

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: WP2 – Webinar (WP2 Public Consultation)

Author: Morten Borrebaek

Date: 2014-11-20

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

Contents

Scope.....	5
1 Definition and abbreviations.....	5
1.1 Definitions.....	5
1.2 Abbreviations.....	6
2 ELF modelling approach.....	6
2.1 Model structure.....	6
2.2 UML modelling principles.....	7
2.3 Naming conventions.....	13
3 Documentation.....	14
3.1 Definitions.....	14
3.2 ELF Diagrams.....	15
3.3 Feature catalogue.....	18
4 ELF UML Profile.....	18
4.1 Description.....	18
4.2 Implementation.....	21
4.3 Colour scheme.....	26
5 Step by step approach.....	26
6 Bibliography.....	27
Annex A – ELF configuration file.....	28

ELF WP 2 – Modelling guidelines ¹

Author: Morten Borrebæk

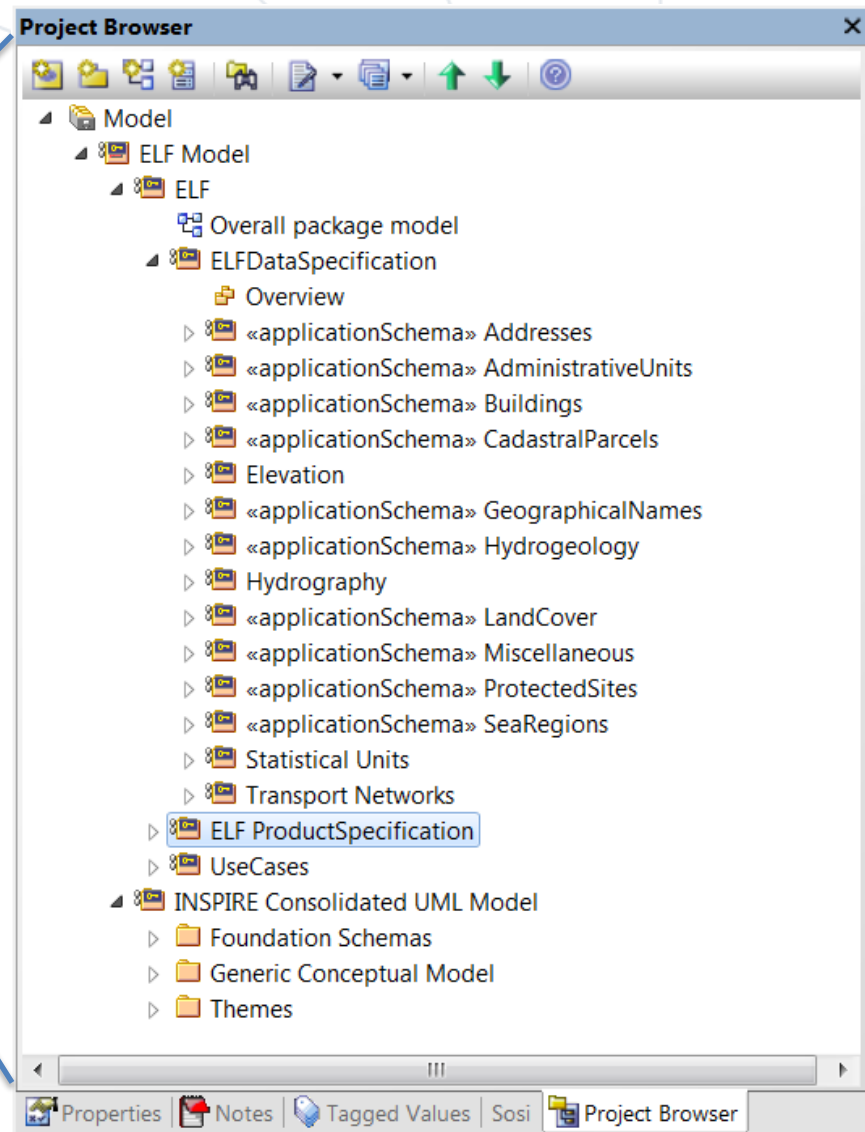
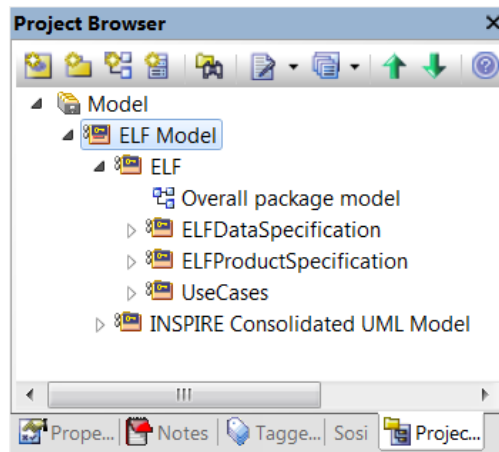
Date: 19.11.2014

