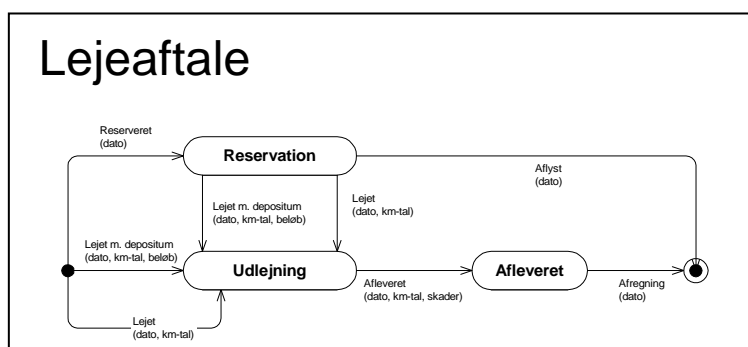
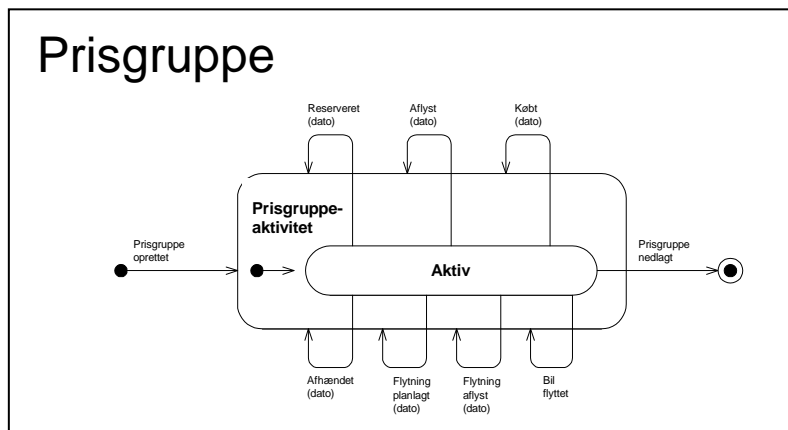
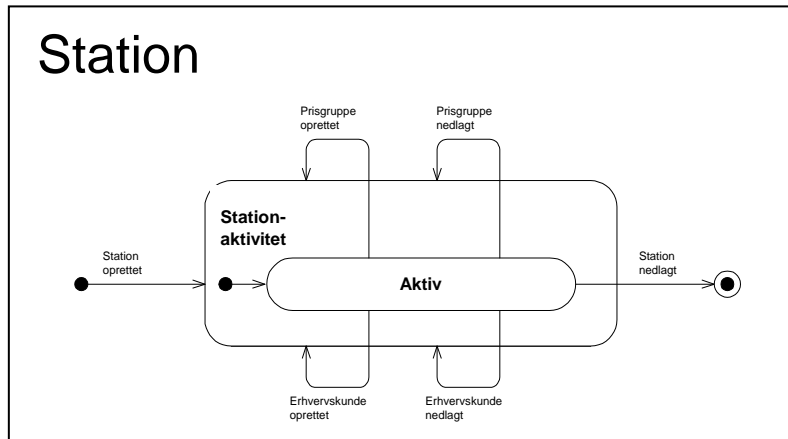
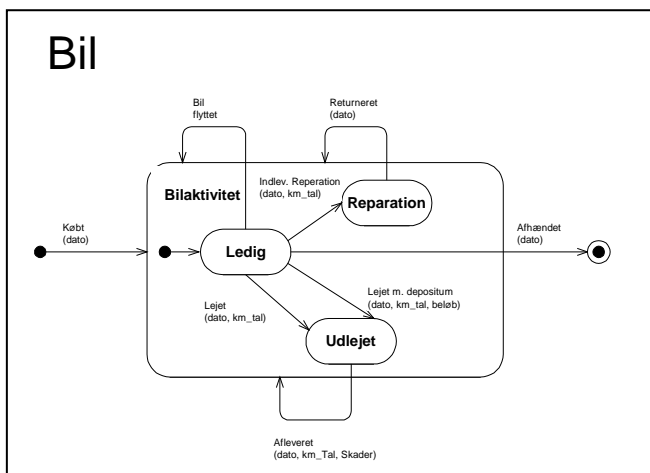
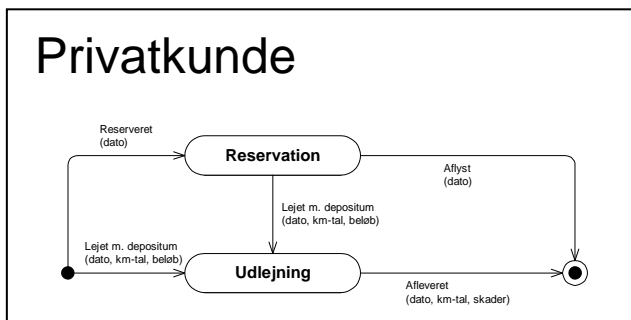
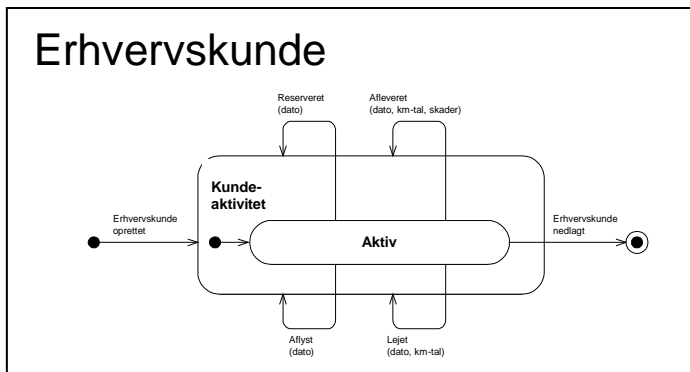
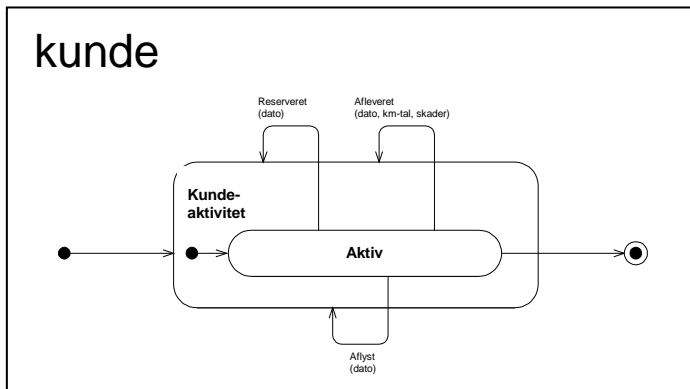


