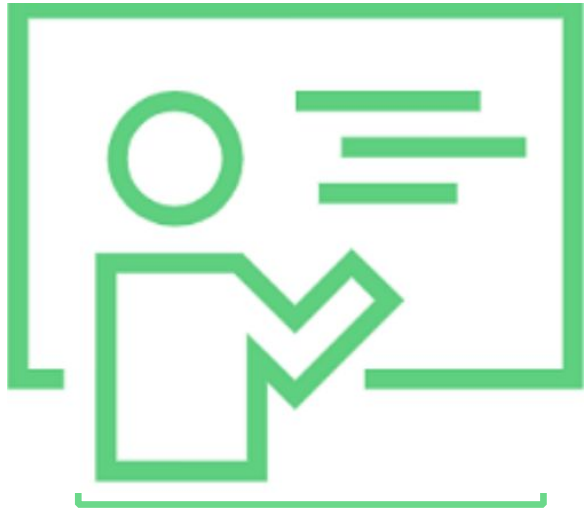




Tjänsteorienterad arkitektur och systemintegration

Joakim Englund, Systemmentor AB

Repetition från dag 4

- Intern datamodell och extern representation
- Validering före serialisering
- Parsning och validering efter mottagning
- JSON eller XML enligt ett gemensamt kontrakt

Dagens IoT-flöde

- Sensor eller simulator lämnar en mätning
- Broker förmedlar meddelandet
- Adapter anropar API:t
- Konsument läser och visar värdet

Dagens mål

- Förklara hur tjänster samverkar
- Integrera minst två tjänster
- Dokumentera gränssnitt och ansvar
- Lokalisera ett fel i flödet

Tjänsteorienterad arkitektur

- Aka SOA - Service Oriented Arcitecture
- Avgränsat ansvar
- Beskrivna gränssnitt
- Samverkan mellan tjänster
- Möjlighet att byta implementation bakom kontraktet

Distribuerade komponenter

<i>Komponent</i>	<i>Möjlig plats</i>
Sensor	Rum eller maskin
Broker	Lokal gateway
API	Server eller moln
Konsument	Driftens dator

Request och response

GET /api/readings/latest

- Klienten ställer en fråga (eller begäran)
- Servern svarar med status och data
- Klienten behöver hantera väntan och fel

Publish och subscribe

- Publisher väljer ett topic
- Subscriber prenumererar via brokern
- Sensorn behöver ingen lista över mottagare
- Leverans beror på val av QoS och session

Adapter mellan MQTT och HTTP

1. Ta emot payload
2. Parsa och validera
3. Serialisera enligt API-kontraktet
4. Skicka POST och kontrollera svaret

Kontrakt över tjänstegränser

```
{"sensorId":"temp-01","value":21.7,"unit":"C"}
```

- Fältnamn och typer
- Enhet och tillåtna värden
- Topic eller endpoint
- Felbeteende och formatversion

REST, SOAP och befintliga system

<i>Situation</i>	<i>Möjligt gränssnitt</i>
Ny app läser mätdata	REST-API
Befintligt verksamhetssystem	SOAP-tjänst
Olika kontrakt ska samverka	Adapter

Partiella fel

- Broker fungerar
- API är avstängt
- MQTT kan ändå ta emot en publicering
- Konsumenten får ingen ny mätning

Timeout och omsändning

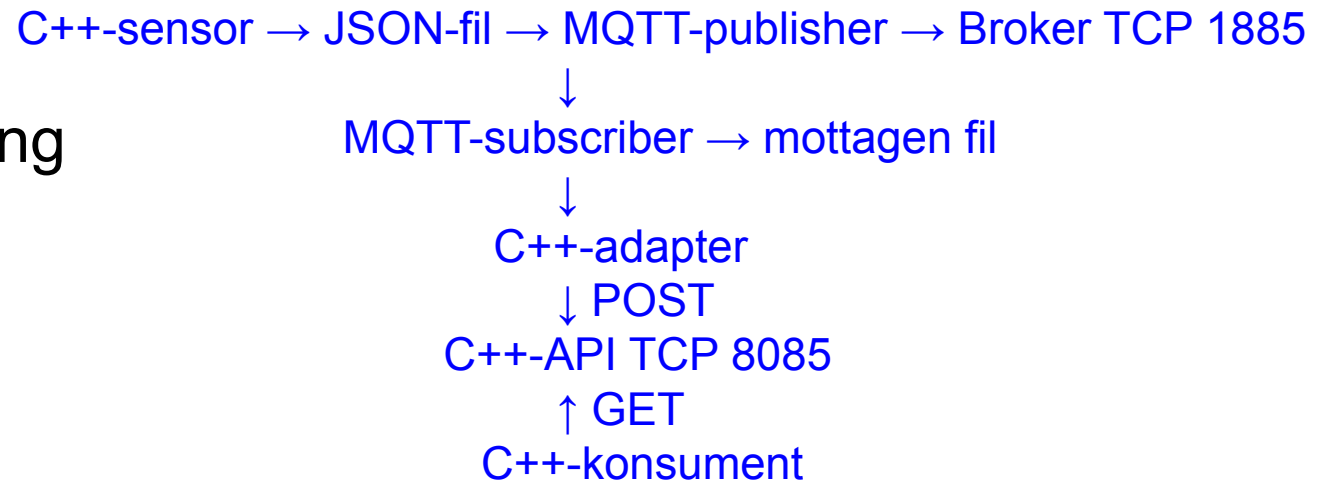
- Timeout begränsar väntan
- Ett uteblivet svar kan ha flera orsaker
- Omsändning kan ge dubletter
- Fel behöver dokumenteras/loggas

Senaste värde och gammalt värde

- Senaste mottagna värde kan vara gammalt
- Minneslagring försvinner vid omstart
- Tidsstämpel kan hjälpa konsumenten
- Nåbarhet och datans aktualitet är olika frågor

Demo: Test av hela flödet

1. Skicka en känd giltig mätning
2. Jämför värdet hos konsumenten
3. Skicka ogiltig data
4. Stoppa en tjänst
5. Återställ och skicka en ny mätning



Eftermiddagens arbete

- Kör referensflödet
- Anpassa gruppens IoT-case
- Behåll datakontraktet från dag 4
- Spara skiss, tester och startkommandon

Anslutningar inför dag 6

- Källa och mål
- Vem som initierar anslutningen
- Transportprotokoll och målport
- Vilken trafik systemet behöver