Version: 1.14

Scope – important policy statement

The revised modelling guidelines in this document 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.

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

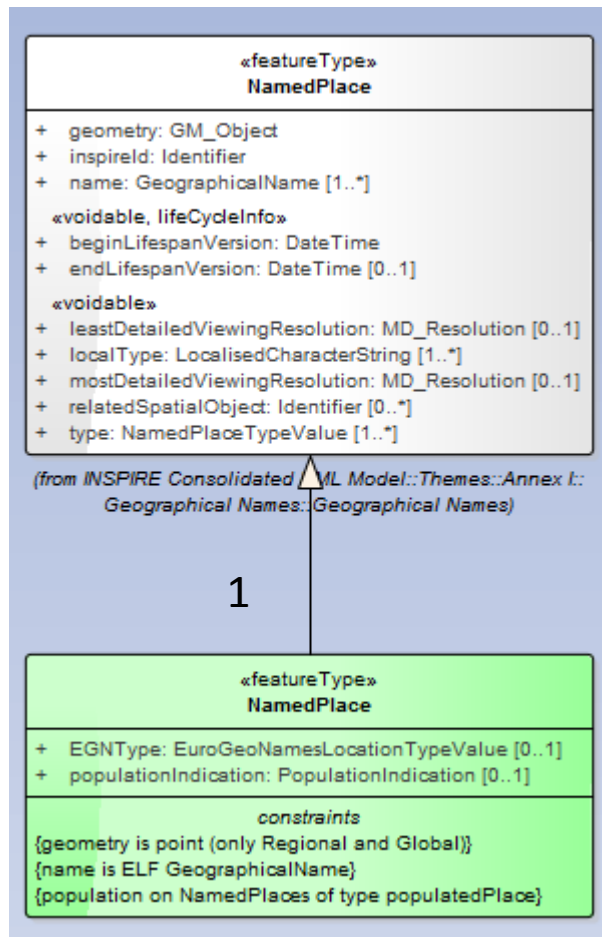
- Add new application schemas importing INSPIRE or other schemas as needed
- Add new types and constraints in the new application schemas
- extend INSPIRE code lists if not centrally managed

ELF UML modelling Principles (cont)

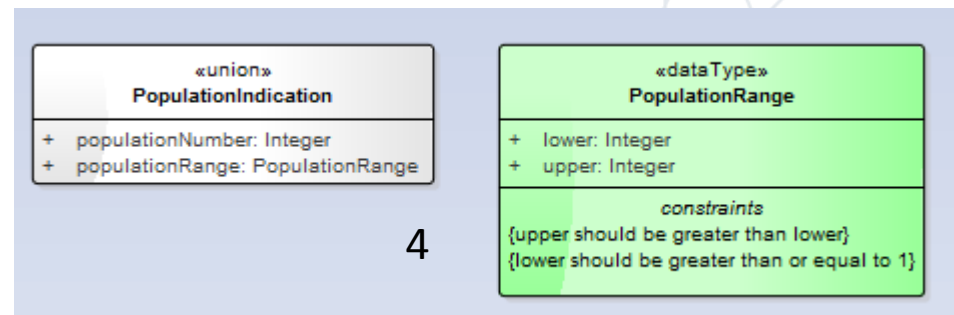
For each matching concept of INSPIRE and ELF identified from the analysis of the matching tables, a corresponding concept is created in ELF. Where possible, these concepts should be sub-classes of existing INSPIRE concepts (feature or data type, code lists etc.) by:

- 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.

