

Navigo

# articy:draft - Query Language

Reference and Tutorial

Version: 2.2

## Table of contents

1	Articy query language.....	5
1.1	Using queries in the 'Search' tab of articy:draft.....	5
1.2	Using queries in calculated reference strips .....	5
1.3	Notation of values .....	5
1.3.1	Logical values (booleans).....	5
1.3.2	Numbers .....	5
1.3.3	Text .....	6
1.3.4	Colors.....	6
1.3.5	Enumerations .....	6
1.3.6	Objects.....	6
1.3.7	Lists of objects .....	6
1.4	Using a SELECT query .....	6
1.4.1	Starting point.....	8
1.4.2	Where clause .....	8
1.4.3	Fieldnames and properties.....	9
1.4.4	Self .....	9
1.4.5	Escaping of identifiers .....	9
1.4.6	Conditions (relational operators) .....	9
1.4.7	Transform Clause.....	10
1.4.8	Group clause.....	10
1.4.9	Order clause .....	10
1.4.10	Limit clause.....	10
1.5	Extended relational operators.....	11
2	Operator .....	12
2.1	Object based query functions .....	12
2.1.1	IsObjectType .....	12
2.1.2	IsDescendantOf .....	13
2.1.3	IsAncestorOf .....	13
2.1.4	ClaimedBy.....	14
2.1.5	Template.....	14
2.1.6	Feature .....	14
2.1.7	HasConflict.....	14
2.2	Property based query functions .....	15
2.2.1	Compare .....	15
2.2.2	Function calls supporting “Self” .....	17
2.3	Handling of NULL and non-existing values.....	17
2.4	Transform functions .....	18

- 2.4.1 UniqueReferences ..... 18
- 2.4.2 UniqueAncestors ..... 18
- 2.4.3 UniqueParents ..... 18
- 3 Examples ..... 19
- 4 Overview of Object Properties ..... 21
  - 4.1 Project ..... 21
  - 4.2 Assets (The top level system folder containing all assets) ..... 22
  - 4.3 AssetsUserFolder ..... 22
  - 4.4 Asset ..... 22
  - 4.5 Flow (The top level flow fragment containing all flow objects) ..... 23
  - 4.6 FlowFragment ..... 23
  - 4.7 Connection ..... 24
  - 4.8 Pin ..... 24
  - 4.9 Dialogue ..... 24
  - 4.10 DialogueFragment ..... 25
  - 4.11 Hub ..... 25
  - 4.12 Jump ..... 26
  - 4.13 Comment ..... 26
  - 4.14 Condition ..... 27
  - 4.15 Instruction ..... 27
  - 4.16 Documents (The top level system folder containing all documents) ..... 27
  - 4.17 DocumentsUserFolder ..... 28
  - 4.18 Document ..... 28
  - 4.19 TextObject ..... 28
  - 4.20 Entities (The top level system folder containing all entities) ..... 29
  - 4.21 EntitiesUserFolder ..... 29
  - 4.22 Entity ..... 29
  - 4.23 InvalidObjectReference ..... 30
  - 4.24 Journeys (The top level system folder containing all journey objects) ..... 30
  - 4.25 JourneysUserFolder ..... 30
  - 4.26 Journey ..... 31
  - 4.27 JourneyPoint ..... 31
  - 4.28 Locations (The top level system folder containing all location objects) ..... 31
  - 4.29 LocationsUserFolder ..... 32
  - 4.30 Location ..... 32
  - 4.31 LayerFolder (former name LocationGroup) ..... 33
  - 4.32 LocationImage ..... 33
  - 4.33 LocationText ..... 34
  - 4.34 Link ..... 35

4.35	Path.....	36
4.36	Spot.....	36
4.37	Zone.....	37
4.38	Anchor .....	38
4.39	Global Variables (The top level system folder containing all location objects) .....	38
4.40	Variable.....	38
4.41	VariableSet .....	39
5	Changes from articy:draft 1.x to 2.x.....	40

# 1 Articy query language

The articy query language is structured like the well known structured query language (SQL), but tailored to work best with the special structure of the data found in articy:draft.

Currently only the “SELECT” command is available and only literal values or references to a special contextual object are allowed on the right hand side of an operation.

When looking at the data types described in section 1.3 you will see that those are not common for SQL but make sense when working with articy:draft data.

Reserved words like “SELECT” or “WHERE” are handled in a case insensitive way. Object type names or system folder references can be used in a case insensitive manner, too.

Feature names and properties are considered to be case-sensitive which is in line with articy:draft conventions: Naming one feature field “Age” and naming another field “age” is allowed here and will define two distinct fields.

## 1.1 Using queries in the 'Search' tab of articy:draft

The query language can also be used from the 'Search' tab in the UI of articy:draft.

In this case, the “SELECT” and “FROM” clauses are not available, as they are implied. “TRANSFORM”, “GROUP”, “ORDER” and “LIMIT” clauses will be ignored.

The following example shows how to search for dialogue fragments with the speaker’s technical name being “Manfred” from inside the 'Search' tab:

```
ObjectType = DialogueFragment AND Speaker = Manfred
```

Please note that connections are omitted from the result set in this context.

## 1.2 Using queries in calculated reference strips

When using queries in calculated reference strips the results are filtered to only show items that make sense in this context. As of this, folders, the project node itself and connections are automatically omitted from the result.

## 1.3 Notation of values

This paragraph shows how to denote the different data types available in query functions or on the right hand side of an operation.

### 1.3.1 Logical values (booleans)

The literal names `true` and `false` can be used.

### 1.3.2 Numbers

Numbers can be formatted by using the default US number format without any additional literal (e.g. no “f” to denote floating point numbers). For example, `123` and `4.3` are valid numbers.

### 1.3.3 Text

Texts can be enclosed in single quotes (') or double quotes (").

To use single quotes inside a text that is enclosed in single quotes, add an additional single quote beforehand (e.g. 'It''s been a long time'). Using double quotes inside a text that is enclosed in double quotes, add a backslash beforehand (e.g. "He said \"Hello.\"").

All other characters can be used without any further escaping.

### 1.3.4 Colors

Colors can either be written in HTML rgb-notation (e.g. #FF0000) or by using a known name (e.g. yellow), if the color is part of the standard Microsoft Windows color list.

### 1.3.5 Enumerations

Values for enumerated types (called 'drop-down list' in articy:draft) can be written by using their numeric value or by using the value's technical name without quotes (e.g. Male).

Both variants can be displayed in the template design area of articy:draft.

### 1.3.6 Objects

Objects are referenced by their 64-Bit long ID (e.g. 0x01000001000010BC) or by their technical name, optionally prefixed with a dollar sign (e.g. \$Manfred).

The "\$" is required to force an identifier to be interpreted as a "technical name reference" and to distinguish it from other reserved identifiers. This can be helpful when using technical names that collide with reserved names like "Project". In that case "FROM Project" and "FROM \$Project" will lead to different results. The first will denote the "Project" folder, while the second one denotes an object with the technical name "Project". Section 1.4.5 covers additional escaping when working with reserved names.

### 1.3.7 Lists of objects

Lists are currently supported only when using the compare function. In that case the elements of the list are separated by a comma while each element itself follows the same rules as single, non-list elements.

## 1.4 Using a SELECT query

Queries are written similar to standard SQL queries. You typically start with a "SELECT \* FROM", followed by the folder which should be searched (e.g. "Project" for searching the whole project). Next you can refine the query by adding conditions with a "WHERE" clause, change how the result set is sorted with the "ORDER" clause and several other options used in SQL Queries.

