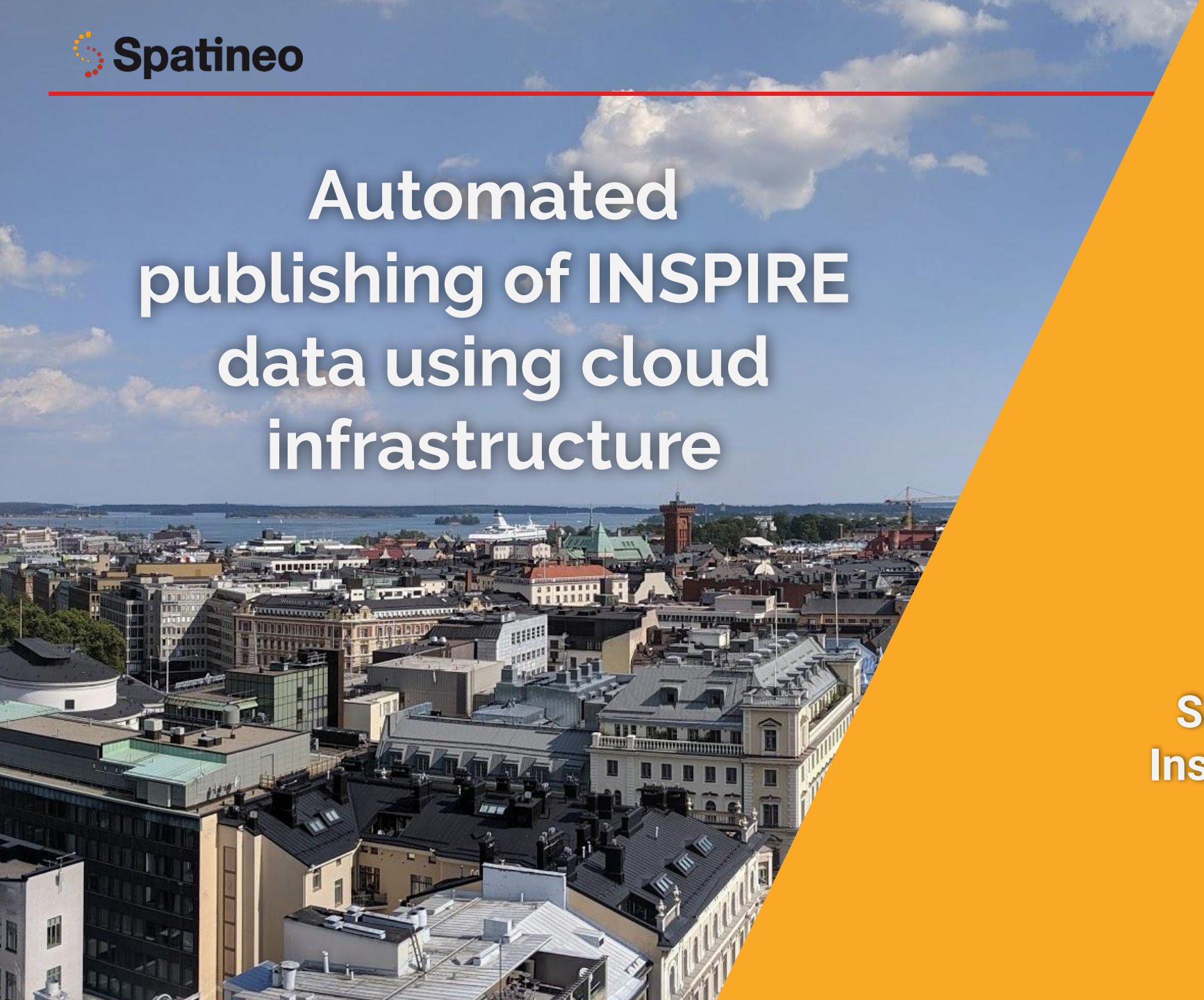
Spatineo

Automated publishing of INSPIRE data using cloud infrastructure



Sampo Savolainen

Inspire Helsinki 2019

www.spatineo.com

Spatineo

Spatineo Monitor & Performance (SaaS)

- Quality assurance and analytics for spatial web services

Services

- Software, GIS & Cloud development
- Impact analysis and data driven decision making
- Machine learning