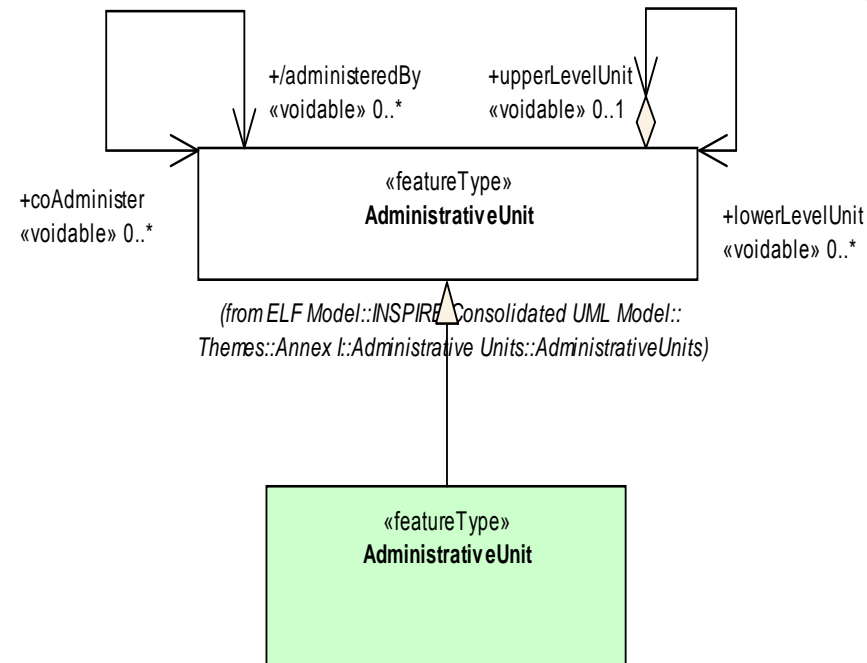
Example: NamedPlace



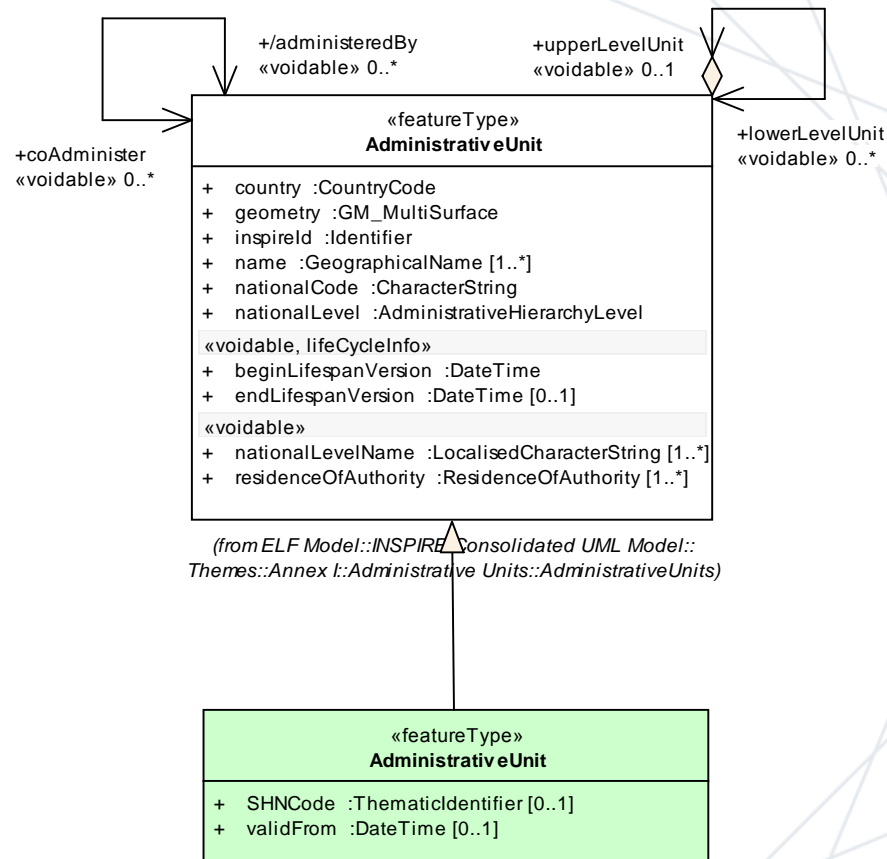
1. Subtype all INSPIRE feature types valid for ELF (topographic reference data)
2. Define additional attributes
3. Add constraints
4. Define new ELF classes (feature types, datatypes, codelists)
5. Associate feature types (not in figure)