The following is an overview of what is possible with select statements:

**SELECT \* FROM <Starting point>**

**[WHERE <Condition>]**

**[TRANSFORM <Transformation>]**

**[GROUP BY <Fieldname>]**

**[ORDER BY <Fieldname> (ASC|DESC) [, <Fieldname> (ASC|DESC)]\***

**[LIMIT <Count>]**



To clarify the use of select statements, have a look at the following examples:

```
SELECT * FROM Project WHERE LocationInfo.IsStartLocation
```

This query collects all objects that have a feature "LocationInfo" and whose property "IsStartLocation" is true.

```
SELECT * FROM Entities WHERE Color = red
```

This query collects all objects beneath the "Entities" folder which are colored red. This will not only return Entities, but also folders that are colored red.

To prevent folders from showing up you can add an additional filter:

```
SELECT * FROM Entities WHERE ObjectType = Entity AND Color = red
```

The following sections will explain the different parts of a select statement in more detail.

### 1.4.1 Starting point

The starting point refers to an object in the navigator tree where the search should start.

The value is either an object reference or one of the following predefined names that reflect the top level, static system folders in articy:draft.

Project - start from the root node of the navigator tree  
Flow  
Entities  
Assets  
Notes  
Journeys

If you want to search for e.g. an asset it is recommended to use "Assets" as starting point. This will speed up the search operation as only assets will be searched.

Please note that the 'Object Customization' and 'Settings' sub-trees are excluded from searches as these areas do not contain any project relevant content.

### 1.4.2 Where clause

The **WHERE** clause is an expression of Boolean terms that can be connected with "AND" or "OR" and can be grouped by using parenthesis.

Each term of the expression can either be

- The name of a field with data type boolean
- A term in the form <field> <operator> <value> (e.g. "color = red")
- The call of a query function that returns a boolean value ( e.g. compare( ... ) )

### 1.4.3 Fieldnames and properties

The fields of object base properties are written with one identifier (e.g. `DisplayName`) while properties of a feature need two identifiers separated with '.' (e.g. `CharacterProps.Gender`). In this case the first identifier is the technical name of the feature and the second name is the technical name of a property in that feature.

```
SELECT * FROM Flow WHERE DisplayName <> 'Manfred' AND CharacterProps.Gender = Male
```

### 1.4.4 Self

The keyword "self" can be used to refer to a special object from the application context. For example, if the query language is used as filter expression for calculated strips, "self" would denote the object containing the strip.

```
SELECT * FROM Flow WHERE ObjectType = DialogueFragment AND Speaker = Self
```

You can also refer to properties of "self" by dereferencing them via '->' or ':'. Both operators are equivalent:

```
SELECT * FROM Flow WHERE ObjectType = DialogueFragment AND Speaker = Self->Entity
SELECT * FROM Flow WHERE ObjectType = DialogueFragment AND Speaker = Self:Entity
```

As the "self" keyword refers to an object it can be used as a starting point, too:

```
SELECT * FROM Self WHERE ObjectType = DialogueFragment
```

### 1.4.5 Escaping of identifiers

When using technical names, features or properties with identifiers matching reserved words of the query language (e.g. a feature named "select") they have to be escaped.

This is done by enclosing the name in backticks (e.g. ``group`.`select`` if you have a feature "group" with a property "select").

### 1.4.6 Conditions (relational operators)

The set of supported operators depends on the data type of the field to compare.

To compare for equality within the query you can use either the SQL like "=" or the C-style "==". Same is true for inequality operators "<>" and "!=" and the usage of "AND" (&) or "OR" (|). The following context/operator combinations exist:

Object references, colors, enumerations, Booleans	=, <>
Date/time, numbers	=, <>, >=, <=, >, <
Text	=, <>, >=, <=, >, <, *=, EndsWith, =*, StartsWith, *, Contains

More complex compares can be done by using query functions (see section 0).

Compares on text fields are case insensitive by default. If you need a case-sensitive comparison, use the compare query function (see section 2.2.1).

```
SELECT * FROM Entities WHERE TemplateName =* 'Item'
```

This query returns all entities whose template name starts with 'Item', 'item', 'itEm', etc..

### 1.4.7 Transform Clause

The transform clause can be used to modify the result of your query by applying a function on each element of the result set. All supported transform functions are described in Section 2.4.

For example: You are interested in all speakers taking part in a conversation. In that case you would search for all dialog fragments in a conversation and transform them into the entity linked to their "Speaker" field. The function "UnquieReferences" will filter out any duplicates, so each speaker will be returned only once:

```
SELECT * FROM $Conversation WHERE ObjectType = DialogueFragment TRANSFORM  
UniqueReferences(Speaker)
```

### 1.4.8 Group clause

The group clause is used to sort the result set so that objects with the same value in a certain field are listed subsequently. The order between groups however is not defined.

Grouping can be helpful when your results can be broken down to only a few different groups. In that case sorting by group is much faster than doing a full sort by using the order clause.

The following example lists all assets grouped by their category (image, audio, etc.)

```
SELECT * FROM Assets GROUP BY AssetCategory
```

### 1.4.9 Order clause

The order clause is used to sort the result set according to one or more fields. You can choose wether to sort ascending (ASC) or descending (DESC), too.

The following example orders all entities by their Age (older persons first) and name (ordered from A to Z):

```
SELECT * FROM Entities ORDER BY Person.Age DESC, DisplayName ASC
```

When using the order clause in combination with the group clause, the ordering will be applied after the grouping.

### 1.4.10 Limit clause

The limit clause is used to limit the total number of results produced by the query.

This can be useful when e.g. searching for one specific entity, so the search can stop as soon as the entity is found:

```
SELECT * FROM Entities WHERE DisplayName = 'Manfred' LIMIT 1
```

## 1.5 Extended relational operators

Some data types (especially texts) provide more than the standard compare operations.

For those cases query functions can be used that require special operators as one of their parameters. The following table is a list of known operators and their meaning.

Operator names are recognized in any casing, but are written in mixed case for better readability.

The inverse operations can either be written as "!Ends" or "NOT Ends".

See the query function reference (section 2) for more information about which operators are allowed on what function.

2 Operator	Meaning
Eq, Equal	Same semantic as "="
Neq, !Equal	Same semantic as "!="
Lt, Less	Same semantic as "<"
Lte, LessOrEqual	Same semantic as "<="
Gt, Greater	Same semantic as ">"
Gte, GreaterOrEqual	Same semantic as ">="
Begins, BeginsWith, StartsWith	A list or text starts with the given value
!Begins, !BeginsWith, !StartsWith	A list or text does not start with the given value
Ends, EndsWith	A list or text ends with the given value
!Ends, !EndsWith	A list or text does not end with the given value
Contains	A list of values contains a given element or text
!Contains	A list of values does not contain a given element or text
AnyOf	The value to compare is one of the elements given
NoneOf	The value to compare is none of the elements given
AllOf	The value to compare has all of the given elements
Matches	String only: the value matches the given regular expression
!Matches	String only: the value does not match the given regular expression
IsSet	The property has a set value
!IsSet	The Property has not a set value
IsEmpty	Reference strip only: returns true if the strip does not have any elements
!IsEmpty	Reference strip only: returns true if the strip has any elements

The following example selects all dialogue fragments where the speaker is set:

```
SELECT * FROM Flow WHERE ObjectType = DialogueFragment AND Compare(Speaker, IsSet)
```

## 2.1 Object based query functions

Object based functions – as their name already implies – work directly on the properties of each object processed during a query. This includes editable properties as well as descriptive information like the claiming state or object type.

Most of these functions use an operator (see 1.5) as their first parameter, followed by a comma separated list of additional parameters required for that operator. The list of additional parameters may be empty, too.

### 2.1.1 IsObjectType

This query function filters objects by their type.

The following operators are supported:

```
Equal, Not Equal, AnyOf, NoneOf
```

In case of `Equal` or `Not equal` exactly one additional parameter must be provided.

All other operators allow a list of type names. Take a look at the [Articy.Api.ObjectType](#) enumeration for a list of possible values.

The following example filters all objects that are either an asset or an entity:

```
SELECT * FROM Project WHERE IsObjectType(AnyOf, Asset, Entity)
```

### 2.1.2 IsDescendantOf

This query function filters objects that are descendants of a given object or a given list of objects. The check will be started from the tested object “upwards” (using its parent property) until one of the given, potential parent objects is found.

