



Version 2.5.0

# INSPIRE Solution Pack for FME

## User Manual

# Table of Contents

<b>1 Introduction</b>	<b>4</b>
<b>2 Preconditions and Installation</b>	<b>5</b>
<b>3 General Concept</b>	<b>14</b>
<b>4 Using the Workbench</b>	<b>16</b>
<b>5 Target Schema</b>	<b>26</b>
<b>6 Mapping Tables and Predefined Lists</b>	<b>27</b>

# User Manual

## Introduction

The directive for an Infrastructure for Spatial Information in the European Community (INSPIRE) requires that spatial information is harmonized and served by standardized services. Data transformation compliant with the INSPIRE Annexes is therefore one of the main tasks for data providers in this context.

The INSPIRE Solution Pack supports this task in various ways. It extends the spatial ETL (Extract Transform Load) capabilities already provided by FME with additional tools. The INSPIRE Solution Pack (ISP) for FME provides support for the transformation of source datasets into the INSPIRE data models. FME's ability to work with more than 350 formats make the INSPIRE Solution Pack for FME a very flexible tool for mapping to the INSPIRE data models. The ISP provides new transformers, tutorial and template workspaces and supports the INSPIRE Esri Geodatabase schema and writing INSPIRE GML directly.

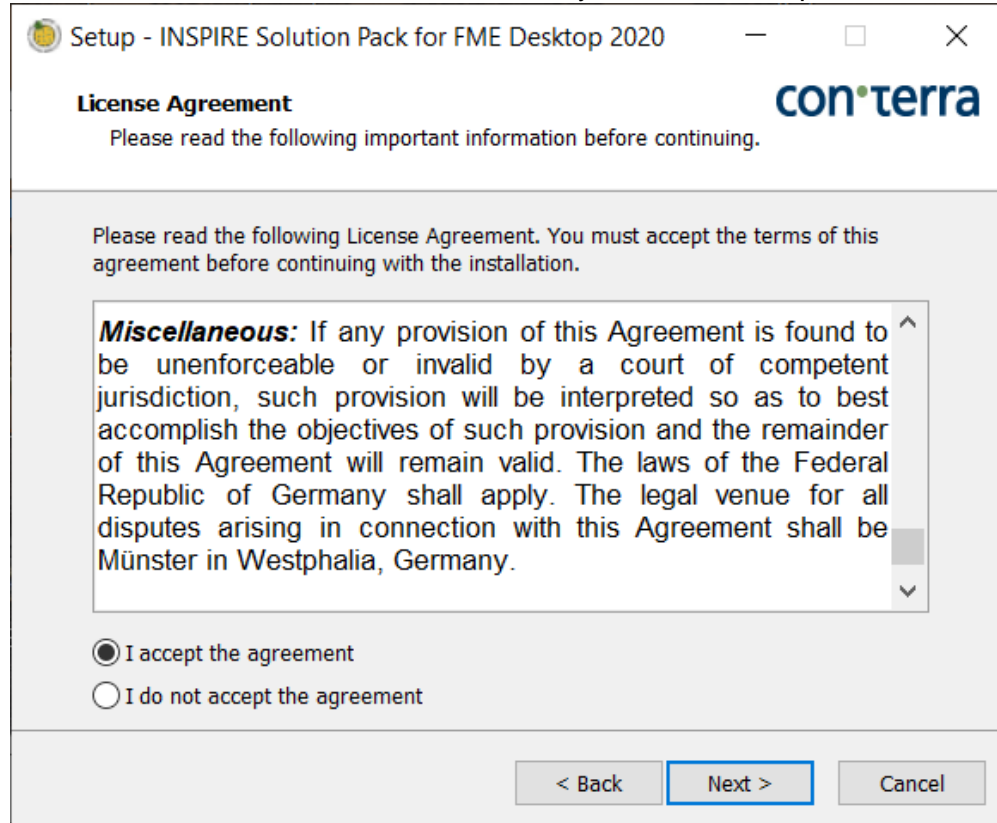
## Preconditions and Installation

The INSPIRE Solution Pack 2020 requires FME 2020. ArcGIS needs to be installed and licensed in order to write INSPIRE GDBs for ArcGIS for INSPIRE.

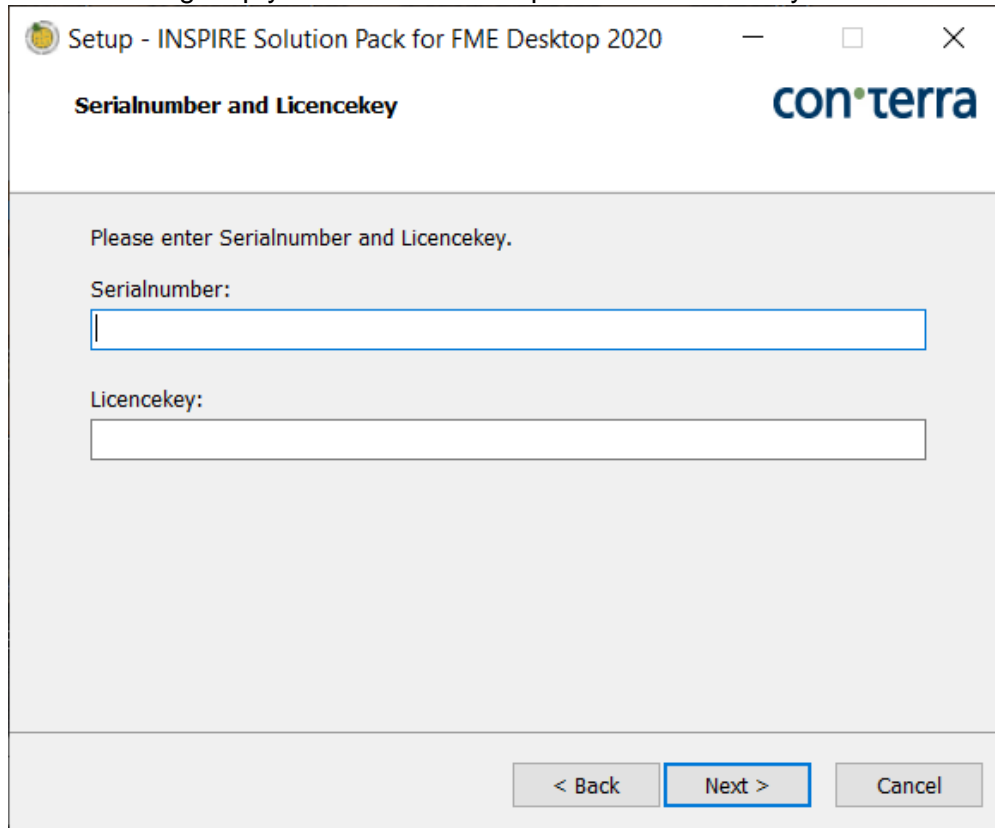
To install the INSPIRE Solution Pack run the 'INSPIRESolutionPack2020.exe' and follow the Installation Wizard.



