
Elmarknadshubben: Kompetensbaserad upphandling



**SVENSKA
KRAFTNÄT**
SWEDISH NATIONAL GRID

AGENDA

- > Projekt Elmarknadshubben; bakgrund
- > Förutsättningar och utgångspunkter
- > Traditionell upphandling jämfört med kompetensbaserad
- > Strategiska val
- > Vad upphandlade vi?
- > Upphandlingens olika steg
- > Arkitekturuppgift
- > Code Camp uppgift
- > Utmaningar och lärdomar

Syfte:

Dela med oss av våra erfarenheter av
upphandlingen i projekt
elmarknadshubben

Projekt Elmarknadshubb; bakgrund

Svenska kraftnät

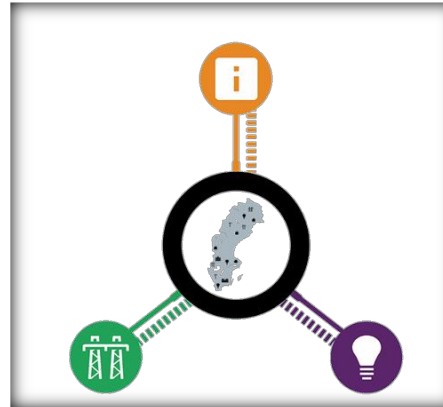
"Uppdrag att utveckla och driva en hubb."



Energimarknadsinspektionen

"Uppdrag att utreda författningsändringar för införande av en hubb och en elhandlarcentrisk marknadsmodell."

Elmarknadshubben



- Ny kommunikationslösning
- Centralisering av processer
- Elhandlarcentrisk marknadsmodell
- Semi publikt API för 400 aktörer
- 5 miljoner mätpunkter
- Faktureringsunderlag på ca 37 miljarder per år

Förutsättningar och utgångspunkter inför upphandling

- > Svenska kraftnät ska inte utveckla hubben
- > Genomförd RFI och erfarenheter från andra nordiska länder visade att det inte fanns några färdiga system på marknaden
- > Oklar kravbild pågående att den nya lagen inte är på plats
- > Stor komplexitet
- > Kostnad över tid

Traditionell upphandling

- Kravställ i detalj (missa inget....)
- Välj leverantör som uppfyller flest krav och har lägst pris
- Inköp och försäljning



- Svårt att ändra krav
- Dyrare lösning

Kompetensbaserad upphandling

- Kravställ utifrån behov och mål
- Välj leverantör utifrån kompetens och förmåga
- Projektteam (leverantör och kund)



- Fatta viktiga beslut sent
- Billigare lösning

Graden av komplexitet en viktig faktor!!

Strategiska val

Strategin utgår från att vi inte vet allt och det effektivaste sättet att få reda på mer är att bygga delar av systemet

- > I ett första steg utveckla en prototyp – därefter option om fullskalig implementering
- > Helhetsåtagande för lösningen
- > Upphandla ett agilt team med rätt kompetens för att utveckla elmarknadshubben
- > Vi ska inte göra en bra upphandling

Minska risk genom delleveranser

- > **Leverans 1**
 - > Prototyp
 - > SAD
 - > SOW
- > **Leverans 2**
 - > Implementation av hela elmarknadshubben

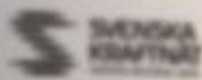
Hur vi upphandlade och varför?

- > Konkurrens (enkelhet och liten insats från leverantörerna)
- > Validera teamets kompetens (Arkitekturuppgift samt en code camp)
- > Medverkande; det framtida projektteamet
- > Namnge nyckelpersoner i teamet (två IT-arkitekter och två seniora utvecklare)
- > Vikta kvalité högt
- > Förhandling och dialog viktig del av utvärderingsprocessen

Hur ser "drömlieferantören" ut?

Röd text utvärderas i pre kval
Blå text utvärderas i utvärdering 1

Plus	Minus
Backend-orienterad	<i>Produktbolag (licenser)</i>
Erfarenhet av publika API:er	<i>Nätverks/paraplybolag</i>
BPM erfarenhet	
Erfarenhet av olika tekniska ekosystem	
Erfarenhet av continuous integration	
Erfarenhet av continuous delivery	
Kompetensorienterat	
Innovationsdrivet	
<i>Kapacitet och redundans</i>	
<i>Struktur och ordning och reda</i>	
<i>Lagom stor jämbördig i en kund- leverantörsrelation</i>	
<i>Myndighetsvana</i>	
<i>Erfarenhet av att realisera mot detaljerade säkerhetskrav</i>	
<i>80 % av personal svenskt medborgarskap</i>	
<i>80 % av ägare svenskt medborgarskap</i>	



