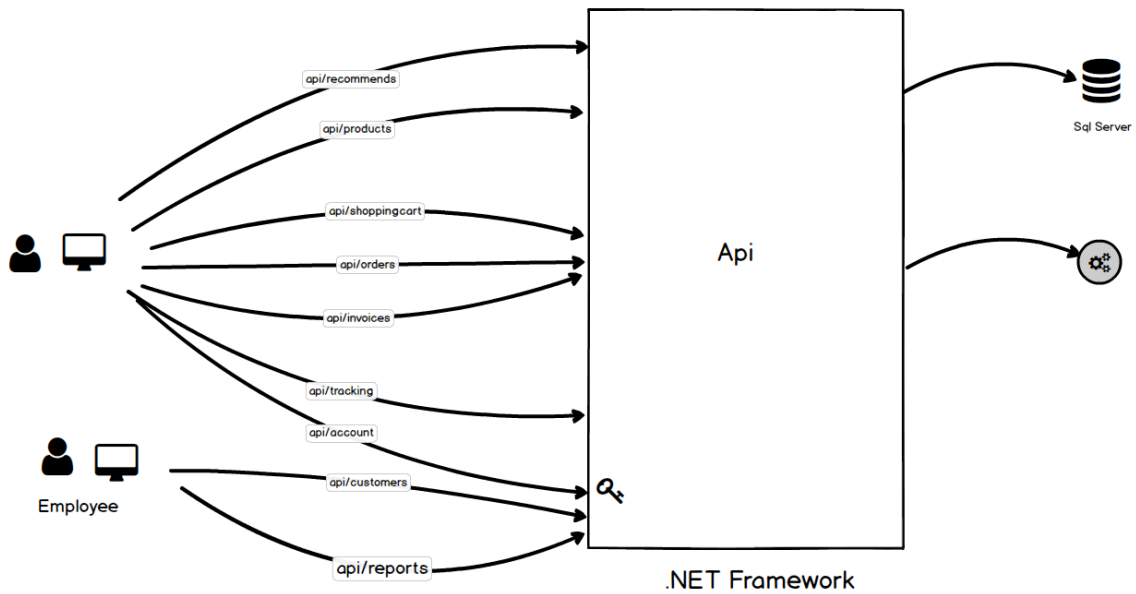
