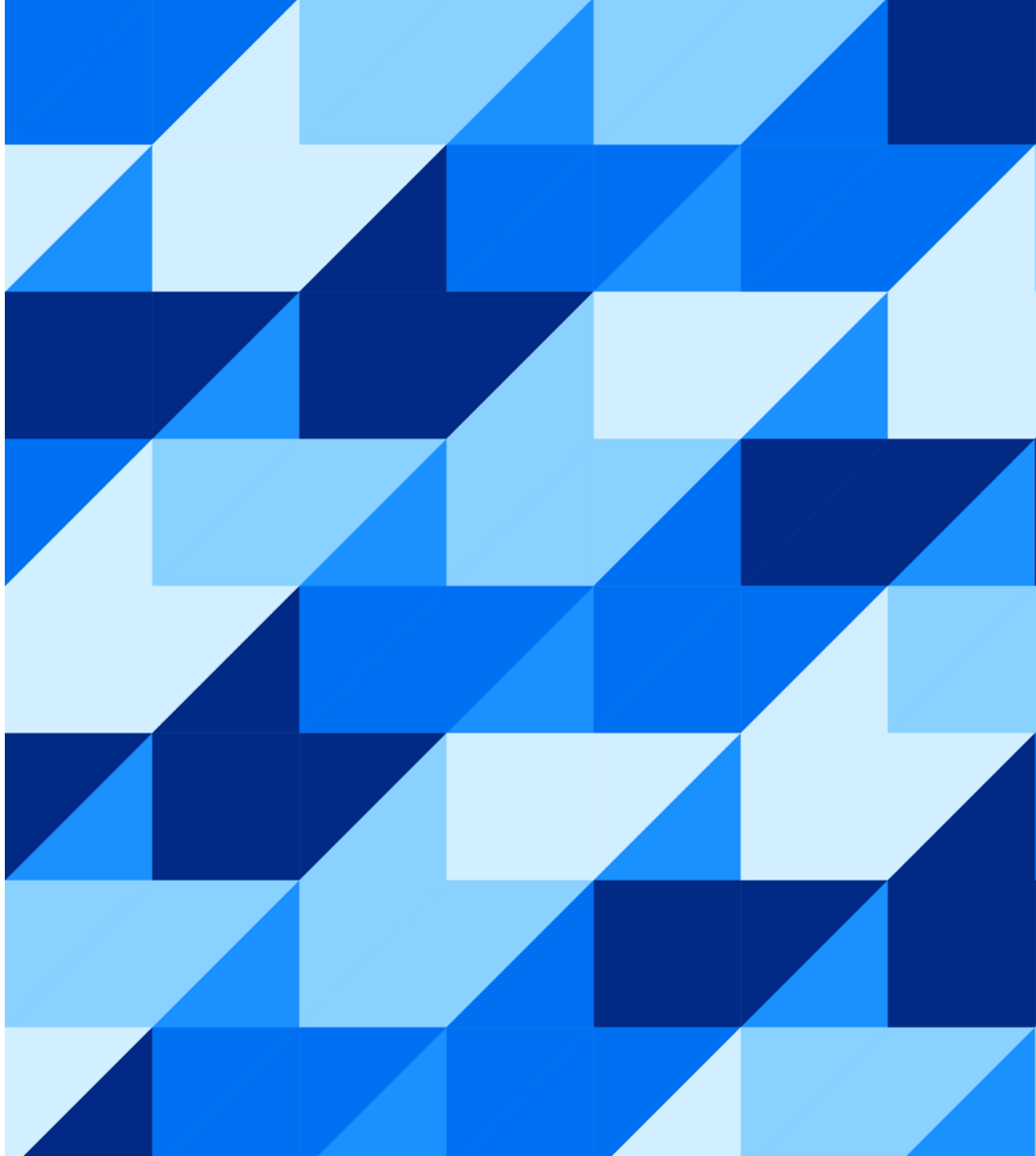
