bp OSDU Data Governance- Key Principles & Standards

Lineage & Ancestry Capture (incl. Activity Model & Business Decisions) (v1)

Document Ownership

Owner	Andrew Flack
Contributors	Chris Hough
	Andrew Kerr
	Paul Stapleton
	Thibault Uzu
	Greg Harris
	Chris Rose

Document Revisions – Change log

Date	Name	Change Comments
01/11/2024	Andrew Kerr	Document creation

Document Overview

This document outlines the OSDU platform capabilities that support data lineage, ancestry & provenance, in addition to how data items are grouped as part of repeatable activities and/or related to ultimate business outcomes and key decisions.

The goal of this standards is to provide guidance on the use of data ancestry, direct & indirect data lineage, and the Activity Model (including the specialised "Business Decisions" Activity Template).

Commented [CH1]: Do we need to include this particular schema kind? I know it's a specialised Activity Template, but others wouldn't now an this could be confusing

Commented [AB2R1]: Updated to be more explicit - only reason to leave it is it might be of particular interest to Subsurface users as a particular high-value use-case to explore)

Key principles include:

- Lineage Definition & Usage (direct & indirect relationships)
- Ancestry Definition & Usage (and it's relation to Legal Tag inheritance)
- Complex lineage capture within the Activity Model (capturing standardised & repeatable business processes)
- Relation of data to key Business Decisions

Contents

1. Introduction 3
2. Lineage & Ancestry 4
   2.1 Definitions 4
      Derivatives 4
      Ancestry 4
      Lineage 4
   2.2 Ancestry Usage 4
      Ancestry Use-Case Example 5
   2.3 Lineage Assertions Usage 5
      Lineage Use-Case Examples 6
3. Activity Model & Business Decisions 7
   3.1 Definitions 7
      Activity Model 7
      Activity Template 7
      Activity Template Arc 7
      Business Decisions 7
   3.2 Activity Model, Activity Templates & Activity Template Arc Usage 7
      Activity Model 7
      Activity Templates 8
      Activity Template Arcs 9
      Activity Use-Case Examples 10
   3.4 Business Decisions Usage 11
4. Mandatory Relationship Capturing 12
5. Related Principles & Standards 12

Appendix 1 – Relevant OSDU Forum Standards Documents (for Reference) 13

1. Introduction

Consistency around how and when we capture lineage and ancestry information is critical to leveraging the maximum value from the OSDU Data Platform. The OSDU Data Platform offers bp the opportunity to capture key relationships between data records, enabling us to track the providence of data, provide information on where and how data has been derived, and ultimately to tie the data used to make critical business decisions directly to the decision outcomes.

The OSDU Data Platform provides multiple ways to capture providence/lineage information, which may be used in combination or individually:

- Ancestry provides information on the Parent records from which a data record was derived, and it's dependence on the parent Legal Tag information
- Lineage Assertions enable direct or indirect relationships to be defined at the record level (for Work Product Components)
- Activity Model provides the means to define more complex lineage relationships, providing full providence and process history for any data record (Master or WPC)
- Business Decisions provide the means to capture information on the data records used to reach a particular business outcome

Rules around what ancestry or lineage information needs to be captured within the OSDU data platform should be defined on a use-case or data kind basis, by the appropriate data owner, and enforced where possible (to ensure consistency across systems).

Commented [CH3]: The "OSDU Data Platform" is generally capital letters

Commented [CH4]: Above you say Providence here you say lineage, is that intentional?

Commented [AB5R4]: Added both

Commented [CH6]: Ancestry is also the name of particular property `Ancestry`. Which itself is simply an array of `LegalParents`. My point is that rather just a concept, it is a specific property within an OSDU record.

FYI - It is applicable to all group types too

Commented [AB7R6]: I feel this is expanded on in more detail further into the document

Commented [CH8]: A key aspect here is that L.A. are only applicable to WPCs. They can point to any object/kind, but only a WPC group type can do the pointing. It is also designed to only point to the Ancestors, not the children. The L.A. simply says I was born from record X,Y,Z and no process history is included - that's the roles of the Activity Model.

Commented [AB9R8]: NOTE - I've added a qualifier here and will explicitly state this later in the document

Commented [CH10]: The Activity Model (A.M) is model, whereas the ancestry and L.A. are properties within a schema. The goal of A.M is to pick up the slack from the L.A. and provide full provenance and process history. The A.M can be used by any group type, unlike the L.A. which is only applicable to WPCs

Commented [AB11R10]: Added clarity here

2. Lineage & Ancestry

2.1 Definitions

Derivatives

In the context of the OSDU data platform, the term "derivative data" is data that has been derived from primary data sources.

