
rmAgro/drmAgro/drmCrop

Standardisation of electronic data exchange and architecture

Live webinar on Wednesday 15 February

Daan Goense. (daan@pragmaas.com)



Daan Goense

- Retired from Wageningen University & Research (WUR)
- Hired by WUR for the Farm Digital Project of AgroConnect
- Member of the the Ad Hoc Arbeitsgruppe Bus Schitstelle, LBS, (1987- 1993)
- Member of ISO/TC23/SC19/WG1 & WG5 (1993 -)
- Research in Farm Machinery Management, Precision Agriculture and ICT in Agriculture.



What triggered rmAgro?

- Changes in technologies over time. (ADIS, EDIFACT, XML, JSON, API)
 - → Domain reference model should be independent from implementations.
- Recent additional scope's.
 - Precision Agriculture, Tracking and tracing, Guidance, etc.
- → **One common basis that defines the whole Agricultural Production Domain.**
 - First focus on crop production

Why one model for whole Agriculture

- Different branches of agriculture share objects.
 - Organisations, people, etc.
- There is a significant percentage of mixed farms.

- Branches in rmAgro
 - Crop production
 - Greenhouse production,
 - Animal husbandry
 - Aqua culture

rmAgro; a model suite

- Business Process Model (**BPMN**), mainly for FISpace
- **Use case** model, mainly for ISO/TC23/SC19WG5
- Domain Reference Model (**drmAgro**)
- Dynamic view (**sequence diagrams** for FISpace)
- **DDL model** (*transformed from drmAgro*)
- External models (ISO19107, Fertilizer, Crop Protection)
- External XSD's (ISO11783, GML)
- Mapping (drmAgro/drmCrop to other models)
- **Java Model** (interface model & *implementation model* transformed from drmAgro)
- **WSDL** (defines messages for FISpace)
- **XSD model** (transformed from drmAgro)

Modelling conventions for the domain model

- It is a **platform independent** model !
- No id's or keys, except for a Global Unique Identifier (GUID) as an attribute.
- No foreign keys.
- Limited set of generic datatypes (*no language specific datatypes*)
- Many to many relations stay as they are, no association class (*except when it has attributes*)

Some starting points

- Use existing standards when appropriate
 - **ISO191xx** and **GML** for geometry
 - **SensorML** for sensor data

Structure of the domain model (1)

■ drmAgro

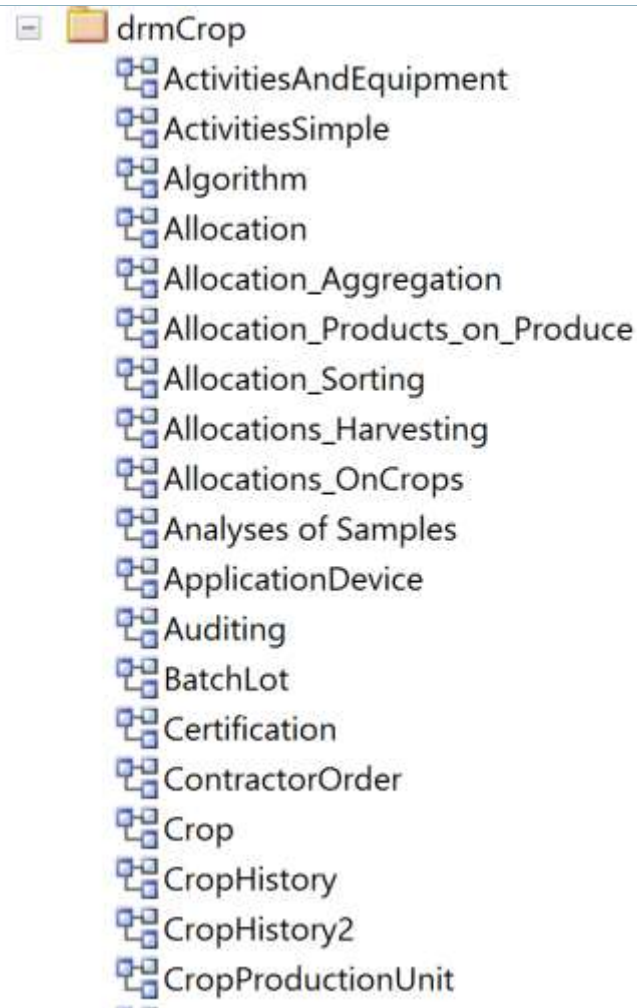
- DataTypes
- Enumerations
- Geometries (→ GML or → ISO19107)
- SWE types (DataArrayType)
- XSD types (token, ncName, anyURI)

Structure of the domain model (2)