In order to use the INSPIRE Solution Pack, you have to accept the license agreement.



In the following step you will be asked to provide a license key.



Setup - INSPIRE Solution Pack for FME Desktop 2020

**Serialnumber and Licencekey** **con•terra**

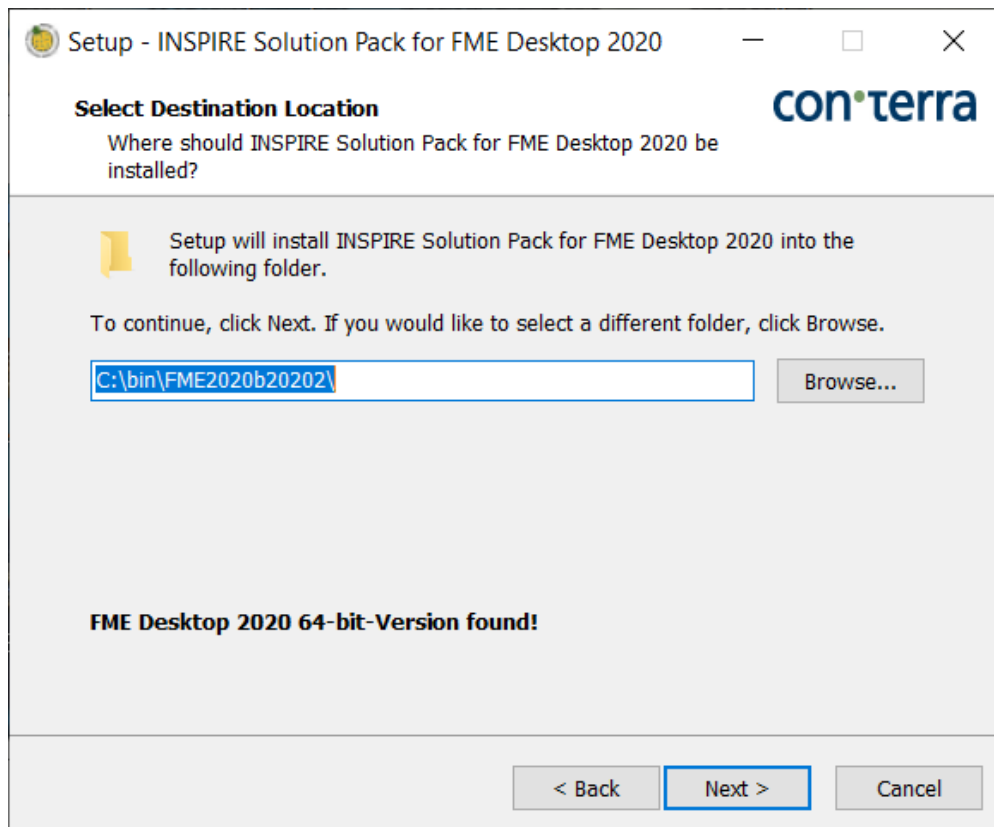
Please enter Serialnumber and Licencekey.

Serialnumber:

Licencekey:

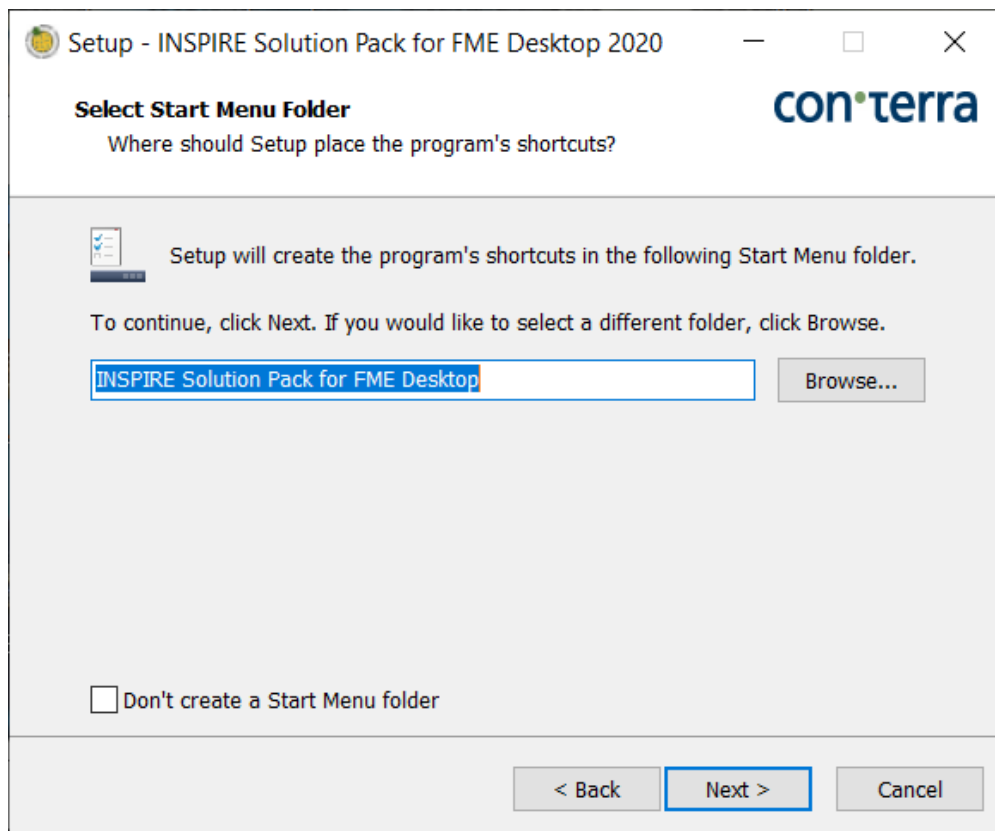
< Back Next > Cancel

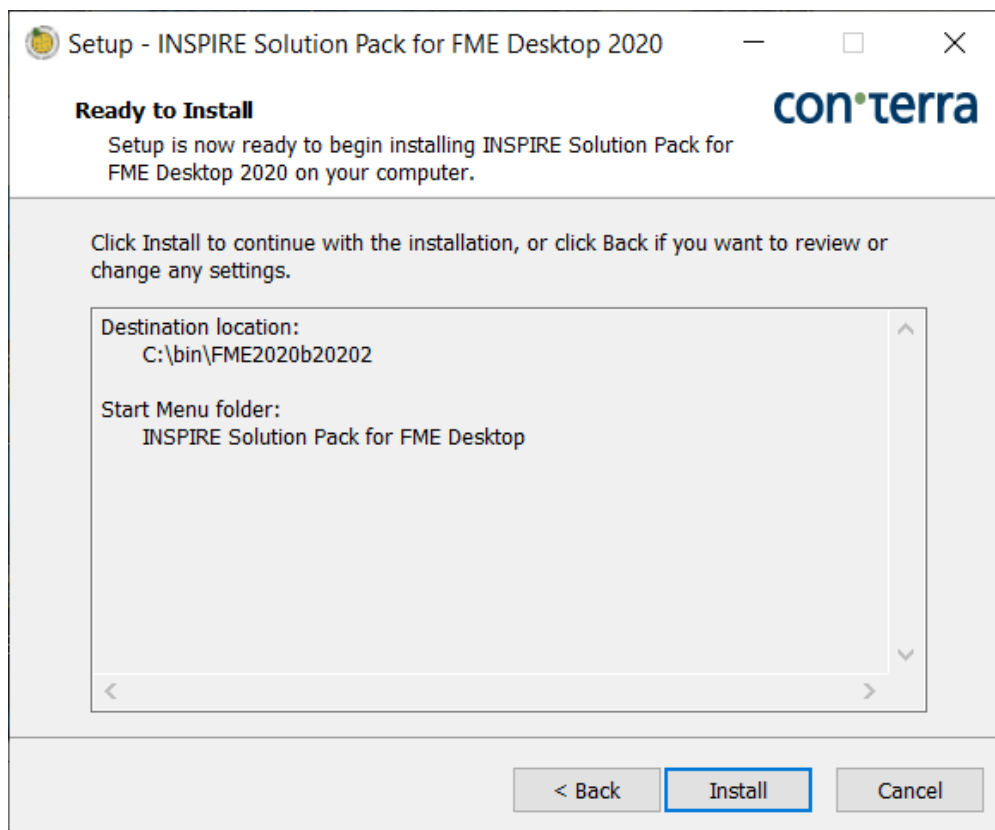
Finally select the home directory of the FME installation you would like to use with the Solution Pack. If you have multiple versions of FME installed select the one you would like to use with the Solution pack. The Installer automatically detects if your version of FME is suitable for the INSPIRE Solution Pack.

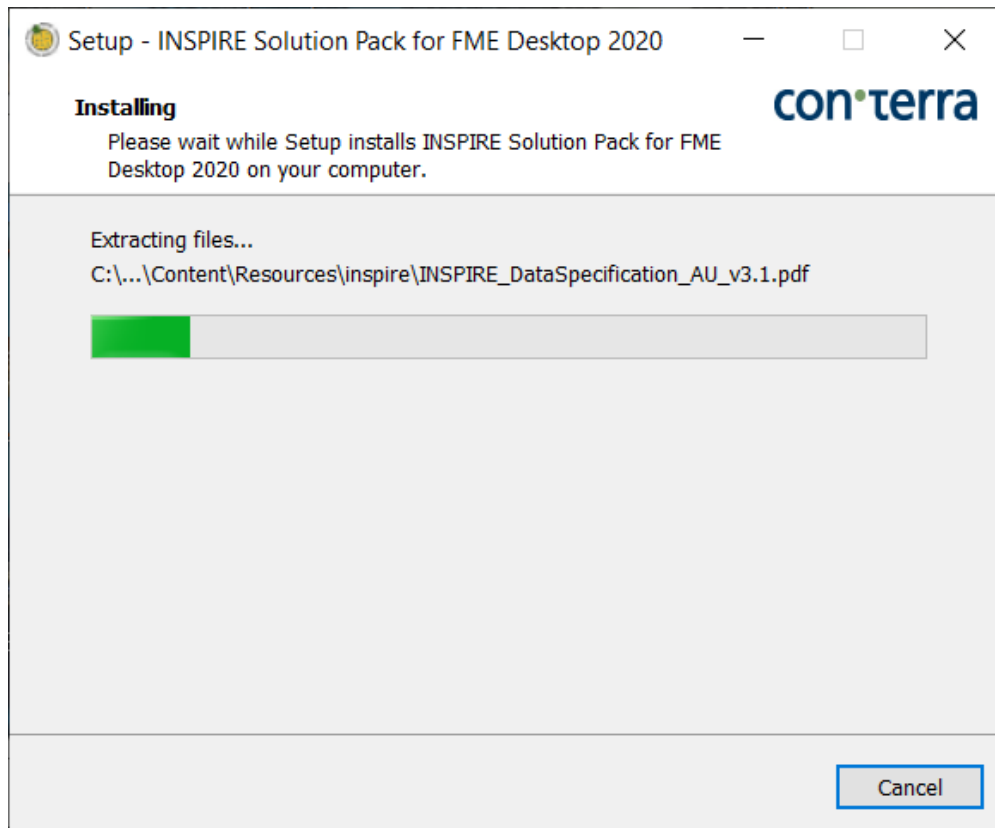




During the following steps follow the instructions of the installer software:









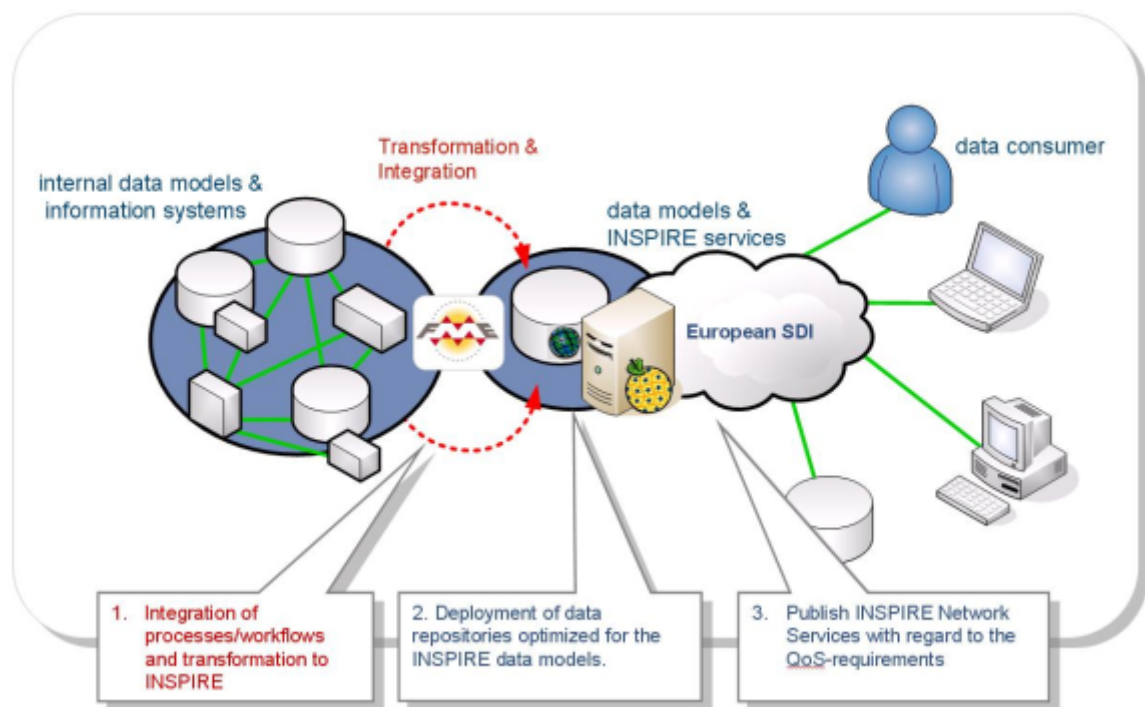
After the installation has finished you are ready to start. There are new start menu entries that contain links to the documentation and additional resources. To use the Inspire Solution Pack for FME simply start FME. The following chapters give an overview of the functionalities and provide instructions on how to use the Solution Pack.



## General Concept

The process of integrating data from a local information system, with its own data model, consists of three main steps. Step one is the transformation of the source data set into an INSPIRE-compliant format. Step two is the deployment of the transformed data into a data store that is optimized for INSPIRE and step 3 involves the final publishing of the transformed data into the European SDI. These three steps are illustrated in the figure below.

The INSPIRE Solution Pack for FME targets the first two steps. The third step is covered by solutions like ArcGIS for INSPIRE.



The transformation of the source data is a complex process that cannot be fully automated, since it requires knowledge of both the local data and the complex INSPIRE data models. The transformation consists of several tasks itself. The source data has to be imported, a target data model needs to be available, the mapping needs to be created, and finally the transformation can be executed.

Relying on FME, the import of the source data is in the most cases simply carried out by the selection and configuration of the right FME readers. THE INSPIRE Solution Pack delivers pre-configured workspaces for writing to the ESRI INSPIRE Geodatabase as well as for writing INSPIRE compliant GML. Thus the main work remaining is the mapping task. Features and attributes need to be analyzed and connected with attributes in the target data model. In many cases, additional processing is necessary; attributes need to be created, joined, transformed and manipulated to fit into the INSPIRE data model. All this happens in the FME Workbench, which provides a set of transformers to define the mapping process via a graphical user interface. FME standard transformers provide many

useful operations, like attribute mapping, attribute creation or attribute renaming. The INSPIRE Solution Pack extends these basic transformers with additional transformers, which are optimized to support the mapping process in many ways. There are transformers for ID creation, versioning and support for more than a hundred predefined value domains inside the INSPIRE data specification. All of these transformers aim at moving as much INSPIRE logic into the workbench as possible.

The following parts of this manual provide practical information how to use the INSPIRE Solution Pack and provide practical guidance on how to create mappings. It can also be helpful to check the Tutorial that comes with the INSPIRE Solution Pack, as it provides a practical example how to use the Solution Pack. It covers Administrative Units, and describes the complete mapping process step by step.

## **Using the Workbench**

### **Additional INSPIRE Mapping Transformers**

The FME Workbench is extended in two ways. There are new transformer categories, with transformers for the different INSPIRE Annex Themes, plus one category for general purpose INSPIRE transformers.

### **The Transformer in the Annex theme categories**

The transformers in the categories named following the INSPIRE annex themes, have the function to create the values from the enumerations and code lists defined in the data specifications. The transformers provide the lists of predefined attribute values as they are defined in the INSPIRE UML-data models and INSPIRE Registry. The naming of a transformer also indicates for which attribute and feature it should be used:



