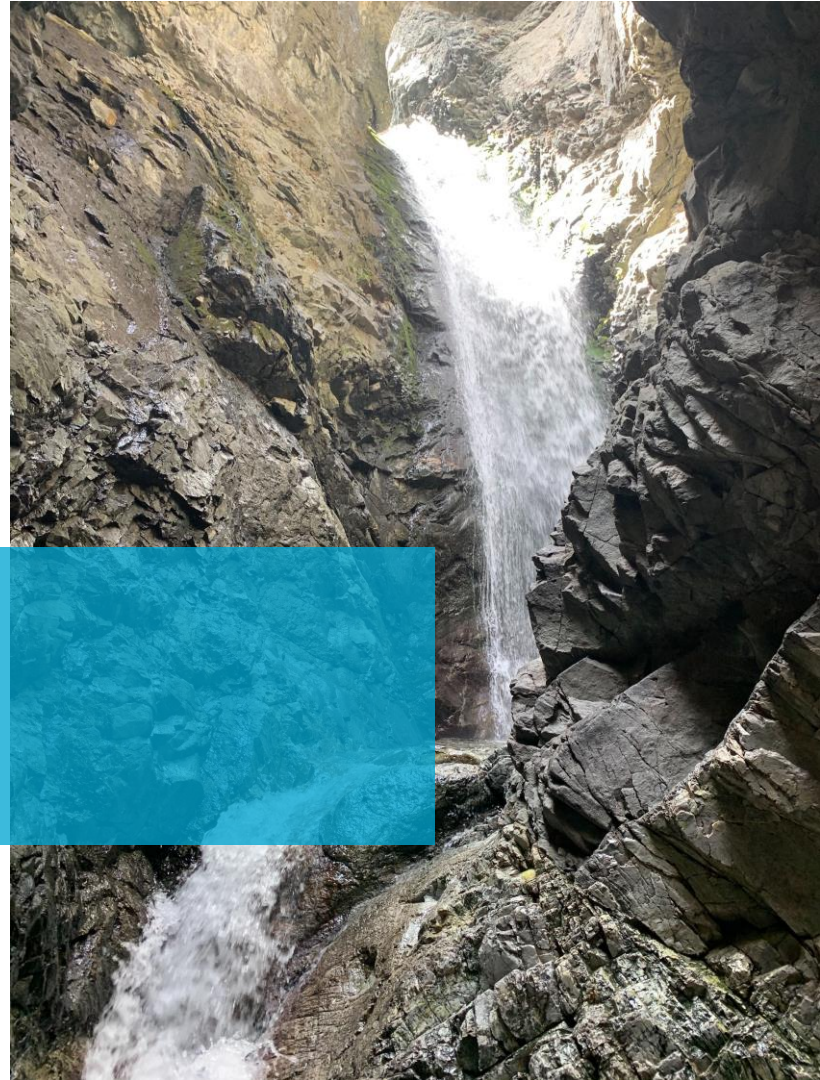




# *OSDU Data Platform Application Developer Training*

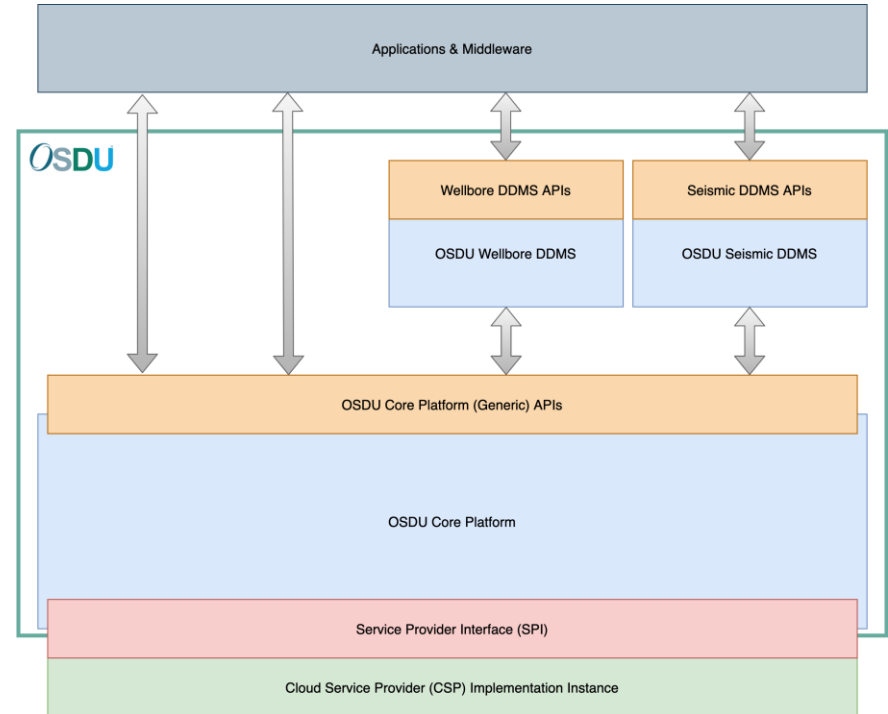
## *DDMS Core Concepts*



# DDMS: Concepts

## A DDMS...

- » delivers optimized handling of data for each (non-overlapping) “OSDU domain” (e.g. vertical discipline or horizontal functional capability)
- » delivers capabilities not supported by OSDU generic normal APIs; A DDMS delivers exclusive access for a set of defined kind/type representations: \* e.g. WPC--SeismicTraceData in OVDS form.
- » can help achieve the extension of OSDU Data Platform scope to new business areas
- » may be developed in a distributed manner with separate resources/sponsors
- » is part of the overall OSDU Data Platform’s common code (open source)
- » is defined owned, governed, distributed and maintained by the OSDU Forum



# OSDU DP Data Principles

## All migrated, acquired, and created data is preserved

Ingestion and retention minimize data loss.  
[Exception: During DDMS activity, data content being built is not considered 'retained' until fully built]

## Data is globally identifiable

Context specific data identity prevents compromising data.

## Data is immutable

Data content and context (metadata) always.  
[Low value intermediate data may be soft deleted, judgement subject to cost trade-offs.]

## Data is access controlled

Complies with Data Platform Authentication and Authorization

## Data is governed for right of use

Implements Data Platform Policy based Entitlements

## Data is discoverable

All data context (metadata) are harmonized and indexed for Search. Master/reference data content as well. [A DDMS may offer optimized indexed deep search capabilities.]

## Data is consumable

All data persisted must be exposed in a way that is consumable to other users of the OSDU Data Platform.

## Improved data is new data

Enrichment results in new data honoring immutability of existing data; link through versioning (master/reference) and lineage (components).

## Data lineage is tracked

All transformations and workflows provide lineage assertions from new data back to predecessors, sources, etc.

## Data and service are managed by Platform Instance

Operational procedures apply consistently across the platform

