

bp OSDU Data Governance- Key Principles & Standards Schema Standards (v1)

Document Ownership

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

Document Revisions – Change log

Date	Name	Change Comments
09/07/2024	Andrew Kerr	Document creation
10/07/2024	Chris Hough	Incorporated comments/feedback from Chris Hough
17/07/2024	Andrew Kerr	Added links to OSDU Forum Standards Documentation
31/10/2024	Andrew Kerr	Tags, Custom SideCar Schema, Preferred Schema Versions
06/12/2024	Andrew Kerr	Further expansion on Custom Side-Car Schema
16/12/2024	Andrew Kerr	Clarification on Custom (& side-car) schema naming standards, namespaces & impact on schema upgrades

Document Overview

This document outlines the need for a managed process and set of standards for OSDU schema management and custom schema authoring at bp. It emphasizes the roles and responsibilities of Data Owners/Data Product Owners in data mapping, schema extensions, and custom schema authoring.

The goal of this standards is to ensure data integrity, consistency, and alignment with OSDU standards while maintaining flexibility to meet bp's specific needs. It provides key OSDU Schema definitions, including principles around the adoption and usage of OSDU Well Known Schemas (WKS), guidance on how to extend the WKS, and when bp may choose to author custom schemas, along with the key considerations, risks and implications of each option.

Key principles include:

- The adoption of Well Known Schemas (WKS) where possible (including the importance of defining minimum mandatory attributes for each schema kind)
- When / how bp may extend schemas, author custom schemas and manage schema upgrades
- Avoid long-term reliance on custom schemas to minimize technical debt and incompatibility risks (including complex custom Domain Data Management Services (DDMS))
- Engagement with / contribution to the OSDU Forum for schema development to reduce technical debt and increase data standards adoption over time

Contents

1.	Introduction	3
2.	Schema Usage	4
2.1	Definitions	4
	Well Known Schemas (WKS)	4
	Extension Properties	4
	Custom “Sidecar” Schema Extensions	4
	Data “Tags”	4
2.2	Schema Usage Principles	5
2.3	Risks & Considerations	6
3.	Authoring OSDU Custom Schemas	6
3.1	Custom Metadata Schemas	6
	bp-Preferred Approach to Custom Schema Usage	7
3.2	Custom Sidecar Schema (custom extensions to WKS)	9
	bp-Preferred Approach to Custom Side-Car Usage	10
	Worked Example (ERD)	12
	Worked Example (Schema)	12
	Considerations when Authoring Custom Side-Car Schema	13
3.3	Custom DDMS	14
3.4	Additional Risks & Considerations	14
4.	Minimum Required Attributes	15
4.1	Related Principles & Standards	15
5.	Schema Upgrades	16
5.1	Preferred Schema Version	16
	Example reference-data--PreferredSchemaVersion record	17
5.2	Custom Schema Upgrades	17
5.3	Custom Side-Car Schema Upgrades	17
	Appendix 1 – Relevant OSDU Forum Standards Documents (for Reference)	19

1. Introduction

It has been recognized that we need a **managed process/set of standards around OSDU schema management and custom schema authoring.**

Regular engagement (and ultimately ownership) of each data type in OSDU is expected from the bp Data Product Line, where existing data products are transitioned to OSDU. This includes the change management and requirements prioritization of their existing Digital Product **Customers** (e.g. ADI, Falcon, OWA, etc.).

As such, final decisions on data mapping, schema extensions or authoring (especially where data mapping is not clear, or there may be potential data loss in adoption of OSDU WKS), should be owned by the respective **Data Product Owner(s).**

- **NOTE - It should be acknowledged that OSDU data modelling is a particular and specialized skill set. This work needs to be carried out by suitably trained and skilled individuals with standardized deliverables such as documentation and worked examples as best practice.**

Where OSDU schemas exist but there is **no current equivalent bp data product** (e.g. when the OSDU Forum is further ahead than bp within a domain or sub-domain), these schemas should **be transitioned to the best appropriate bp data product lines ownership** (or a new data product line established, where necessary).

In general, “already deployed” schemas are mostly static, ready to code against. In-progress schema development is more dynamic, and thus has more scope for influence by bp. Therefore, the earlier bp gets involved in schema development, the more bp will likely be able to influence the schema design such that it minimizes our technical debt or risk of incompatibility.

Commented [CH1]: What's the thought about OSDU schemas not yet transitioned to bp product lines, for example when the OSDU Forum is further ahead than BP. How will bp influence these schemas to minimise technical debt and incompatibility?

Already deployed schemas are generally static, ready to code against, and in-progress schema development is dynamic.