Udgangspunktet for designet er informationer fra analysen, samt nedenstående justerede tilstandsdiagrammer:





I det følgende argumenteres for hvordan hændelserne kan repræsenteres på klasserne. Der tages udgangspunkt i hændelsesoversigten.

Reserveret:

Dato registreres på klassen Lejeaftale, som altid vil blive involveret og hvor den kun optræder en gang og hermed giver den mest enkle repræsentation. Lejeaftalen tilføjes attributten: ReserveretDato. Da hændelsen ikke altid forekommer på lejeaftale skal der også tilføjes en valgattribut Reserveret (ja/nej), så man kan registrere om der har været en reservation.

Aflyst:

Dato registreres på klassen Lejeaftale, som altid vil blive involveret og hvor den ligesom reserveret kun optræder en gang. Lejeaftalen tilføjes attributten: AflystDato. Da hændelsen ikke altid forekommer på lejeaftale skal der også tilføjes en valgattribut. Denne kan kombineres med om valget af udlejning med valgattributten Udlejningstype (aflyst/med depositum/uden depositum), idet lejeaftalens tilstandsattribut så anvendes til at registrere om hændelsen er sket.

Lejet med depositum og Lejet:

Dato, km-tal og beløb registreres på klassen Lejeaftale, som altid vil blive involveret og hvor den kun optræder en gang. Lejeaftalen tilføjes attributterne: UdlejetDato, Udlejet-Km-tal og Depositumbeløb. Da kun den ene af hændelserne kan forekomme kan attributterne deles. Dette registreres med valgattributten Udlejningstype (aflyst/med depositum/uden depositum) samt med lejeaftalens tilstandsattribut.

Afleveret:

Dato, km-tal og skader registreres på klassen Lejeaftale, som altid vil blive involveret og hvor den kun optræder en gang. Lejeaftalen tilføjes attributterne: AfleveretDato, Afleveret-Km-tal og Skader. Registrering af, at hændelsen har fundet sted, sker med valgattributten Udlejningstype og lejeaftalens tilstandsattribut.

