Statoil

Data modelling challenges in industrial domains

Trond Solberg

Lead Solution Architect, PLANT Integration

Backdrop

- GODI = Global Operation Data Integration
- The GODI solution originated in the TAIL research programme
- The MapIT project was launched in 2009 to commercialise the GODI solution
- After an extensive RFP process, Statoil chose the Information Integration Core (IIC) solution from IBM
- We are currently executing a pilot project where we implement the GODI solution at four assets

Why GODI?

The GODI vision:

Provide enterprise-wide access to plant and equipment related data, through standardised information models combining data from different sources, to end-user applications

What are we aiming for?

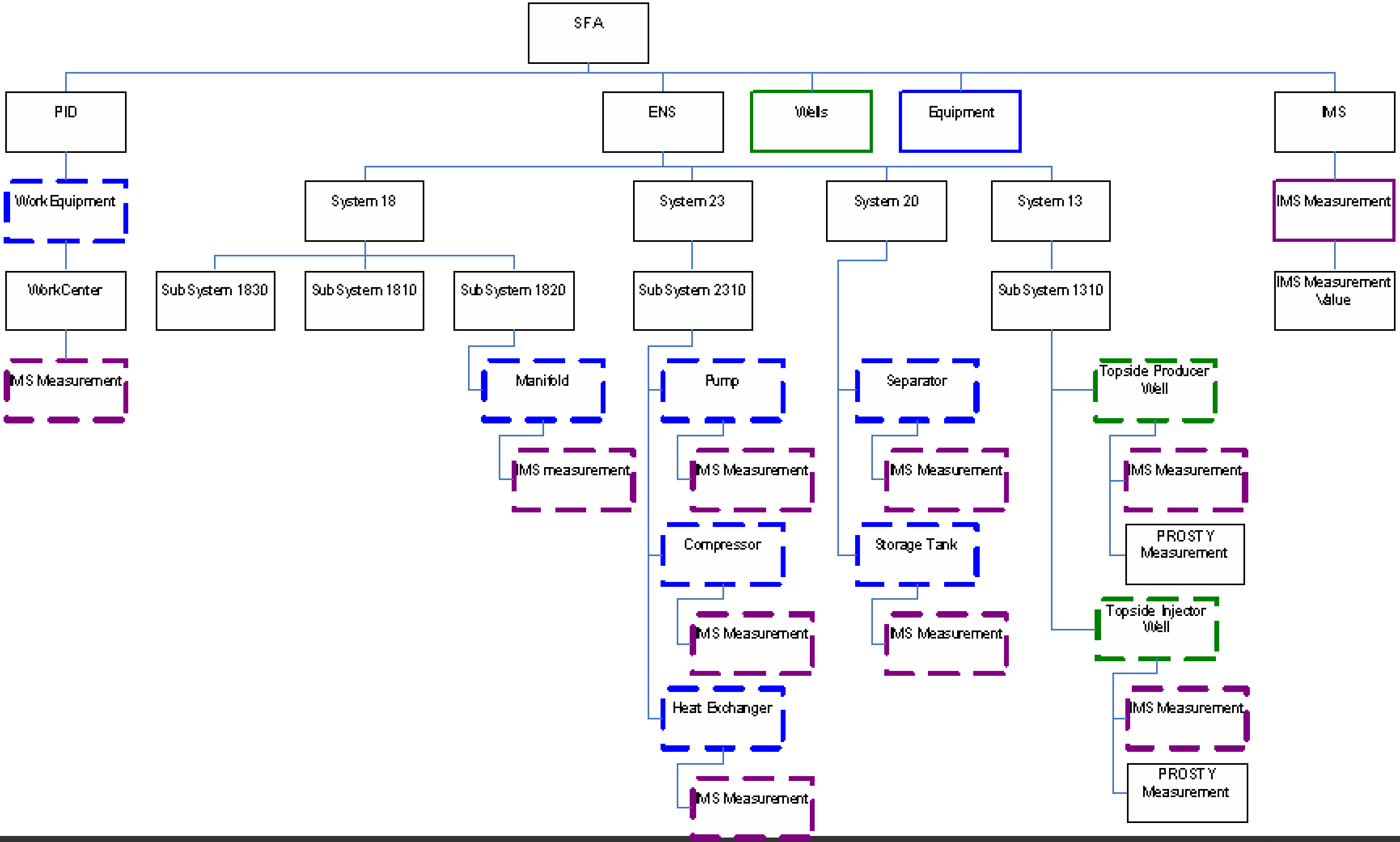
- Introduce an integration layer for plant related data
    - Providing data through a model that sets data in context
        - In our case this means to connect a number of data sources through an equipment-centric model
            - In this context, “equipment” is anything from a single valve to a complete well

