ELF WP2 Modeling Guidelines

This document gives a short introduction to modelling of application schemas for the ELF specifications. The application schemas shall build upon the INSPIRE specifications and should therefore adhere to the requirements and recommendations in the INSPIRE Generic Conceptual Model (GCM).

Presentation to: DM-Workshop Copenhagen January 29 2015

Author: Morten Borrebaek, Anja Hopfstock, Heidi Vanparys

Date: 2015-01-29

the Competitiveness and Innovation framework Programme (CIP)
ICT Policy Support Programme (PSP) Call 6 (Grant 325140)

the Competitiveness and Innovation framework Programme (CIP)
ICT Policy Support Programme (PSP) Call 6 (Grant 325140)

https://service.projectplace.com/pp/pp.cgi/r934779671

Contents

ELF WP 2 – Modelling guidelines [1]

ELF_Modelling guidelines_v1.15 is available as a public document at:

https://service.projectplace.com/pub/english.cgi/0/865376303

Author: Morten Borrebæk

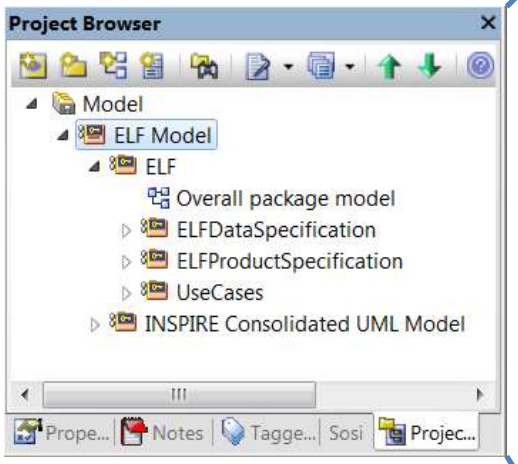
Date: 19.11.2014
Version: 1.14

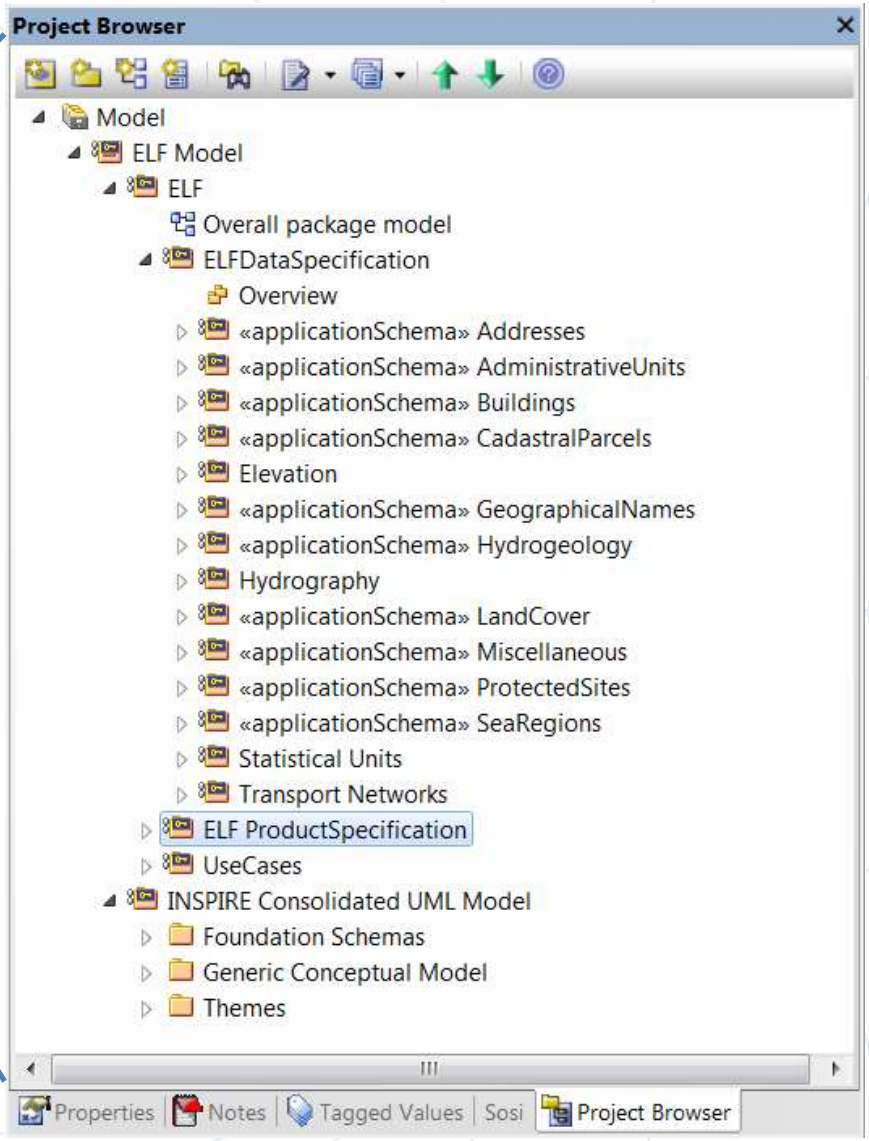
Scope – important policy statement

The revised modelling guidelines are formed in such a way that an existing INSPIRE implementation by default is conformant to an ELF specifications for the themes that are in the remit of the ELF data specifications. The impact of such a precondition is that all ELF additions have to be optional (not even voidable), and that there should be no constraints on the INSPIRE that affects the INSPIRE GML application schema.