Founded in 2011

OGC® Member

aws partner network
Standard Technology Partner

Case: Municipal Planning Pilot

Ympäristöministeriö
Miljöministeriet
Ministry of the Environment

Municipal planning in Finland

- Software assisted process
- Resulting plans are “drawings”, not data

Ministry of Environment is renewing legislation

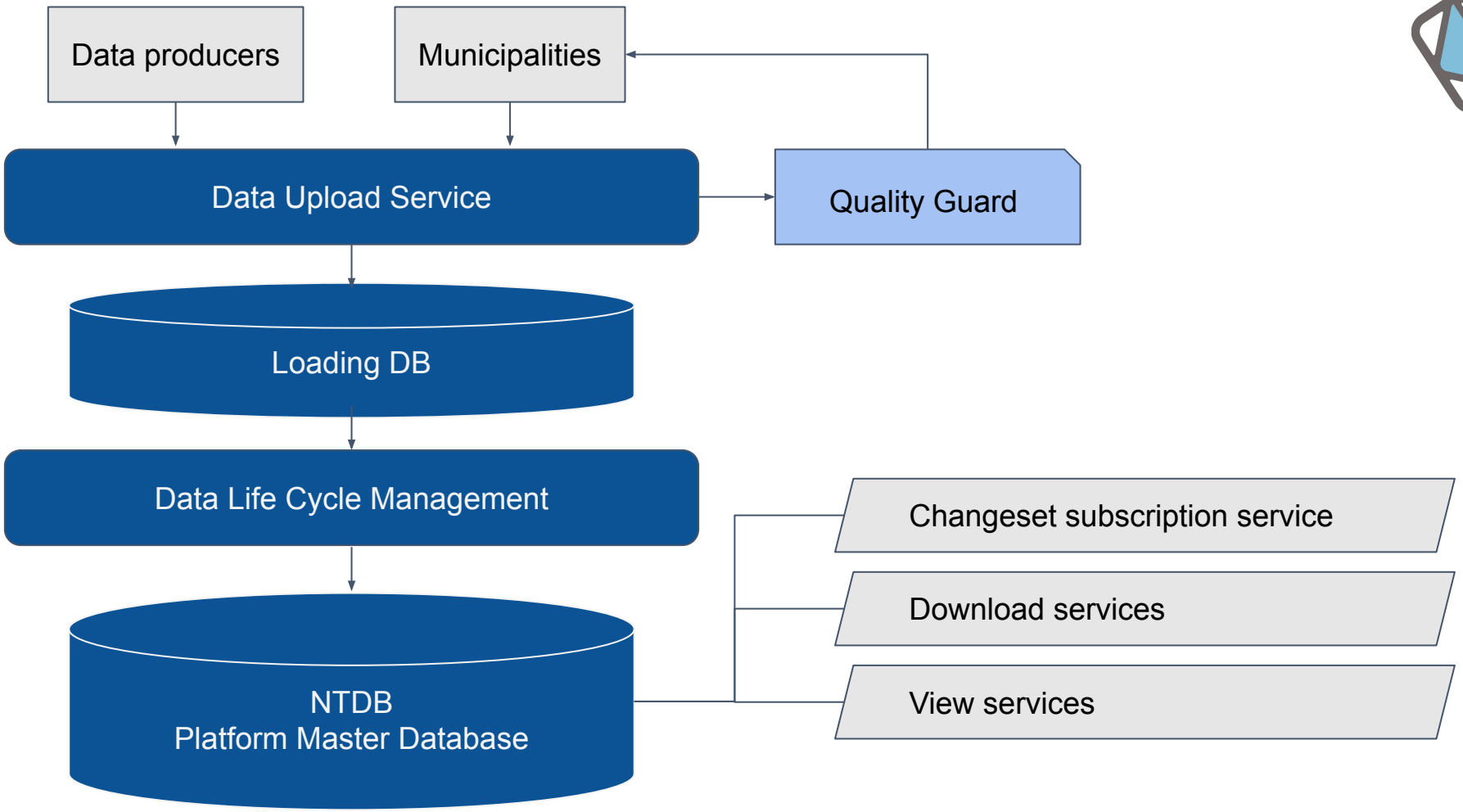
- Common data model for municipal planning
- Same format for interchange and planning software
- INSPIRE PLU compatibility

Part of the Geospatial Platform Project

The purpose of the Geospatial Platform is to:

- harmonise and improve e-services in public administration,
- improve and enhance the transparency of data-based decision-making,
- create cost savings in public administration,
- make the core spatial datasets available on a platform, which the private sector can use as a basis for constructing its own service ecosystem.