The following example will return any objects that are part of a flow fragment called “MainStory”.

```
SELECT * FROM Flow WHERE IsDescendantOf($MainStory)
```

In this example the property “Parent” is used to search for the flow fragment “MainStory”. If another object reference property is to be used instead of “Parent”, it can be specified as an optional, second parameter:

```
SELECT * FROM Flow WHERE IsDescendantOf(Self, Speaker)
```

This example checks if the object referenced by “self” (e.g. the object holding a reference strip) can be reached from a tested object when traversing the project via the property “Speaker”.

In order to limit the distance between the filtered candidate object and the assumed, further optional parameters for maximum and minimum distance can be added. Note that you will always have to specify the traversed field in this case:

```
SELECT * FROM Flow WHERE IsDescendantOf($MainStory, Parent, 4, 2)
```

This example checks if “MainStory” can be reached from the tested object over a minimum of 2 or a maximum of 4 steps.

### 2.1.3 IsAncestorOf

This function is similar to IsDescendantOf, but changes the order of testing. It checks if the given object is able to reach the tested object by traversing “upwards” (using its parent property).

So the following example will return all objects that are parents of “SideStory”:

```
SELECT * FROM Flow WHERE IsAncestorOf($SideStory)
```

IsAncestorOf supports the same additional parameters as IsDescendantOf for using a different property or limiting the traversal count:

```
SELECT * FROM Flow WHERE IsAncestorOf(self, Speaker, 3, 2)
```

The above example checks if the tested object can be reached from “self” when traversing the project via the property Speaker over a minimum of 2 or a maximum of 3 steps.

### 2.1.4 ClaimedBy

This query function filters objects by using the claiming state of the partition an object belongs to. The following operators are supported:

Me, AnyOne, NoOne, AnyOf, NoneOf

In case of `Equal` or `Not equal` exactly one additional parameter must be provided. All other parameters support one or more additional parameters. The identifier “me” can be used to refer to the current user.

Examples:

```
SELECT * FROM Project ClaimedBy(Me)
```

```
SELECT * FROM Project ClaimedBy(AnyOf, Me, 'Peter', 0x01000001000010bc)
```

### 2.1.5 Template

This query function filters objects by testing on the template of an object.

The following operators are supported:

IsSet, NOT IsSet, Equal, NOT Equal, AnyOf, NoneOf

The following example returns all entities with any template set:

```
SELECT * FROM Entity WHERE Template(IsSet)
```

The following example returns all entities that implement either the “MainCharacterTemplate” or the “BossTemplate”.

```
SELECT * FROM Entity Template(AnyOf, MainCharacterTemplate, BossTemplate)
```

### 2.1.6 Feature

This query function filters objects by checking on features used by the template assigned to an object. The following operators are supported:

AnyOf, AllOf, NoneOf

The next example returns a list of all objects implementing a template containing either the “InteractionZone” feature or the “LocationInfo” feature:

```
SELECT * FROM Project WHERE Feature(AnyOf, InteractionZone, LocationInfo)
```

### 2.1.7 HasConflict

This query function searches if an object contains a conflict. Its functionality is the same as it can be found in the conflict navigator tab in the articy:draft UI. The following names match the checkboxes that can be found there. Using other expressions in the query the check can be restricted to a limited set of objects rather than checking all objects as it is done in the UI.

When used without parameters any type of conflict is checked.

If you want to search for specific conflicts only use a comma separated list of the following conflict type names:

Name	Description
References	Searches for broken references.
FeatureValues	Searches for any feature value that does not obey its check constraints.
TechnicalNames	Searches for duplicate technical names for user given technical names.
Asset	Searches for Assets that were not properly imported or have failed thumbnailing.

The following example returns a list of all objects of the Flow tree that have a conflict:

```
SELECT * FROM Flow WHERE HasConflict()
```

This example returns a list of all Asset objects that are not in the proper state:

```
SELECT * FROM Assets WHERE HasConflict( Asset )
```

## 2.2 Property based query functions

Property functions require a field name passed as their first parameter. The second parameter is the name of a compare operator, followed by values that are defined by the context of the function.

### 2.2.1 Compare

The compare function is the Swiss army knife for comparisons. Text, object references and object lists are supported.

#### 2.2.1.1 Compare for Text

As an exception to the other property functions a specialized function called `AnyText()` is supported to be used instead of a field. This function expands to a list of all string fields collected from the tested object before it is passed to the compare operator.

After the second parameter denoting the compare operator, a literal string is expected to be used for comparison. A second literal string may be provided containing additional compare options.

Currently there is only one option ('s') to compare case sensitive texts:

```
SELECT * FROM Project WHERE Compare(DisplayName, StartsWith, 'Move', 's')
```

The above example will search for display names that start with the word “move”, ignoring case.

When using the `AnyOf`, `NoneOf` or `AllOf` operator the compare string (3<sup>rd</sup> parameter) is splitted into single words on each space character and that list is used. For example:

```
SELECT * FROM Project WHERE Compare(AnyText(), AnyOf, 'Manfred Hamster Bomb')
```

This searches every text field of an object for any of the words “Manfred”, “Hamster” or “Bomb”. When using the “Matches” operator, regular expressions can be used:

```
SELECT * FROM Project WHERE Compare(Text, Matches, 'Talk to (Manfred|Hamster)')
```

This example checks if the “Text” property of an object matches the regular expression “Talk to (Manfred|Hamster)”.

### 2.2.1.2 Compare for references

The following operators are supported:

IsSet, NOT IsSet, Equal, NOT Equal, AnyOf, NoneOf

After passing the operator a list of object references is expected, either in denoted by id or by technical name:

```
SELECT * FROM Project WHERE Compare(Speaker, Equal, $Ntt_Hamster)
```

This will test if the “Speaker” field (as present in dialog fragments) equals an object with the technical name “Ntt\_Hamster”.

### 2.2.1.3 Compare for reference lists

The following operators are supported:

`Equal`, `NOT Equal`, `AnyOf`, `NoneOf`, `AllOf`, `IsEmpty`, `NOT IsEmpty`

After passing the operator a list of object references is expected, either in denoted by id or by technical name.

## 2.2.2 Function calls supporting “Self”

By default, functions are evaluated by comparing against the object that is currently processed. Alternatively, query functions can be evaluated with by using the “Self” object, too:

```
SELECT * FROM Flow WHERE Self->Compare(Speaker, Equal, $Ntt_Hamster)
```

Assume that “Self” is a dialogue fragment. In that case this query will either return all objects from the flow (if the speaker of “Self” is \$Ntt\_Hamster) or no objects (in all other cases). This can be useful in combination with `ClaimedBy(Me)` or other special cases.

## 2.3 Handling of NULL and non-existing values

Most queries will at some point try to reference a field that is not present on all objects (e.g. when testing for “Speaker” on all flow objects, which is only present in dialogue fragments). This is especially true when working with feature properties.

Objects not containing the field to be tested will always be excluded from the result.

When looking at the compare operations themselves, objects not containing a requested field will force the result to be false for positive tests like `=` or `AnyOf` or true for negative tests like `<>` or `NoneOf`. Compare operators like `>` or `<` will always evaluate to false.

The same set of rules applies when testing feature properties that do not define a default value and where no value was set in the object’s property sheet.

## 2.4 Transform functions

Transform functions modify the result set of a query by returning objects based on the original result. This can be helpful when looking for objects that are e.g. referenced by other objects.

### 2.4.1 UniqueReferences

This function accepts the name of an object reference or object strip property as its only parameter. The result set will be transformed to return all objects that were encountered to be referenced by the passed property exactly once.