Commented [CH2]: It should be acknowledged that data modelling and in particular OSDU data modelling is a particular and specialised skill set. There should be a plan for this.

2. Schema Usage

2.1 Definitions

Well Known Schemas (WKS)

- **Well Known Schemas (WKS)** – are the canonical OSDU Standard schemas that need to be adopted to enable many of the benefits brought by storing data in an OSDU Standard

Extension Properties

- **Extension Properties** – All WKS have a section called “Extension Properties” into which any non-canonical attributes can be added to extend the WKS; however, these extension properties are not indexed and therefore are not searchable (not recommended as a long-term approach – see comments [Custom “Sidecar” Schema Extensions](#) below)

Commented [CH3]: Extension property are not indexed and therefore are not searchable

Custom “Sidecar” Schema Extensions

- **Custom Sidecar Schema** – Alternatively, attributes that do not fit to the OSDU WKS can be added to a custom “Sidecar” schema. This schema holds all additional properties and refers to the WKS. Therefore, all WKS attributes can be accessed and used by any OSDU-aware application or process, while bp custom attributes can be accessed and used independently, without affecting the standard. Sidecar schemas need to be built and data objects modeled as per OSDU schema framework and schema guidelines. They should not be built in isolation. This work needs to be carried out by suitably trained and skilled individuals with standardized deliverables such as documentation and worked examples as best practice. Note - these custom schema are indexed and are therefore searchable. (recommended approach)
- Further information provided in [3.2 Custom Sidecar Schema \(custom extensions to WKS\)](#)

Commented [CH4]: Yes. Sidecar schemas need to be built and data objects modeled as per OSDU schema framework and schema guidelines. They should not be built in isolation. This work needs to be carried out by suitably trained and skilled individuals with standardised deliverables such as documentation and worked examples as best practice

Commented [CH5]: Both these approaches should NOT be undertaken under a osdu:wks namespace and authority. Rather, any alteration of any schema should be done under a custom namespace and authority e.g. bp:dsbp

Commented [AB6R5]: Even a WKS + some values stored in the extension properties? If so what's the point in a WKS having the extension properties if, as soon as they're used, the schema would be considered as “custom”

Data “Tags”

- **Tags** – Tags are a generic dictionary of string keys to string value pairs, and can be added to any record within OSDU. Unlike [Extension Properties](#), Tags are indexed; however, there are still similar risks to overuse or over-reliance on the use of tags for critical business data. Tags are not governed by any referential integrity and as such are not easily managed, resulting in the potential for multiple inconsistent, duplicate or conflicting tags being applied across different data kinds. Therefore, the use of tags in any automated or app-driven use-case or workflow is not recommended.

2.2 Schema Usage Principles

The following set of principles should be followed when adopting (or extending) any OSDU WKS:

- Where an appropriate or equivalent WKS exists, this should be adopted as the schema standard for any data product held in OSDU
- Where a WKS (or WKS + extension properties) has been adopted, there should only be one schema used per Data Type (e.g. duplicate/conflicting custom schemas for the same data type should not exist, even if registered under another schema authority – e.g. “bp”).
- Multiple “versions” of each schema *may* exist, where required to manage or stagger upgrades across consuming systems and applications (see following section on “Schema Upgrades”)
- Where the WKS are not sufficient for bp’s needs (e.g. there are attributes “missing” from the WKS), missing attributes should be brought forward to the appropriate data definitions group within the OSDU Forum for inclusion in a future WKS version (where appropriate to do so). Additionally, missing attributes may be added to the WKS as “extension properties” or through the use of a custom “Sidecar” Schema. This should be done by exception, with the following considerations (to be owned by the relevant data product owner):
 - General attributes (non bp-specific, that may be applicable to other operators or ISVs – i.e. deemed to be an omission from the WKS)
 - These attributes should be added to a managed backlog, and brought to the appropriate Data Definitions group within the OSDU Forum, with an aim to have these adopted and incorporated into the WKS in future WKS versions
 - Proposals should be presented to the Forum following the Proposal process and template (this generally means that an issue is created and a proposal workbook is generated and populated - when raising an issue, take note of the release you want/aim to have the change included into)
 - Once the proposal workbook is approved documentation and work examples should be generated too
 - In the interim, until these changes are made, missing attributes may be either
 - (Recommended) Temporarily added to a custom Sidecar Schema (with the risk of incurring technical debt), registered under a namespace “bp:custom-schema” (see [Custom Side-Car Schema](#)), or
 - (Not Recommended) Temporarily added as extension properties to a standards WKS schema (with the risk of incurring technical debt and potential additional complications), which can be re-loaded as custom and registered under a namespace “bp:custom-schema”, or
 - (Not Recommended) Temporarily omitted from the OSDU data record (with the risk of data loss during the interim period)
 - bp-specific attributes (attributes specific to bp’s digital product needs, for example internal IDs or internal tool-specific attributes). These attributes may never need to be adopted by the OSDU Forum, and may exist indefinitely:
 - (Recommended) A [custom Sidecar Schema](#) should be authored to hold these properties/attributes (registered under a “bp:custom-schema” namespace)
 - Responsibility to maintain/upgrade these custom schemas sits with the relevant data product owner