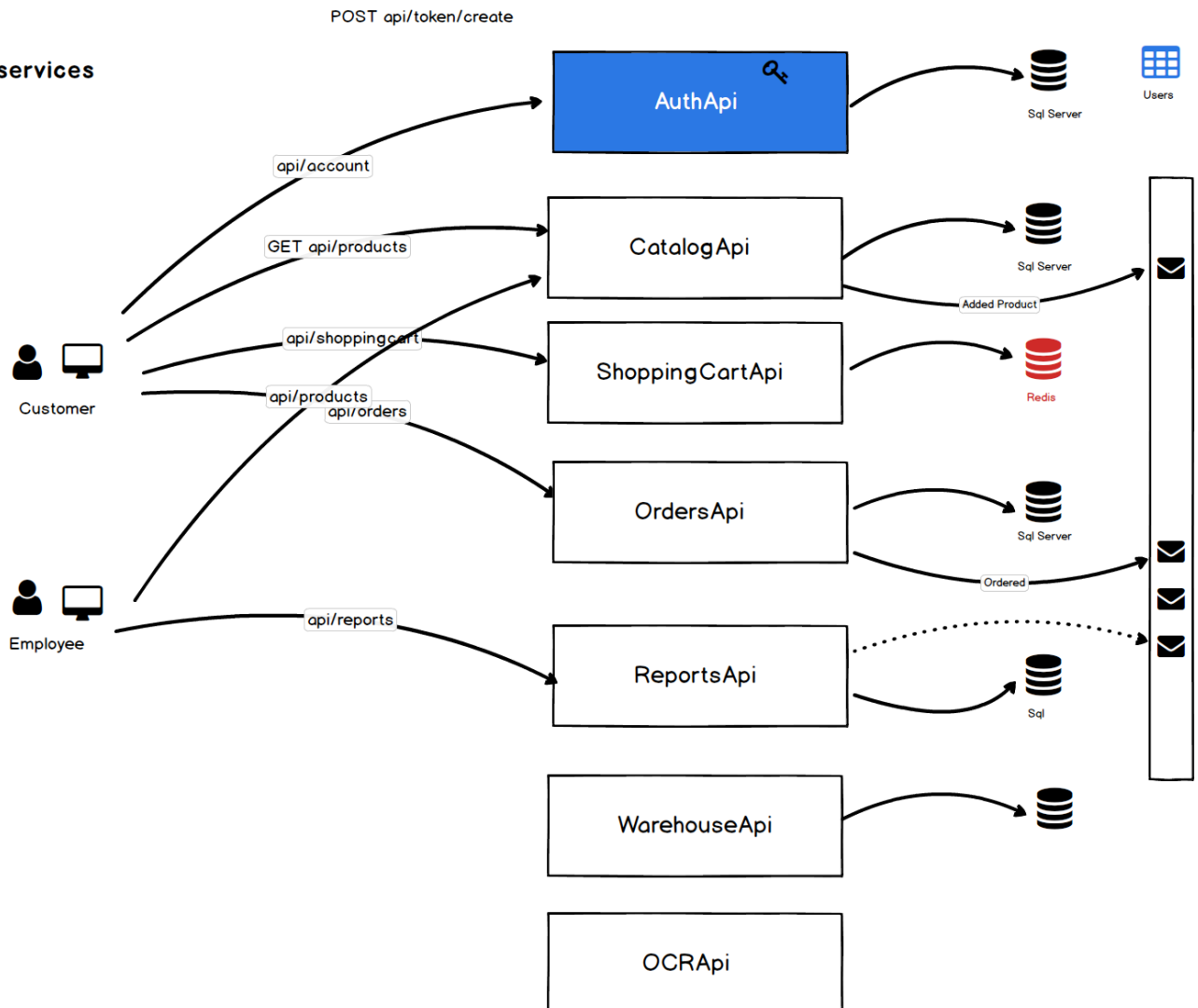