## Transformer Gallery

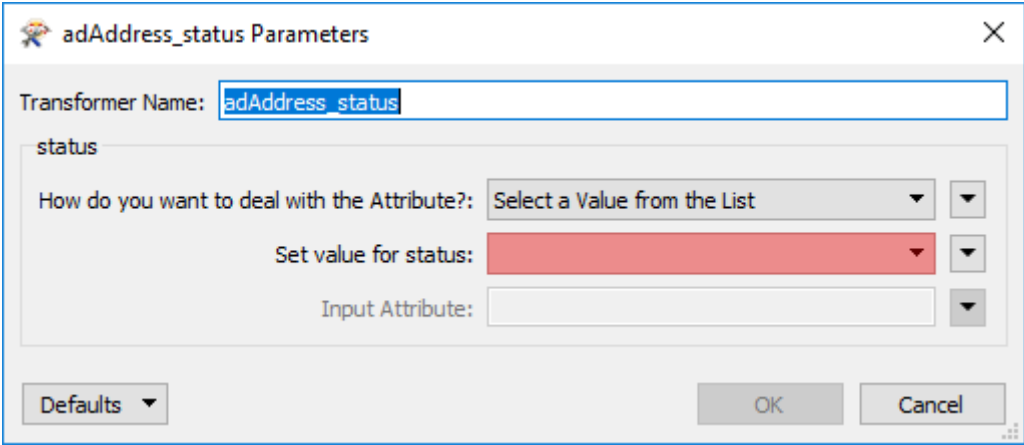


There are transformers for both supported formats. The screenshot shows the inspire categories. The transformers can be used like any other FME transformer. To add one, drag it into the workbench or use the quick-add by typing the first letters while the mouse is inside the workbench. The categories with 'GDB' in the name contain transformers that support writing the ESRI GDB model. The transformers in the categories with 'GML' in the name support writing GML using the INSPIRE Writer or the GML writer.

The name of the transformers follow a given structure. The name of the transformer start with lowercase letters that identify the Annex Theme. 'ad' for Addresses, 'cp' for Cadastral Parcels and so on. For the GDB transformers the name until the last underscore identifies the feature type. For example the 'adAddress\_status' contains values for the attribute 'status' from the feature type 'adAddress'. The 'psSite\_siteld\_siteldScheme' contains values for the attribute 'siteldScheme' in the feature class 'psSite\_siteld'.

The naming of the GML transformers follows a similar principle. The part of the name after the dot is the attribute name from the UML model. The part before the dot is the name of the Datatype or Feature Type that this transformer belongs to. For example the 'adAddress.status' contains values for the attribute 'status' from the feature type 'Address'. However the name of the Datatype or Feature Type does not always equal the name of the GML writer feature type. This is because of the complex structure of the GML model. In general the UML diagram that is linked from the templates should help to identify the correct attribute.

The transformers for both formats have a very similar user interface:



The field 'how do you want to deal with...' specifies what general action should be carried out for the attribute. It allows you to specify whether the parameter receives a value or is left empty. In case the field is voidable and is going to be left empty, a reason can be provided. The reason specifies why a field is empty in accordance with the INSPIRE Generic Conceptual Model which specifies the following choices:

- [Unknown](#)
- [Unpopulated](#)
- [Withheld](#)

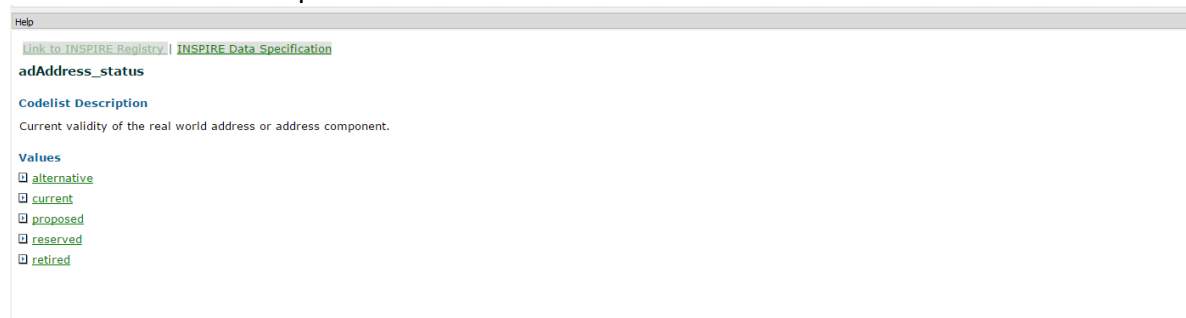
For the GDB transformers for the Annex 1 themes it provides the value No reason instead of Withheld. This list is preliminary and may be extended in the future.

In the Geodatabase the reasons are mapped to the values 0, 1, 2 for the reasons, but this is handled internally by the transformer. If the field is not void, select a value from the list.

In this case the status of an Address. Possible values are:

- [current](#)
- [retired](#)
- [proposed](#)
- [reserved](#)
- [alternative](#)

If the source data already has an attribute that holds the correct values for the status attribute you can select this attribute in the 'Input Attribute' field and the transformer takes the values from the 'Input Attribute'.



There is help available for all the attributes inside the workbench. The transformer description contains additional information about the allowed values and further notes on when to choose which attribute.

## INSPIRE\_IFCID

The Category INSPIRE contains a transformer IFCID. This transformer is used to create a unique Identifier for each feature in a table. This Identifier is simply a count starting from 0 and assigning an increasing number to each processed feature. The IFCID is not an attribute required by INSPIRE, but by the Geodatabase model. It is, for example, used to maintain the references between different features.

## INSPIRE\_IdentifierSetter and INSPIREGML\_IdentifierSetter

This transformer sets the attributes namespace, version and localId. Set the namespace and version by specifying a value or select an attribute. For the localId an attribute that provides the values is expected. The field versionId is voidable, if you chose to leave the version attribute empty select one of the predefined reasons. INSPIRE describes the attributes as follows:

### **localId**

A local identifier, assigned by the data provider. The local identifier is unique within the namespace, i.e. no other spatial object carries the same unique identifier.

NOTE: It is the responsibility of the data provider to guarantee uniqueness of the local identifier within the namespace.

### **namespace**

Namespace uniquely identifying the data source of the spatial object.

NOTE: The namespace value will be owned by the data provider of the spatial object and will be registered in the INSPIRE External Object Identifier Namespaces Register.

### **versionId**

The identifier of the particular version of the spatial object, with a maximum length of 25 characters. If the specification of a spatial object type with an external object identifier includes life-cycle information, the version identifier is used to distinguish between the different versions of a spatial object. Within the set of all versions of a spatial object, the version identifier is unique.

NOTE: The maximum length has been selected to allow for time stamps based on ISO 8601, for example, "2007-02-12T12:12:12+05:30" as the version identifier.