Geospatial Platform Project Architecture



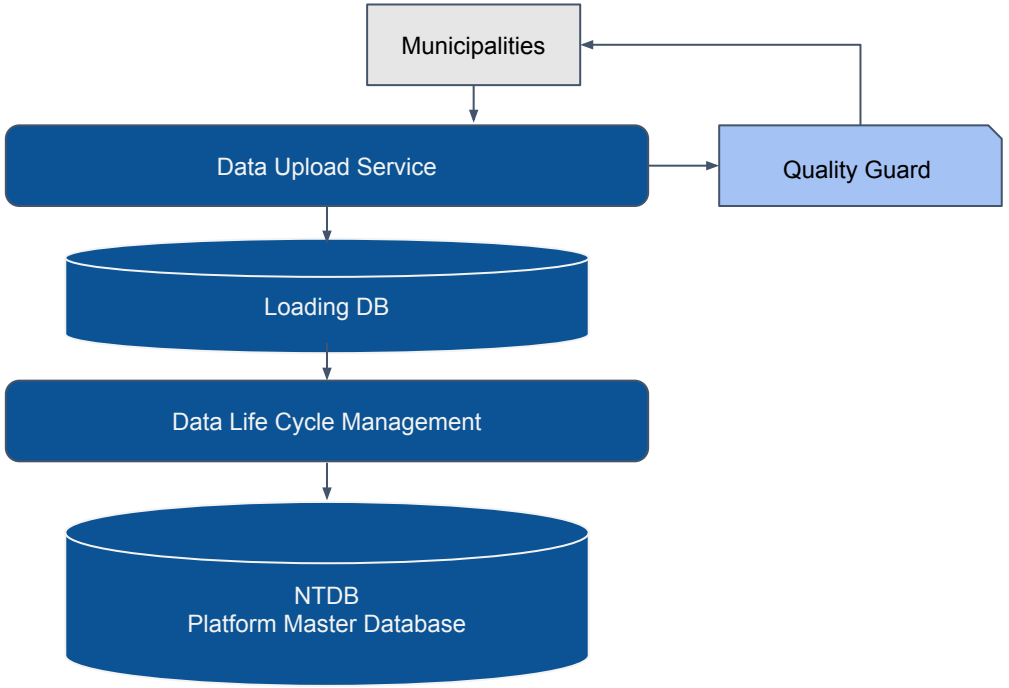
NLS
NATIONAL
LAND SURVEY
OF FINLAND

Mismatched schedule & priorities

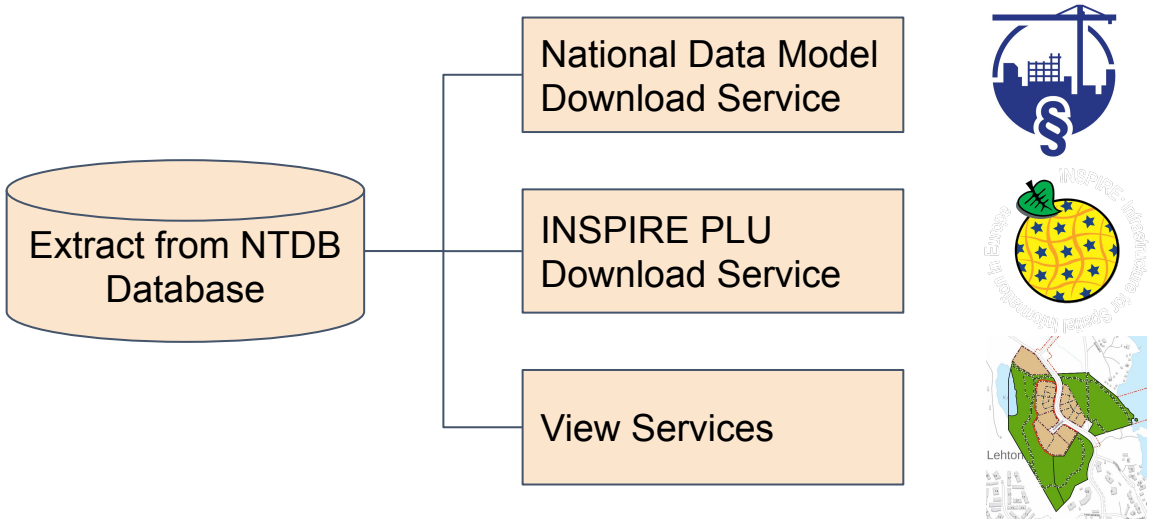
Geospatial Platform Project	Pilot project requirements
Development focused on data input and quality	Missing feature: services for data delivery
Multiple parallel interconnected development efforts, release schedule fixed to pre-defined milestones	Need to evolve quickly as data model was still in flux
Platform goal is to get a generic system that needs to work for all use-cases - takes time to develop	Well defined use-case Services required ASAP
Service focus on simple features & OGC API Features	Complex data model & current generation INSPIRE services