- We are not “just” trying to model pieces of equipment and how they relate to plants and to each other
    - The main purpose of the integration layer and its model is to facilitate easy data capture for applications, without requiring detailed data source knowledge

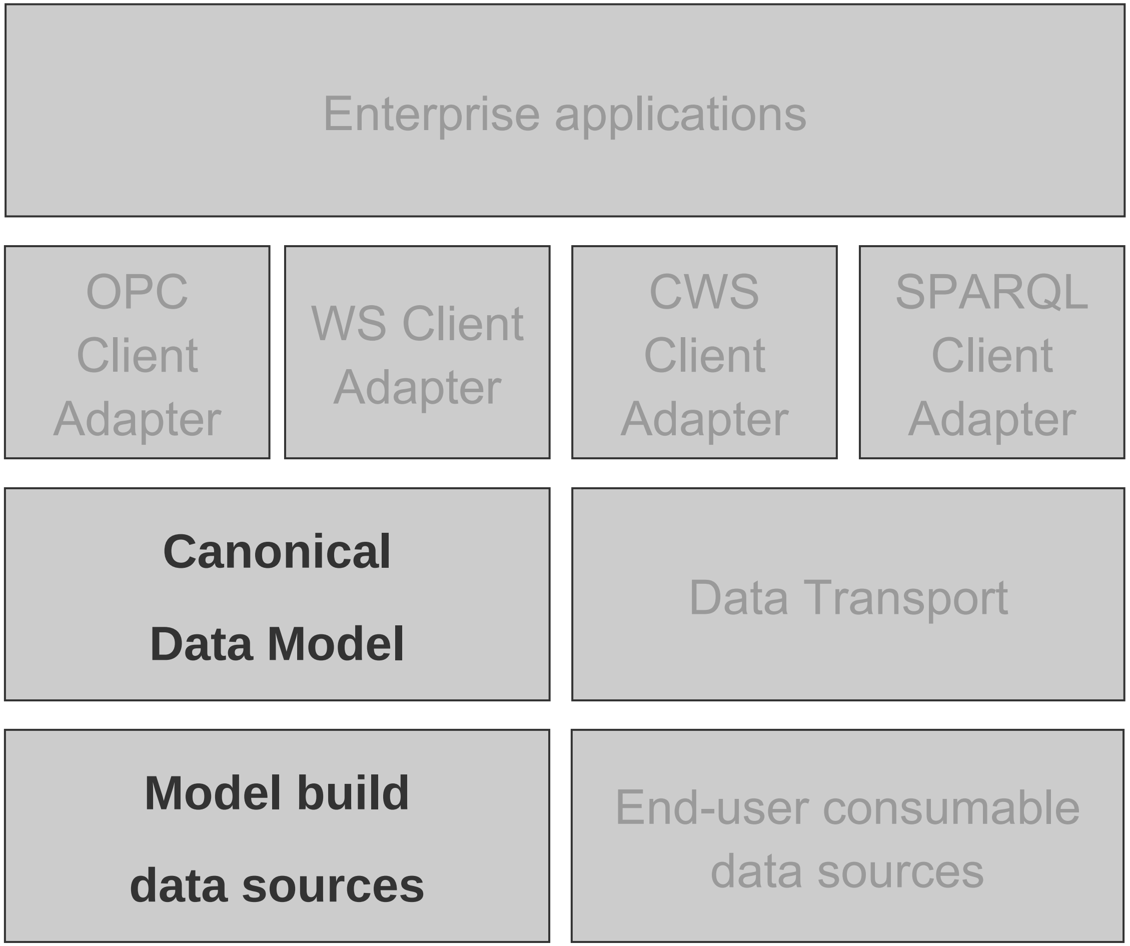
The MapIT method

- Learn to crawl before attempting to fly
  - Only model what you know will add value to a work process
    - Ensure that the model is able to grow into new requirements
  - Aim for a well-defined scope
    - For the pilot phase we are closely connected to a single end-user visualisation project
      - However, we try to make sure that we don’t do anything that prohibits a more extensive usage
  - Find a balance between the need to avoid scope creeping and the need to uncover problem areas (a.k.a. “challenges”)