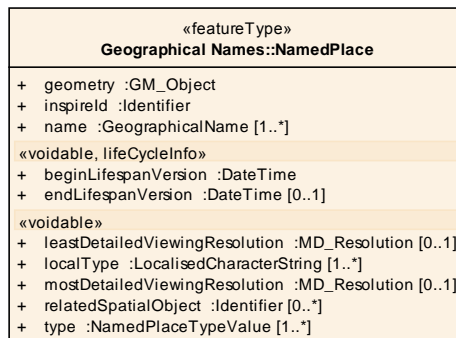
Step 1 - Subtyping INSPIRE feature types



Step 2 - Adding new attributes to ELF feature types



Step 3 - Adding constraints



Class : NamedPlace

Properties: General, Details, Templates, Advanced, Tagged Values

Rules: Requirements, Constraints, Scenarios

Related: Files, Links

Constraint: populationNumberPopulatedPlace

Type: OCL

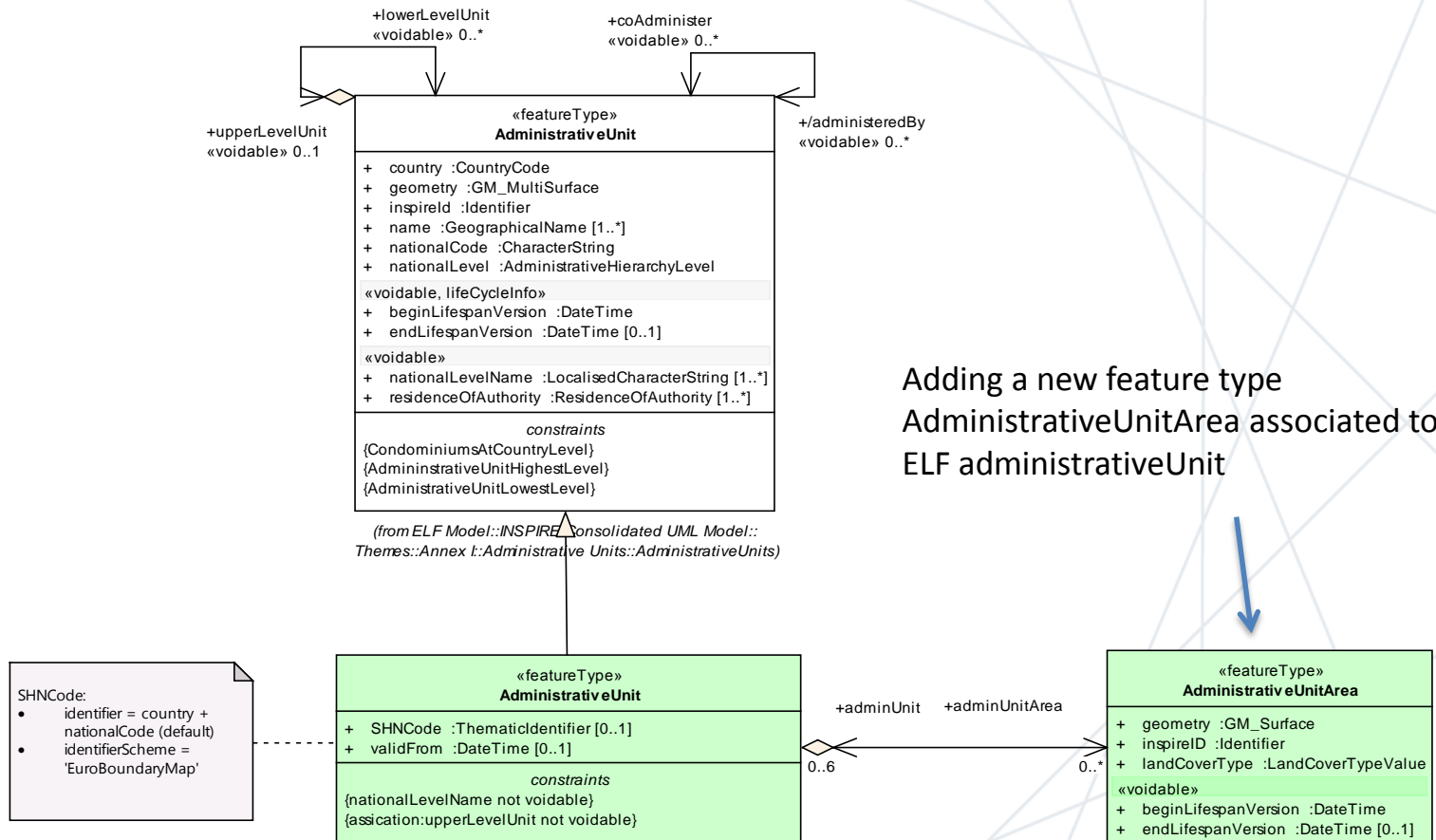