Clause 2 - ELF modelling approach

Model structure / UML repository





ELF UML modelling Principles

To achieve INSPIRE compliancy in data modelling the following principles should be observed:

Extensions shall not

- Change the specification but normatively reference it with all its requirements
- Set any additional requirements that break any requirement of the INSPIRE data specification
- Add concepts that overlap with existing INSPIRE concepts
- Make a pure INSPIRE implementation non-conformant to the ELF specifications

Extensions may

- Add new application schemas importing INSPIRE or other schemas as needed
- Add new types and constraints in the new application schemas
- extend INSPIRE code lists if not centrally managed

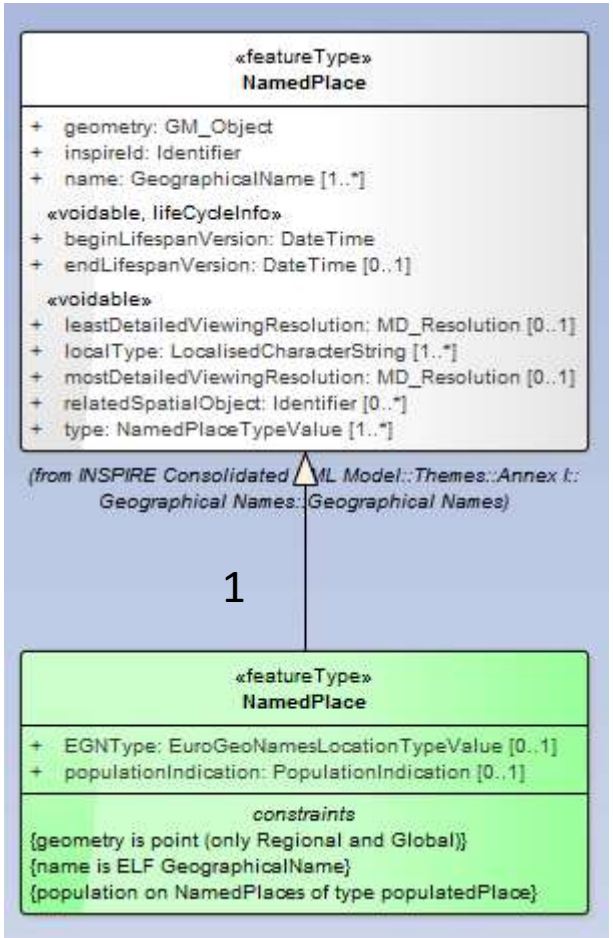
ELF UML modelling Principles (cont)

For each matching concept of INSPIRE and ELF identified from the analysis of the matching tables, a corresponding concept is created in ELF. Where possible, these concepts should be sub-classes of existing INSPIRE concepts (feature or data type, code lists etc.) by:

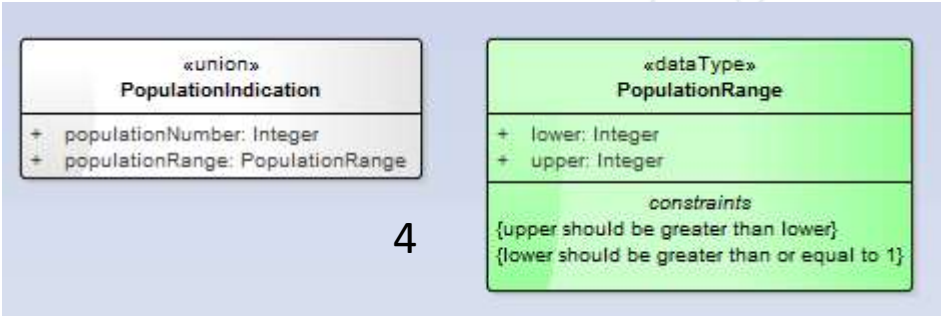
- Define additional optional attributes that are present in existing data or required by users but missing in INSPIRE
- Add constraints to ensure that ELF meets the user requirements where applicable
- Define new ELF feature types for concepts that are present in existing data or required by users but missing in INSPIRE
- For code lists in INSPIRE, identify matching codes and define additional codes where missing – reuse as many values from INSPIRE as possible and define new values only if no existing value can be matched. Describe any additional constraints (e.g. sometimes a code list value cannot be mapped or a code list value depends based on the value of another property).
- Add optional associations where required
- Avoid the stereotype <voidable> for new attributes and associations, to ensure that a 'pure' INSPIRE implementation conforms to ELF.