Example from the current model



The GODI stack - simplified



Enterprise applications

OPC Client Adapter

WS Client Adapter

CWS Client Adapter

SPARQL Client Adapter

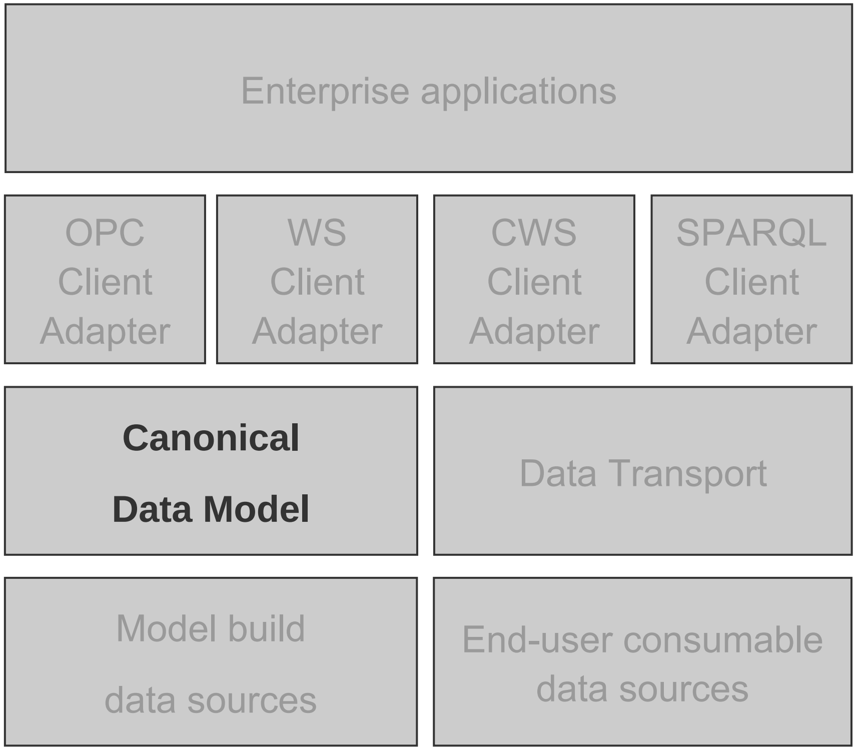
Canonical Data Model

Data Transport

Model build data sources

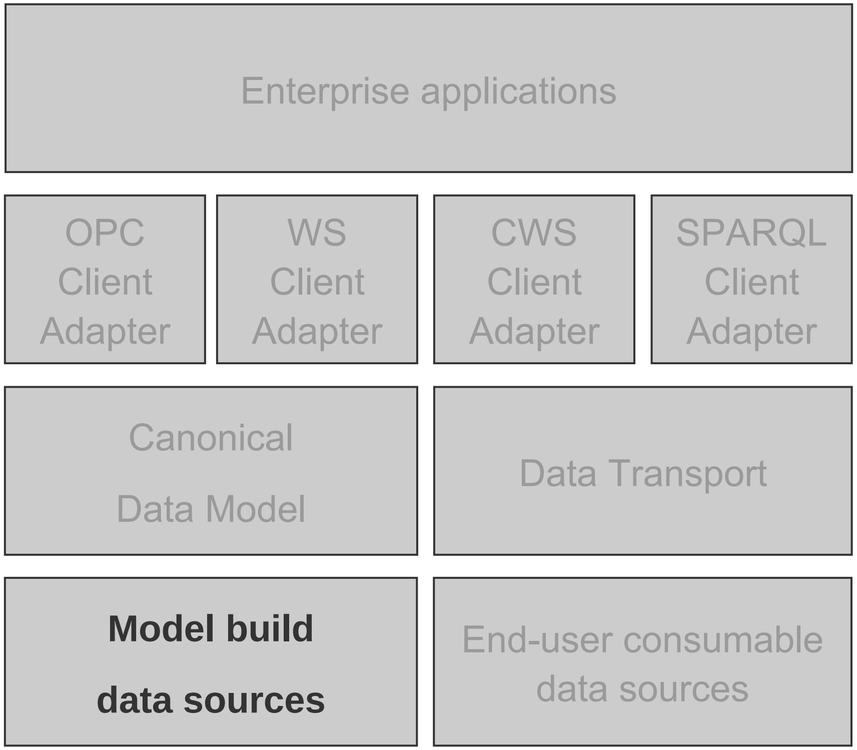
End-user consumable data sources

Model design challenges



- Enterprise applications need to be **detailed** with regards to data requirements
  - “We want all data about everything” is not detailed enough
- No single **standard** covers our requirements
- How can a client **discover** which interface to use?
- How do we **name** things?
- How do we **combine** several physical assets into a single logical asset?
  - E.g. multiple ID’s in SAP
- How do we **organise** the data?

Model instantiation challenges



- How can we **automatically** extract equipment of various kinds and connect them to data series without
  - **Enough information** in our technical information systems?
  - Usable and agreed-upon **naming standards**?
  - Enterprise-wide **standards compliance**?
  - **Coherence** between data sources with regards to definitions and identifiers?

Lesson learned

Get it right!

Lesson #1 – Model the right things

- Don’t bite off more than you can chew, and don’t attempt to model everything at once
  - Decide on the use-cases for which you require data
  - Determine the essential pieces of equipment for the problem at hand
    - Decide what they are and what they should be called
  - Determine the necessary data series/attributes for each piece of equipment
    - Decide what they are and what they should be called
- Model this, and nothing else, but keep in mind that you will have to expand the model later so don’t do anything that will restrict expansion