Often when ingesting data into the platform, it is the raw data itself. In these scenarios, bp ***may*** associate a single Legal Tag with this data.

However, in the case when the data to be ingested comes from multiple sources, it is the case of derivative data.

Example use-cases:

- A bp practitioner uses multiple OSDU records to create a new interpreted record
- A bp practitioner runs an algorithm over some seismic data to process it and create a new attribute volume

Ancestry

- Applies to ALL OSDU records
- Property: ancestry.parents[] (in AbstractSystemProperties schema fragment)
- An array of none, one or many entity references of 'direct parents' (id:version references) in the data platform, which mark the current record as a derivative. In contrast to other relationships, the source record version is required.

Lineage

- Applies to all Work Product Component (WPC) records (not applied to Master Data)
- Property: data.LineageAssertions[] (in AbstractWorkProductComponent schema fragment)
- This is lineage in it's most basic form and provides the ability to define the relationship between a data record and one or more related data records. Each relationship can be set as "Direct", "Indirect" and "Reference".

2.2 Ancestry Usage

The purpose of data Ancestry is primarily to support **legal** lineage and entitlements, by providing information on the record & version of all directly related data from which the record was derived.

The Ancestry schema fragment is available for **all** group types. During record creation or update, the ancestry.parents[] relationships are used to collect the legal tags from the sources and aggregate them in the legal.legaltags[] array.

As a consequence, should e.g., one or more of the legal tags of the source data expire, the **access to the derivatives is also terminated**.

*Note - this is **optional** to use WHERE APPROPRIATE – examples exist where bp has **DIFFERENT** legal obligations for derivatives vs original source data – **therefore** you would NOT want to use ancestry capability in these cases.*

When creating Records that represent derivative data, the following must be assigned:

- The Record Id and version of all the Records that are the ***direct parents*** of the new derivative (added to the ancestry section)
- The Alpha-2 country code of where the derivative was ***created***

Commented [HC12]: Data Platform

Commented [HC13]: This would be the more common use case. Another common term maybe processed or interpreted

Commented [CH14]: Might be worth explicitly saying that it applies to all OSDU records

Commented [AB15R14]: Done - even if stated below - makes it clear in the definition

Commented [CH16]: Might be worth explicitly saying that it applies only to WPCs

Commented [AB17R16]: Done - even if stated below - makes it clear in the definition

Commented [CH18]: Cool. Sorry 😉

If populating "ancestry", the Legal Tag values should be populated, based on the parents, which is then checked against the Legal Tag service.

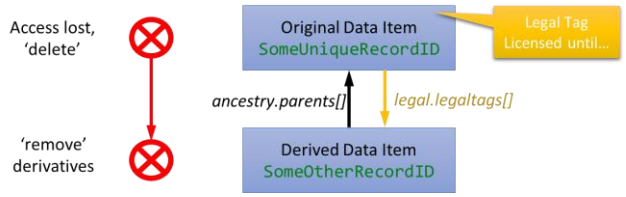
Commented [CH19]: I'm confused about the intent here. The schema can hold the value and that value is checked against the E&O (legal tag) service

e.g.

```
"legal" :{
"otherRelevantDataCountries": ["US"]
//the physical location of where the derivative was created
 },
"ancestry" :{
"parents": ["osdu:id:1:version", "osdu:id:2:version"]
//the record ids and versions of the Records this derivative was created from
}
```

The impact of this Ancestry dependency is outlined in the **Entitlements & Legal Tag Management (Security & Compliance) Standard.**

Ancestry Use-Case Example




```
"ancestry": {
  "parents": [
    "partition-id:group-type--IndividualType:SomeUniqueRecordID:1562066009929332"
  ]
}
```

2.3 Lineage Assertions Usage

Lineage Assertions provides basic lineage information in the form of **simple object tracking** and history of transformations.

Lineage Assertions is **limited** in it's complexity:

- Scope is limited to **Work Product Components (WPC), only**
- It does not capture methodology(ies) used, parameters used, parameter values used or any further context behind ***how*** the derived or related data was generated from source.
- Each relationship can be set, simply, as **"Direct", "Indirect" and "Reference"**.