## Monolit



## Microservices



# SOAP

```
POST /service.svc HTTP/1.1
host: localhost
Content-Type: xml+soap

<xml>
<soap:Envelope>
<soap:Body>
  <AddProduct>
    <Product>product 1</Product>
  </AddProduct>
</soap:Body>
</soap:Envelope>
```

POST

# RPC

+

AddProduct(Product product)

-

RemoveProduct(int id)

Q

Get(string color)

Q

Get(int id)

# RESTFull

```
GET /api/products/1 HTTP/1.1
host: localhost
Accept: application/xml

200 OK
Content-Type: application/json

{
  "Id": 1,
  "Name": "Product 1"
}

POST /api/products/1 HTTP/1.1
host: localhost
Content-Type: application/json
{
  "Name": "Product 2"
}
```

GET api/products?page=10

GET api/products?fields=Name,Price

GET api/products

GET api/products/{id}

POST api/products

PUT/PATCH api/products

DELETE api/products/{id}

GET api/products/{id}/customers

# Resources

Products

[HttpGet]

Q

Get(string color)

[HttpGet]

Q

Get(int id)

+

AddProduct(Product product)

UpdateProduct(Product product)

-

RemoveProduct(int id)

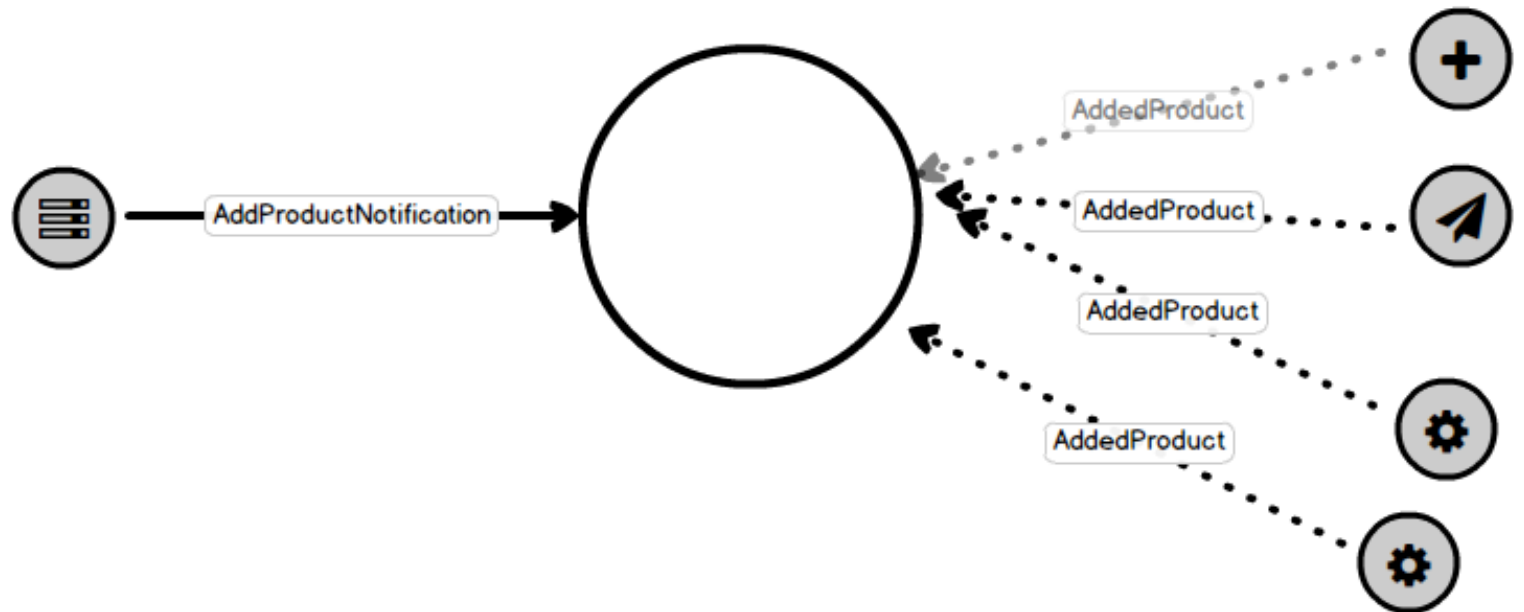
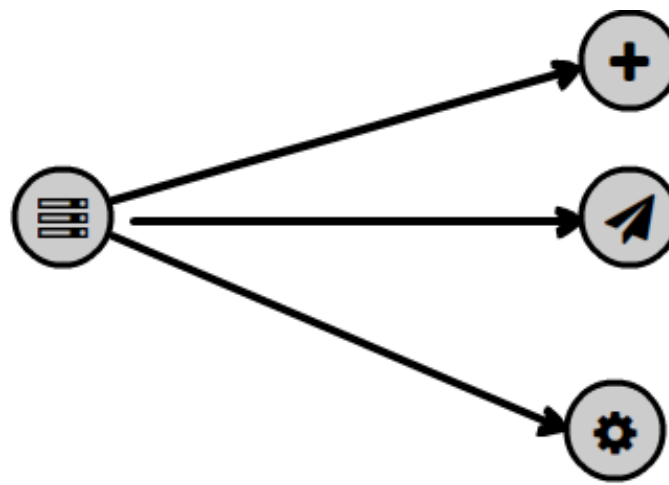
# OData

GET api/products?select=Name,Priceorder by=Pricefilter color eq red

# GraphQL

```
POST /graphql HTTP/1.1
host: localhost

{
  "Products": { name, price,
  Customers }
}
```



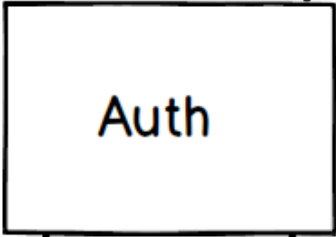
GET /api/products/1 HTTP/1.1  
host: localhost  
Accept: application/xml