The following example will return all speakers that participate in the “MainDialog”:

```
SELECT * FROM $MainDialog TRANSFORM UniqueReferences(Speaker)
```

The following example will return all unique objects that are used within the attachment strips of the flow objects:

```
SELECT * FROM Flow TRANSFORM UniqueReferences(Attachments)
```

### 2.4.2 UniqueAncestors

This function accepts a list of objects and behaves similar to the function “IsAncestorOf”. The result set will be replaced by those objects that are ancestors of the objects passed to this function. Every ancestor will only be returned once.

When having a hierarchy like this: “FlowFragment01” -> “Dialog01” -> “DialogFragment01” the following example will return “Flow” (as it is the parent of “FlowFragment01”), “FlowFragment01” (as it is the parent of “Dialog01”) and “Dialog01” (as it is the parent of “DialogFragment01”):

```
SELECT * FROM $FlowFragment01 TRANSFORM UniqueAncestors()
```

The first parameter can be used to restrict the ancestor object type. Assume we want to exclude everything but FlowFragments from the result. This could be achieved by using:

```
SELECT * FROM $FlowFragment01 TRANSFORM UniqueAncestors(FlowFragment)
```

This would return only “FlowFragment01”.

As with IsAscendantOf, another property instead of “Parent” can be used by supplying a second parameter:

```
SELECT * FROM $FlowFragment01 TRANSFORM UniqueAncestors(Entity, Speaker)
```

This query will return all entities that are linked to the “speaker” property.

Finally, the search distance in the hierarchy can be supplied with upper and lower bounds:

```
SELECT * FROM $FlowFragment01 TRANSFORM UniqueAncestors(FlowFragment, Parent, 3, 2)
```

### 2.4.3 UniqueParents

This function is an alias of `UniqueAncestors()`.

### 3 Examples

This section shall give you some practical examples for using the articy:draft query language. Assume we have an entity with technical name "Manfred". Additionally, there is a dialogue called "Dialogue1" containing a dialogue fragment in which Manfred is speaking. Let's also assume our entity "Manfred" uses the template "Person" with a feature called "BasicProperties" containing the property "Age".

The following query examples are meant to be used to fill a calculated reference strip in a flow fragment template. As of this, the "Self" object reference will always point to some flow fragment.

Let's start with a simple one: Listing all objects in the current project:

```
SELECT * FROM Project
```

To get all entities and entity user folders, we could use:

```
SELECT * FROM Project WHERE ObjectType = Entity OR ObjectType = EntitiesUserFolder
```

Instead, we could also use the IsObjectType query function:

```
SELECT * FROM Entities WHERE IsObjectType(AnyOf, Entity, EntitiesUserFolder)
```

In order to get all dialogue fragments in which Manfred is talking, we simply use:

```
SELECT * FROM Flow WHERE ObjectType = DialogueFragment AND Speaker = $Manfred
```

To restrict this to only those dialogue lines being somewhere underneath the flow fragment which is referenced by "Self", we have to extend the query in this way:

```
SELECT * FROM Flow WHERE ObjectType = DialogueFragment AND Speaker = $Manfred AND IsDescendantOf(Self)
```

To make the query faster and more compact, we use the knowledge, that our "Self" object is a flow fragment and we are only interested in objects that are contained in that flow fragment. We restrict the search to start at the "Self" object to avoid traversing the whole flow.

```
SELECT * FROM Self WHERE ObjectType = DialogueFragment AND Speaker = $Manfred
```

To find all Flow-Fragments which have no references in their attachment strip use:

```
SELECT * FROM Self WHERE ObjectType = FlowFragment AND Compare( Attachments, IsEmpty )
```

To find all dialogue lines in which the words "apple" or "peach" occur, we use:

```
SELECT * FROM Flow WHERE ObjectType = DialogueFragment AND Compare(Text, AnyOf, 'apple peach')
```

To find all occurrences of "apple" and "peach" in any text based property we can use AnyText() instead of referencing the "text" field:

```
SELECT * FROM Flow WHERE ObjectType = DialogueFragment AND Compare(AnyText(), AnyOf, 'apple peach')
```

We are also interested in all entities that use the “Person” template and whose age is at least 25:

```
SELECT * FROM Entities WHERE TemplateName = 'Person' AND BasicProperties.Age >= 25
```

We decided to add one more feature to the template “Person” called “Workflow” containing a Boolean property called “Tested”. Let's find all Person entities which have not yet been tested, sorted by their DisplayName:

```
SELECT * FROM Entities WHERE TemplateName = 'Person' AND Workflow.Tested = false ORDER BY DisplayName ASC
```

Finally we want to gather the attached object references from all untested entities:

```
SELECT * FROM Entities WHERE TemplateName = 'Person' AND Workflow.Tested = false TRANSFORM UniqueReferences(Attachments)
```

## 4 Overview of Object Properties

In this section we list all properties available for use in query expressions. Some properties have different names in the user interface of articy:draft than in the query language:

User interface	Query language
Description (not in dialogue fragments)	Text
Full text (of dialogue fragments)	Text

The following table gives context on some non-obvious properties available in queries:

Property	Description
DisplayName	This is the name of an object as shown in the navigator
FilePath	The file path to the Asset file relative to the project's asset directory
AbsoluteFilePath	The absolute file path to the Asset file
OriginalSource	The original location from which the Asset was imported
FileExtension	The file extension of the Asset
AssetCategory	A value from: Image, Video, Audio, Document or Misc
AssetState	Current processing status, values are: NeedsImport, Importing, Imported, NeedsUpdate, Updating, Updated, NeedsAdding, Adding, Added, NeedsUpdateFromRepository, UpdatingFromRepository, UpdatedFromRepository, Deleting, Deleted, Reimporting, Reimported, DeleteFailed, Ready, ImportFailed, UpdateFromRepositoryFailed, ReimportFailed, AddingFailed, UpdateFailed
FileType	File type as reported by Windows

### 4.1 Project

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	✗	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
Children	✓	✗	ObjectList	List<Articy.Api.ObjectProxy>
DisplayName	✗	✗	Text	System.String
TechnicalName	✗	✗	Text	System.String
Color	✗	✗	Color	System.Windows.Media.Color
ExternalId	✗	✗	Text	System.String
ShortId	✗	✗	Integer	System.UInt32

## 4.2 Assets (The top level system folder containing all assets)

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	✗	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
Children	✓	✗	ObjectList	List<Articy.Api.ObjectProxy>
DisplayName	✗	✗	Text	System.String
TechnicalName	✗	✗	Text	System.String
Color	✗	✗	Color	System.Windows.Media.Color
Text	✗	✗	Text	System.String
ExternalId	✗	✗	Text	System.String
ShortId	✗	✗	Integer	System.UInt32

## 4.3 AssetsUserFolder

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	✗	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
Children	✓	✗	ObjectList	List<Articy.Api.ObjectProxy>
DisplayName	✗	✗	Text	System.String
TechnicalName	✗	✗	Text	System.String
Color	✗	✗	Color	System.Windows.Media.Color
Text	✗	✗	Text	System.String
ExternalId	✗	✗	Text	System.String
ShortId	✗	✗	Integer	System.UInt32

## 4.4 Asset

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	✗	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
TemplateName	✓	✓	Text	System.String
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
DisplayName	✗	✗	Text	System.String
TechnicalName	✗	✗	Text	System.String
Color	✗	✗	Color	System.Windows.Media.Color
Text	✗	✗	Text	System.String
ExternalId	✗	✗	Text	System.String
ShortId	✗	✗	Integer	System.UInt32
AssetFilename	✓	✗	Text	System.String
OriginalSource	✓	✗	Text	System.String
FileSize	✓	✓	Integer	System.Int64
FileType	✓	✓	Text	System.String
FileExtension	✓	✓	Text	System.String
FilePath	✓	✗	Text	System.String

Name	Read only	Calc.	Data Type	.NET Type
AbsoluteFilePath	✓	✓	Text	System.String
AssetCategory	✓	✓	Enum	System.Int32
AssetState	✓	✓	Enum	System.Int32