Solution

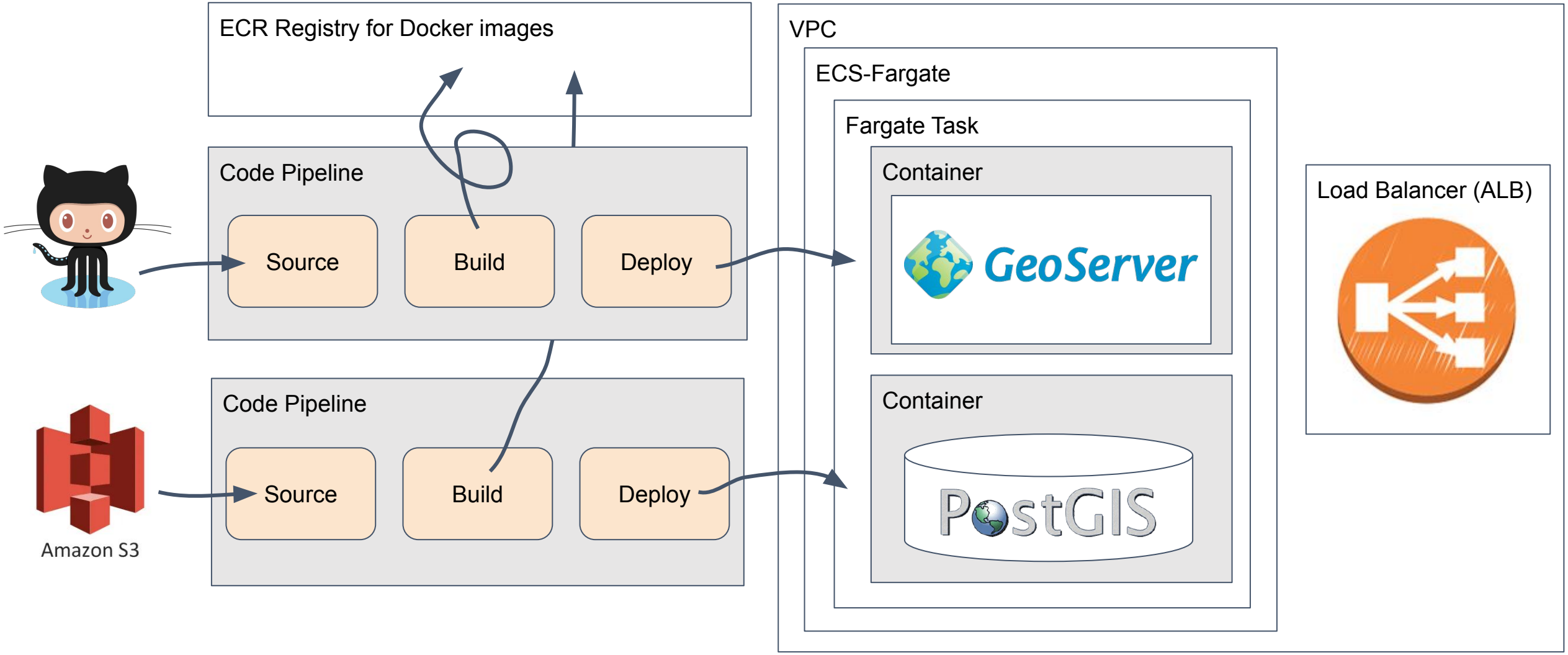
Geospatial Platform = input and data quality



Spatineo & AWS = services and INSPIRE harmonization



Deployment architecture



Deployment using CloudFormation

Define required component and infrastructure stack in YAML

- VPC deployment
- ECS + Fargate deployment
- CodePipeline deployment

CloudFormation is able to update / modify a running stack

S3 triggers built using serverless framework

PostGIS

CodePipeline builds a PostGIS Docker image with the latest data export

Containerized database has exact copy of NTDB database

- Tables
- Data
- Indexes, constraints, triggers, etc.

Build trigger: new data export saved to S3

GeoServer

GitHub project with GeoServer configuration (“data directory”):

- Workspace and datastore configuration
- App-schema configuration
- Layer configuration
- Styles

CodePipeline builds geoserver from source, injects the data directory and pushes Docker image to ECR

Build trigger: commit pushed to GitHub

GeoServer - data formats

GeoServer uses the app-schema extension to produce complex features

Hale Studio was used to map the relational NTDB data model to

- National Data Model 
- INSPIRE PLU 

Some caveats: missing information in NTDB data model can lead to “missing” features in complex feature types

Seamless updates

Deploy phase in CodePipeline creates a new FarGate task

- New task with latest images (GeoServer and PostGIS)
- Health checks test when new version is up
- Switches load balancer to new version (removes old version)

The health check ensures that if GeoServer or PostGIS becomes unresponsive, the FarGate task is replaced with little downtime.