Upphandlingens olika steg

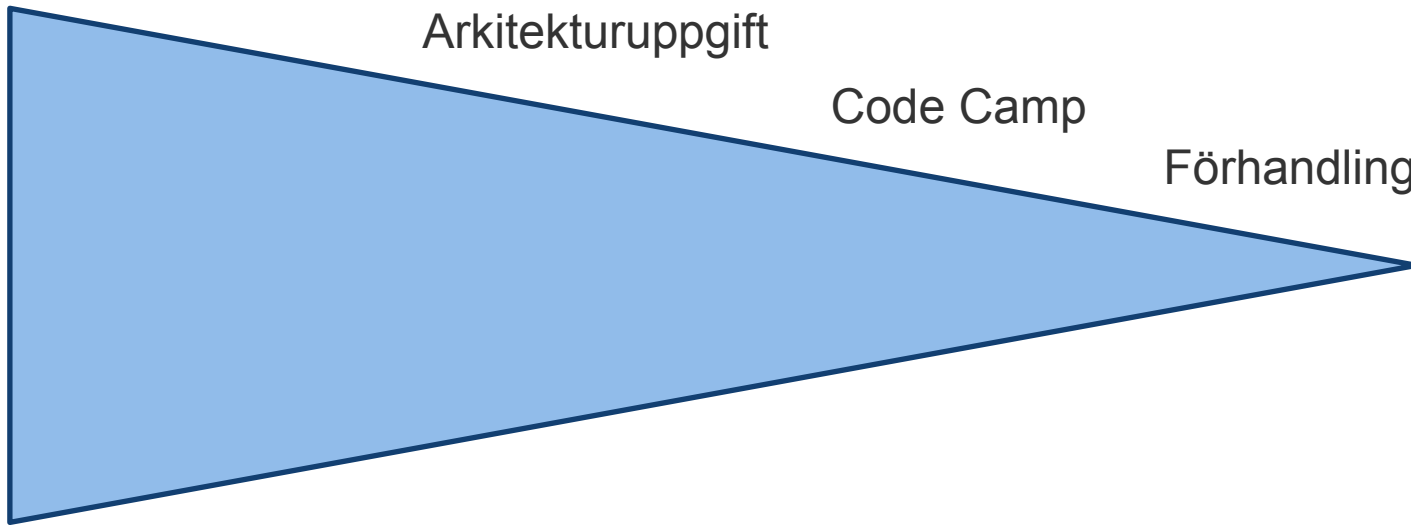
Pappersutvärdering

Arkitekturuppgift

Code Camp

Förhandling

Vald leverantör



Antal
leverantörer

16

10

6

4

3

1

Pappersutvärdering

> eQual

- > Företag
- > Referensprojekt

> ITT

- > CVs på core team medlemmar
- > Project setup
 - > Förslag på team och arbetssätt
 - > De största utmaningarna för lev resp kund

Arkitekturuppgift

- > **Syfte: utvärdera leverantörens arkitekturförmåga/-kompetens**
 - 1) Design av uppgift
 - 2) Test av uppgift och utvärdering
 - 3) Genomförande: 2 arkitekter
1+1 dag
 - 4) Utvärdering



Arkitekturuppgift - Design



- > Max tidsåtgång? **Leverantörerna måste kunna göra tidsinvesteringen!**
- > Karaktären på kompetensområdet? **Går uppgiften att utvärdera kvantitativt eller måste den vara kvalitativ?**
- > VAD vill man utvärdera? **Enbart lösningresultatet eller också hur leverantören arbetar på vägen mot lösningen?**
- > Vilka är utvärderingskriterierna och hur är de viktade? **Hög kvalitet viktigast? Tillräcklig kvalitet och hög produktivitet?**
- > Skall uppgiften vara fiktiv eller utgå från ett verkligt problem?