NOTE 2: The property is void, if the spatial data set does not distinguish between different versions of the spatial object. It is missing, if the spatial object type does not support any life-cycle information.

The localId and the namespace shall only use the following set of characters: {"A" ... "Z", "a" ... "z", "0" ... "9", "\_", ".", "-"}, i.e. only letters from the Latin alphabet, digits, underscore, point, and dash are allowed.

## INSPIRE\_LifespanSetter and INSPIREGML\_LifespanSetter

Set the beginLifespan and endLifespan Attributes that are used for INSPIRE Feature Types with the lifeCycle stereotype. The Attributes are voidable. If no values should be set, chose one of the void reason values. There are two options to set the lifespan values. If an attribute exists that already has the right value, select it as the input attribute. The other option is to use the TimeStamper which is integrated in this transformer. If you want the transformer to set the attribute to the actual date and time choose "yes" for the option "Set beginLifespan to actual date and time?".

The INSPIRE Data specifications on Temporality representation

The application schemas use the derived attributes "beginLifespanObject" and "endLifespanObject" to record the lifespan of a spatial object.

The attribute "beginLifespanVersion" specifies the date and time at which this version of the spatial object was inserted or changed in the spatial data set. The attribute "endLifespanVersion" specifies the date and time at which this version of the spatial object was superseded or retired in the spatial data set.

NOTE 1 The attributes specify the beginning of the lifespan of the version in the spatial data set itself, which is different from the temporal characteristics of the real-world phenomenon described by the spatial object. This lifespan information, if available, supports mainly two requirements: First, knowledge about the spatial data set content at a specific time; second, knowledge about changes to a data set in a specific time frame. The lifespan information should be as detailed as in the data set (i.e., if the lifespan information in the data set includes seconds, the seconds should be represented in data published in INSPIRE) and include time zone information.

NOTE 2 Changes to the attribute "endLifespanVersion" does not trigger a change in the attribute "beginLifespanVersion".

Recommendation - If life-cycle information is not maintained as part of the spatial data set, all spatial objects belonging to this data set should provide a void value with a reason of "unpopulated".

## INSPIRE ValiditySetter and INSPIREGML\_VValiditySetter

Set the validFrom and validTo Attributes that are used for a couple of INSPIRE Feature Types. The Attributes are voidable. If no values should be set choose one of the void reason values. There are two options to set the lifespan values. If an attribute exists that already has the right value, select it as the input attribute. The other option is to use the TimeStamper which is integrated in this transformer. If you want the transformer to set the attribute to the actual date and time choose "yes" for the option "Set validFrom/validTo to actual date and time?". Please keep in mind that the intention of the Attributes is different and the purpose of the time stamping option is only for testing issues. The intended use of the validity fields is described in the next paragraph. If more detail is required please check the official INSPIRE data specifications.

The INSPIRE Data specifications on validity representation

The attribute "validFrom" specifies the date and time at which the real world phenomena that are represented by the spatial objects in a spatial data set started to exist in the real world. The attribute "validTo" specifies from which date and time these phenomena no longer exist.

Recommendation - If information about the actual start and end of the existence of elements is not maintained as part of the spatial data set, all spatial objects belonging to this data set should provide a void value with a reason of "unknown".

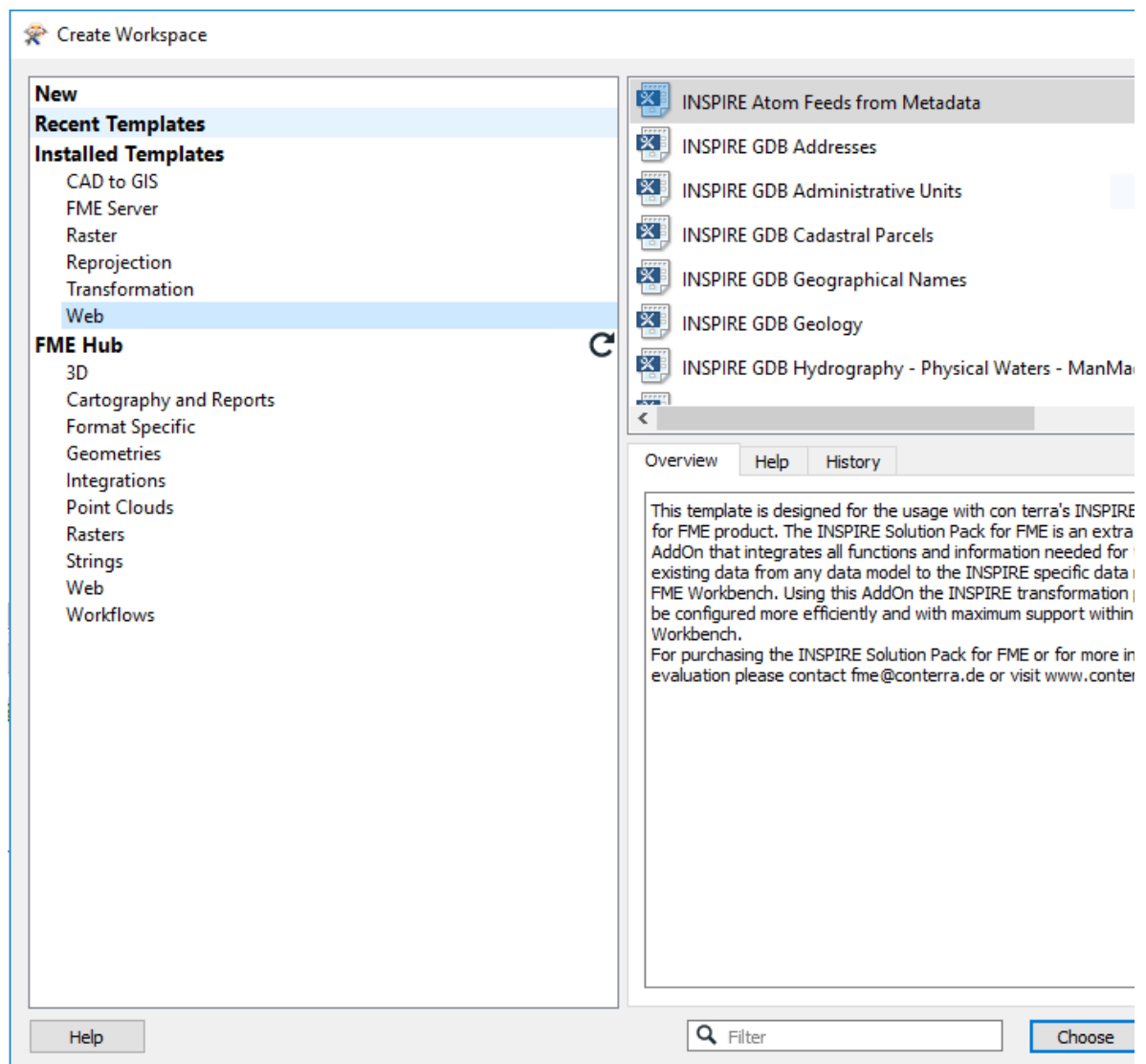
## Installed Template Workspaces & FME Store Templates