Commented [CH7]: This should be done under a custom namespace and authority. Only schema approached by the forum can have the osdu:wks

Commented [CH8]: Yes. This is where engagement with the Forum for already deployed schema exists. If we/bp need an added property or schema (not sure if you’ve covered whole schemas elsewhere) the chances are other operators need it too. Added properties, reference values (or schemas) should be view as temporary in nature, until they can be added into the official osdu:wks schema - enabling code in broader ecosystem to be developed against them. Oops you’ve got that below

Commented [CH9]: Proposals should be presented the Forum following the Proposal process and template. This generally means that an issue is created and a proposal work book is generated and populated. When raising an issue take note of the release you want/aim to have the change included into. Once the proposal workbook is approved documentation and work examples should be generated too

Commented [CH10]: Yep. And there not an osdu:wks schema, they would need their own custom namespace and authority. I’m guessing this maybe something like bp:dsbp but it could be different.

2.3 Risks & Considerations

Significant risks come with long-term over-reliance on attributes held as either **extension properties** or **custom sidecar schemas** to a WKS:

- Risks to both Extension Property & Custom Sidecar Schema usage
 - bp may deviate from the OSDU canonical WKS structure over time
 - **Increased maintenance effort/cost** to “carry” extension properties between each schema version upgrade or “upgrade” custom sidecar schemas in-line with SODU milestone releases
 - By default, applications will **not recognize attributes held by bp as extension properties or in any Custom Sidecar Schema** (resulting in the need for customization to app deployments or a reliance on developing custom plug-ins over “native” application connectivity)
 - **Increased “regret spend”** from internal tools/application/integration infrastructure if built with an over-reliance on extension properties
- Risks Specific to Extension Property usage
 - All **extension properties are not indexed** and therefore are **not returned in the OSDU Search** (therefore cannot be leveraged for data discovery). They are only persisted in the storage record and NOT the indexed record.
 - As mentioned above - by default, applications will unlikely recognize attributes held by bp as extension properties - As such, there is significant risk that, should these applications then write back an update to the OSDU WKS, these extension properties may be omitted – this could break down any internal bp dependencies on these attributes

Commented [HC11]: Just to clearly spell it out. Extended properties (and values) will NOT be return in search. The extended properties are only persisted in the storage record and NOT the indexed record.

Commented [CH12]: I would say that externally developed applications WILL NOT recognise these items

Commented [CH13]: Technically I believe it's the same record now. The Forum dropped separated index schema. But this is splitting hairs, the concept is correct, they're not indexed

3. Authoring OSDU Custom Schemas

3.1 Custom Metadata Schemas

This section deals with the authoring of custom **metadata** schemas. These refer to descriptive attributes, describing the data record, which enable search and discovery of files or bulk content held in OSDU storage. In the case of a Work Product Component (WPC) this is **NOT** the bulk data content itself. For bulk content storage, see “[Custom DDMS](#)”.

Commented [CH14]: Should this and others similar be subheadings? It's becoming easy to lose my place in the document. I am being immersed in this word document ☺

Commented [AB15R14]: Will sort the heading/sub-heading formats to be clearer!

Commented [CH16]: This a specific to group-type WPCs. Master data is the content.... That may be to much for the audience.

Commented [CH17]: “can be delayed”??? I don’t get it.

Commented [AB18R17]: This is pandering to bp IT speak... as they often just view the OSDU project as a mass-migration exercise...