Clause 2 – UML modelling principles

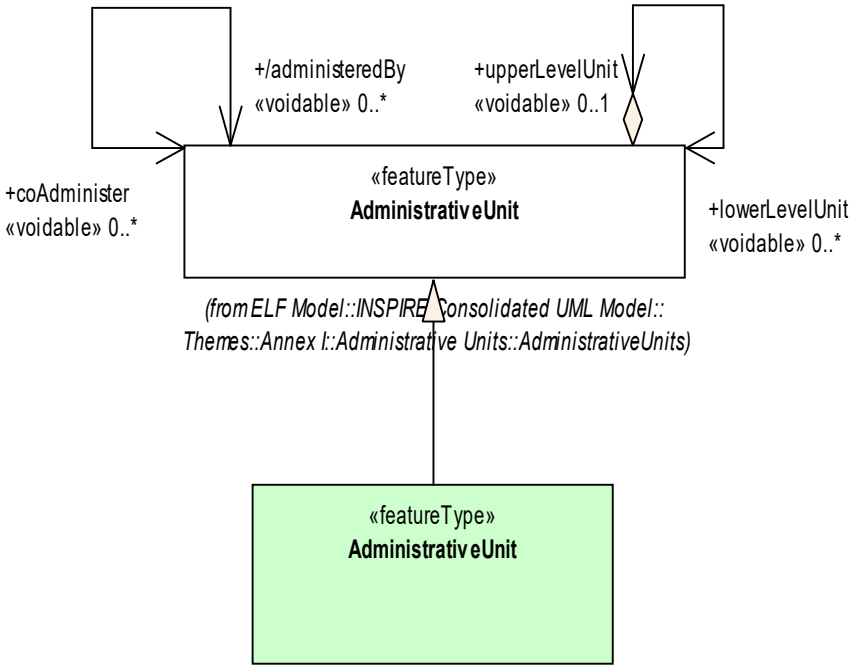
Example: NamedPlace



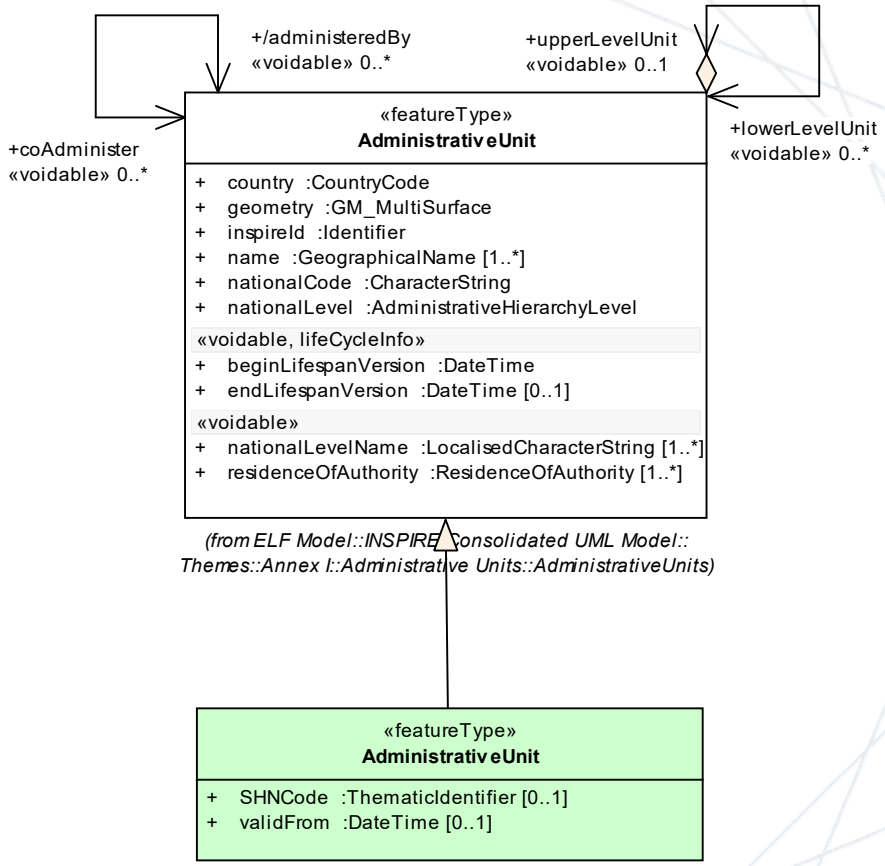
1. Subtype all INSPIRE feature types valid for ELF (topographic reference data)
2. Define additional attributes
3. Add constraints
4. Define new ELF classes (feature types, datatypes, codelists)
5. Associate feature types (not in figure)