#### 4.5 Flow (The top level flow fragment containing all flow objects)

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	✗	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
Children	✓	✗	ObjectList	List<Articy.Api.ObjectProxy>
DisplayName	✗	✗	Text	System.String
TechnicalName	✗	✗	Text	System.String
Color	✗	✗	Color	System.Windows.Media.Color
Text	✗	✗	Text	System.String
ExternalId	✗	✗	Text	System.String
ShortId	✗	✗	Integer	System.UInt32

#### 4.6 FlowFragment

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	✗	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
TemplateName	✓	✓	Text	System.String
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
Children	✓	✗	ObjectList	List<Articy.Api.ObjectProxy>
InputPins	✓	✗	ObjectList	List<Articy.Api.ObjectProxy>
OutputPins	✓	✗	ObjectList	List<Articy.Api.ObjectProxy>
DisplayName	✗	✗	Text	System.String
TechnicalName	✗	✗	Text	System.String
Color	✗	✗	Color	System.Windows.Media.Color
Text	✗	✗	Text	System.String
ExternalId	✗	✗	Text	System.String
ShortId	✗	✗	Integer	System.UInt32
Attachments	✗	✗	ObjectList	List<Articy.Api.ObjectProxy>
PreviewImageAsset	✗	✗	ObjectReference	Articy.Api.ObjectProxy
PosX	✗	✗	Double	System.Double
PosY	✗	✗	Double	System.Double
ZIndex	✗	✗	Integer	System.Int32
Width	✗	✗	Double	System.Double
Height	✗	✗	Double	System.Double

## 4.7 Connection

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	✗	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
DisplayName	✗	✗	Text	System.String
TechnicalName	✗	✗	Text	System.String
Source	✓	✗	ObjectReference	Articy.Api.ObjectProxy
Target	✓	✗	ObjectReference	Articy.Api.ObjectProxy
SourcePin	✓	✗	ObjectReference	Articy.Api.ObjectProxy
TargetPin	✓	✗	ObjectReference	Articy.Api.ObjectProxy
Label	✗	✗	Text	System.String
ShowLabel	✗	✗	Boolean	System.Boolean
Color	✗	✗	Color	System.Windows.Media.Color
ShortId	✗	✗	Integer	System.UInt32

## 4.8 Pin

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	✗	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
DisplayName	✗	✗	Text	System.String
PinIndex	✗	✗	Integer	System.Int32
PinType	✗	✗	Enum	System.Int32

## 4.9 Dialogue

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	✗	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
TemplateName	✓	✓	Text	System.String
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
Children	✓	✗	ObjectList	List<Articy.Api.ObjectProxy>
InputPins	✓	✗	ObjectList	List<Articy.Api.ObjectProxy>
OutputPins	✓	✗	ObjectList	List<Articy.Api.ObjectProxy>
DisplayName	✗	✗	Text	System.String
TechnicalName	✗	✗	Text	System.String
Color	✗	✗	Color	System.Windows.Media.Color
Text	✗	✗	Text	System.String
ExternalId	✗	✗	Text	System.String
ShortId	✗	✗	Integer	System.UInt32
Attachments	✗	✗	ObjectList	List<Articy.Api.ObjectProxy>
PreviewImageAsset	✗	✗	ObjectReference	Articy.Api.ObjectProxy
PosX	✗	✗	Double	System.Double
PosY	✗	✗	Double	System.Double

Name	Read only	Calc.	Data Type	.NET Type
ZIndex	x	x	Integer	System.Int32
Width	x	x	Double	System.Double
Height	x	x	Double	System.Double

#### 4.10 DialogueFragment

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	x	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
TemplateName	✓	✓	Text	System.String
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
InputPins	✓	x	ObjectList	List<Articy.Api.ObjectProxy>
OutputPins	✓	x	ObjectList	List<Articy.Api.ObjectProxy>
DisplayName	x	x	Text	System.String
TechnicalName	x	x	Text	System.String
Color	x	x	Color	System.Windows.Media.Color
Text	x	x	Text	System.String
ExternalId	x	x	Text	System.String
ShortId	x	x	Integer	System.UInt32
MenuText	x	x	Text	System.String
StageDirections	x	x	Text	System.String
Speaker	x	x	ObjectReference	Articy.Api.ObjectProxy
PosX	x	x	Double	System.Double
PosY	x	x	Double	System.Double
ZIndex	x	x	Integer	System.Int32
Width	x	x	Double	System.Double
Height	x	x	Double	System.Double

#### 4.11 Hub

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	x	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
TemplateName	✓	✓	Text	System.String
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
InputPins	✓	x	ObjectList	List<Articy.Api.ObjectProxy>
OutputPins	✓	x	ObjectList	List<Articy.Api.ObjectProxy>
DisplayName	x	x	Text	System.String
TechnicalName	x	x	Text	System.String
Color	x	x	Color	System.Windows.Media.Color
Text	x	x	Text	System.String
ExternalId	x	x	Text	System.String
ShortId	x	x	Integer	System.UInt32
PosX	x	x	Double	System.Double
PosY	x	x	Double	System.Double
ZIndex	x	x	Integer	System.Int32

Name	Read only	Calc.	Data Type	.NET Type
Width	x	x	Double	System.Double
Height	x	x	Double	System.Double

## 4.12 Jump

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	x	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
TemplateName	✓	✓	Text	System.String
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
InputPins	✓	x	ObjectList	List<Articy.Api.ObjectProxy>
OutputPins	✓	x	ObjectList	List<Articy.Api.ObjectProxy>
DisplayName	x	x	Text	System.String
TechnicalName	x	x	Text	System.String
Color	x	x	Color	System.Windows.Media.Color
Text	x	x	Text	System.String
ExternalId	x	x	Text	System.String
ShortId	x	x	Integer	System.UInt32
Target	x	x	ObjectReference	Articy.Api.ObjectProxy
TargetPin	x	x	ObjectReference	Articy.Api.ObjectProxy
PosX	x	x	Double	System.Double
PosY	x	x	Double	System.Double
ZIndex	x	x	Integer	System.Int32
Width	x	x	Double	System.Double
Height	x	x	Double	System.Double

## 4.13 Comment

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	x	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
DisplayName	x	x	Text	System.String
Color	x	x	Color	System.Windows.Media.Color
Text	x	x	Text	System.String
ExternalId	x	x	Text	System.String
ShortId	x	x	Integer	System.UInt32
CreatedBy	✓	x	Text	System.String
CreatedOn	✓	x	DateTime	System.DateTime
PosX	x	x	Double	System.Double
PosY	x	x	Double	System.Double
ZIndex	x	x	Integer	System.Int32
Width	x	x	Double	System.Double
Height	x	x	Double	System.Double

## 4.14 Condition

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	✗	Integer	System.UInt64
DisplayName	✗	✗	Text	System.String
TechnicalName	✗	✗	Text	System.String
DisplayNameOverride	✗	✗	Text	System.String
ShowCalculatedDisplayName	✗	✗	Boolean	System.Boolean
Color	✗	✗	Color	System.Windows.Media.Color
Text	✗	✗	Text	System.String
ExternalId	✗	✗	Text	System.String
ShortId	✗	✗	Integer	System.UInt32
ObjectType	✓	✓	Enum	System.Int32
TemplateName	✓	✓	Text	System.String
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
Expression	✗	✗	Text	System.String
PosX	✗	✗	Double	System.Double
PosY	✗	✗	Double	System.Double
ZIndex	✗	✗	Integer	System.Int32
Width	✗	✗	Double	System.Double
Height	✗	✗	Double	System.Double

## 4.15 Instruction

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	✗	Integer	System.UInt64
DisplayName	✗	✗	Text	System.String
TechnicalName	✗	✗	Text	System.String
DisplayNameOverride	✗	✗	Text	System.String
ShowCalculatedDisplayName	✗	✗	Boolean	System.Boolean
Color	✗	✗	Color	System.Windows.Media.Color
Text	✗	✗	Text	System.String
ExternalId	✗	✗	Text	System.String
ShortId	✗	✗	Integer	System.UInt32
ObjectType	✓	✓	Enum	System.Int32
TemplateName	✓	✓	Text	System.String
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
Expression	✗	✗	Text	System.String