A couple of template workspaces are delivered with the INSPIRE Solution pack. These templates can be found in the Create Workspace dialog under 'Installed Templates'. These are the templates that were available with the release of your version of the INSPIRE Solution Pack. If you are looking for the latest version of the templates or new templates that were not available at the time of the release, you can find the latest versions of all templates in the FME Store. The category is name 'con terra (INSPIRE Solution Pack)' in both cases.

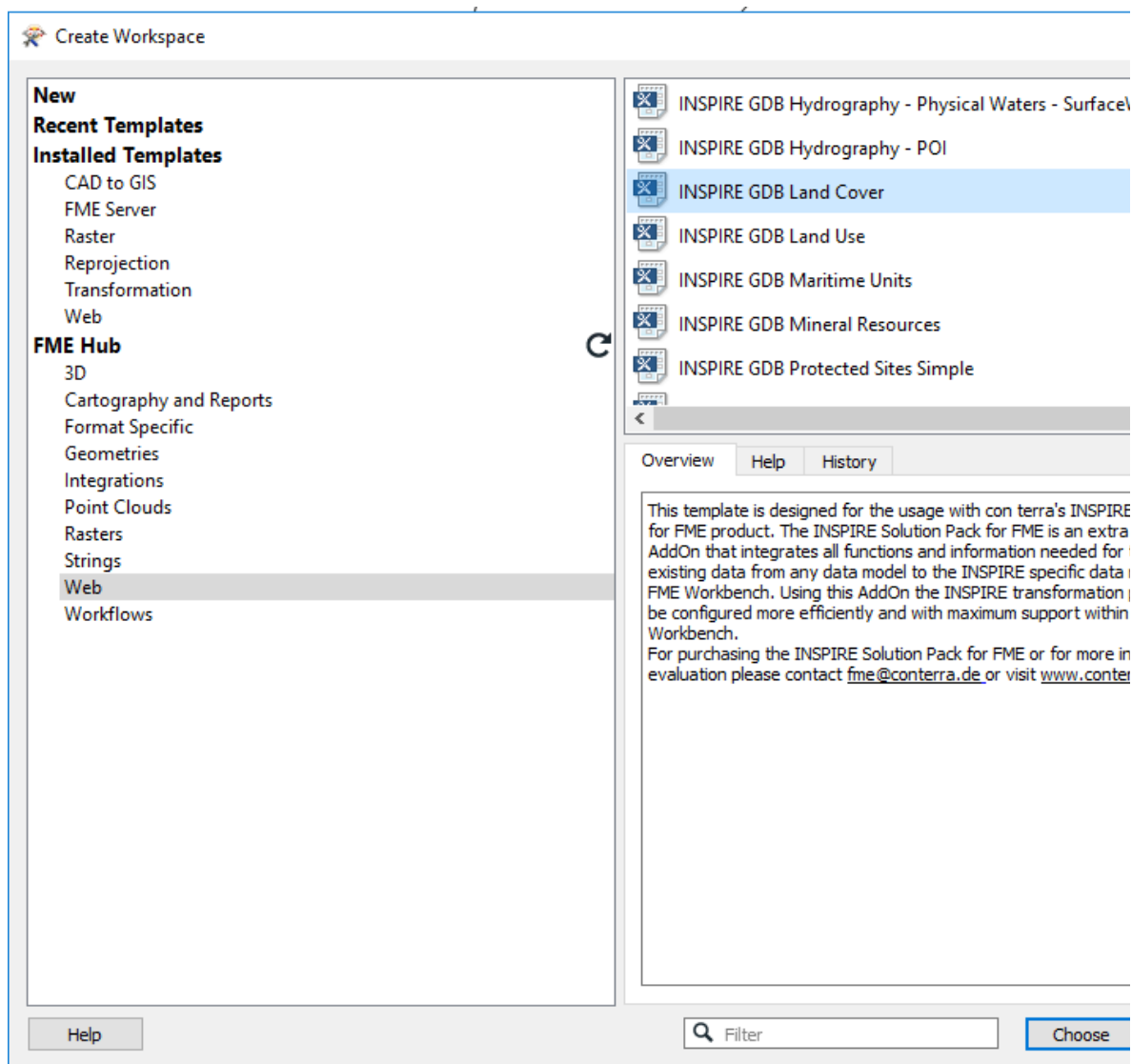
The template workspaces contain pre-configured writers for the target database which are already connected to necessary transformers in the context of that theme. Additionally the workspaces give instructions on how to proceed to conduct the mapping. In this document, we use the 'administrative units' template workspace to describe the general procedure.