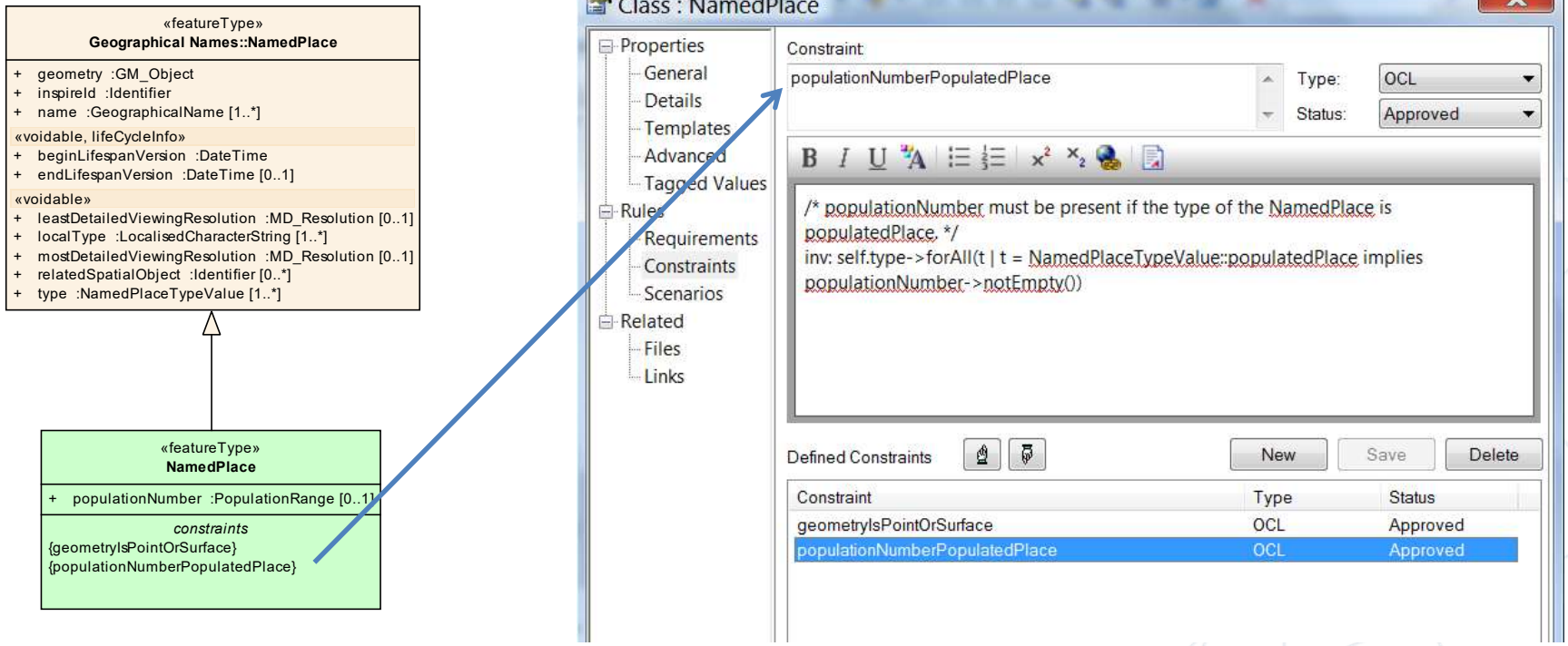
Step 1 - Subtyping INSPIRE feature types



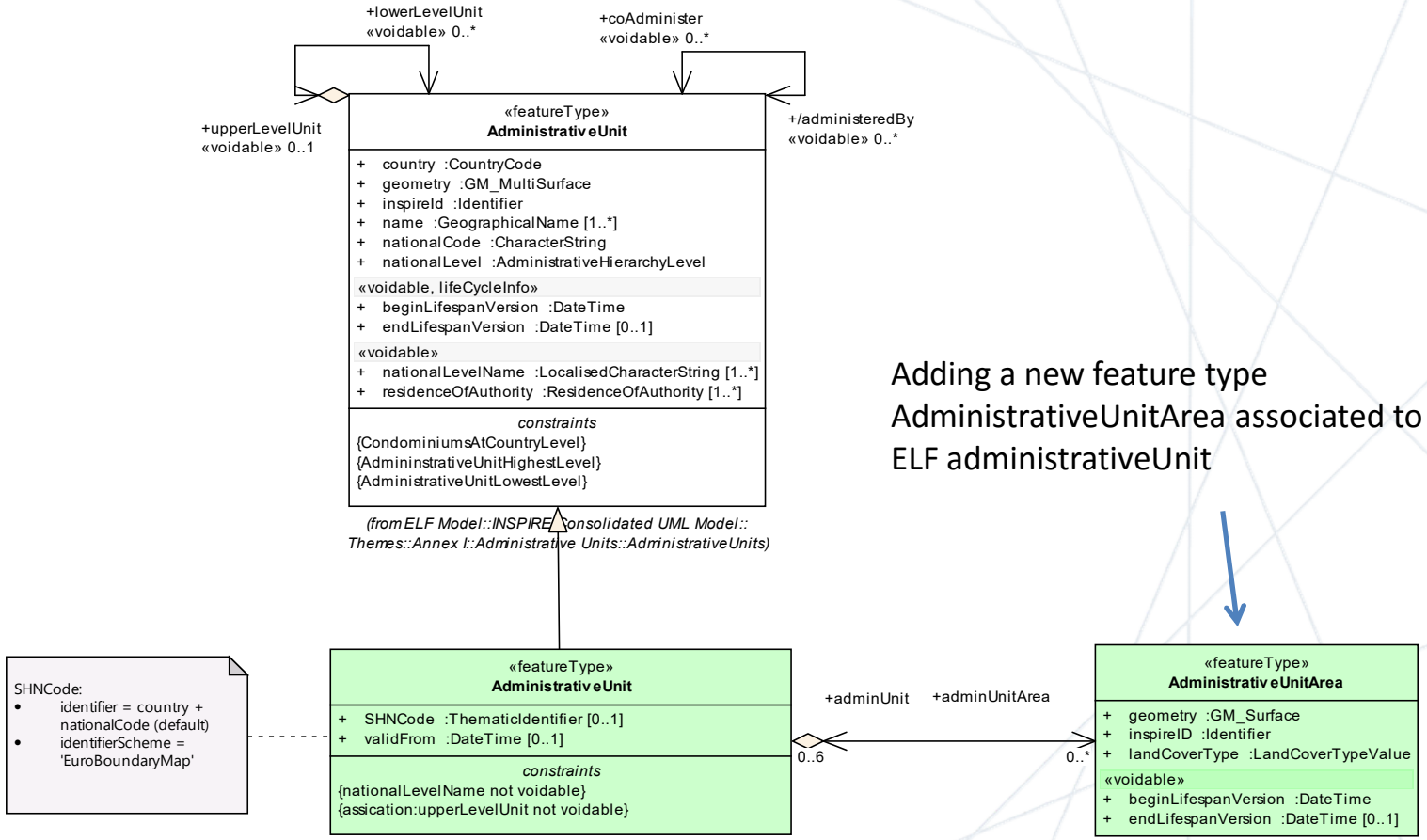
Step 2 - Adding new attributes to ELF feature types