Commented [CH19]: Reducing technical debt and incompatibility and increasing adoption of data standards, speed of technical innovation and general participation in the broader ecosystem and market place

This approach promotes the following benefits:

- Reducing technical debt and incompatibility between digital tools and data by increasing adoption of data standards
- Speed of technical innovation through general participation in the broader ecosystem and industry market place

Where this is **not possible** and an exception is made, the following set of principles should be followed when authoring any “Custom” schema within the bp OSDU platform:

- Where a Custom schema is created by bp, it should be registered under a consistent namespace **“bp:custom-schema”**
- Where a custom schema has been authored for a data type, there should only be one schema used per Data Type (e.g. duplicate/conflicting custom schemas for the same data type should not exist)

Alternatively - where there is ongoing data definition work for a data type not yet available in the latest OSDU release, there may be occasions where bp is dependent on **attributes available in a future WKS version** (not yet available for use in our Production OSDU instance). In this case, we may opt to **author a future WKS schema version early as a custom schema**, so as to enable “early access”. This should only be done on a **use-case basis (by exception)** and a systematic process should be established to **migrate from this custom schema to its equivalent WKS**, once it become available for use:

- As above, until this schema is available in line with the active OSDU platform release being used by bp, then this schema should be registered as “custom” under a consistent schema namespace of **“bp:custom-schema”**
 - Pro – aims to avoid deviating too far from existing work by the OSDU Forum
 - Con – Schemas not yet incorporated into a future platform release may still change before being adopted as WKS

[bp-Preferred Approach to Custom Schema Usage](#)

As above, Custom Schema must only be authored by exception and must be actively managed / governed by appropriate data owners. Custom schemas need to be built, and data objects modeled, **as per OSDU schema framework and schema guidelines**. **This work needs to carried out by suitably trained and skilled individuals with standardized deliverables such as documentation and worked examples as best practice.**

NOTE – Conforming with OSDU standardized deliverables when authoring custom schema with OSDU the schema service ensures that standard OSDU platform Data Quality assessments may be automatically performed by suitable data governance technology (ensuring only conformant data is added to these schema).

bp's recommended approach is outlined below:

1. Custom Side-Car schema **must** conform with OSDU Schema expectations:
 - They **must contain standard OSDU platform system attributes** (e.g. id, entitlements, legal tags, etc.), in conformance with the expected schema structure
 - Ideally custom schema should be **kept as simple as possible** – and leverage existing, available OSDU standards schema fragments to support consistency and efficiency (e.g. leverage, AbstractCommonResources)
2. Any bp-authored custom schema should still confirm to OSDU standards (e.g. **naming standard for any bp-authored custom schema has been agreed**):
 - **“master-data--<Schema-Name>:x.x.x” (for Master Data)**
 - **“work-product-component--<Schema-Name>:x.x.x” (for Work Product Components)**
 - Where “<Schema-Name>” refers to the name of the custom schema
 - And “x.x.x” refers to the managed major and minor versions of the schema

3. Any new bp custom schema must be authored under the following namespace:
 - “***bp:custom-schema:***”
 - (e.g. “***bp:custom-schema:work-product-component--<Schema-Name>:x.x.x***”)
4. When any new attribute needs to be added to a custom schema, ***a new minor version of that schema needs to be authored***, which would then contain that additional attribute – this has an impact on [schema upgrade management](#), and as such should ***ideally be done in managed batches***, rather than on a sporadic / ad hoc basis.

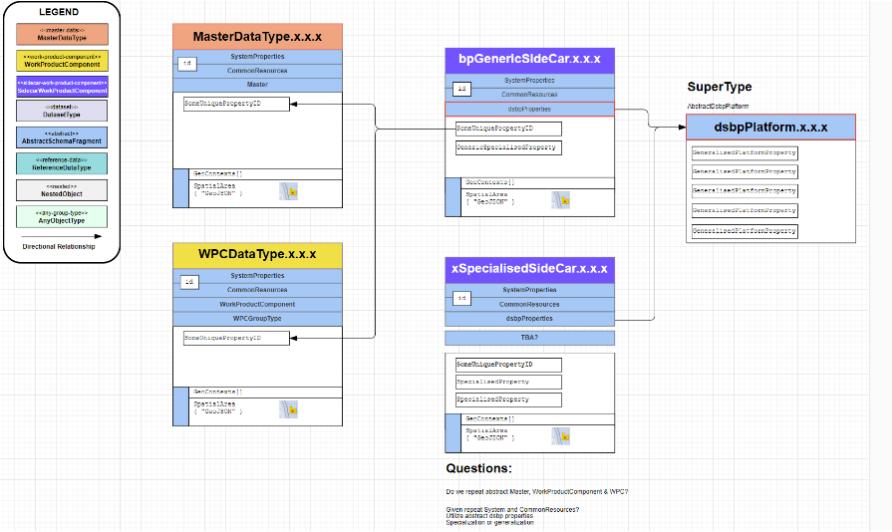
3.2 Custom Sidecar Schema (custom extensions to WKS)

*The concept is that of a Motorcycle (the WKS of a particular data kind) that contains the OSDU Standard attributes, that is connected to one or more Side-Cars (which could be an OSDU standard or a **bp custom** side-car schema), containing any managed extensions to the OSDU Standard attributes held by the “Motorcycle”/WKS.*