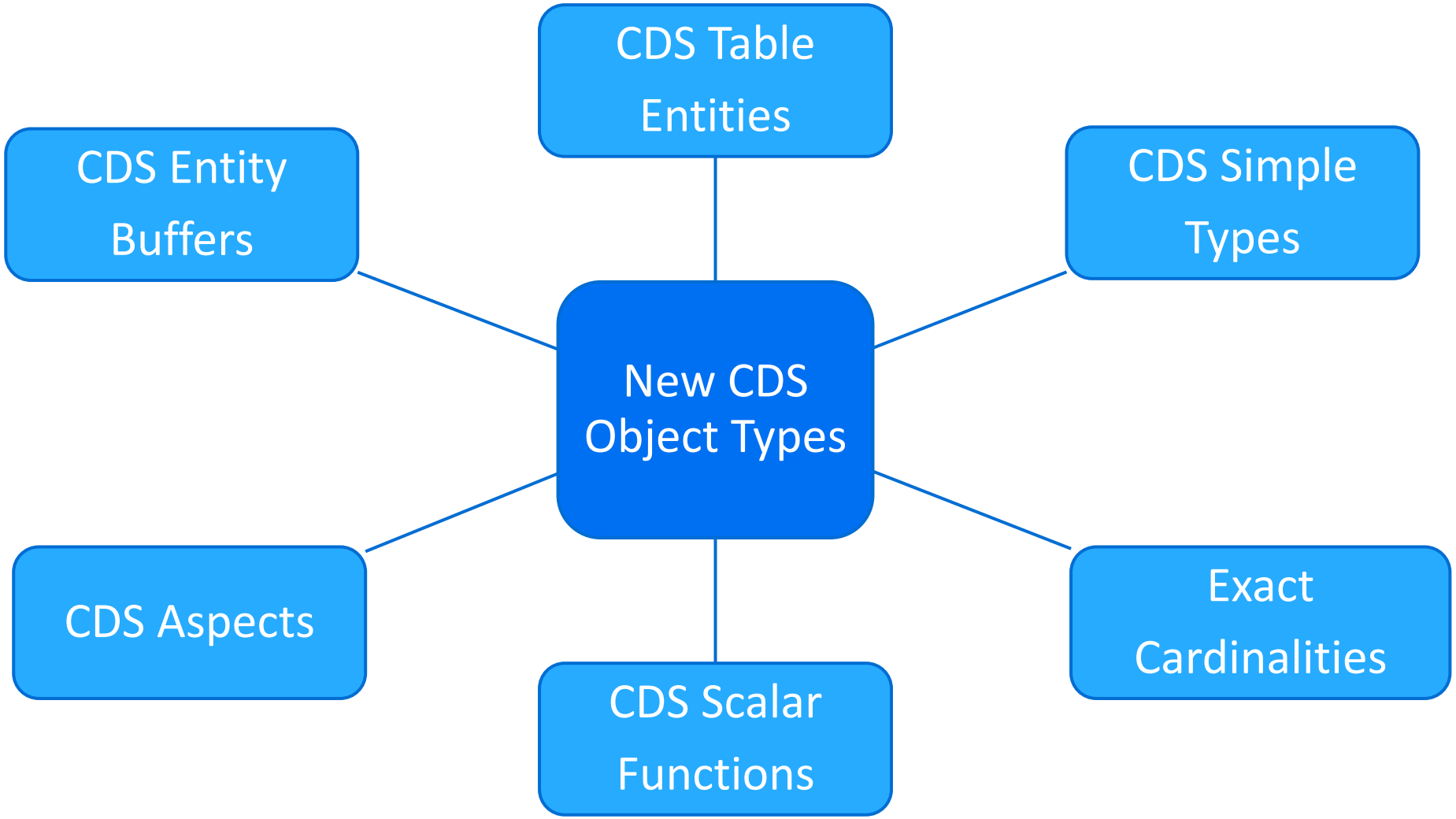