Step 3 - Adding constraints

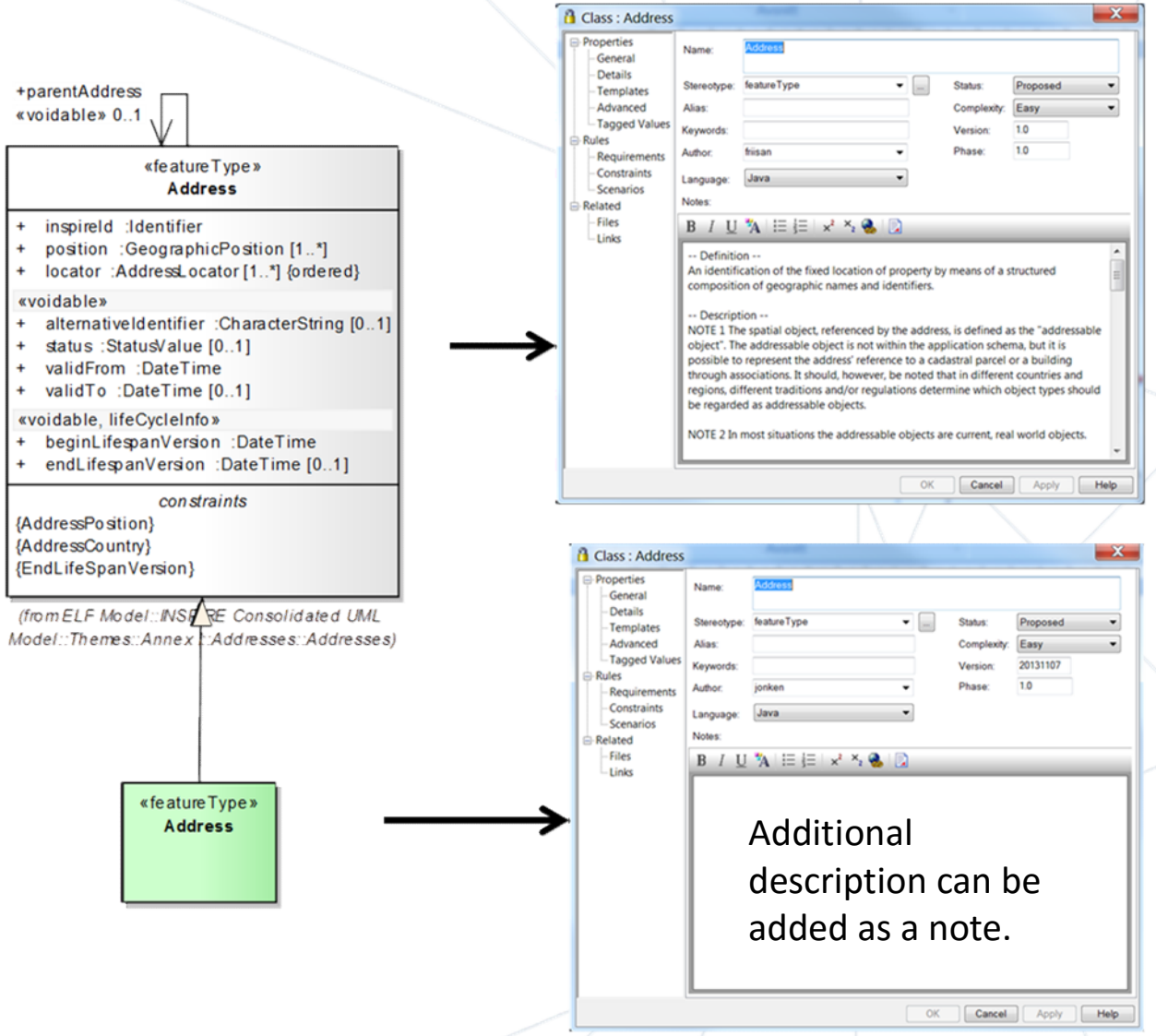


Step 4 and 5 - Define new ELF feature type and associate it to another ELF feature type