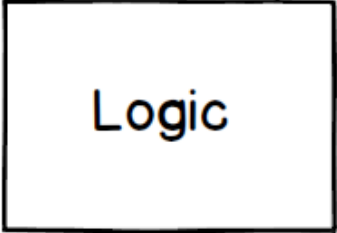
Use()

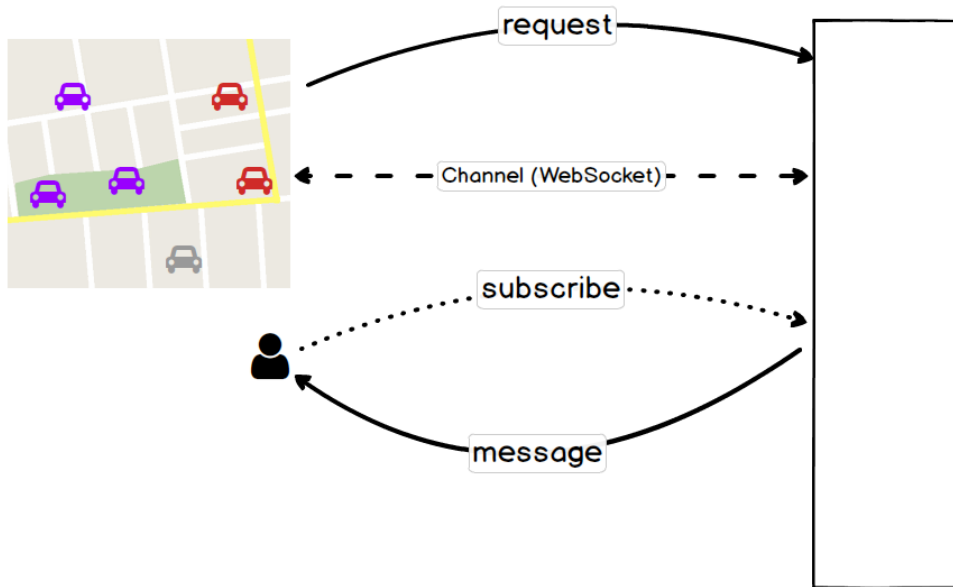
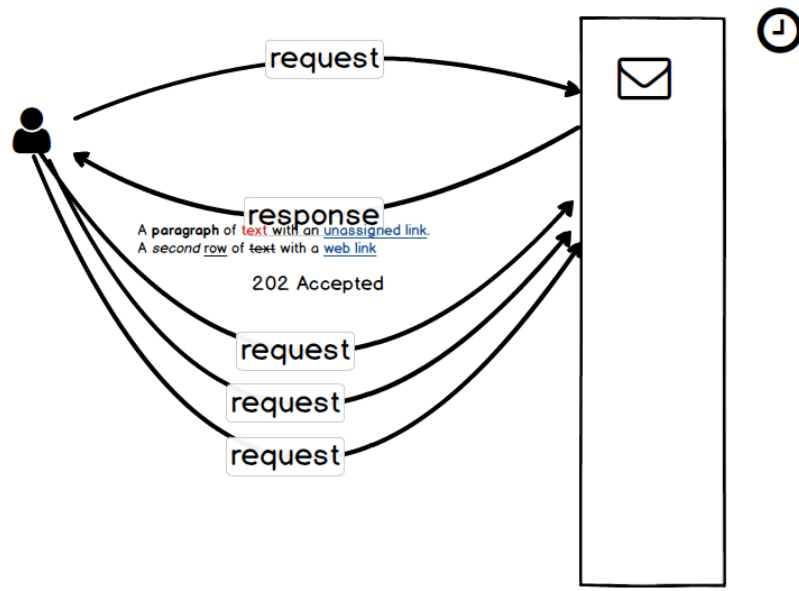


Use()