CDS-only Data Modeling -
Harnessing New ABAP CDS Object Types for Modern RAP Applications

Katharina Altrichter, Developer
Matthias Herchenroether, Development Manager
June 4th, 2025

PUBLIC


CDS Table Entities
CDS Entity Buffers
CDS Simple Types
New CDS Object Types
CDS Aspects
Exact Cardinalities
CDS Scalar Functions

Similar look and feel across whole data model

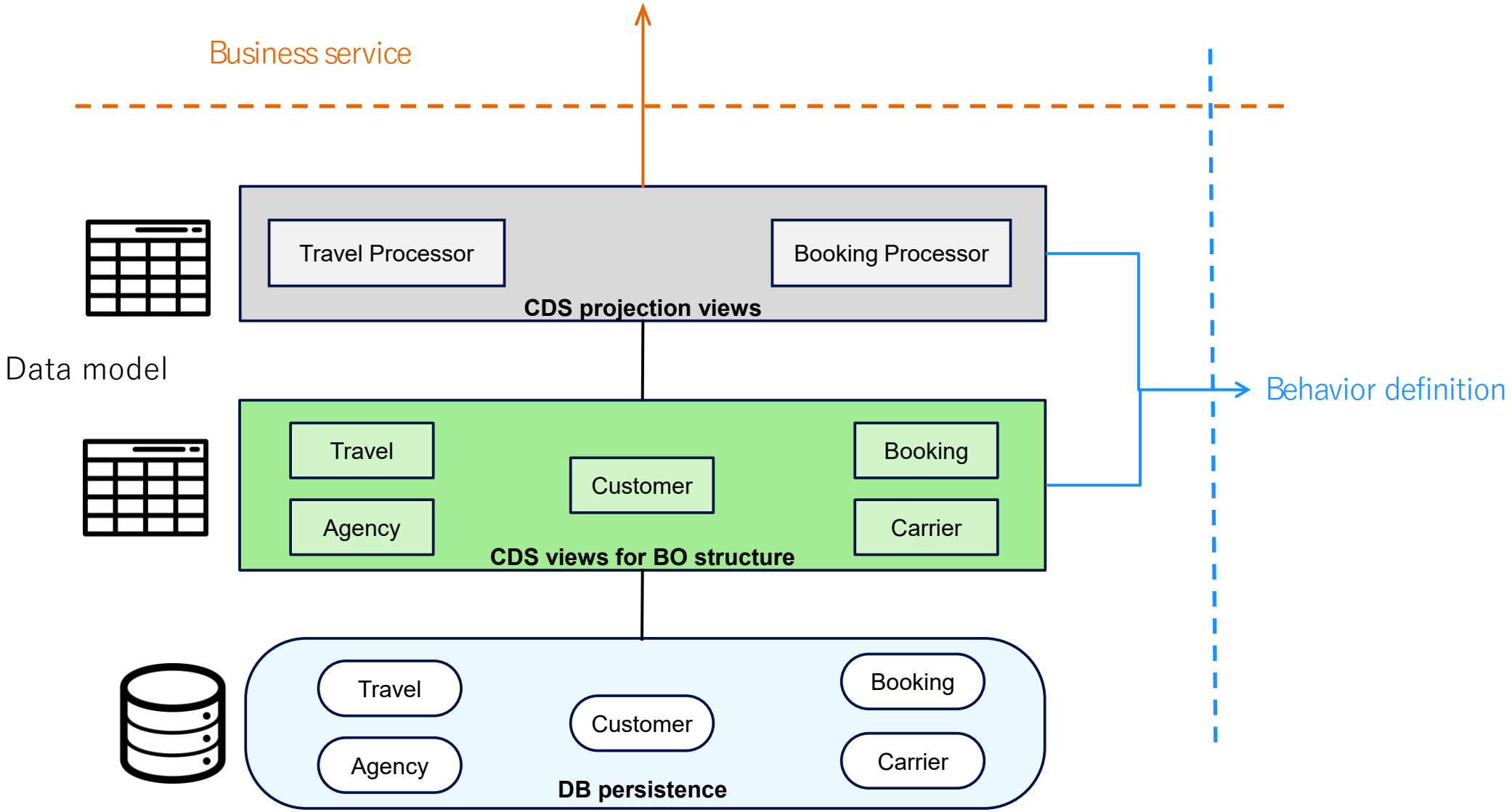
Better integration of different CDS types

