bort.

**Vad är styrkorna med modellen, vad tycker du är riktigt bra och varför?**

- Modellen är till synes enkel och överskådlig.
- Objektnamnen är överlag enkla och tydliga.
- Meeting och dess attribut.

**Vad är svaghetera med modellen, vad anser du skulle ändras och varför?**

- Metoderna som finns på Secretary borde fördelas mellan Dock och Calendar.
- Secretary behöver en association till Calendar.
- "Slip" är förvirrande: varför används metaforen för en bokningslapp/bokningstillstånd? Kanske är mooring (kajplats) mer passande? Associationen Pays skulle förslagsvis kunna vara pays for.
- *Angående domändesign:* Metoderna på Member borde fördelas mellan Boat (eller någon form av båthanterare) och Calendar.ShowMeeting() och ViewCalendar(), vad är skillnaden?
- *Angående domändesign:* PayFee() däremot bör vara kvar på Member.
- *Angående domändesign:* Märklig datatyp 0..1 på Boat Image.

**Anser du att modellens utformning når upp till betyg 3?**

Vissa korrigeringar skulle behövas för en bra domändesign. Genom att skala av den nuvarande domändesignen kan en bra domänmodell skapas. Om metoder och datatyper tas bort anser vi att modellens utformning når upp till betyg 3.