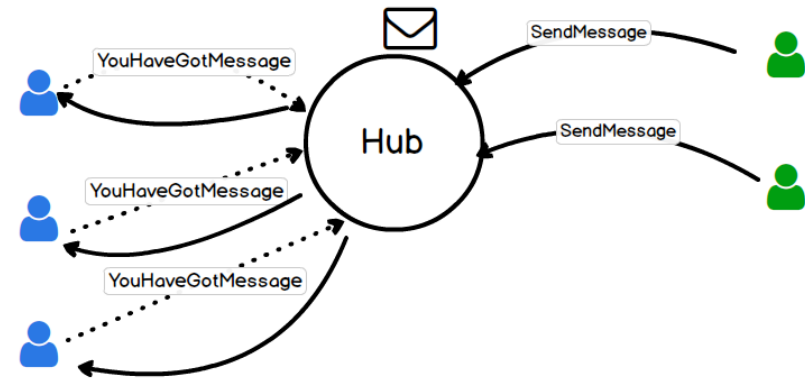


next

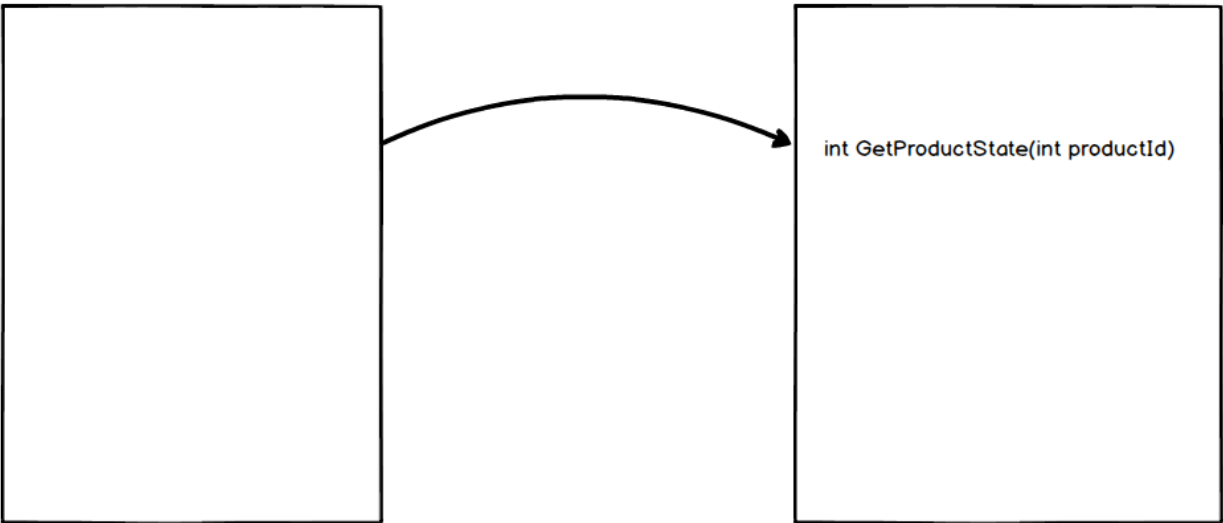




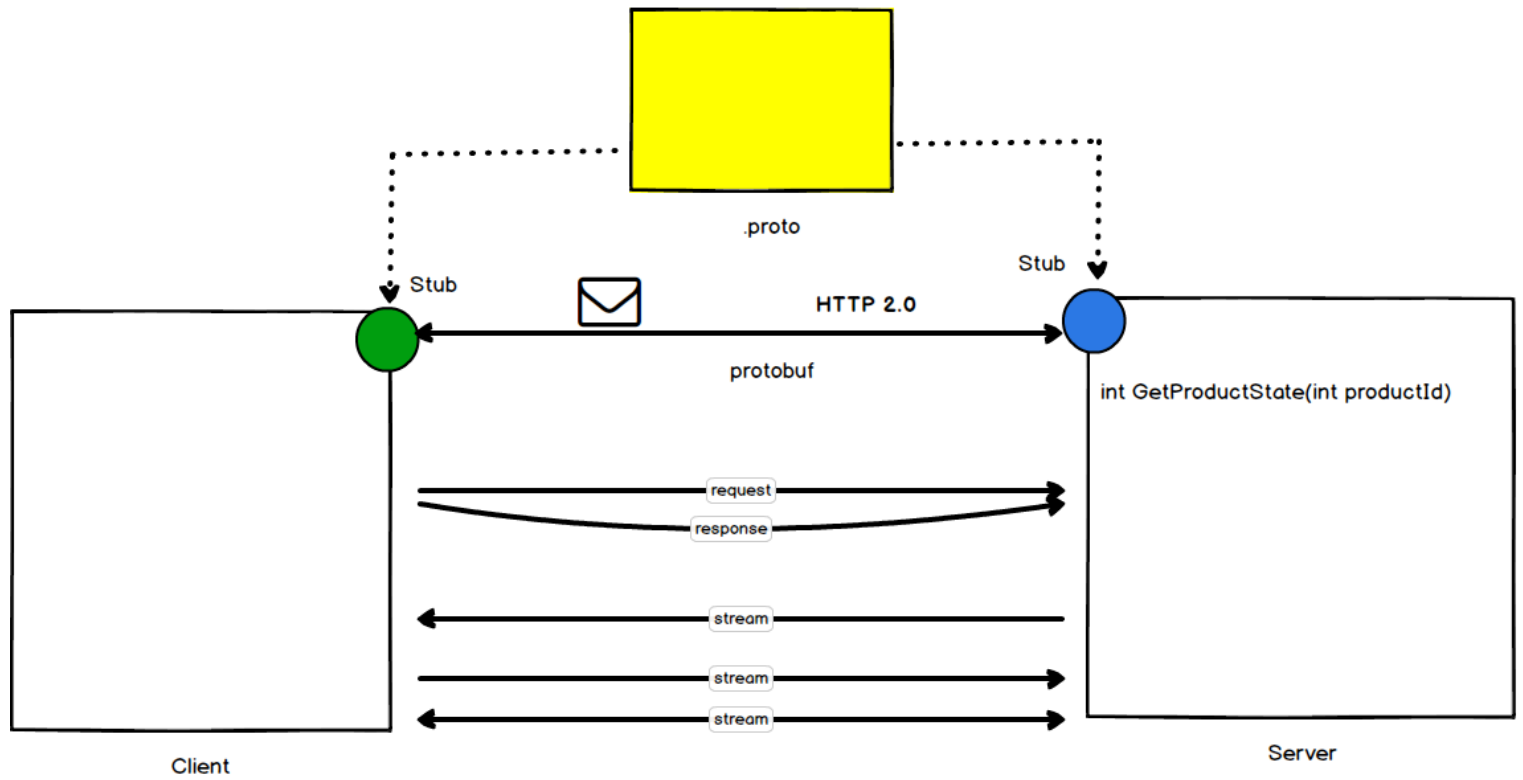
## Signal-R

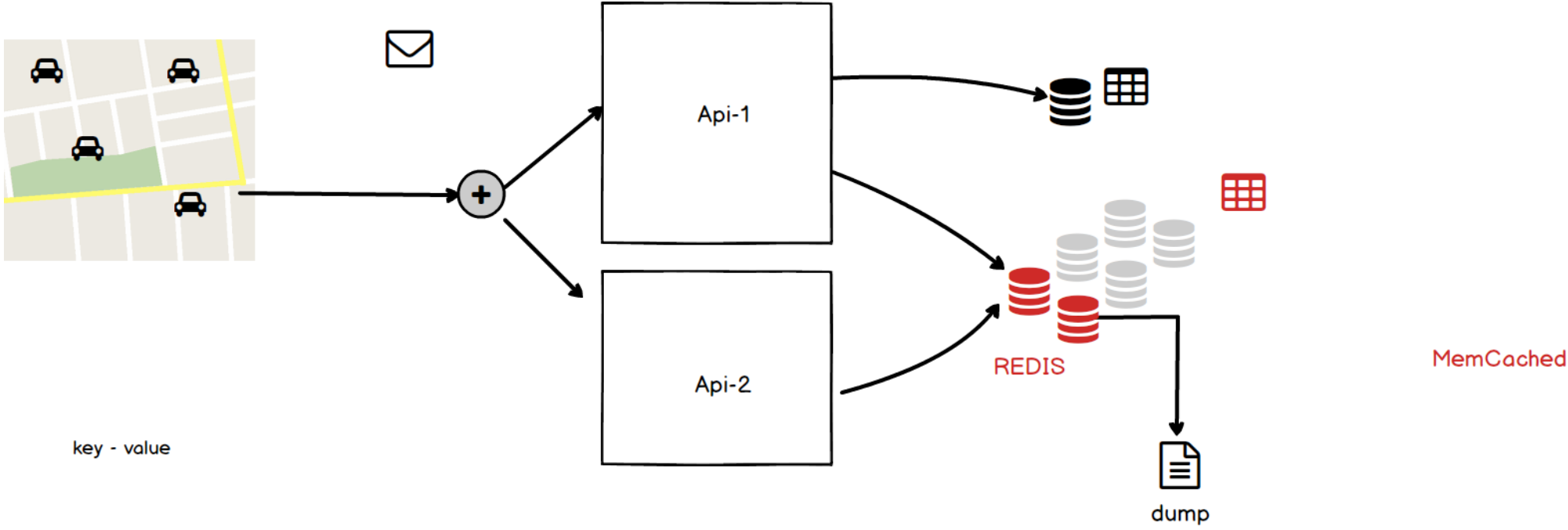


# RPC

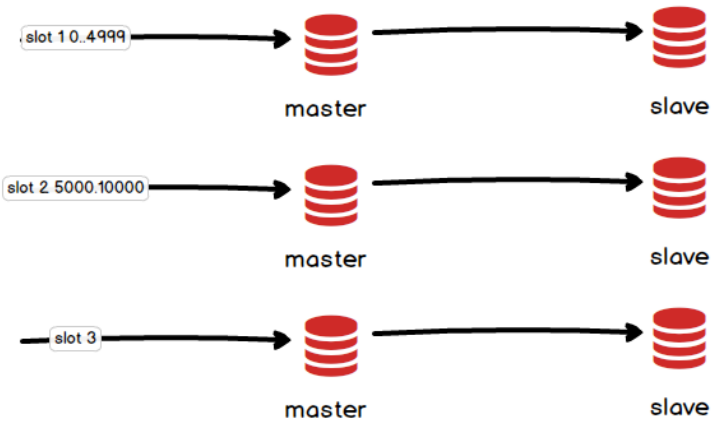


# gRPC

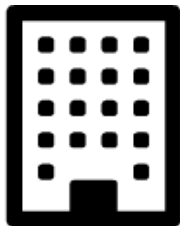




key - value







### *IPrincipal*

`IIdentity Identity { get; }`

`bool IsInRole(string role)`



### *IIdentity*

`AuthenticationType { get; }`

`bool IsAuthenticated { get; }`

`Name { get; }`



Name: John Smith

Typ	Value
-----	-------

Category: A

Category: B

Category: C

Due date: bezterminowe

Role: developer

Role: trainer

Claim



Login

Password

Handler



Token



ACCCCC

exp:

BCCCC  
XCCCC  
YCCCC  
ZCCCC  
QCCCC

POST /service.svc HTTP/1.1  
host: localhost  
Content-Type: application/json  
Authorization: Basic Base64({login}:{password})

GET /service.svc HTTP/1.1  
host: localhost  
Content-Type: application/json  
Authorization: Basic Base64({login}:{password})

GET /service.svc HTTP/1.1  
host: localhost  
Content-Type: application/json  
Authorization: Bearer {token}  
Cookies: {value}



Authorization: {scheme} {parameter}