CDS-only Data Modeling

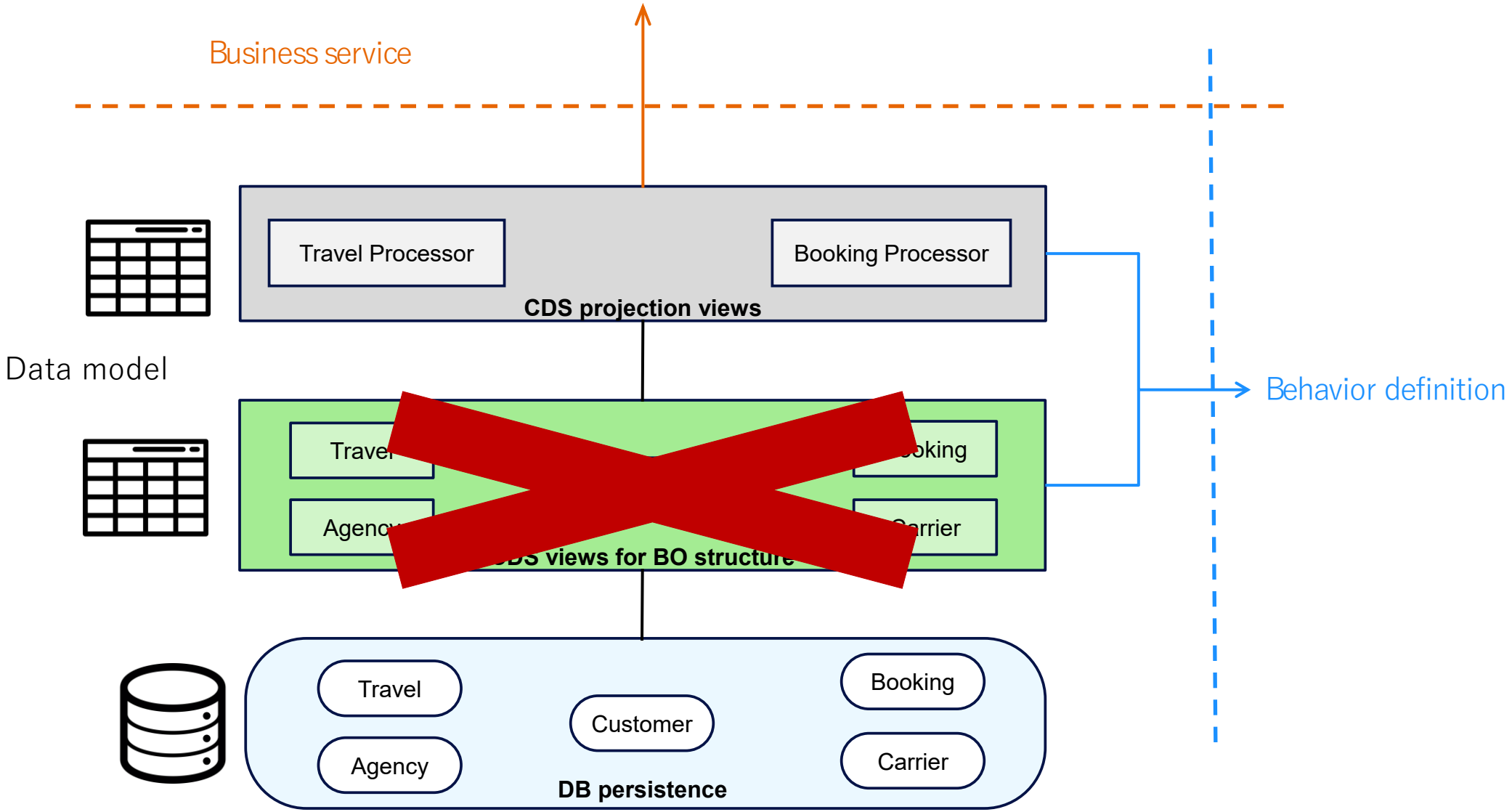
Reduced complexity of data models