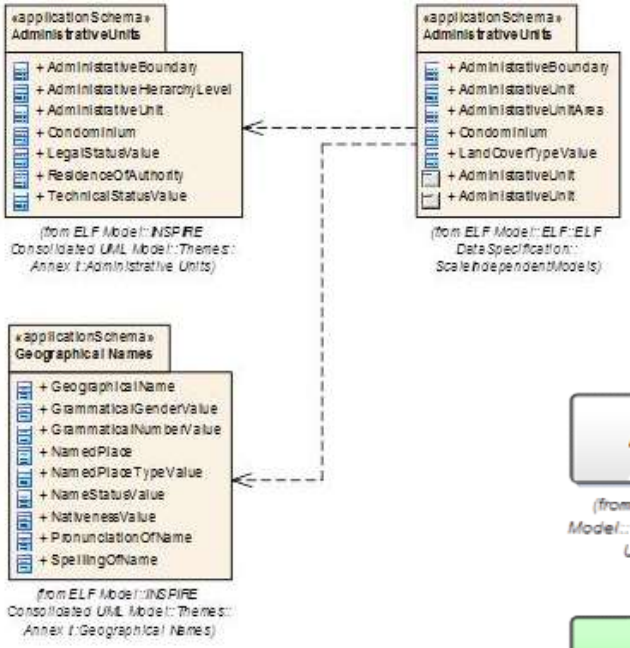


Clause 3 – Documentation

All additional ELF classifiers shall contain definitions sufficient for understanding of all classes, attributes, associations, operations and appropriate data type definitions.