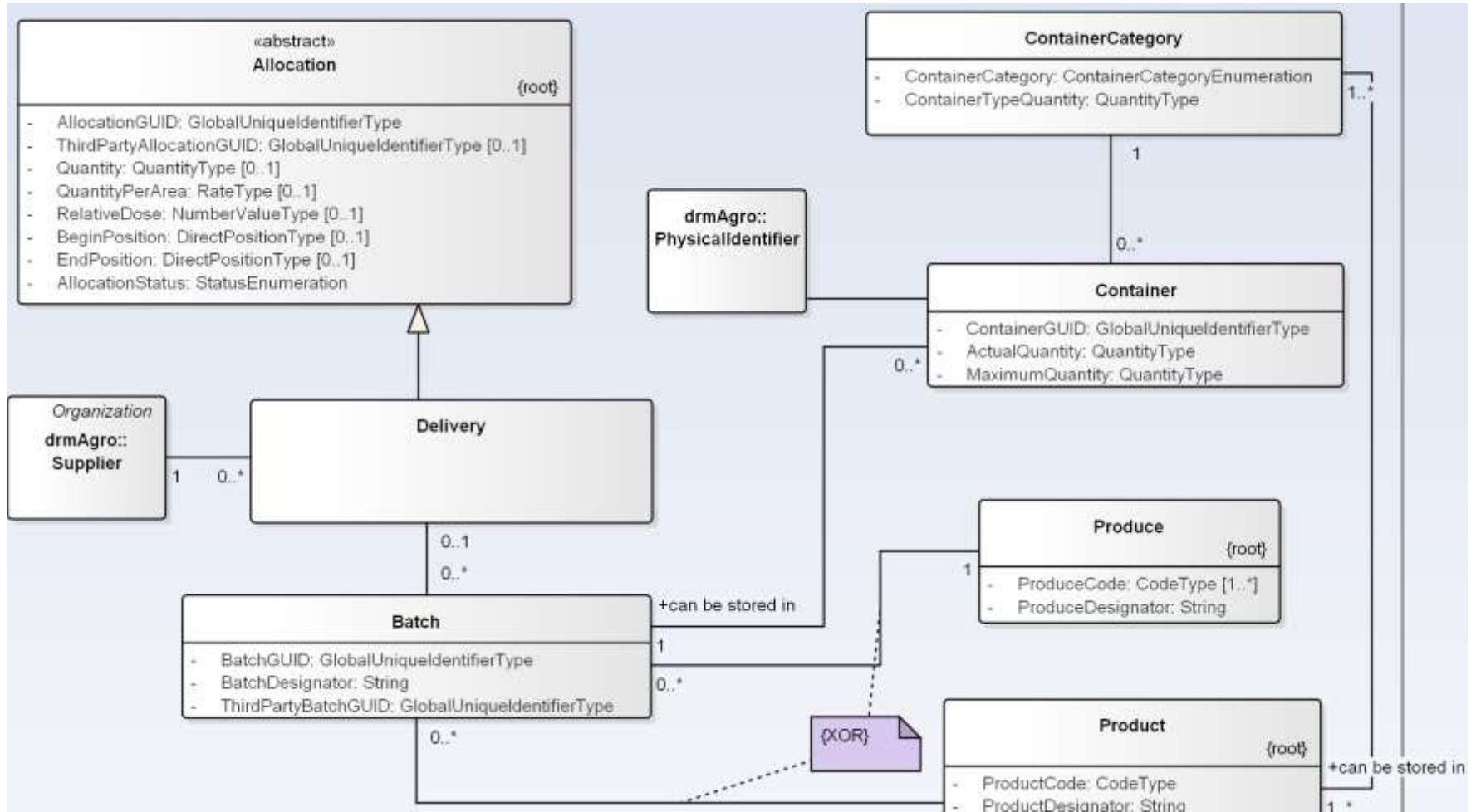
■ drmAgro

- *All common classes* (i.e. Party, Organization, .)
-
-
- drmCrop
- drmAnimal
- drmGreenHouse
- drmInfrastructure (yards, trees, roads, etc)
- drmPostharvest

Diagrams for different scopes of the model



Example for Batch



All classes have definitions, evt. remarks and examples

The screenshot shows a software interface for editing a class named "CropField". On the left, a navigation pane lists "Properties" (General, Templates), "Rules" (Requirements, Constraints, Scenarios), and "Related" (Files, Links). The main area displays the class name "CropField" and a rich text editor with a toolbar. The editor contains the following text:

Definition.
CropField describes the continuous surface of land which is used during a period of time by a certain **CropProductionUnit**.

Remark.
The surface of the **CropField** is always within the borders of a Field.

As a **CropProductionUnit** applies to only one **CropType**, a **CropField** is grown by one **CropType**.

It is the farmer who decides when to split up to different **CropFields**. That can be a different **Variety**, a different purpose for which the crop is grown, a different class in seed production, the need to keep it separate for tracking and tracing purposes etc.

On the right side, there is a properties panel with the following fields:

- Stereotype:
- Status:
- Alias:
- Keywords:
- Author:
- Complexity:
- Language:
- Version:
- Phase:
- Package:

Scopes covered by drmAgro

- Parties

- Party, Organization, Person, Department, Farm, etc.

- Fields

- Plot, Field, CropField, GreenhouseFloor, ActivityField, KadastralField

- Activities on the farm

- Job, Task, Operation

- Data processing

- DataSet, DataAggregation, Algorithm, DataProcess

Scopes covered by drmAgro (2)

- Handling of products and produce
 - ProductAllocation, Product, Batch, TreatmentZone
- Sampling and analyses
 - Sample, Analyses, PropertyValue, Laboratory, Container, VerticalLayer
- CropRecording
 - CropProductionUnit, CropField, Operation, AbsoluteTiming, CulturalPractise, OperationTechnique, ProductAllocation, TreatmentZone, Batch, etc.

Scopes covered by drmAgro (3)

- Farm machinery
 - Equipment, Implement, Tractor, ManMachineSystem
- Ordering
 - Order, OrderItem, Delivery, Invoice, Customer, Supplier
- Product composition
 - Product, ProductAllocation, Batch, ProductElement

Relevant Packages

- drmAgro/drmCrop
 - Allocation***
 - Auditing & Certification
 - Crop & CropRecording
 - Operation***
 - ProductApplicationOnCrops
 - PropertyValue
 - Site
 - Zone

- drmAgro/drmGreenhouse
 - Greenhouse

Availability of the model

- rmAgro snapshot:
ftp://pragmaas.com/rmCrop/rmAgro_SNAPSHOT
- Enterprise Architect model: **rmAgro.eap**
- Description of background: **rmAgroGuideline.docx**