Custom Sidecar Schema are the **recommended way to incorporate additional properties to an existing WKS**. These are entities that extend the definition of an existing entity in a specific functional area. As mentioned in the definition section - Sidecar schemas need to be built and data objects modeled [as per OSDU schema framework and schema guidelines](#). They should not be built in isolation. **This work needs to carried out by suitably trained and skilled individuals with standardized deliverables such as documentation and worked examples as best practice.**

NOTE – Conforming with OSDU standardized deliverables when authoring custom schema with OSDU the schema service ensures that standard OSDU platform Data Quality assessments may be automatically performed by suitable data governance technology (ensuring only conformant data is added to these schema).

Optional – sidecar schema could be developed in different ways (e.g. bpGenericSideCar.x.x.x vs xSpecialisedSideCar.x.x.x). Therefore, **decisions over how a sidecar schema is developed should be “use-case driven”** (e.g. we may opt to create a “bp<KIND>SideCar.x.x.x”, with a generic structure, to house bp-specific attributes specific to a particular master or WPC WKS record, to accommodate ALL attributes for that Kind that specifically support the needs of bp tools (e.g. ADI or Falcon, etc.), OR we may opt to create a “xSpecialisedSideCar.x.x.x” to incorporate just the attributes specific to the needs of a particular app or system, for any one data kind). Examples of both approaches are shown below.



bp-Preferred Approach to Custom Side-Car Usage

Below are some key concepts & guidelines that should be followed **consistently** across bp's implementation of the OSDU data platform, when authoring Custom Side-Car Schema:

- OSDU WKS (the “motorcycle”) may have multiple side-car schema
 - o Side-car schema, themselves, can have related side-car schema
 - o Side-car Schema **cannot** be related to multiple WKS (“motorcycles”)
- Side-car schema **must** relate to the “motorcycle” WKS
 - o The “motorcycle” WKS (by definition of being a WKS) **will not** relate directly to any associated side-car schema
- Custom Side-Car schema **must** conform with OSDU Schema expectations
 - o They **must contain standard OSDU platform system attributes** (e.g. id, entitlements, legal tags, etc.), in conformance with the expected schema structure
 - o Ideally – these could be inherited from the “motorcycle” (though this is not currently automatic/driven by any kind of service etc.) – this would avoid divergence of entitlements between the WKS and associated side-car data.
 - o Ideally custom side-car schema should be **kept as simple as possible** – with minimal duplication from the “motorcycle” (other than system properties, if necessary). **By exception**, some “duplication for convenience” may be preferable to support data search & discovery (for example, AbstractCommonResources)
 - o By default, custom side-car schema should **avoid replicating the “GeoContext[]” schema fragment, nor include any “SpatialArea[]” data** (except via use-case-driven exception), as **these values should reside in the “motorcycle”**.
- bp **may** opt to define some **additional custom abstracted schema fragments** (SuperTypes) to house **common bp system attributes that may be relevant across multiple custom side-cars** (e.g. if all dsbp data products share common dsbp-specific properties [that we wish to retain post-migration] these may be stored in a common schema abstraction that can be referred to in each data-type-specific Custom Side-Car schema, to avoid replication).

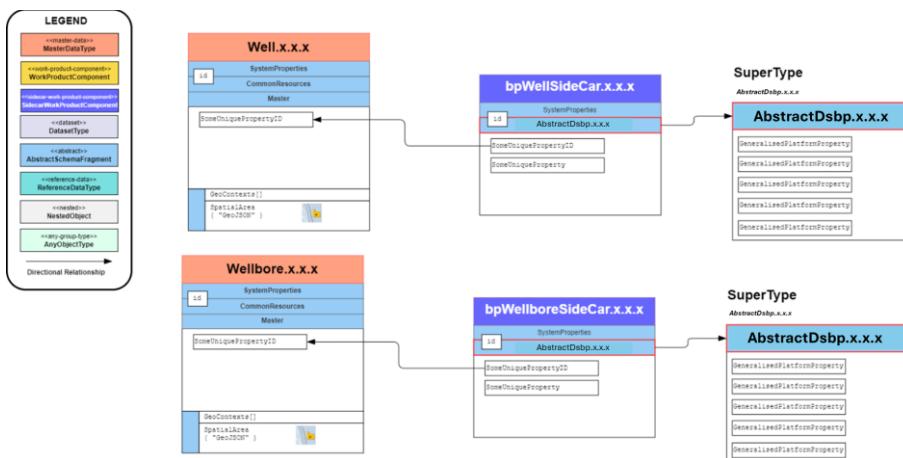
bp's recommended approach is outlined below:

1. A **standard, "generic", custom side-car schema format has been agreed** (that can be used for any bp-authored custom side-car schema), in a simplified sense, this structure includes:
 - Standard OSDU "SystemProperties"
 - "dsbpProperties" SuperType abstraction (optional – use-case driven)
 - id reference to data kind WKS ("motorcycle")
 - Set of attributes (main body of the Side-Car)
2. A **naming standard for any bp-authored custom side-car schema has been agreed**:
 - "**master-data--bp<KIND>SideCar:x.x.x**" (for Master Data)
 - "**work-product-component--bp<KIND>SideCar:x.x.x**" (for Work Product Components)
 - Where "<KIND>" refers to the associated WKS (motorcycle)
 - And "x.x.x." refers to the managed major and minor versions of the side-car
3. Any new bp custom schema must be authored under the following namespace:
 - "**bp:custom-schema:**"
(e.g. "bp:custom-schema:master-data--bp<KIND>SideCar:x.x.x")
4. A generic Custom Side-Car Schema should be authored for each relevant Data Kind, which **can then be used to house any bp-specific attributes** (that do not conform to the WKS for that kind)
5. When a new attribute is found that needs to be added to the side-car schema for that data kind, **a new minor version of that schema needs to be authored**, which would then contain that additional attribute – this has an impact on [schema upgrade management](#), and as such should **ideally be done in managed batches**, rather than on a sporadic / ad hoc basis.

Worked Example (ERD)

Below is a worked example showing bp custom side-car schema for Well and Wellbore WKS:

- Authored under the “bp:custom-schema:” namespace
- Related to their respective bp-authored custom side-car schema (“master-data--bpWellSideCar” & “master-data--bpWellboreSideCar”, respectively)
- Related to their (optional) “AbstractDsdp” SuperType abstraction



Worked Example (Schema)

Below are the proposed Custom Side-Car Schema for:

- **bpWellSideCar.1.0.0.json**



- **bpWellboreSideCar.1.0.0.json**



- **AbstractDsdp.1.0.0.json**



Considerations when Authoring Custom Side-Car Schema

In the same vein as [Custom Schemas](#), Custom Side-Car Schema should only be authored in OSDU **by exception**, and where an appropriate WKS does not have suitable OSDU-equivalent attributes to meet bp's needs.

Ideally, prior to making this exception, if migration/introduction of said attributes to OSDU can be delayed or deferred, bp's preference is to bring SME requirements to the OSDU Forum by engaging with the appropriate Data Definitions group to collaboratively agree to explain the required OSDU WKS without industry peers, such that these previously-missing attributes can be used by bp in a future milestone release of OSDU.]

This approach promotes the following benefits:

- Reducing technical debt and incompatibility between digital tools and data by increasing adoption of data standards
- Speed of technical innovation through general participation in the broader ecosystem and industry market place

Commented [CH20]: "can be delayed"??? I don't get it.

Commented [AB21R20]: This is pandering to bp IT speak... as they often just view the OSDU project as a mass-migration exercise...

Commented [CH22]: Reducing technical debt and incompatibility and increasing adoption of data standards, speed of technical innovation and general participation in the broader ecosystem and market place

3.3 Custom DDMS

This section deals with the authoring of a custom **Domain Data Management Service (DDMS)**. These refer to the domain optimized data storage solutions that hold the bulk data content, itself.

The strong advice from the OSDU forum is that a **custom DDMS should NOT be created**, even if an appropriate DDMS does not yet exist to store the bulk content for a data type.

Instead, migration of said data type to OSDU **must** be delayed or deferred. **Best practice should then be for bp (via the relevant data product owner) to bring SME requirements to the OSDU Forum by engaging with the appropriate Data Definitions or DDMS “Capability” group to donate, build or contribute towards the building of a new OSDU DDMS (or incorporation of additional data types into an existing DDMS)**, which can then be adopted by bp in a future milestone release of OSDU.