# DDMS: Detailed definitions (1 of 2)

- » **Service:** A DDMS is an **OSDU application accessible service** with one set optimized APIs:
  - A DDMS implementation shares in all relevant aspects of OSDU DP architecture, governance, infrastructure, technology, and behavior; and calls on OSDU DP internal services, as appropriate.
  - A DDMS implementation is part of the overall OSDU DP common code, is governed by the OSDU Forum, and is distributed in each OSDU DP release.
  
- » **Domain:** A DDMS has a **scope** corresponding to:
  - Single vertical discipline or business area, or (Ex: Petrophysics, Geophysics/seismic)
  - A functional aspect of one or more vertical disciplines or business areas (Ex: Earth Model)
  
- » **Optimised Domain APIs:** A DDMS delivers **high performance capabilities** not available using generic normal DP APIs, such as:
  - DDMS APIs to access or build partial content defined OSDU DP data kinds/types, including very large content datasets
  - DDMS APIs to access or build derived data structure instances for application use from one or more defined OSDU data kinds/types
  - DDMS APIs to accessing intelligently selected subsets of content based on ‘deep search’ content-based criteria, including criteria based on bulk/array data structures

# DDMS: Detailed definitions (2 of 2)

- » **Seamless Layered Linkage for data content:**
  - DDMS API implementations handle domain API behavior according to defined content data models and mappings to data storage actions.
  - Each data storage action is handled in one of three ways: an OSDU DP internal storage service, an OSDU DP resident 'access library' in a DDMS, or directly through the SPI (Service Provider Interface) to a CSP storage technology (where applicable).
  
- » **Application / DP Interface Design Pattern:** Using both DP generic normal APIs and a DDMS's optimized APIs:
  - Application uses OSDU DP generic normal APIs to discover and access the data needed to prepare and carry out application data actions.
  - Application uses DDMS optimized APIs to read/write new, potentially partial data content to perform workflow steps, algorithms, visualizations, derivations, etc., accessing and building.
  - Note that DDMS API service implementations call a generic OSDU DP service to handle support for 'context' (dataset metadata, WPC and WP metadata, master data, reference data).