Update time < 5 min for data updates, 20 min for GeoServer updates. Update speed was not a concern and can be improved.

Conclusions

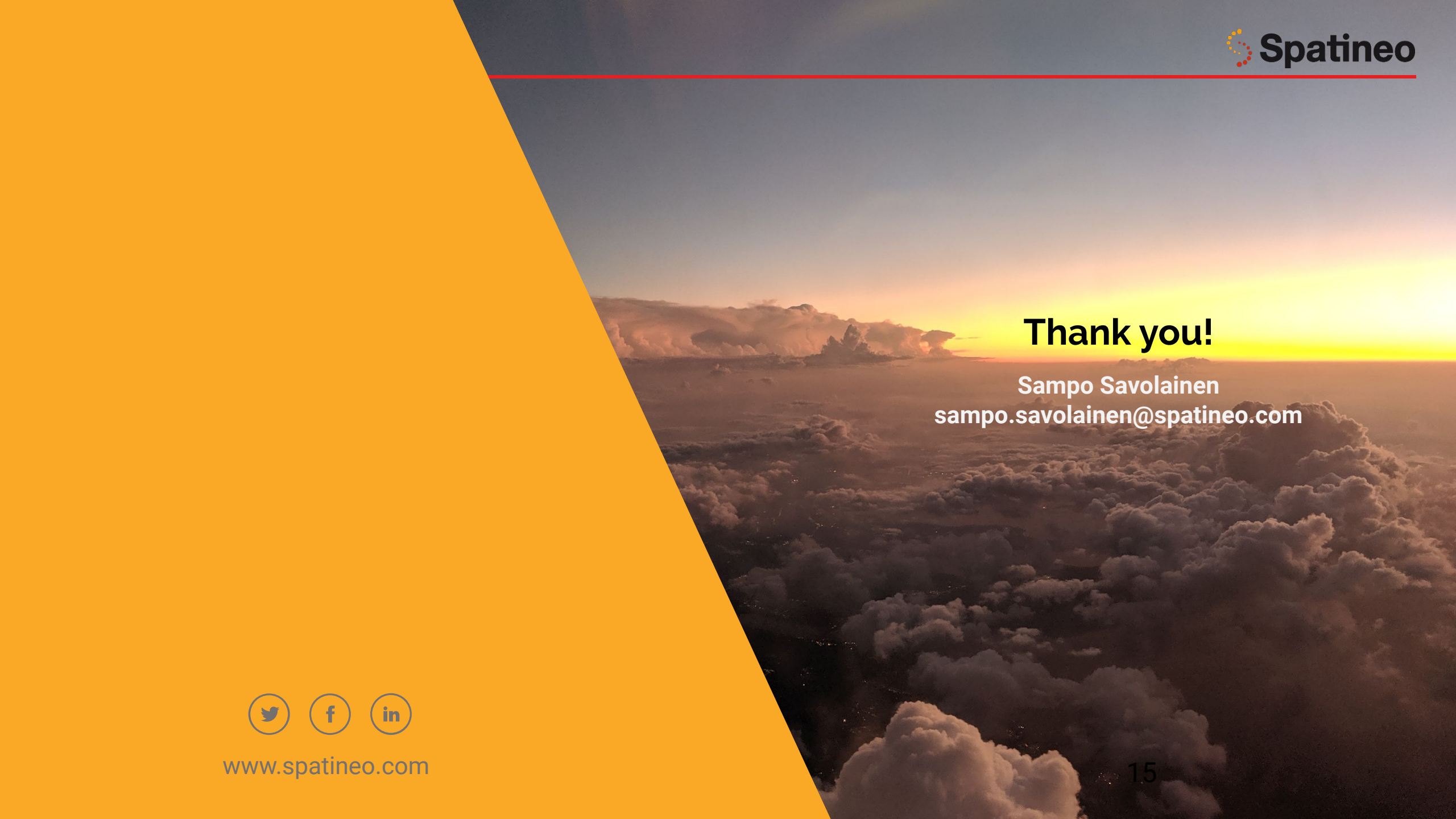
Cloud automation tools have become very capable:

- Automated deployment of software & open data in the cloud
- Configuration of complex infrastructure has become easy

Auto scaling resources easy to implement, requires testing

- Optimize price & ramp up capacity for peak loads

Instead of wondering if you can use the cloud for open data, one should question if there is any reason to use on-premise.



Thank you!

Sampo Savolainen
sampo.savolainen@spatineo.com

www.spatineo.com