Where the above is not possible, or **bp chooses to act regardless of this advice**, the following points should be carefully addressed when considering creating any custom DDMS within the bp OSDU platform:

- The process of defining and authoring a bulk content store as a DDMS in OSDU is vastly more complex than creating a custom metadata schema (requiring significant time and resource to develop and implement)
- A DDMS is non-standard, and is a full set of services and APIs, not just a schema. And thus there is **significant development work required to build and maintain a custom DDMS**, e.g.
 - Detailed consideration around technology & architecture
 - Interaction with the OSDU core services - A DDMS also requires integration back to all the core platform services, including honouring the legal and access permissions
 - A DDMS record must also create an equivalent metadata catalogue record and cannot exist in isolation from the core platform
- The above development work is deemed to be “regret spend” due to incurring significant technical debt that will, in time, be superseded by an OSDU Standard.
- Where a custom DDMS has been created to house a data type, that DDMS should be the only place bulk content for that data type is held

3.4 Additional Risks & Considerations

Note – **significant risks come with long-term over-reliance on custom schemas (metadata & DDMS)**:

- bp will deviate from the OSDU canonical WKS structure over time
- bp will be solely responsible for the maintenance effort/cost to maintain custom schemas through each milestone release (this is significant in the case of a custom DDMS)
- Applications will not recognize any data held in bp-authored custom schemas or DDMS (resulting in a reliance on developing custom plug-ins vs “native” application connectivity)
- Significant “regret spend” from internal tools/application/integration infrastructure if built on custom schemas or custom DDMS, and then subsequently are superseded by a WKS or OSDU DDMS over time (in addition to secondary migration and change management, etc.)

Commented [CH23]: DDMS schema's are not even a thing!

Commented [CH24]: Custom domain optimised data storage solutions should NOT be created

Commented [CH25]: A DDMS also requires integration back to all the core platform services, including honoring the legal and access permissions. A DDMS record must also create an equivalent metadata catalogue record.

I think “Best Practice” should be engage with the Forum and the dedicated DDMS groups. These are called Capability groups and are not the same as Data Definitions

Commented [AB26R25]: Second part is incorporated in the section above

4. Minimum Required Attributes

The OSDU forum only prescribes the **bare** minimum mandatory attributes for a data record to be "valid" in OSDU (e.g. attributes such as the record ID, Entitlements & Legal Tags). However, for data to be functionally useful in OSDU, many applications, processes or workflows (either bp specific or 3rd party) will likely depend on a specific set of agreed "mandatory" attributes.

Secondly, where certain data properties are held within the schema (i.e. to which OSDU WKS attribute these properties are assigned or mapped to) can oftentimes be subjective and **open to interpretation**. Additionally, certain properties may need to exist in **multiple places** within a schema (e.g. the well UWI could exist as the Well ID, but equally, if a well is deemed to be a facility, it could/should also exist as the Facility ID).

If there is no agreement across OSDU members around what properties should be supplied as a minimum, or where they should be held in the schema – there is a risk the entire value-case for conforming to an industry standard may break down. Therefore, it is crucial that bp takes the below steps to mitigate this risk (to be owned by the relevant data product owner):

- bp defines a clear set of minimum mandatory attributes for all master and WPC data kinds
 - o This should be clearly communicated as our baseline expectations for internal bp and 3rd party consumers of OSDU data (mandating that any data written to OSDU contains these attributes for it to be a viable data record)
 - o The minimum mandatory attributes may change over time, in collaboration with the OSDU Forum and our partners (**and must align with an OSDU standard once set**)
- bp actively engages with the OSDU Forum (in addition to our strategic ISV partners and Operator peers) to agree a standard set of mandatory attributes (by data kind), to align across the industry (including principles around schema usage – esp. where attribute definitions are ambiguous or open to interpretation)

4.1 Related Principles & Standards

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

- **Lineage & Ancestry, Activity & Business Decisions Standards**
 - o Considerations around impact of Lineage & Ancestry on Legal / Contractual restrictions of any derivatives or related data
- **Mapping Standards (incl. Minimum Required Attributes)**
 - o Defining bp's expectations for Minimum Required Attributes, by data kind
- **Data QC/QA & Technical Assurance Standards**
 - o Data Quality & Technical Assurance technology implementation (governance around pre- & post-ingestion data completeness requirements)

Commented [CH27]: The Bear or Bare rawww!

Commented [AB28R27]: rawr

Commented [CH29]: I haven't seen this topic come up yet.... How will bp implement and manage these?

Commented [AB30R29]: Held in a separate standards document - as probably a whole topic in itself?

Commented [CH31]: This sentence confuses me. Maybe I'm still thinking about that bear.

Commented [AB32R31]: Hopefully better?

Commented [CH33]: To me, that's a different topic and not minimal recommended properties. We changed the term in the forum

Commented [AB34R33]: Agree, but with nowhere else to put this, just wanted to mention it here (for anyone who has got this far in the doc!)

Commented [CH35]: I think the benefit here is that other companies may have already done the work to identify the minimal properties.

Typically, we all work with the same vendors, e.g. core lab, TGS, weatherford, etc. and I'll be easier if we partner with our partners on this, external facing side.