Status: Approved

/* populationNumber must be present if the type of the NamedPlace is populatedPlace */
 inv: self.type->forAll(t | t = NamedPlaceTypeValue::populatedPlace implies
 populationNumber->notEmpty())

Defined Constraints

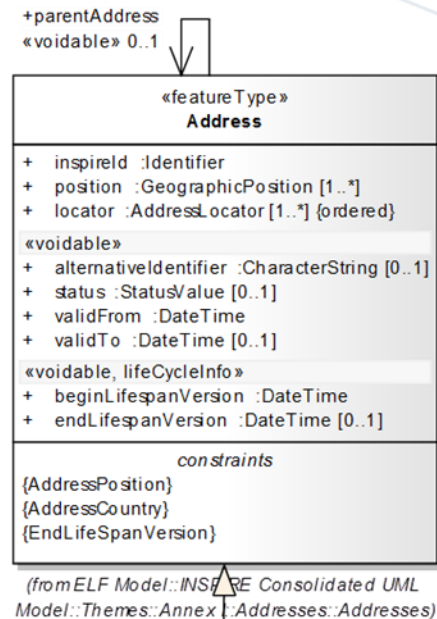
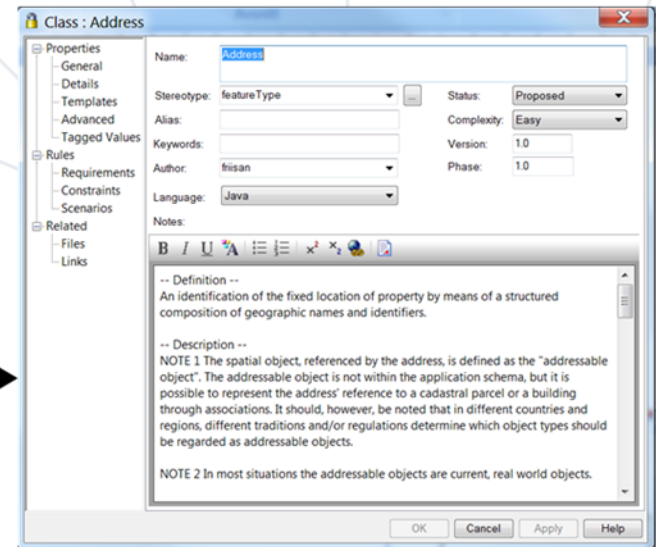
Constraint	Type	Status
geometryIsPointOrSurface	OCL	Approved
populationNumberPopulatedPlace	OCL	Approved