## 4.16 Documents (The top level system folder containing all documents)

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	✗	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
Children	✓	✗	ObjectList	List<Articy.Api.ObjectProxy>

Name	Read only	Calc.	Data Type	.NET Type
DisplayName	x	x	Text	System.String
TechnicalName	x	x	Text	System.String
Color	x	x	Color	System.Windows.Media.Color
Text	x	x	Text	System.String
ExternalId	x	x	Text	System.String
ShortId	x	x	Integer	System.UInt32

#### 4.17 DocumentsUserFolder

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	x	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
Children	✓	x	ObjectList	List<Articy.Api.ObjectProxy>
DisplayName	x	x	Text	System.String
TechnicalName	x	x	Text	System.String
Color	x	x	Color	System.Windows.Media.Color
Text	x	x	Text	System.String
ExternalId	x	x	Text	System.String
ShortId	x	x	Integer	System.UInt32

#### 4.18 Document

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	x	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
TemplateName	✓	✓	Text	System.String
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
Children	✓	x	ObjectList	List<Articy.Api.ObjectProxy>
DisplayName	x	x	Text	System.String
TechnicalName	x	x	Text	System.String
Color	x	x	Color	System.Windows.Media.Color
Text	x	x	Text	System.String
ExternalId	x	x	Text	System.String
ShortId	x	x	Integer	System.UInt32

#### 4.19 TextObject

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	x	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
TemplateName	✓	✓	Text	System.String
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
Children	✓	x	ObjectList	List<Articy.Api.ObjectProxy>

Name	Read only	Calc.	Data Type	.NET Type
DisplayName	x	x	Text	System.String
TechnicalName	x	x	Text	System.String
Color	x	x	Color	System.Windows.Media.Color
Text	x	x	Text	System.String
ExternalId	x	x	Text	System.String
ShortId	x	x	Integer	System.UInt32
Attachments	x	x	ObjectList	List<Articy.Api.ObjectProxy>
PreviewImageAsset	x	x	ObjectReference	Articy.Api.ObjectProxy

## 4.20 Entities (The top level system folder containing all entities)

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	x	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
Children	✓	x	ObjectList	List<Articy.Api.ObjectProxy>
DisplayName	x	x	Text	System.String
TechnicalName	x	x	Text	System.String
Color	x	x	Color	System.Windows.Media.Color
Text	x	x	Text	System.String
ExternalId	x	x	Text	System.String
ShortId	x	x	Integer	System.UInt32

## 4.21 EntitiesUserFolder

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	x	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
Children	✓	x	ObjectList	List<Articy.Api.ObjectProxy>
DisplayName	x	x	Text	System.String
TechnicalName	x	x	Text	System.String
Color	x	x	Color	System.Windows.Media.Color
Text	x	x	Text	System.String
ExternalId	x	x	Text	System.String
ShortId	x	x	Integer	System.UInt32

## 4.22 Entity

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	x	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
TemplateName	✓	✓	Text	System.String
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy

Name	Read only	Calc.	Data Type	.NET Type
DisplayName	x	x	Text	System.String
TechnicalName	x	x	Text	System.String
Color	x	x	Color	System.Windows.Media.Color
Text	x	x	Text	System.String
ExternalId	x	x	Text	System.String
ShortId	x	x	Integer	System.UInt32
Attachments	x	x	ObjectList	List<Articy.Api.ObjectProxy>
PreviewImageAsset	x	x	ObjectReference	Articy.Api.ObjectProxy

### 4.23 InvalidObjectReference

Name	Read only	Calc.	Data Type	.NET Type
ObjectType	✓	✓	Enum	System.Int32

### 4.24 Journeys (The top level system folder containing all journey objects)

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	x	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
Children	✓	x	ObjectList	List<Articy.Api.ObjectProxy>
DisplayName	x	x	Text	System.String
TechnicalName	x	x	Text	System.String
Color	x	x	Color	System.Windows.Media.Color
Text	x	x	Text	System.String
ExternalId	x	x	Text	System.String
ShortId	x	x	Integer	System.UInt32

### 4.25 JourneysUserFolder

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	x	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
Children	✓	x	ObjectList	List<Articy.Api.ObjectProxy>
DisplayName	x	x	Text	System.String
TechnicalName	x	x	Text	System.String
Color	x	x	Color	System.Windows.Media.Color
Text	x	x	Text	System.String
ExternalId	x	x	Text	System.String
ShortId	x	x	Integer	System.UInt32

## 4.26 Journey

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	✗	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
TemplateName	✓	✓	Text	System.String
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
DisplayName	✗	✗	Text	System.String
TechnicalName	✗	✗	Text	System.String
Color	✗	✗	Color	System.Windows.Media.Color
Text	✗	✗	Text	System.String
ExternalId	✗	✗	Text	System.String
ShortId	✗	✗	Integer	System.UInt32
Attachments	✗	✗	ObjectList	List<Articy.Api.ObjectProxy>
PreviewImageAsset	✗	✗	ObjectReference	Articy.Api.ObjectProxy
JourneyPoints	✓	✗	ObjectList	List<Articy.Api.ObjectProxy>
JourneyState	✓	✗	Enum	System.Int32

## 4.27 JourneyPoint

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	✗	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
TemplateName	✓	✓	Text	System.String
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
DisplayName	✗	✗	Text	System.String
TechnicalName	✗	✗	Text	System.String
Color	✗	✗	Color	System.Windows.Media.Color
Text	✗	✗	Text	System.String
ExternalId	✗	✗	Text	System.String
ShortId	✗	✗	Integer	System.UInt32
Attachments	✗	✗	ObjectList	List<Articy.Api.ObjectProxy>
PreviewImageAsset	✗	✗	ObjectReference	Articy.Api.ObjectProxy
JourneyPointType	✓	✗	Enum	System.Int32
ReachedByActionType	✓	✗	Enum	System.Int32
IdxInJourney	✓	✗	Integer	System.Int32

## 4.28 Locations (The top level system folder containing all location objects)

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	✗	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
Children	✓	✗	ObjectList	List<Articy.Api.ObjectProxy>
DisplayName	✗	✗	Text	System.String
TechnicalName	✗	✗	Text	System.String

Name	Read only	Calc.	Data Type	.NET Type
Color	x	x	Color	System.Windows.Media.Color
Text	x	x	Text	System.String
ExternalId	x	x	Text	System.String
ShortId	x	x	Integer	System.UInt32

#### 4.29 LocationsUserFolder

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	x	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
Children	✓	x	ObjectList	List<Articy.Api.ObjectProxy>
DisplayName	x	x	Text	System.String
TechnicalName	x	x	Text	System.String
Color	x	x	Color	System.Windows.Media.Color
Text	x	x	Text	System.String
ExternalId	x	x	Text	System.String
ShortId	x	x	Integer	System.UInt32

#### 4.30 Location

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	x	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
TemplateName	✓	✓	Text	System.String
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
Children	✓	x	ObjectList	List<Articy.Api.ObjectProxy>
DisplayName	x	x	Text	System.String
TechnicalName	x	x	Text	System.String
Color	x	x	Color	System.Windows.Media.Color
Text	x	x	Text	System.String
ExternalId	x	x	Text	System.String
ShortId	x	x	Integer	System.UInt32
Attachments	x	x	ObjectList	List<Articy.Api.ObjectProxy>
PreviewImageAsset	x	x	ObjectReference	Articy.Api.ObjectProxy
BackgroundOpacity	x	x	Float	System.Double
BackgroundWidth	✓	x	Float	System.Double
BackgroundHeight	✓	x	Float	System.Double
KeepAspectRatio	✓	x	Boolean	System.Boolean
ShowDisplayNameForLink	x	x	Boolean	System.Boolean
ShowDisplayNameForSpot	x	x	Boolean	System.Boolean
ShowDisplayNameForZone	x	x	Boolean	System.Boolean
ShowDisplayNameForPath	x	x	Boolean	System.Boolean
ShowDisplayNameForImage	x	x	Boolean	System.Boolean
DisplayNameColorForLink	x	x	Color	System.Windows.Media.Color
DisplayNameColorForSpot	x	x	Color	System.Windows.Media.Color