Internally for our own apps and workflows we should, as you've said, set, document and enforce those minimal properties going forward

5. Schema Upgrades

With each new milestone release of OSDU there may be new “version” of each WKS. Data may need to be “upgraded” to the most up-to-date schema version. This process is not (currently) automatic, and will need to be done in a systematic way by the data product owners, on behalf of their business customer needs (note - the decision to upgrade a schema should primarily be a business decision; however, there may be a technology component to meet certain vendor support requirements, etc. – e.g. a vendor may denote a minimum schema version with which it’s software is compliant):

- Ensure any **changes** between schema versions are appropriately communicated
- Ensure all consuming digital tools, products or workflows are upgraded to the new schema version **at the same time**
- Where a consuming product/consuming products cannot be upgraded to the new schema version, there may be the need to manage, maintain & keep-in-sync **multiple schema versions** for the same data kind (this may also apply to bp-authored Custom and Custom Side-Car schema)
- Any existing Custom Side-Car Schema **may also need to be “upgraded”** to point to the relevant upgraded Schema version (that it refers to)

Release documentation and information is provided by the OSDU Forum, detailing the schema changes and reference value list content changes to aid data migration per OSDU milestone release.

[OSDU Migration Guides](#)

5.1 Preferred Schema Version

In order to accommodate for the inevitable complexity on handling schema upgrades, the OSDU platform provides a configuration record to identify a minimum, maximum and a preferred schema version for a given schema kind fragment defined by a **major** version number. This kind fragment is held in the data.Code and part of the id of the reference-data--PreferredSchemaVersion record.

- [Link](#) to OSDU documentation on PreferredSchemaVersion.
- [Link](#) to example json.

This configuration allows us to manage staggered schema upgrades (by denoting the min, max & preferred schema versions for a data kind, rather than necessarily defaulting to the latest version for all schema).

e.g. for some schema, for a major version “1” (i.e. osdu:wks:group-type--exampleschema:1.X.X), we can define:

- MinimumKindVersion = osdu:wks:group-type--exampleschema:1.1.0
- PreferredKindVersion = osdu:wks:group-type--exampleschema:1.2.0
- MaximumKindVersion = osdu:wks:group-type--exampleschema:1.3.0
- IsPreferredMajorVersion = true

In this example for exampleschema major version 1 (“osdu:wks:group-type--exampleschema:1.X.X”), bp defining that schema versions “osdu:wks:group-type--exampleschema:1.0.0” to “osdu:wks:group-type--exampleschema:1.3.0” are suitable for use, with the preferred schema version being “osdu:wks:group-type--exampleschema:1.2.0”.

Commented [CH36]: It should primarily be a business decision to upgrade schema once everything is operational... There may be an IT component for vendor support requirements.

This allows us to programmatically specify to any consuming application, software or workflow what schema versions are available for use, and what our preferred schema version should be.

NOTE - PreferredSchemaVersion is a local reference list. As such, OSDU provide a starting (default) list of values, but these need to be amended/updated and added-to, based on bp's own requirements.

There needs to be **ownership/accountability** from each data owner to keep this reference list **current** and **in-line with relevant preferred schema versions, for each data kind, for each major version**.
Updating this configuration should form an integral part of any defined schema upgrade process.

Example reference-data--PreferredSchemaVersion record

Below are examples of the PreferredSchemaVersion format –

Excel representation:



PreferredSchemaVers
ion.1.xlsx

Json record example:



PreferredSchemaVers
ion.1.json

5.2 Custom Schema Upgrades

It should be noted that every time a new attribute is added to a custom schema, a **new minor version of that schema needs to be authored** (which would then contain that additional attribute). And a **new major version should be authored for any significant re-structuring** of a custom schema.

As such, this has an impact on **schema upgrade management**, as any consuming application would need to be **updated to point to the new schema version**, containing the new additional attribute(s) or updated schema structure.

To mitigate the impact of this, the recommended approach is to make changes to custom schema **in managed batches** (rather than sporadic / ad hoc upgrades), and **in line with platform-wide schema upgrade processes**.

5.3 Custom Side-Car Schema Upgrades

It should be noted that every time a new attribute is added to a custom side-car schema for any data kind, a **new minor version of that schema needs to be authored** (which would then contain that additional attribute).

As such, this has an impact on **schema upgrade management**, as any consuming application would need to be **updated to point to the new schema version**, containing the new additional attribute(s).

To mitigate the impact of this, the recommended approach is to make changes to custom side-car schema **in managed batches** (rather than sporadic / ad hoc upgrades), and **in line with platform-wide schema upgrade processes**.

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](#)