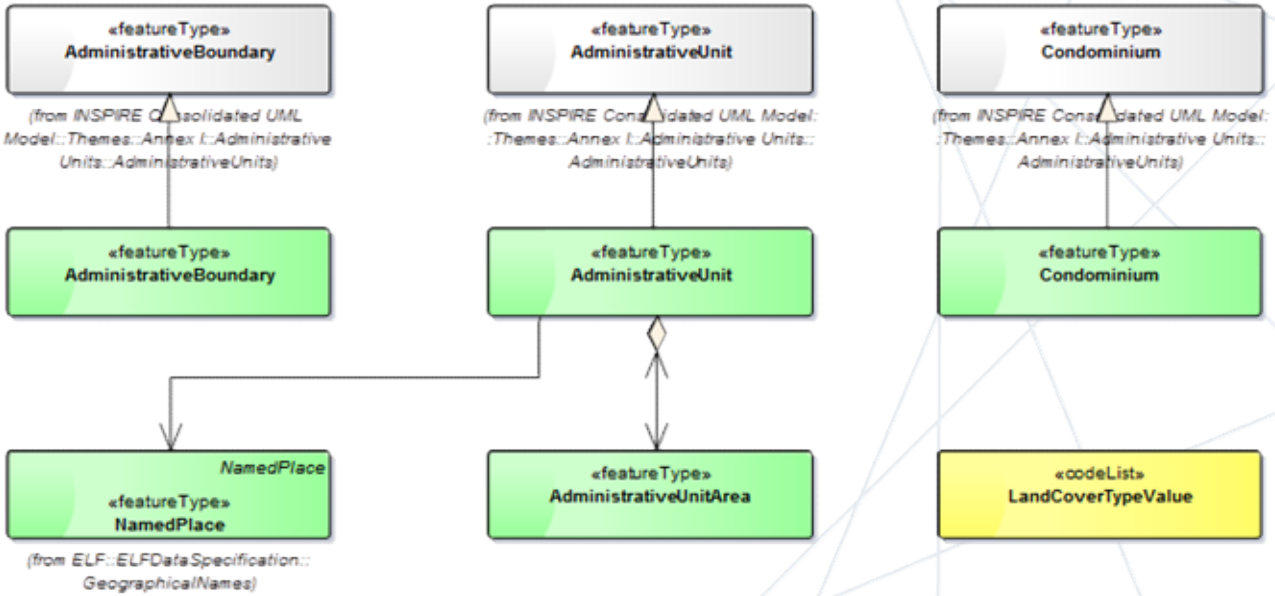


Clause 3 – Documentation - diagrams



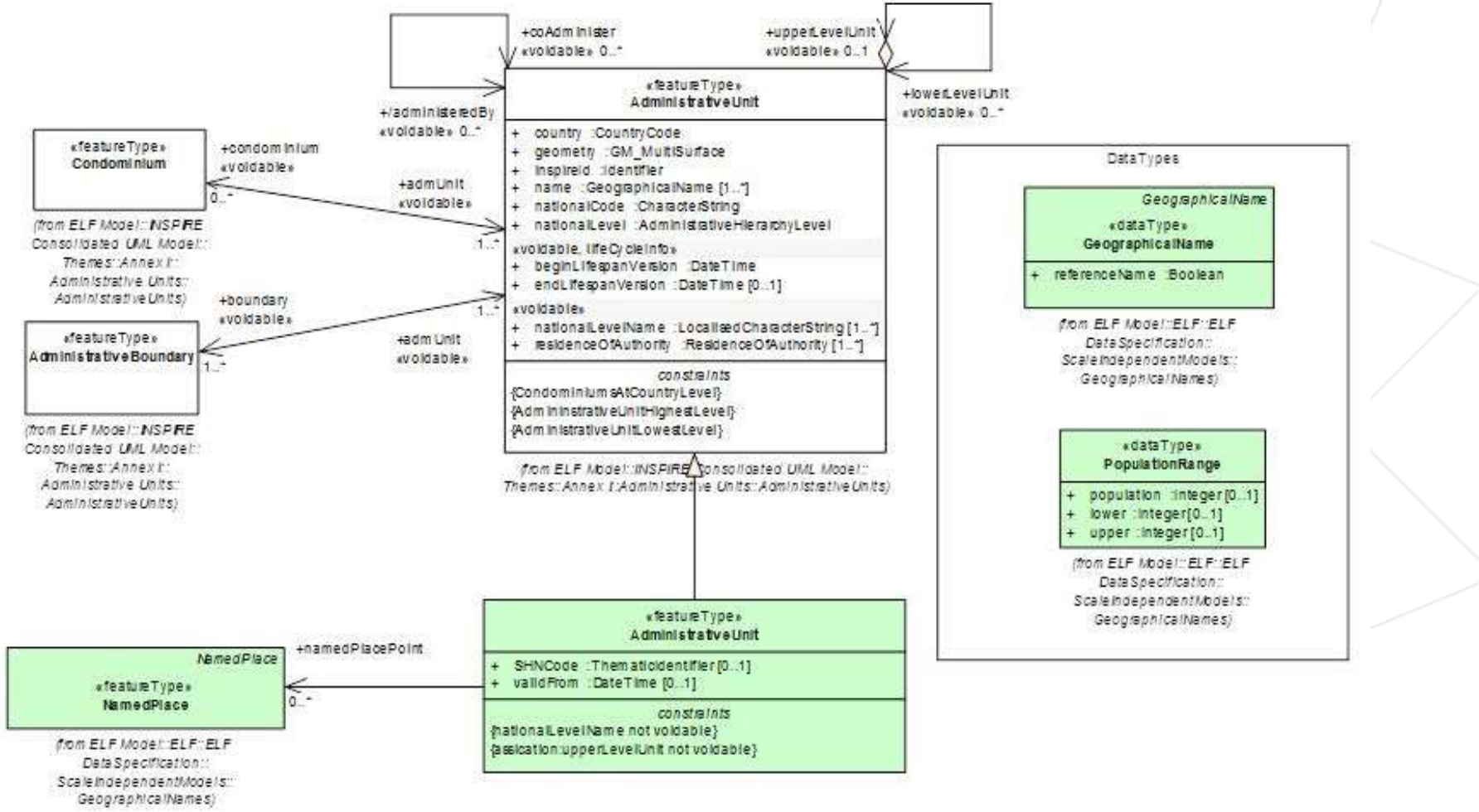
Package dependency diagrams

Overview diagrams



Clause 3 – Documentation - diagrams

Context diagrams



ELF UML profile (technicalities)

The modelling guidelines specifies a profile of UML to be used in ELF, based upon similar UML profile for INSPIRE.

The ELF UML profile adds some additional tagged values to the INSPIRE UML profile. Different name spaces are defined, in addition to the xsdEncodingRule that is specific to ELF.

Stereotype	Tagged Value	Description	Remark	Value and examples
applicationSchema	targetNamespace	Target XML namespace of the application schema [ISO 19136]	Mandatory	http://www.locationframework.eu/schemas/<theme>/<version> Example: http://www.locationframework.eu/schemas/LandCover/1.0 This structure allows one hema for mas have olute and path which ındancy.
	suppress	When true, identifies the feature type as a type that is created only in the ELF UML application schema to indicate that (1) the INSPIRE feature type is included in the ELF data specification and/or (2) to attach a constraint. When true, this feature type will be suppressed in the GML application schema.		values="true | false" default="false"
	profiles	Comma-separated list of profile indicators which associates this model element to one or more levels of detail.		MasterLoD0 MasterLoD1 MasterLoD2 Regional Global If empty, it applies to all LoD's
dataType	xsdEncodingRule	XML Schema encoding rule to apply	Mandatory	iso19136_2007_ELF_Extensions

add
.5]