Name	Read only	Calc.	Data Type	.NET Type
DisplayNameColorForZone	x	x	Color	System.Windows.Media.Color
DisplayNameColorForPath	x	x	Color	System.Windows.Media.Color
DisplayNameColorForImage	x	x	Color	System.Windows.Media.Color
DisplayNameSizeForLink	x	x	Integer	System.Int32
DisplayNameSizeForSpot	x	x	Integer	System.Int32
DisplayNameSizeForZone	x	x	Integer	System.Int32
DisplayNameSizeForPath	x	x	Integer	System.Int32
DisplayNameSizeForImage	x	x	Integer	System.Int32
DropShadowForLink	x	x	Boolean	System.Boolean
DropShadowForSpot	x	x	Boolean	System.Boolean
DropShadowForZone	x	x	Boolean	System.Boolean
DropShadowForPath	x	x	Boolean	System.Boolean
DropShadowForImage	x	x	Boolean	System.Boolean
DropShadowForText	x	x	Boolean	System.Boolean

### 4.31 LayerFolder (former name LocationGroup)

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	x	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
Children	✓	x	ObjectList	List<Articy.Api.ObjectProxy>
DisplayName	x	x	Text	System.String
TechnicalName	x	x	Text	System.String
Color	x	x	Color	System.Windows.Media.Color
Text	x	x	Text	System.String
ExternalId	x	x	Text	System.String
ShortId	x	x	Integer	System.UInt32
Visibility	x	x	Enum	System.Int32
Selectability	x	x	Enum	System.Int32

### 4.32 LocationImage

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	x	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
Children	✓	x	ObjectList	List<Articy.Api.ObjectProxy>
DisplayName	x	x	Text	System.String
TechnicalName	x	x	Text	System.String
Color	x	x	Color	System.Windows.Media.Color
Text	x	x	Text	System.String
ExternalId	x	x	Text	System.String
ShortId	x	x	Integer	System.UInt32
Visibility	x	x	Enum	System.Int32
Selectability	x	x	Enum	System.Int32

Name	Read only	Calc.	Data Type	.NET Type
ShapeType	✓	✗	Enum	System.Int32
PreviewImageAsset	✗	✗	ObjectReference	Articy.Api.ObjectProxy
DisplayNamePosX	✗	✗	Double	System.Double
DisplayNamePosY	✗	✗	Double	System.Double
AutoDisplayNamePosition	✗	✗	Boolean	System.Boolean
TransformRotation	✗	✗	Double	System.Double
TransformXAxisX	✗	✗	Double	System.Double
TransformXAxisY	✗	✗	Double	System.Double
TransformYAxisX	✗	✗	Double	System.Double
TransformYAxisY	✗	✗	Double	System.Double
TransformPivotX	✗	✗	Double	System.Double
TransformPivotY	✗	✗	Double	System.Double
OutlineColor	✗	✗	Color	System.Windows.Media.Color
OutlineSize	✗	✗	Double	System.Double
OutlineStyle	✗	✗	Enum	System.Int32
FillColor	✗	✗	Color	System.Windows.Media.Color
ShowDisplayName	✗	✗	Boolean	System.Boolean
DisplayNameColor	✗	✗	Color	System.Windows.Media.Color
DisplayNameSize	✗	✗	Integer	System.Int32
DropShadow	✗	✗	Boolean	System.Boolean
ImageOpacity	✗	✗	Double	System.Double
ImageAsset	✗	✗	ObjectReference	Articy.Api.ObjectProxy
Url	✓	✓	String	System.String
TemplateName	✓	✓	Text	System.String
PosX	✗	✗	Double	System.Double
PosY	✗	✗	Double	System.Double
Anchors	✓	✗	ObjectList	List<Articy.Api.ObjectProxy>

### 4.33 LocationText

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	✗	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
DisplayName	✗	✗	Text	System.String
TechnicalName	✗	✗	Text	System.String
Color	✗	✗	Color	System.Windows.Media.Color
Text	✗	✗	Text	System.String
ExternalId	✗	✗	Text	System.String
ShortId	✗	✗	Integer	System.UInt32
Visibility	✗	✗	Enum	System.Int32
Selectability	✗	✗	Enum	System.Int32
ShapeType	✓	✗	Enum	System.Int32
PreviewImageAsset	✗	✗	ObjectReference	Articy.Api.ObjectProxy
DisplayNamePosX	✗	✗	Double	System.Double
DisplayNamePosY	✗	✗	Double	System.Double
AutoDisplayNamePosition	✗	✗	Boolean	System.Boolean
TransformRotation	✗	✗	Double	System.Double

Name	Read only	Calc.	Data Type	.NET Type
TransformXAxisX	x	x	Double	System.Double
TransformXAxisY	x	x	Double	System.Double
TransformYAxisX	x	x	Double	System.Double
TransformYAxisY	x	x	Double	System.Double
TransformPivotX	x	x	Double	System.Double
TransformPivotY	x	x	Double	System.Double
OutlineColor	x	x	Color	System.Windows.Media.Color
OutlineSize	x	x	Double	System.Double
OutlineStyle	x	x	Enum	System.Int32
FillColor	x	x	Color	System.Windows.Media.Color
ShowDisplayName	x	x	Boolean	System.Boolean
DisplayNameColor	x	x	Color	System.Windows.Media.Color
DisplayNameSize	x	x	Integer	System.Int32
DropShadow	x	x	Boolean	System.Boolean
Url	✓	✓	String	System.String
TemplateName	✓	✓	Text	System.String
PosX	x	x	Double	System.Double
PosY	x	x	Double	System.Double
ForcedWidth	x	x	Double	System.Double
ForcedHeight	x	x	Double	System.Double
Anchors	✓	x	ObjectList	List<Articy.Api.ObjectProxy>

### 4.34 Link

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	x	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
TemplateName	✓	✓	Text	System.String
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
DisplayName	x	x	Text	System.String
TechnicalName	x	x	Text	System.String
Color	x	x	Color	System.Windows.Media.Color
Text	x	x	Text	System.String
ExternalId	x	x	Text	System.String
ShortId	x	x	Integer	System.UInt32
Target	x	x	ObjectReference	Articy.Api.ObjectProxy
Visibility	x	x	Enum	System.Int32
Selectability	x	x	Enum	System.Int32
LinkStyleType	x	x	Enum	System.Int32
LinkStyleSize	x	x	Enum	System.Int32
ShowDisplayName	x	x	Boolean	System.Boolean
DisplayNameColor	x	x	Color	System.Windows.Media.Color
DisplayNameSize	x	x	Enum	System.Int32
DropShadow	x	x	Boolean	System.Boolean