Arkitekturuppgift – Test

- > Otroligt viktigt: Bägge parter investerar rejält med tid inför och under uppgiften -> Allt måste funka när det är dags!
- > Exempel på brister som vi hittade och åtgärdade:
 - > fel antal personer för att lösa uppgiften => **justerades**
 - > uppgiften var alldeles för svår att lösa utan förberedelser av leverantören
=> **utskick för förberedelser**
 - > instruktioner kring uppgiften var för otydliga => **lösningsmall**
 - > flera av testarna upplevde situationen som stressande. Hur ska det då vara för de som gör uppgiften på riktigt? Så kommer ju inte "en dag på jobbet" alls se ut...
=> **lämna dem ifred så mycket som möjligt**
 - > flera fastnade i detaljfrågor och glömde bort att disponera tiden rätt => **moderera**
 - > snacks & dryck lättillgängligt för att orka tänka under press en hel dag
=> **handla själva** 😊

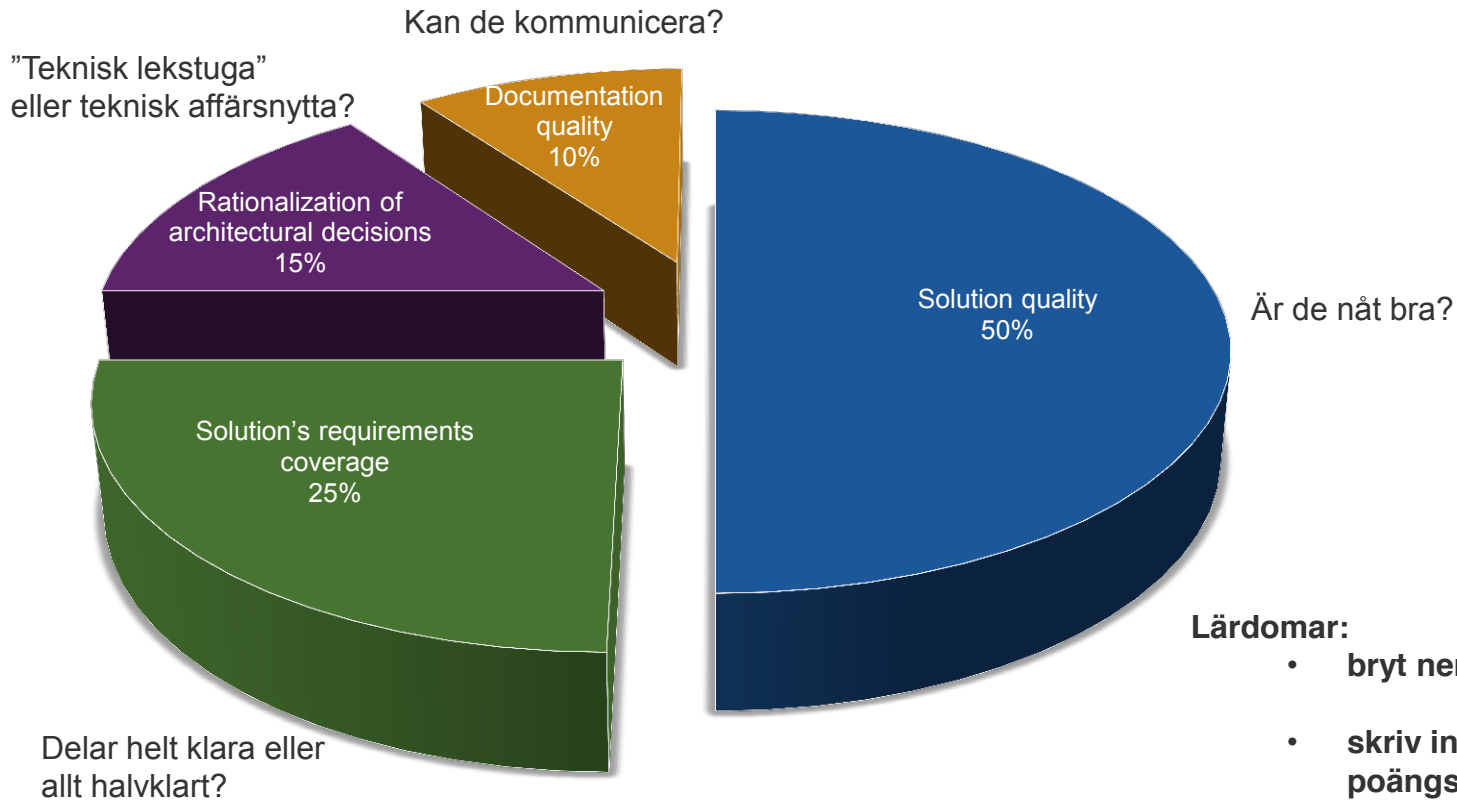


Arkitekturuppgift – Genomförande

- > Plan B måste finnas! Vad gör vi om:
 - > ansvarig person är sjuk? => **vikarie? Reservtider?**
 - > någon hos leverantören får förhinder? => **ska de ha reservpersoner eller reservtid?**
 - > övningen kommer igång en timme försent? => **sluta en timme senare? Reservtid?**
- > Lika behandling:
 - > om förberedelser på hemmaplan => **hur minimeras ”fuskmöjligheter”?**
 - > alla ska ha samma muntliga information och hjälp => **oerhört svårt!**
Minimera muntlig information och sociala kontakter utan att det blir för formellt
 - > våga låta bli att svara på ”fel” frågor: => **”hur vill ni att arkitekturen ska se ut?”**