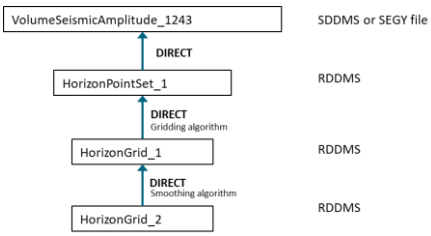
Commented [CH20]: Great. Thanks!

See Activity Model & Business Decisions for advance 'lineage' & workflow capture capability.

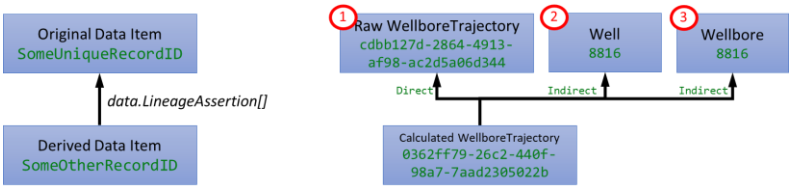
However, even though simple, Lineage Assertions can be **very powerful in supporting simple relationship use-cases and graph visualization**. It allows us to track the "ancestor" from where a resource is derived by capturing the record ID and version of all the Records that are the direct parents of the new derivative, in addition to any *indirect* relationships, such as a related well or wellbore.

Lineage Use-Case Examples

Simple direct parent relationship, captured via Lineage Assertions:



More complex direct & indirect relationships, captured via Lineage Assertions




```
"LineageAssertions": [
  {
1 "ID": "partition_id:work-product-component--WellboreTrajectory:cdbb127d-2864-4913-af98-ac2d5a06d344:1621068020029332",
  "LineageRelationshipType": "partition_id:reference-data--LineageRelationshipType:Direct:"
  },
  {
2 "ID": "partition_id:master-data--Well:8816:1621065020029332",
  "LineageRelationshipType": "partition_id:reference-data--LineageRelationshipType:Indirect:"
  },
  {
3 "ID": "partition-id:master-data--Wellbore:8816:1621059020029332",
  "LineageRelationshipType": "partition_id:reference-data--LineageRelationshipType:Indirect:"
  }
]
```

3. Activity Model & Business Decisions

3.1 Definitions

Activity Model

The Activity Model is an OSDU platform concept that allows the capture of more complex data lineage information.

It enables full traceability of granular operations (including direct/indirect lineage information, providing links to data resources as both as input and output parameters used to create new objects or revise existing data objects).

Activity Templates and TemplateArcs schemas facilitate standardised & repeatable business process, using the activity model concept.

Activity Template

Activity templates provide a standardization mechanism for workflows, by providing "definable templates", corresponding to workflow steps.

Activity Template Arc

TemplateArcs provide the ability to create standardised ***groups*** of Activities and ***nested*** Activities (or Activity Templates) which make up ***end-to-end workflows or sub-workflows***.

Business Decisions

Business Decisions are a specialized Activity Template, that provide a means to relate key data sets, lineage, ancestry and child activities to the direct business decisions/value that they contribute towards, **via a parent "Project"**

- "a record of a technical or business decision, capturing the context of the decision"

3.2 Activity Model, Activity Templates & Activity Template Arc Usage

Activity Model

The Activity Model platform concept allows the capture of more complex data lineage information and facilitates the standardization of repeatable business processes (e.g. the generation of a synthetic seismic log):

- Full traceability of granular operations (e.g. the parameters, controls, inputs & outputs of an operation), capturing the lineage & parameters used to create new, or revise existing, objects
- Provides direct links to the data resources used for both the input and output to an operation

"Activity" is supported by <u>all</u> group types, including master data (unlike Lineage Assertions)

Commented [CH21]: I see things a little different here. I see the Activity Model being the whole system of a Template, Activity Instance and Activity Arc. With those three objects being distinct and different things.