### 4.35 Path

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	✗	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
TemplateName	✓	✓	Text	System.String
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
DisplayName	✗	✗	Text	System.String
TechnicalName	✗	✗	Text	System.String
Color	✗	✗	Color	System.Windows.Media.Color
Text	✗	✗	Text	System.String
ExternalId	✗	✗	Text	System.String
ShortId	✗	✗	Integer	System.UInt32
PreviewImageAsset	✗	✗	ObjectReference	Articy.Api.ObjectProxy
ShapeType	✓	✗	Enum	System.Int32
Visibility	✗	✗	Enum	System.Int32
Selectability	✗	✗	Enum	System.Int32
DisplayNamePosX	✗	✗	Double	System.Double
DisplayNamePosY	✗	✗	Double	System.Double
AutoDisplayNamePosition	✗	✗	Boolean	System.Boolean
TransformRotation	✗	✗	Double	System.Double
TransformXAxisX	✗	✗	Double	System.Double
TransformXAxisY	✗	✗	Double	System.Double
TransformYAxisX	✗	✗	Double	System.Double
TransformYAxisY	✗	✗	Double	System.Double
TransformPivotX	✗	✗	Double	System.Double
TransformPivotY	✗	✗	Double	System.Double
OutlineColor	✗	✗	Color	System.Windows.Media.Color
OutlineSize	✗	✗	Double	System.Double
OutlineStyle	✗	✗	Enum	System.Int32
FillColor	✗	✗	Color	System.Windows.Media.Color
ShowDisplayName	✗	✗	Boolean	System.Boolean
DisplayNameColor	✗	✗	Color	System.Windows.Media.Color
DisplayNameSize	✗	✗	Integer	System.Int32
DropShadow	✗	✗	Boolean	System.Boolean
ImageOpacity	✗	✗	Double	System.Double
ImageAsset	✗	✗	ObjectReference	Articy.Api.ObjectProxy
Url	✓	✓	String	System.String
PosX	✗	✗	Double	System.Double
PosY	✗	✗	Double	System.Double
StartCap	✗	✗	Enum	System.Int32
EndCap	✗	✗	Enum	System.Int32
LineWidth	✗	✗	Double	System.Double

### 4.36 Spot

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	✗	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32

Name	Read only	Calc.	Data Type	.NET Type
TemplateName	✓	✓	Text	System.String
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
DisplayName	✗	✗	Text	System.String
TechnicalName	✗	✗	Text	System.String
Color	✗	✗	Color	System.Windows.Media.Color
Text	✗	✗	Text	System.String
ExternalId	✗	✗	Text	System.String
ShortId	✗	✗	Integer	System.UInt32
Visibility	✗	✗	Enum	System.Int32
Selectability	✗	✗	Enum	System.Int32
OutlineColor	✗	✗	Color	System.Windows.Media.Color
OutlineSize	✗	✗	Double	System.Double
OutlineStyle	✗	✗	Enum	System.Int32
ShowDisplayName	✗	✗	Boolean	System.Boolean
DisplayNameColor	✗	✗	Color	System.Windows.Media.Color
DisplayNameSize	✗	✗	Integer	System.Int32
DropShadow	✗	✗	Boolean	System.Boolean
SpotStyleType	✗	✗	Enum	System.Int32
SpotStyleSize	✗	✗	Enum	System.Int32
PreviewImageAsset	✗	✗	ObjectReference	Articy.Api.ObjectProxy
FillColor	✗	✗	Color	System.Windows.Media.Color

### 4.37 Zone

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	✗	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
TemplateName	✓	✓	Text	System.String
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
DisplayName	✗	✗	Text	System.String
TechnicalName	✗	✗	Text	System.String
Color	✗	✗	Color	System.Windows.Media.Color
Text	✗	✗	Text	System.String
ExternalId	✗	✗	Text	System.String
ShortId	✗	✗	Integer	System.UInt32
PreviewImageAsset	✗	✗	ObjectReference	Articy.Api.ObjectProxy
ShapeType	✓	✗	Enum	System.Int32
Visibility	✗	✗	Enum	System.Int32
Selectability	✗	✗	Enum	System.Int32
OutlineColor	✗	✗	Color	System.Windows.Media.Color
OutlineSize	✗	✗	Double	System.Double
OutlineStyle	✗	✗	Enum	System.Int32
ShowDisplayName	✗	✗	Boolean	System.Boolean
DisplayNameColor	✗	✗	Color	System.Windows.Media.Color
DisplayNameSize	✗	✗	Integer	System.Int32
DropShadow	✗	✗	Boolean	System.Boolean
FillColor	✗	✗	Color	System.Windows.Media.Color
PosX	✗	✗	Double	System.Double

Name	Read only	Calc.	Data Type	.NET Type
PosY	x	x	Double	System.Double
DisplayNamePosX	x	x	Double	System.Double
DisplayNamePosY	x	x	Double	System.Double
AutoDisplayNamePosition	x	x	Boolean	System.Boolean
TransformRotation	x	x	Double	System.Double
TransformXAxisX	x	x	Double	System.Double
TransformXAxisY	x	x	Double	System.Double
TransformYAxisX	x	x	Double	System.Double
TransformYAxisY	x	x	Double	System.Double
TransformPivotX	x	x	Double	System.Double
TransformPivotY	x	x	Double	System.Double

### 4.38 Anchor

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	x	Integer	System.UInt64
ObjectType	✓	✓	Enum	System.Int32
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
Color	x	x	Color	System.Windows.Media.Color
Size	x	x	Enum	System.Int32
PosX	x	x	Double	System.Double
PosY	x	x	Double	System.Double

### 4.39 Global Variables (The top level system folder containing all location objects)

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	x	Integer	System.UInt64
DisplayName	x	x	Text	System.String
Color	x	x	Color	System.Windows.Media.Color
Text	x	x	Text	System.String
ExternalId	x	x	Text	System.String
ShortId	x	x	Integer	System.UInt32
ObjectType	✓	✓	Enum	System.Int32
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy

### 4.40 Variable

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	x	Integer	System.UInt64
DisplayName	x	x	Text	System.String
ScriptingName	x	x	Text	System.String
Description	x	x	Text	System.String

Name	Read only	Calc.	Data Type	.NET Type
DataType	✓	✗	Enum	System.Int32
ObjectType	✓	✓	Enum	System.Int32
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy
DefaultValue	✓	✓	Text	System.String
BooleanValue	✗	✗	Boolean	System.Boolean
NumericValue	✗	✗	Integer	System.Double

#### 4.41 VariableSet

Name	Read only	Calc.	Data Type	.NET Type
Id	✓	✗	Integer	System.UInt64
DisplayName	✗	✗	Text	System.String
ScriptingName	✗	✗	Text	System.String
Text	✗	✗	Text	System.String
Variables	✓	✗	ObjectList	System.Collections.Generic. List<Articy.Api.ObjectProxy>
ObjectType	✓	✓	Enum	System.Int32
Parent	✓	✓	ObjectReference	Articy.Api.ObjectProxy

## 5 Changes from articy:draft 1.x to 2.x

For articy 2, we changed the data type of object IDs from 128 bit Guids to 64 bit unsigned integers. So all references to the usage of object IDs like `0x01000001000010Bc` in this document substitute to Guids like `{ccfa19c5-b056-4759-817c-7c78deb551e1}` when you are using an articy:draft release 1.x, which is every version released before 2.0.

When projects created with articy:draft 1.x are opened with articy:draft 2, they are converted to the new project format. In order to enable you to relate old object IDs to new ones, the file `idmap.xml` is created in the project's directory containing each object ID in the old and new format:

```
<?xml version='1.0' encoding='utf-8' standalone='yes' ?>
<ProjectConversionInfo TargetVersion="47" CreatorTool="articy:draft" CreatorVersion="2.1.0.0">
  <ObjectIdMapping OriginalGuid="c4c77c6a-5489-4e34-b809-31599e9a50aa" NewId="0x030000007992BBC5"/>
  <ObjectIdMapping OriginalGuid="65ec65d5-8aeb-4b4a-aab1-524d877bfc99" NewId="0x03000000FA082513"/>
  <ObjectIdMapping OriginalGuid="2e15a712-2d1d-4048-ba75-59e162a0ba8d" NewId="0x0300000002BE5FD7"/>
  <ObjectIdMapping OriginalGuid="0608dfa1-a5af-4baf-aaa8-97430d6a83df" NewId="0x03000000D1B3B8A9"/>
  <ObjectIdMapping OriginalGuid="25d62ca0-cea9-4613-84b5-90708c9f580f" NewId="0x030000001C0DC801"/>
  [...]
</ProjectConversionInfo>
```