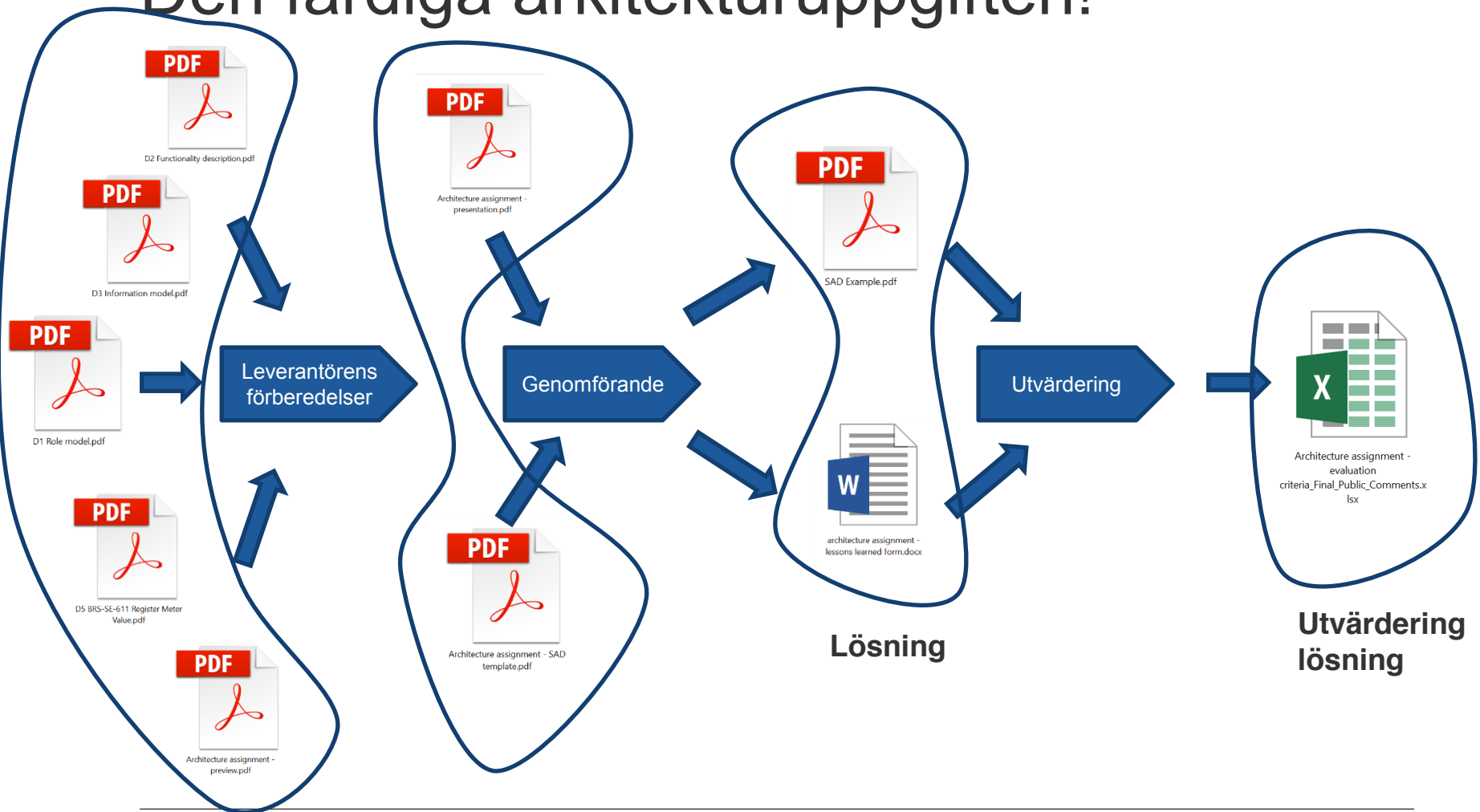
Utvärdering av Arkitekturuppgift



Lärdomar:

- bryt ner kriterier tills mätbara
- skriv instruktioner för poängsättning
- ekvivalera poängsättning

Den färdiga arkitekturuppgiften!