So the Activity Model is the system
The Activity Template is as you said
The Activity Instance is the WPC and
The Activity Arc can join Activity Templates together (FYI a template by itself can do that too)

Happy to talk it over if that helps

Commented [AB22R21]: Understood - changed the wording to reflect the above - thanks much 😊

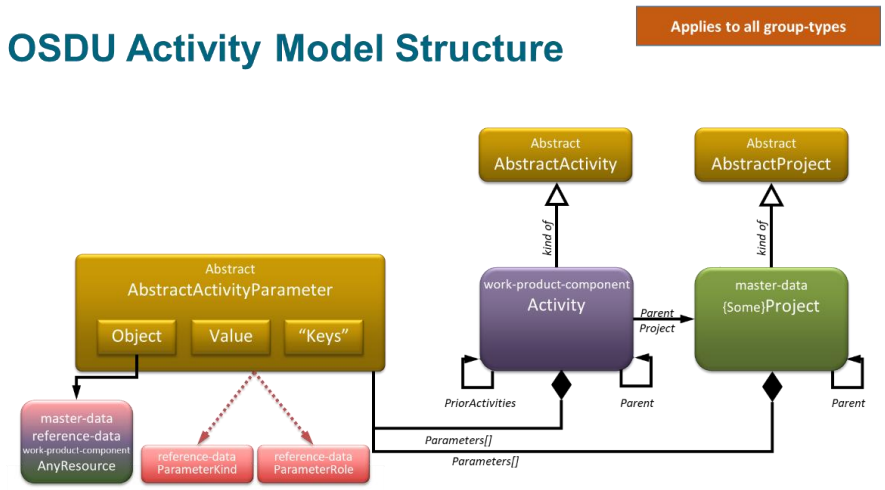
Commented [CH23]: I'm still convinced that this schema should be part of this document

Commented [AB24R23]: Unsure what you mean? As in we should expand on this definition later in the document? Happy to do so, if so

Commented [CH25]: See above

The Activity Model describes **actual parameter values used to perform the activity**:

- Activity Information
  - Reference to an "Parent" Activity (enabling nested activities, or groups of activities that form part of a wider workflow activity)
  - Reference to a standardized "Activity Template" (see Activity Templates)
- Multiple Parameters
  - Associated to individual Parameter descriptors
  - Can capture an array or a (multi-index) map
    - May use keys to associate with complex output (Example: volume per unit, per facies, per lease…)
  - Software specifications, Activity States & more…
- Contains the actual parameter **values**
  - Contains the actual variable values used in the activity or operation, for input into an Activity Template (see Activity Templates)



Activity Templates

The ActivityTemplate schema is Master data, and provides a standardization mechanism for workflows, by providing "definable templates", corresponding to workflow steps.

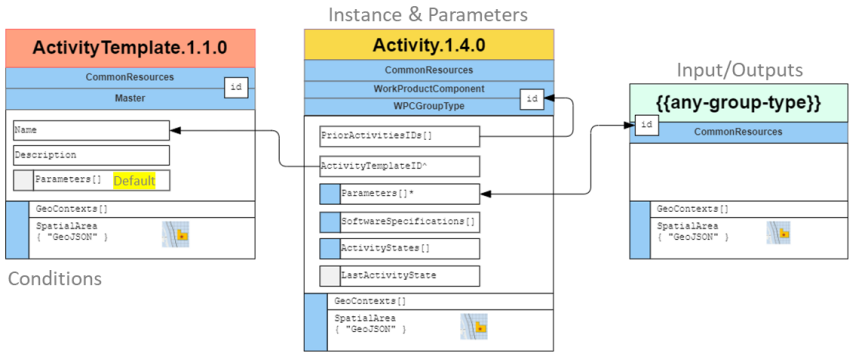
The Schema provides an overall description of an activity or steps in a process:

- Activity Descriptor
  - Name of the Activity Template
- Multiple Parameter descriptor
  - Name of each parameter
  - Content Type: Quantity, Resource (Object), String, TimeIndex

- Number of occurrence allowed: MinOccurs, MaxOccurs
- Limits on allowed Object Types and/or other Constraints
- Usage: In, out, inout
- Any default values

- Sets the conditions of the referenced Activity instance (WPC)
  - e.g. min (1) & max (-1) occurrences, etc

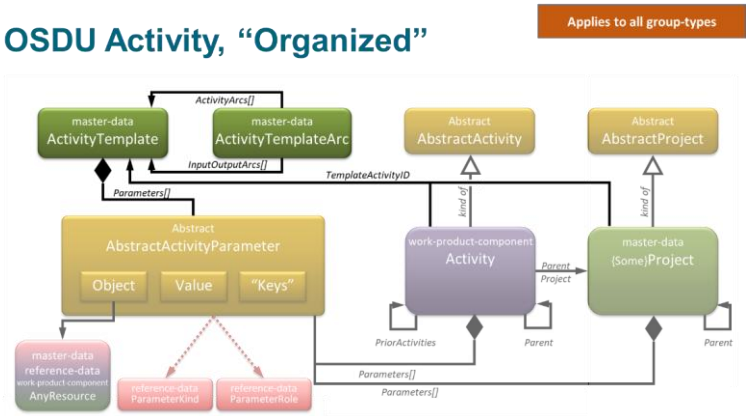
Example relation between an Activity Template, associated activity & any inputs/outputs:



Activity Template Arcs

TemplateArcs allow Activities (and activity templates) to be **nested and/or grouped to form complex workflows and sub-workflows** (rather than purely linear sequences)

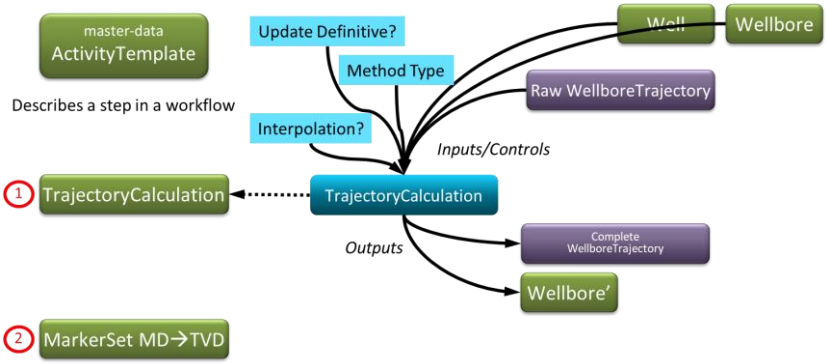
The ActivityTemplateArc provides a means to link multiple activity templates together within a complete end to end workflow:



Activity Use-Case Examples

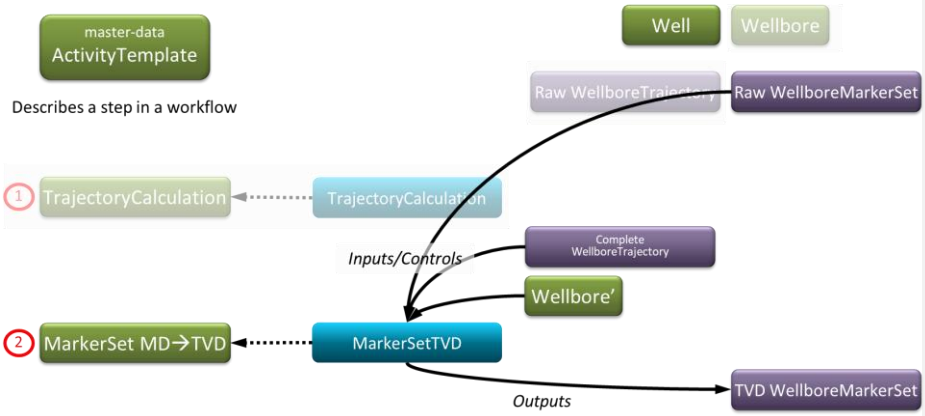
For example, when creating a final, definitive survey, there are many parent data-types that feed into this calculation as inputs, and a number of particular outputs (including the update of existing data kinds with new information).

In this example we can leverage a standardized set of processes (e.g. method types used for interpolation, updates to definitive, etc.), which can be stored as a repeatable "TrajectoryCalculation" **ActivityTemplate** (green). This activity template references the specific instance of this activity as a "TrajectoryCalculation" **Activity** (blue), in which we capture the specific **inputs/controls** for this particular occurrence of the activity (e.g. the Well, Wellbore and Raw WellboreTrajectory), along with reference to the activity outputs (e.g. the Complete WellboreTrajectory and a new updated version of the Wellbore record with reference to the new Definitive trajectory).



Additional complexity can then be added by including further, associated, activities. For example, the outputs from the TrajectoryCalculation activity (above) can be used as inputs for a new activity: "MarkerSet MD-TVD".