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.

Class : Address

Properties

- Name: Address
- Stereotype: featureType
- Status: Proposed
- Complexity: Easy
- Version: 1.0
- Phase: 1.0
- Keywords:
- Author: frisan
- Language: Java

Rules

- Requirements
- Constraints
- Scenarios

Related

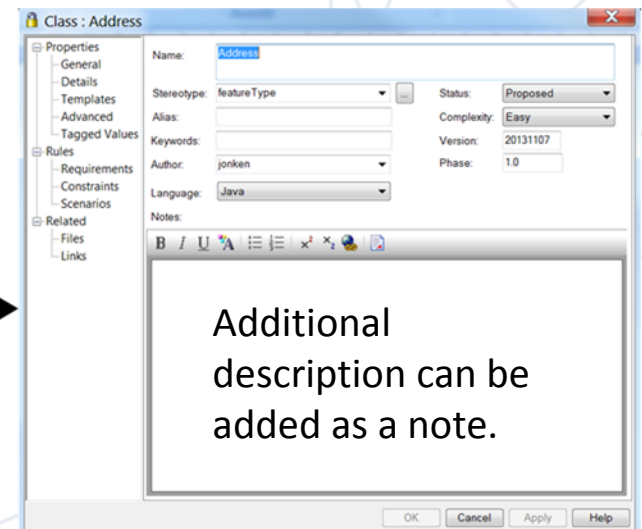
- Files
- Links

Notes

-- Definition --
An identification of the fixed location of property by means of a structured composition of geographic names and identifiers.

-- Description --
NOTE 1 The spatial object, referenced by the address, is defined as the "addressable object". The addressable object is not within the application schema, but it is possible to represent the address' reference to a cadastral parcel or a building through associations. It should, however, be noted that in different countries and regions, different traditions and/or regulations determine which object types should be regarded as addressable objects.
NOTE 2 In most situations the addressable objects are current, real world objects.

OK Cancel Apply Help



Class : Address

Properties

- Name: Address
- Stereotype: featureType
- Status: Proposed
- Complexity: Easy
- Version: 20131107
- Phase: 1.0
- Keywords:
- Author: jonken
- Language: Java