Afregnet:

Dato registreres på klassen Lejeaftale, som altid vil blive involveret og hvor den kun optræder en gang. Lejeaftalen tilføjes attributten: AfregnetDato. Registrering af, at hændelsen har fundet sted, sker med valgattributten Udlejningstype og lejeaftalens tilstandsattribut.

Købt:

Dato registreres på klassen Bil, hvor den den simpleste repræsentation kan foretages. Bil tilføjes attributten: KøbtDato. Registrering af, at hændelsen har fundet sted, sker med bilens tilstandsattribut.

Afhændet:

Dato registreres på klassen Bil, hvor den den simpleste repræsentation kan foretages. Bil tilføjes attributten: AfhændetDato. Registrering af, at hændelsen har fundet sted, sker med bilens tilstandsattribut.

Flytning planlagt:

Denne hændelse kræver en ny klasse, da den er repeterende på Prisgruppe-klassen og ikke kan pladsers andre steder. Der oprettes en ny klasse FLYTNING til hændelsens data. Dato registreres på klassen Flytning som med attributten: FlytPlanDato. Den nye klasse aggregeres ind i klassen Prisgruppe, således at tilknytningen til denne klasse bevares. Da en flytning sker fra en prisgruppe til en anden, vil aggregeringen ske i 2 prisgrupper.

Flytning aflyst:

Denne hændelse repeterende på Prisgruppe-klassen ligesom Flytning planlagt. Hændelsen hænger sammen med Flytning planlagt og kan også pladsers på den nye klasse FLYTNING. Dato registreres på klassen Flytning som med attributten: FlytAflystDato. Registrering af, at hændelsen har fundet sted, sker med tilstandsattributten på klassen Flytning.

Bil flyttet:

Denne hændelse repeterende på Prisgruppe- og bil-klassen. Hændelsen hænger sammen med Flytning planlagt. Hændelsen har ingen attributter, men kan registreres gennem en associering mellem klassen Flytning og klassen Bil samt tilstandsattributten på klassen Flytning.

Indleveret:

Denne hændelse kræver en ny klasse, da den er repeterende på Bil-klassen og ikke kan pladsers andre steder. Der oprettes en ny klasse REPARATION til hændelsens data. Dato og værksted registreres på klassen Reparation som med attributterne: IndleveretDato og Værksted. Den nye klasse aggregeres ind i klassen Bil, således at tilknytningen til denne klasse bevares.

Returneret:

Denne hændelse hænger sammen med Indleveret og kan ligeledes pladsers på klassen REPARATION. Dato registreres på klassen Reparation som med attributten: ReturneretDato. Registrering af at hændelsen har fundet sted, styres af tilstandsattributten på klassen Reparation.

Erhvervskunde oprettet:

Hændelsen har ingen attributter. Registrering af, at hændelsen har fundet sted, sker med Erhvervskundens tilstandsattribut.

Erhvervskunde nedlagt:

Hændelsen har ingen attributter. Registrering af, at hændelsen har fundet sted, sker med Erhvervskundens tilstandsattribut.

Prisgruppe oprettet:

Hændelsen har ingen attributter. Registrering af, at hændelsen har fundet sted, sker med Prisgruppens tilstandsattribut, samt af aggregeringen op i Station.

Prisgruppe nedlagt:

Hændelsen har ingen attributter. Registrering af, at hændelsen har fundet sted, sker med Prisgruppens tilstandsattribut.

Station oprettet:

Hændelsen har ingen attributter. Registrering af, at hændelsen har fundet sted, sker med Stationens tilstandsattribut.

Station nedlagt:

Hændelsen har ingen attributter. Registrering af, at hændelsen har fundet sted, sker med Stationens tilstandsattribut.

Oversigt over klassernes tilstandsattributter:

Erhvervskunde:	KundeTilstand	(Aktiv/Nedlagt)
Privatkunde:	KundeTilstand	(Aktiv/Nedlagt)
Lejeaftale:	AftaleTilstand	(Reservation/Udlejning/Afleveret/Afsluttet)
Station:	StationTilstand	(Aktiv/Nedlagt)
Prisgruppe:	PrisGrTilstand	(Aktiv/Nedlagt)
Bil:	BilTilstand	(Ledig/Reparation/Udlejet/Afhændet)
Flytning:	FlytTilstand	(Planlagt/Aflyst/Flyttet)
Reparation:	RepTilstand	(Reparation/Returneret)

Ud fra denne argumentation følger nu det reviderede klassediagram, hvor der er taget højde for registreringen af hændelserne:

Klassediagram for biludlejning Design - før metoder påsættes