Enable performance tuning

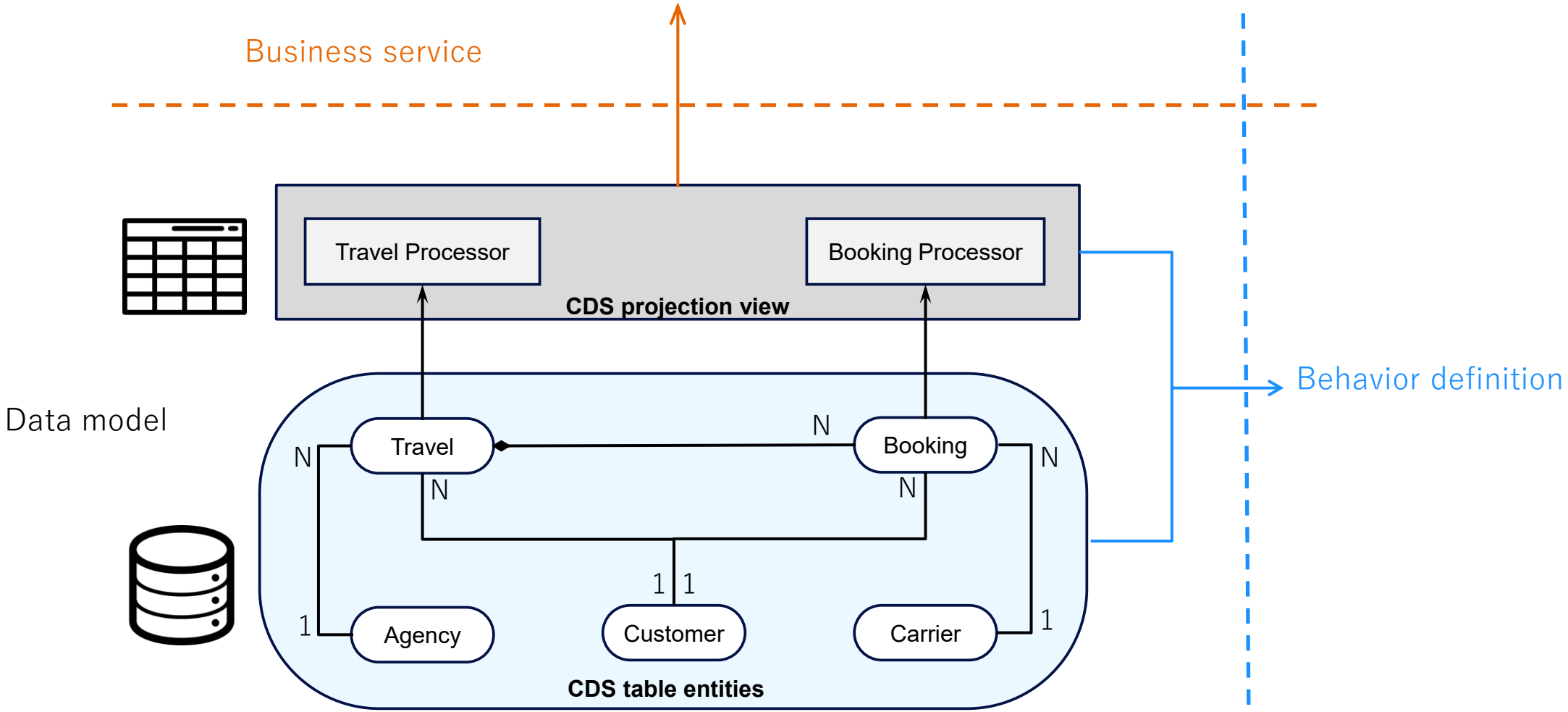
Example data model: SFLIGHT (simplified)