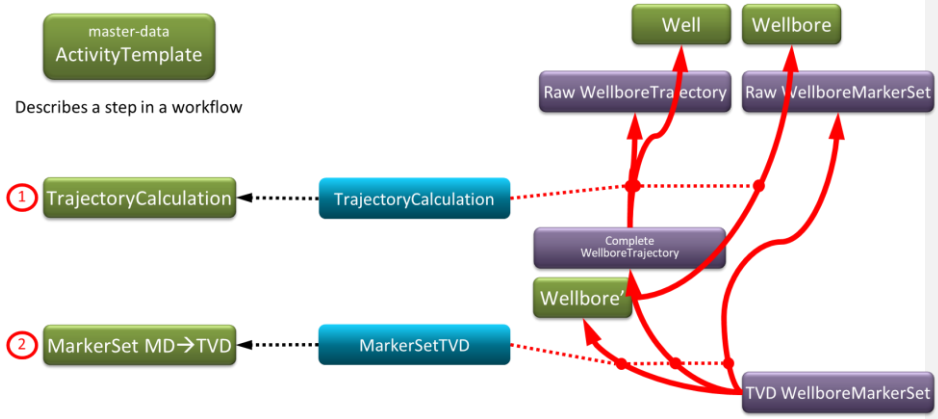
Again, a repeatable **ActivityTemplate** (green) can be defined to specify repeatable methods and a specific **Activity** WPC can capture the **inputs/controls** (e.g. Raw WellboreTrajectory, Complete WellboreTrajectory, updated Wellbore record) and the **output** TVD WellboreMarkerSet.



Commented [CH26]: Might not need to capture it, but a another benefit and use case of the model is the ability to output and ensemble of results and conduct sensitivities analysis. For example, using different "Method Types" in this example and seeing how the result or outcomes differs

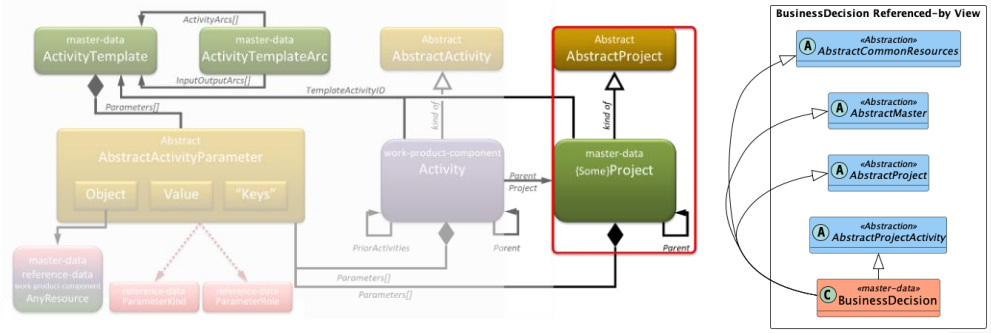
Commented [AB27R26]: Chose not to include - may expand on additional use-case examples in the wiki? One for the future

The end result is a highly detailed set of lineage and ancestry relations between each data type, the direct, indirect and derived parent records and the actual methods and activity instance used to create each new record output.



3.4 Business Decisions Usage

The Business Decisions Master Data schema, embedded within an Activity Template, provides a means to relate key data sets, lineage, ancestry and activities to **the direct business decisions/value that they contribute towards**. This can be done via a parent "Project".



The Business Decisions schema can be used to capture the decision outcomes, key dates, associated risks, key personas & accountable individuals (owners, contributors, etc.), alternatives, and much more.

This ultimately enables us to tie activities and activity outputs to the key business outcomes that they contribute towards.

4. Mandatory Relationship Capturing

As a **minimum** at bp, we should aim to capture the **basic direct and indirect parental lineage and ancestry information for any derived record.** This should be captured within the LineageAssertions and Ancestry (ancestry.parents[]) schema fragments.

Ownership should be taken by domain data owners to **define standardised, repeatable ActivityTemplates** (where applicable), and as such **mandate more complex lineage capture for specific, high-value business processes** by leveraging the Activity WPC schema.

Additionally, business owners should strive to capture specific, high-value business decisions within the OSDU data platform, that **are related back to the underlying data and/or activities that underpin these decisions** by leveraging the BusinessDecisions Master Data schema.

5. Related Principles & Standards

The section above relates to topics also covered in the below Principles & Standards:

- ***Entitlements & Legal Tag Management (Security & Compliance) Standard***
    - How derivatives relate to LegalTag implementation & Ancestry

Appendix 1 – Relevant OSDU Forum Standards Documents (for Reference)

OSDU Forum Documentation URL - https://osduforum.org/getting-started/osdu-documentation/#

Key Documents (links to bp internal copies – refer to above URL to check latest updates):

- OSDU Reference Architecture
- OSDU Schema Usage Guide
- OSDU Technical Standard
- OSDU System Concept