Problemförutsättningar

Problemet

Exempel på lösning:

4 Logical view

4.1 Archite

[Your architect

Nr	Decision
	Stand replica
	Micros auton
	Load E functio
	Event multip
	User a indivi

4.2 Archite

4.2.1 Micro s

Figure 4-1. Log

5 Use case realization

5.1 Scenario 1: Collection

Figure 5-1. Collection of meter val

6 Realization plan

The realization plan and the way forward for the suggested solution would involve a couple of transitional steps. The agile project team tasked with realizing the solution can see these steps as an organized way forward.

6.1 Initial setup

Setup the relevant environment in terms of development environment and any kind of deployable infrastructure, may it be one of the major cloud vendors, private cloud or on premise (own computer). Suggested technology for the microservices is Spring Boot and MySQL for the microstorage. Create one initial microservice together with its microstorage and setup a Docker container for that microservice. Utilize Kubernetes and setup the master and nodes accordingly to encapsulate the docker container. When Kubernetes and Docker are operational, create an API for the microservice to realize the flow of data through the microservice's API. This REST API can be a test interface.

6.2 Intermediate steps

Next step is to select one of the identified alternatives for a microservice and use the subset of the information model that relates to what the current microservice will handle. So, for example if it is the Meter Point Service (Register and dismantle metering point) that is being realized, implement the necessary tables for the microstorage and the internal DAO's in the Spring Boot project. Create a REST API endpoint for the microservice based on the given functional requirements (D2 Functional Description.pdf) that will enable API service consumers to perform the given operation.

Functions related to this Meter Point service (D2 Functional Description.pdf):

- BRS-SE-111 Register customer metering point
- BRS-SE-112 Dismantle customer metering point
- BRS-SE-113 Register exchange metering point
- BRS-SE-114 Dismantle exchange metering point
- BRS-SE-115 Register submetering point
- BRS-SE-116 Update submetering point
- BRS-SE-225 Reverse action on a metering point

The full information model can be viewed in D3 Information model.pdf. See chapter 2.2. for specific information on the Meter Point related information. Given the Meter Point information model, delimit the

information model further to only handle meter related data, e.g., Tariff data should be handled by the Tariff service.

Examples of another microservice is: Tariff service.

Furthermore, when the Meter Point microservice is realized with the related REST API, internal DAO's, and microstorage tables are completed, the next step will be to setup Kafka and enable the microservice to publish data to not only to its microstorage but also to Kafka. For this, Kafka needs to be setup accordingly and enable the microservice to write to a specified Kafka Topic handling the delimited information model for the Meter Point microservice.

Code camp uppgift 1/2

- Syfte: utvärdera utvecklingskompetens & **samarbetsförmåga**
- I stort samma förberedelser, genomförande och uppföljning som arkitekturuppgiften
- Men att utvärdera **samarbetsförmåga** gjorde denna uppgift svårare att genomföra:
 - "Deltagande observation" => **kan ge onaturliga situationer**
 - Kräver interaktion => **hur interagerar man likadant varje gång?**
 - Svårt att kvantifiera => **värdera utfall i grupp!**

```
ws.on("message", m => {
  let a = m.split(" ")
  switch(a[0]){
    case "connect":
      if(a[1]){
        if(clients.has(a[1])){
          ws.send("connected");
          ws.id = a[1];
        }else{
          ws.id = a[1]
          clients.set(a[1], {client: {position: {x: 0, y: 0}, id: a[1]}})
          ws.send("connected")
        }
      }else{
        let id = Math.random().toString().slice(2, 8)
        ws.id = id;
        clients.set(id, {client: {position: {x: 0, y: 0}, id: id}})
      }
    }
  }
})
```

Code camp uppgift 2/2

- Lärdomar:
 - Vi ville mäta generell utvecklingskompetens (oberoende av teknologi)
 - Vad vi faktiskt mätte var hur bra man kunde Springboot
 - > alla "genade" mha Spring Boot 😊
 - Vi borde ha "snävat in" förutsättningarna mer
 - Men: Elmarknadshubben kommer sannolikt byggas baserat på Spring-teknologi hursomhelst!



Utmaningar och lärdomar

- > Förändring är svårt!
 - > Extern hjälp
 - > Test
 - > Erfaret team
- > Offentlig upphandling vs privat
- > Hur få bästa leverantör/kompetens?
- > Hur gick det?