

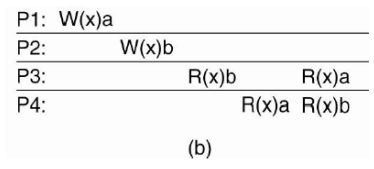
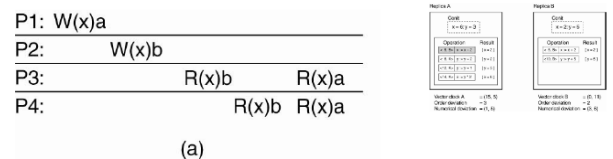
7. Consistency

Consistency models (Data)

Data-store holder data. En process der kan tilgå noget af den, kan tilgå det hele.

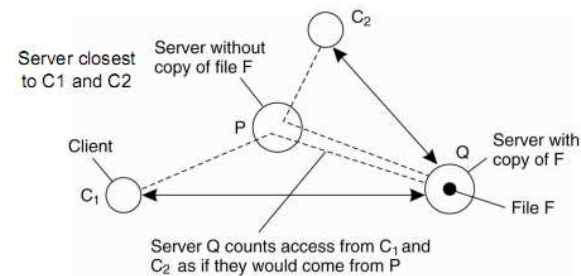
Modellen beskriver samarbejdet mellem processer og data-storen. Inkonsistens og afvigelser kan måles på forskellige måder:

- Inkonsistens og afvigelser kan måles på forskellige måder
 - **Numerisk:** Bruges i numerisk data. Angiver max afvigelse mellem kopier, enten absolut eller relativt. Kan også være opdateringer der endnu ikke er set af andre.
 - **Staleness:** Hvornår blev der sidst udført opdateringer. Kan kræve opdatering efter x tid.
 - **Ordrede operationer:** Ordningen af opdateringer kan variere lokalt, men skal stemme når de commites globalt.
- **Conit:** God måleenhed for hvor meget kopier afviger. En conit bør kun dække sammenhængende data. Der skal angives, hvor stor afvigelse der må være, conits imellem. **[TRANSPARENT]**
- **Sekventiel consistency:** Alle processer ser den samme rækkefølge af operationer på en data store. **[TRANSPARENT]**



Replica management

- **Serverplacering:** Administrativ/kommerciel beslutning end optimering.
- **Indholdsreplikering og -placering [TRANSPARENT]**
 - **Permanent:** Flere servere på en lokation, eller mirroring på flere servere fordelt geografisk.
 - **Server-initiated:** Ved "flash-crowds" laves en midlertidig kopi i det område. Flere læsninger end skrivinger. Tæller hvor mange gange en "migrerende" fil bliver hentet og hvor mange gange den kunne være blevet hentet – afgør på baggrund af dette, hvad der skal ske.
 - **Client-initiated:** Cacher lokale kopier. Helt op til klienter.
- **Indholdsdistribuering** (Hvordan sendes opdateringer rundt?)



1. **Invalidation:** Sender notification rundt om at en del af dataen er opdateret, og ikke konsistent mere.
2. **Overfør data:** Send data rundt
3. **Active replication:** Send instruksen rundt til alle kopier.
 - **Push:** Send updates til andre. Høj grad konsistens. Server-init.
 - **Pull:** Ofte brugt til klient-caching.
 - **Lease** er et løfte fra serveren om, at den i et tidsrum sender opdateringer. Kombi af push og pull.

| Issue | Push-based | Pull-based |
|-------------------------|--|-------------------|
| State at server | List of client replicas and caches | None |
| Messages sent | Update (and possibly fetch update later) | Poll and update |
| Response time at client | Immediate (or fetch-update time) | Fetch-update time |

Primary-based protocols

- **Remote-write protocol:** (primary backup protocol). Sekventiel konsistens. Nonblocking ved at sende ACK tilbage med det samme. Ved ikke om operation rent faktisk er udført.