To open a template workspace, select 'File->New' from the menu. The INSPIRE Templates are in the 'XML' Category:

Use either installed templates:



Or templates from the FME Store:

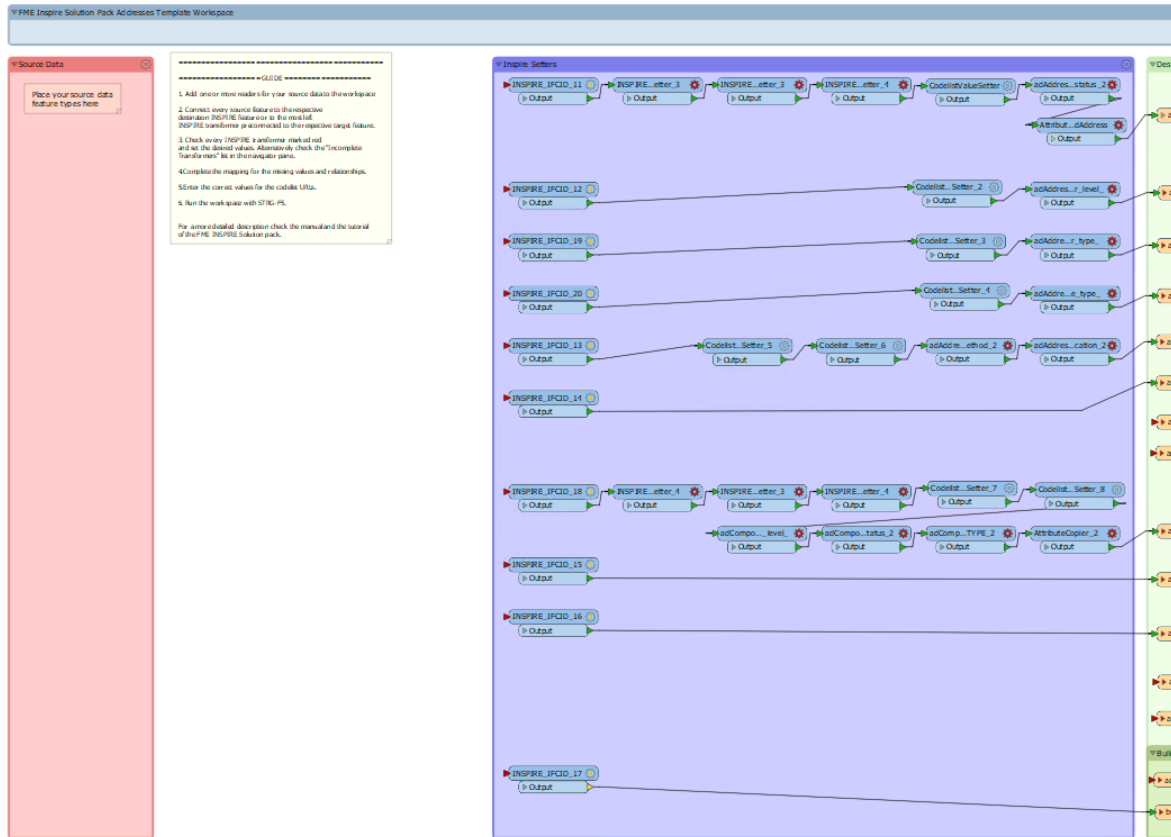


Similar as for the transformers there are different templates for writing GDB and GML.

The Workspace is depicted in the next Figure is the 'INSPIRE GDB Administrative Units' template. Bookmarks are used to indicate different purposes of areas. The green bookmark marks the area where the writers for the target geo-database are preloaded. Usually the writers here have to be modified to point to the desired local or remote geo-database, which serves as the target database. The blue bookmark indicates the area where pre-defined transformers are placed in order to make the transformation easier. The red area is the starting point to insert the input data readers. The input data may come from several sources.

After inserting the readers everything is ready to create the mapping and fill the target database. Use the empty space in the template. If you require more space drag and drop the blue and green areas further to the right.





## Tutorial Workspaces for Administrative Units (GDB & GML)

There are tutorials including completed workspaces that illustrate the necessary steps to create a mapping from a source dataset to the INSPIRE GDB or INSPIRE GML. The tutorial begins with the administrative units template workspace. The Tutorials are described in a separate document that is linked from the INSPIRE Solution Pack start menu entry, or directly from the following path: <FME\_FOLDER>/inspire solution pack /tutorial. The PDF documents describe the necessary steps to create a mapping between OpenStreetMap data and the INSPIRE Geodatabase Model used by ArcGIS for INSPIRE or INSPIRE GML.

## Target Schema

The preconfigured target schemas are implementations of the conceptual INSPIRE UML model. There are currently two implementations available.

### ESRI INSPIRE GDB

The target schema used is an implementation of the INSPIRE conceptual model as an Esri geo-database. It does not contain INSPIRE compliant data itself, but provides an empty, INSPIRE-compliant container. ArcGIS for INSPIRE uses this data model to provide the INSPIRE compliant view and download services. Since it is a database implementation of an object oriented model the hierarchical elements and relationships from the original data model are represented by the use of multiple tables which have to be set into relation, to connect attributes with their parent feature type. The relations are set in two possible ways: Either by setting a reference to the IFCID of the target feature type or in an external table. The actual relationships are specified in the Annex of this document. The documentation that comes with the data model describes the attributes and relationships more specifically.

### INSPIRE GML

The other target writer is based on the implementation as GML. The INSPIRE Writer and the GML Writer provided in FME can both be used to write GML that is compliant with the INSPIRE GML schemas. The schemas are available from <http://inspire.ec.europa.eu/schemas/> For further information on the GML Writer and the INSPIRE Writer please refer to the FME Documentation provided by Safe Software: [https://docs.safe.com/fme/html/FME\\_Desktop\\_Documentation/FME\\_ReadersWriters/Home.htm](https://docs.safe.com/fme/html/FME_Desktop_Documentation/FME_ReadersWriters/Home.htm).

# Mapping Tables and Predefined Lists

## Mapping Tables

Mapping tables are a tool to document the mapping. You can find the mapping tables in the Start Menu under "INSPIRE Solution Pack/MatchingTables GML" or from the INSPIRE Website. The current version of the mapping tables is available from: <http://inspire.ec.europa.eu/data-model/approved/r4618-ir/mapping/>

## Predefined Lists

Predefined Lists contain allowed CodelistValues for the GDB and GML model. This type of mapping table are for use inside the FME Workbench e.g. inside a Schemamapper, or AttributeValueMapper Transformer. The documents can be edited to describe the mapping of values from the local data to the predefined values from an INSPIRE codelist and then be imported into the FME workspace.

Phone: +49 251 59689 400  
Email: [support@conterra.de](mailto:support@conterra.de)  
Web: [www.conterra.de](http://www.conterra.de)