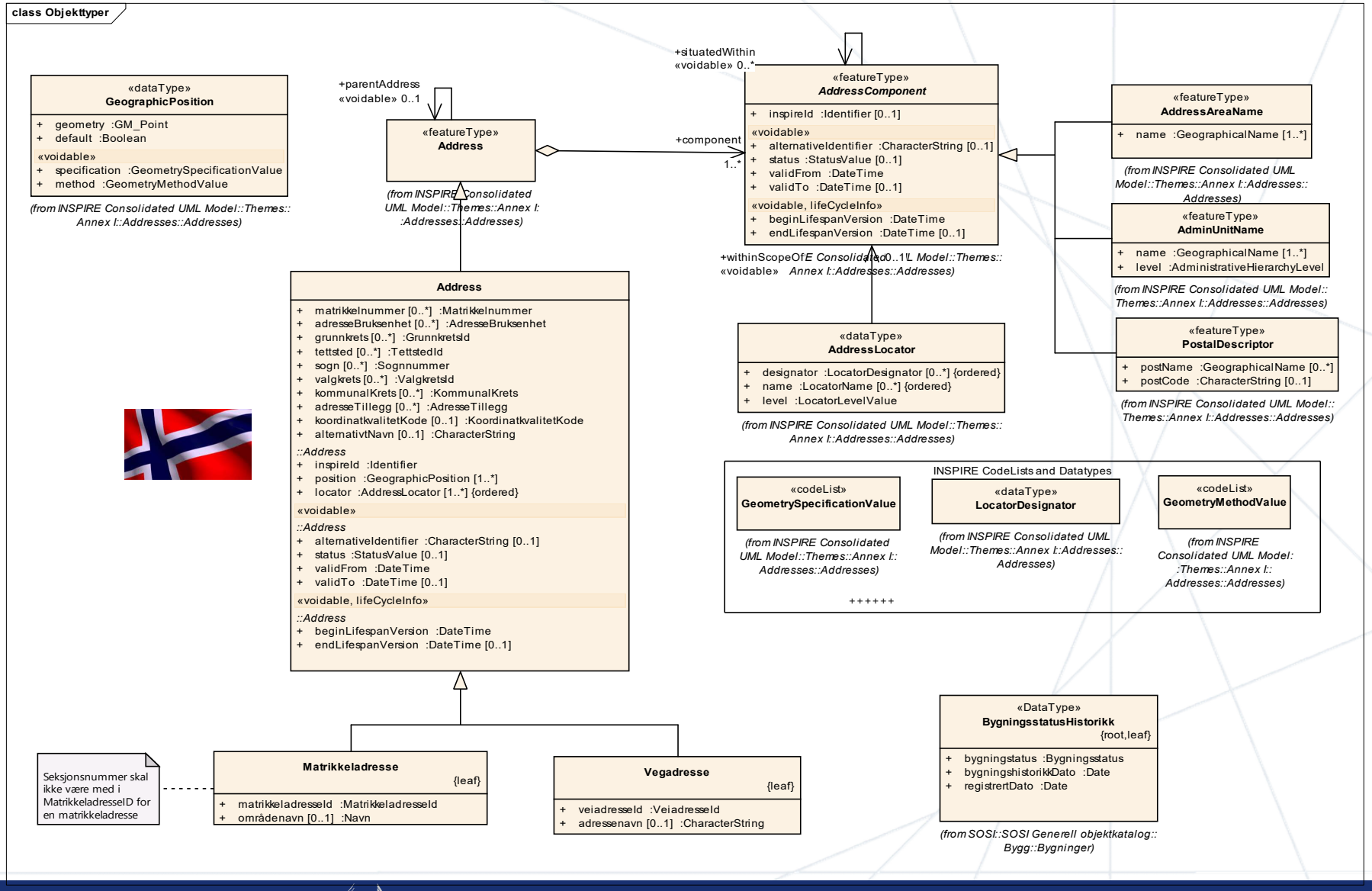
Implementation

https://service.projectplace.com/pp/pp.cgi/r1025631019

Colour scheme

Overview/Feature Types & data types	Apply color coding system for Feature Types and data types
	- INSPIRE white
	- ELF light green
codelists	Apply color coding system for codelists
	- INSPIRE codelists light yellow
	- ELF additional codelists yellow

Possible migration path for national specifications? [not within the remit of ELF]. Virtual example for national address model.



Possible migration path for national specifications. Virtual example for national address model.

Advantages:

- Increase the national awareness of European requirements.
- Takes INSPIRE specifications seriously when revising national specifications.
- Over time, reduce the need for transformations.
- Language independence, can use the alias mechanism in Enterprise Architect , or the tagged values for multilingual packages, featureTypes and PropertyTypes as described in ISO 19109 Rules for application schemas.
- Could use a subversion controlled version of the INSPIRE schemas to ensure that updates / bugfixes are implemented.
- May use the same schema for both European and national specifications by using the tagged value «Profile»
- May use the same schema for generalised products by using the tagged value «Profile»

Disadvantages:

- More complex UML models (and GML application schemas)
- Introduces the «voidable» mechanism in national models
- Not all models can use this methodology due to their complexity.

Annex A – ELF configuration file

Configuration files for the generation of GML application schemas and feature catalogues will be made available by request.

Questions?