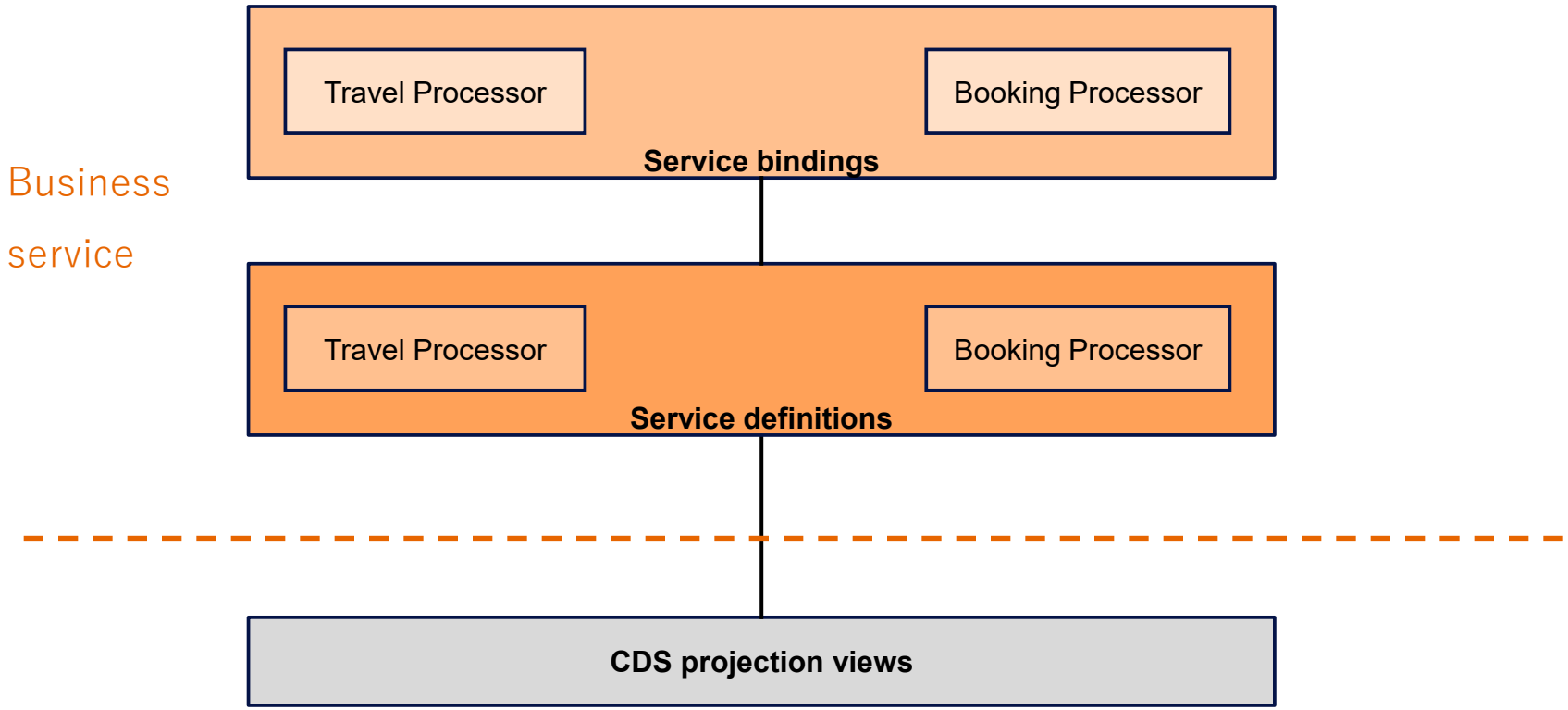
Example data model: SFLIGHT (simplified)