# DDMS: Examples

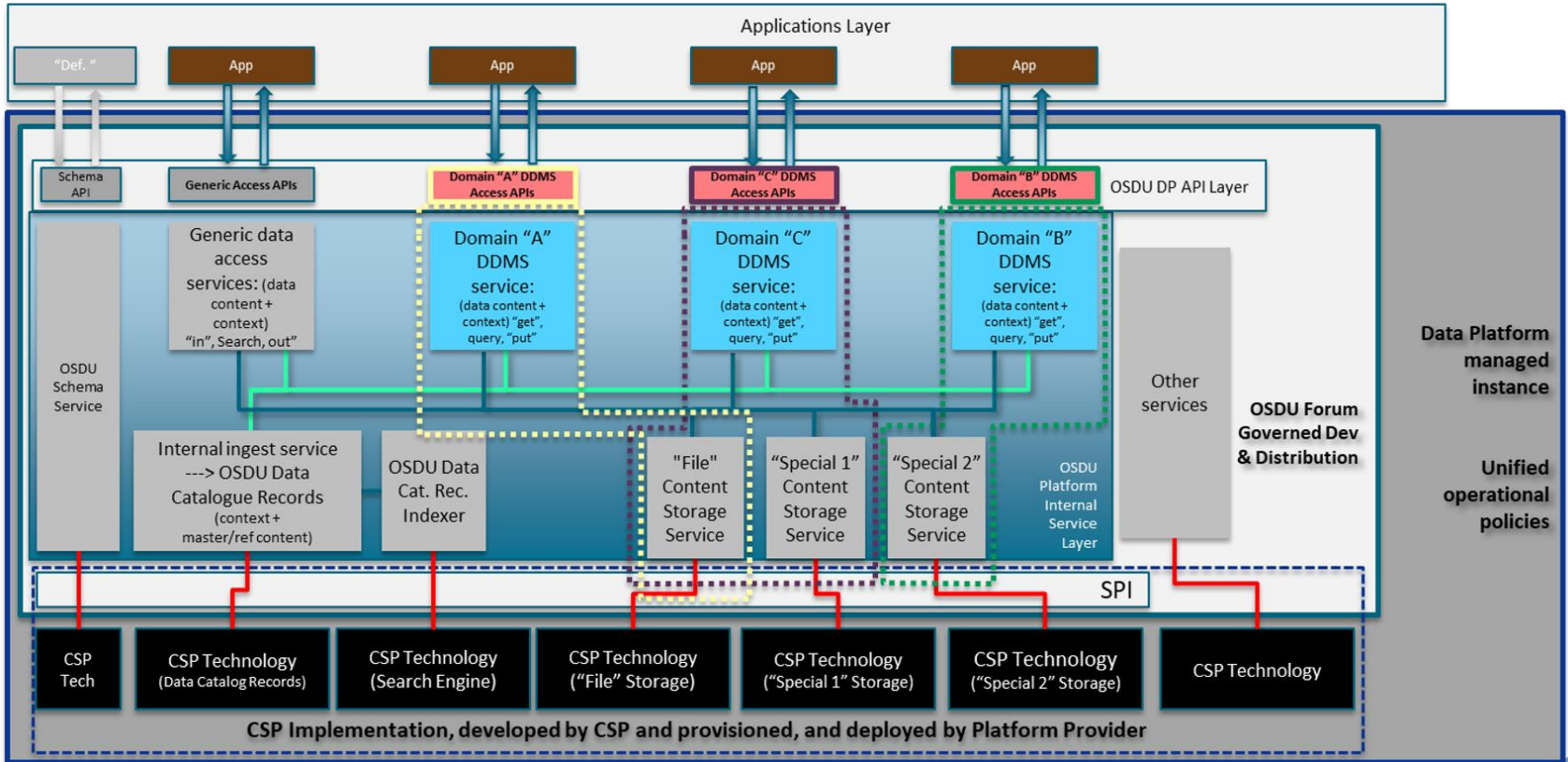
## » Discipline Examples:

- **Seismic / Geophysics DMS** including seismic with provision to grow to no-seismic Geophysics; offering high-performance selective and subset access to super-large datasets
- **Petrophysics / Well Log (Wellbore) DMS** including well log; offering high-performance searchability of curve values across logs, wells, etc. integrating references to property values from wellbores and more
- **Reservoir / Modeling / Geology DMS** offering high-performance support for earth models building, interpretation, optimization, and simulation
- **Well Delivery / Drilling DMS** offering high-performance support for planning, execution, and analytics
- **Production DMS** offering efficient support for reporting and operational planning, including suitable reservoir and flow models

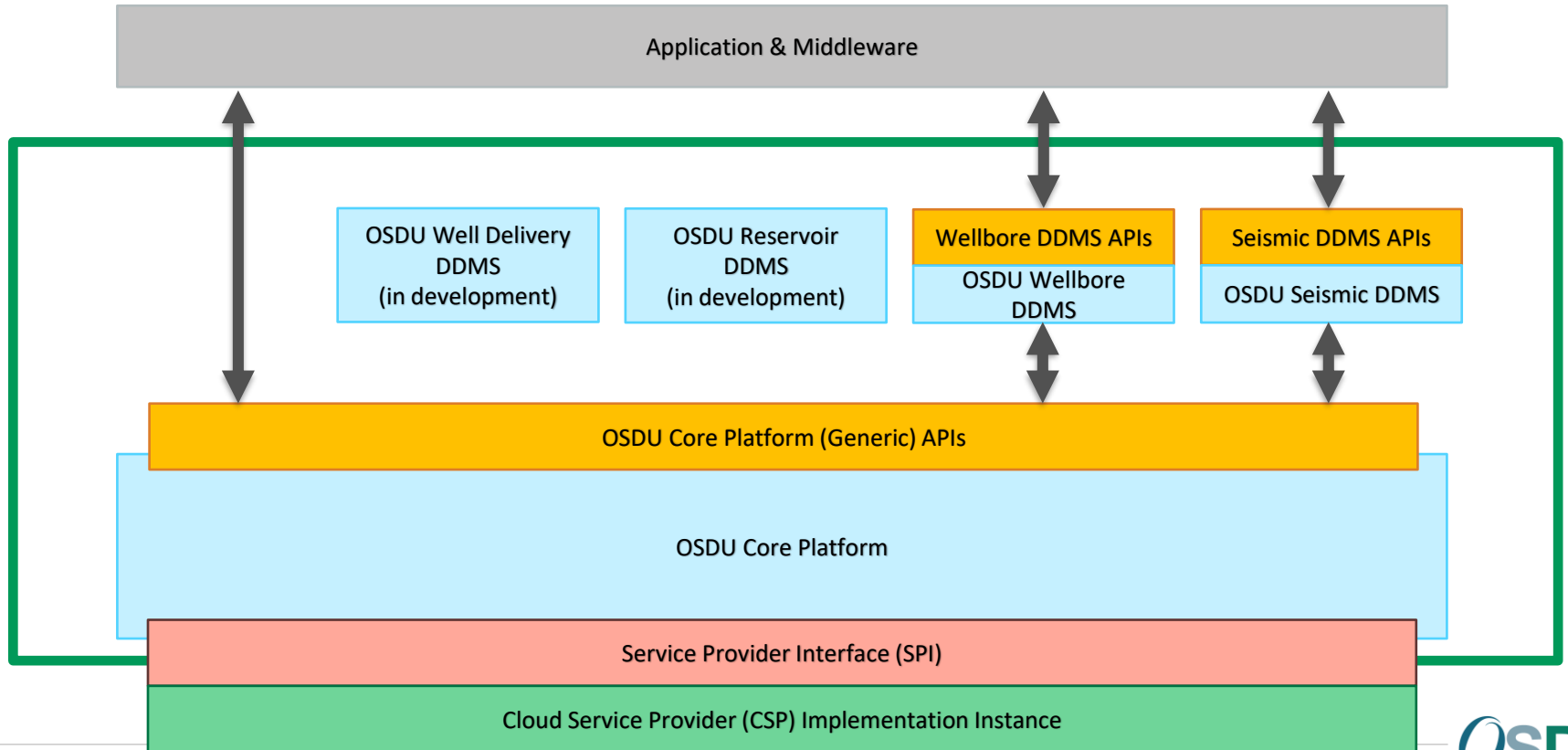
## » Functional Example:

- **Earth Model DMS** offering support for subsurface model description, visualization, and interpretation relevant for geological, reservoir optimization, production modelling, and other purposes.

# DDMS + Generic: Architecture Diagram



# DDMS: R3 Status





# Thank you!