Rules

- Requirements
- Constraints
- Scenarios

Related

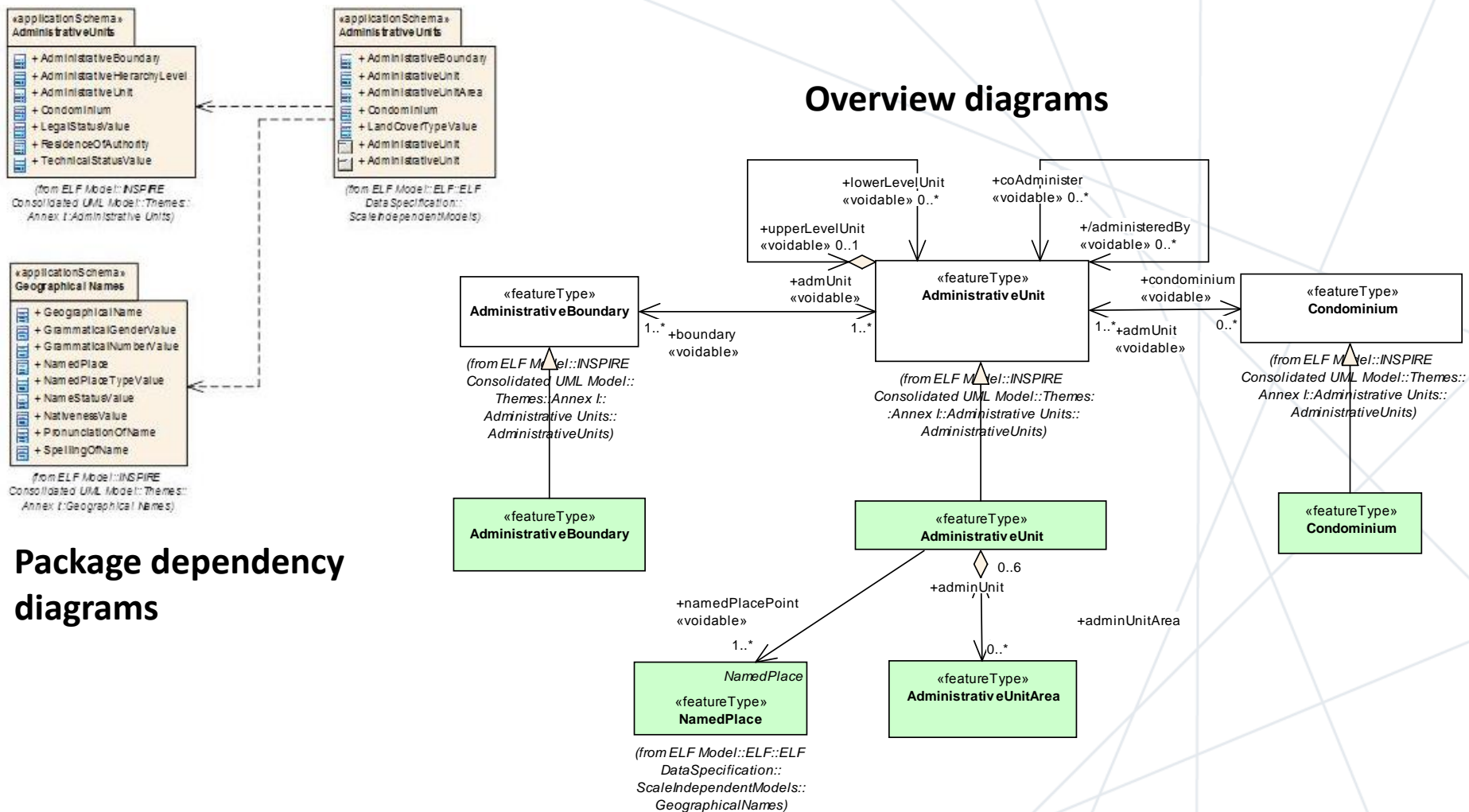
- Files
- Links

Notes

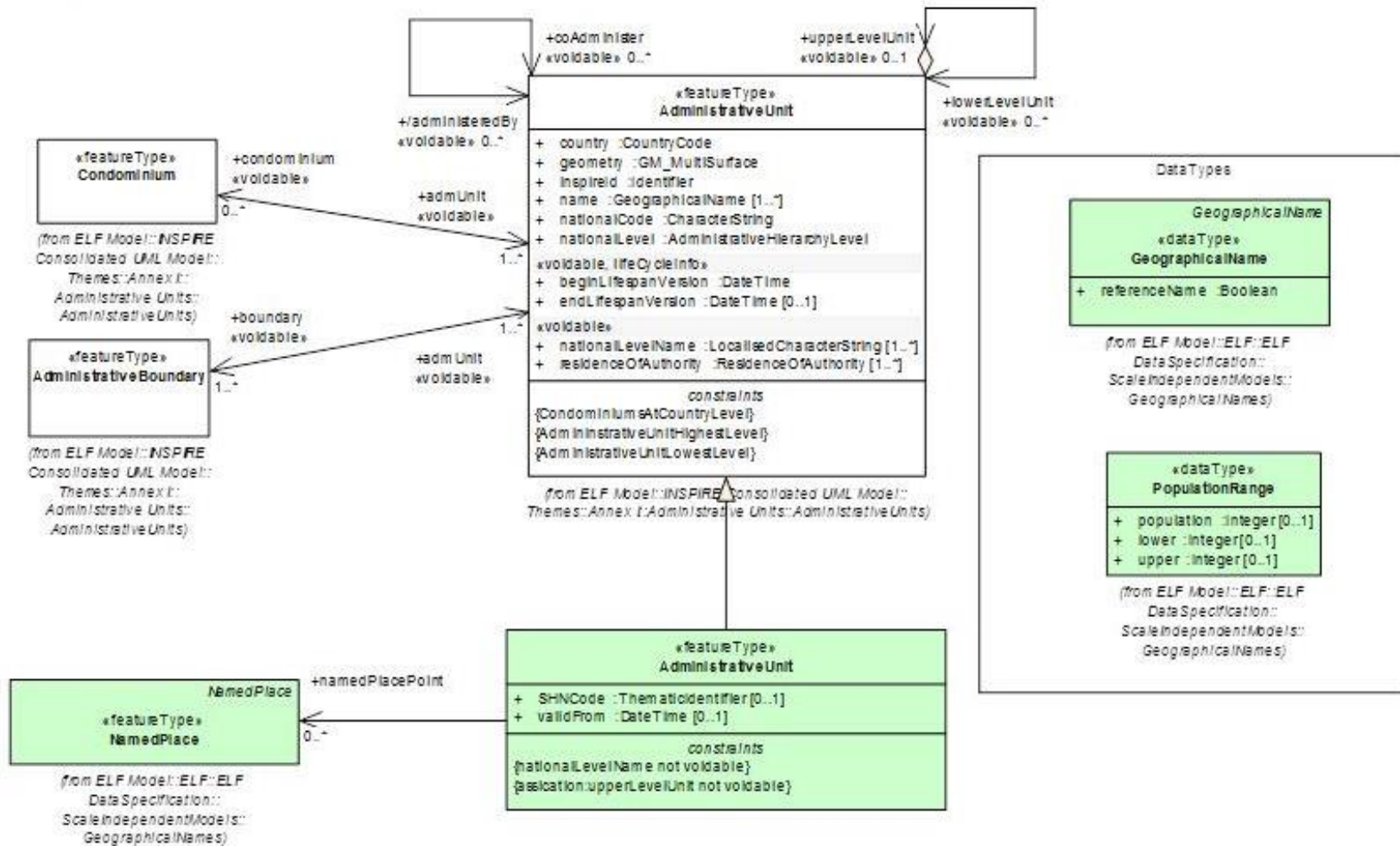
Additional description can be added as a note.

OK Cancel Apply Help

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
<u>applicationSchema</u>	<u>targetNamespace</u>	Target XML namespace of the application schema [ISO 19136]	Mandatory	<a href="http://www.locationframework.eu/schemas/<theme>/<version>">http://www.locationframework.eu/schemas/<theme>/<version> Example: http://www.locationframework.eu/schemas/LandCover/1.0 This structure allows one
	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" has a relative path which indicates its location.
	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
<u>dataType</u>	<u>xsdEncodingRule</u>	XML Schema encoding rule to apply	Mandatory	iso19136_2007_ELF_Extensions add [2.5]

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
<u>codelists</u>	Apply color coding system for <u>codelists</u>
	- INSPIRE <u>codelists</u> light yellow
	- ELF additional <u>codelists</u> yellow

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

Questions?