Lesson #2 – Use the right standards

- Use the right **combination** of industry modelling standards and custom classes
  - Neither ISO 15926, ISA95, MIMOSA nor any of the other available standards are able to provide all of the required functionality
    - Currently we are using only one ISO15926 class and a few ISA95 classes, the rest are custom classes
- The custom classes are used to model equipment and their related data
  - To keep the number of classes down, we decided on weak typing for both equipment and measurements
  - We need to include data source identifiers within the model

Lesson #3 – Keep the right focus

- As with other integration projects this is, and should always be, a business driven project and not technology driven
- Everything, from the data sources through the model to the client endpoints, should be included only when it adds value to do so
- An integration layer is not always the correct answer to a given problem

Lesson #4 – Right scope

- The integration layer, and especially the model, will never be “completed”
  - There is no point in attempting to model everything from the start
  - Do only what is needed for a given task, and do that well!
- Baseline scope for version one needs to be relatively narrow
  - Narrow enough to ensure success for the first production-ready version
  - Broad enough to investigate the most important possible problem areas
- While keeping a narrow scope, keep the bigger picture in mind
  - Ensure, as far as possible, that you don’t do anything that restricts expansion into new usage areas

Data modelling challenges in industrial domains

Questions?

Thank you

Data modelling challenges in industrial domains

Trond Solberg

Lead Solution Architect, PLANT Integration

soltro@statoil.com

www.statoil.com