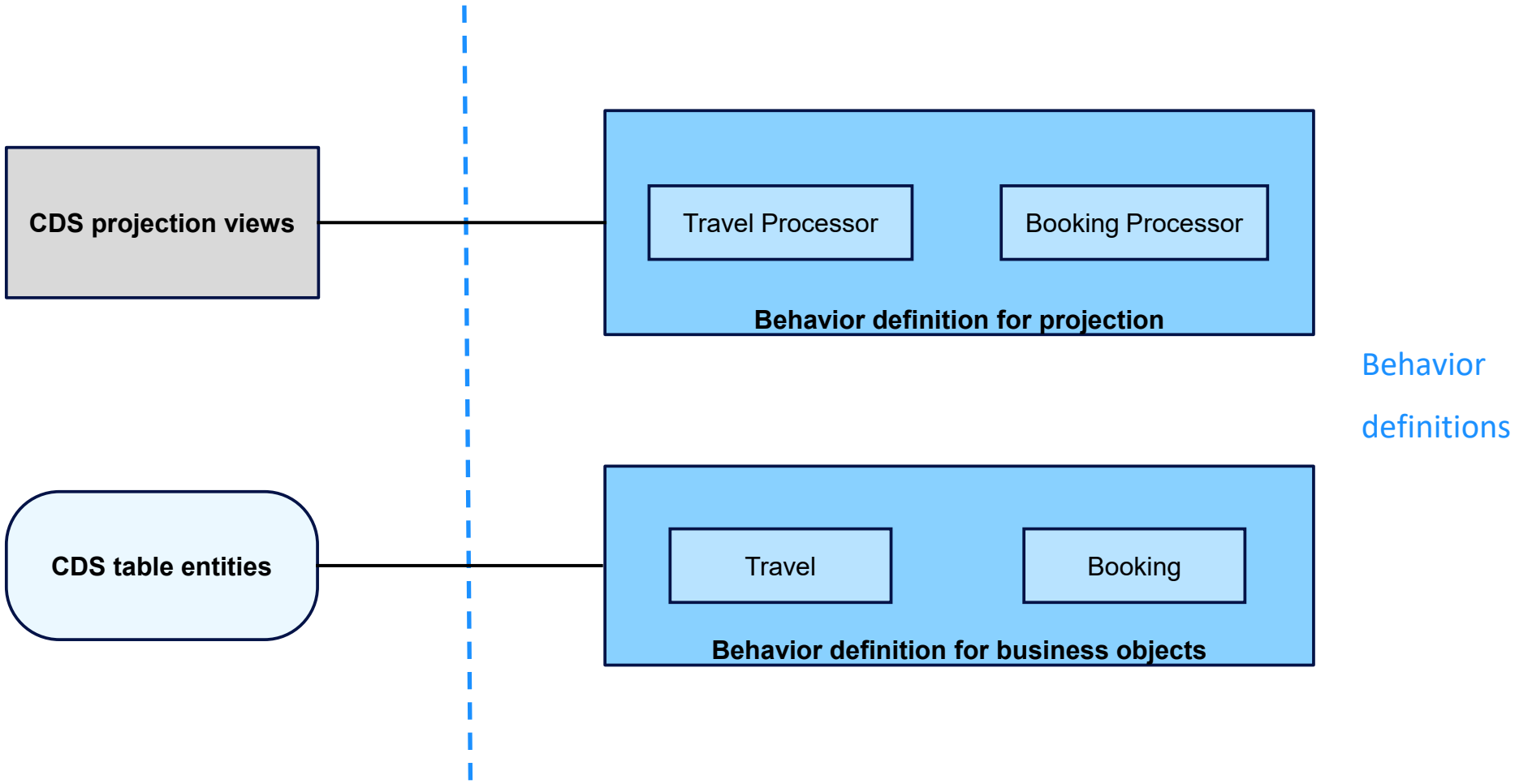
Example data model: SFLIGHT (simplified)



Example data model: SFLIGHT (simplified)

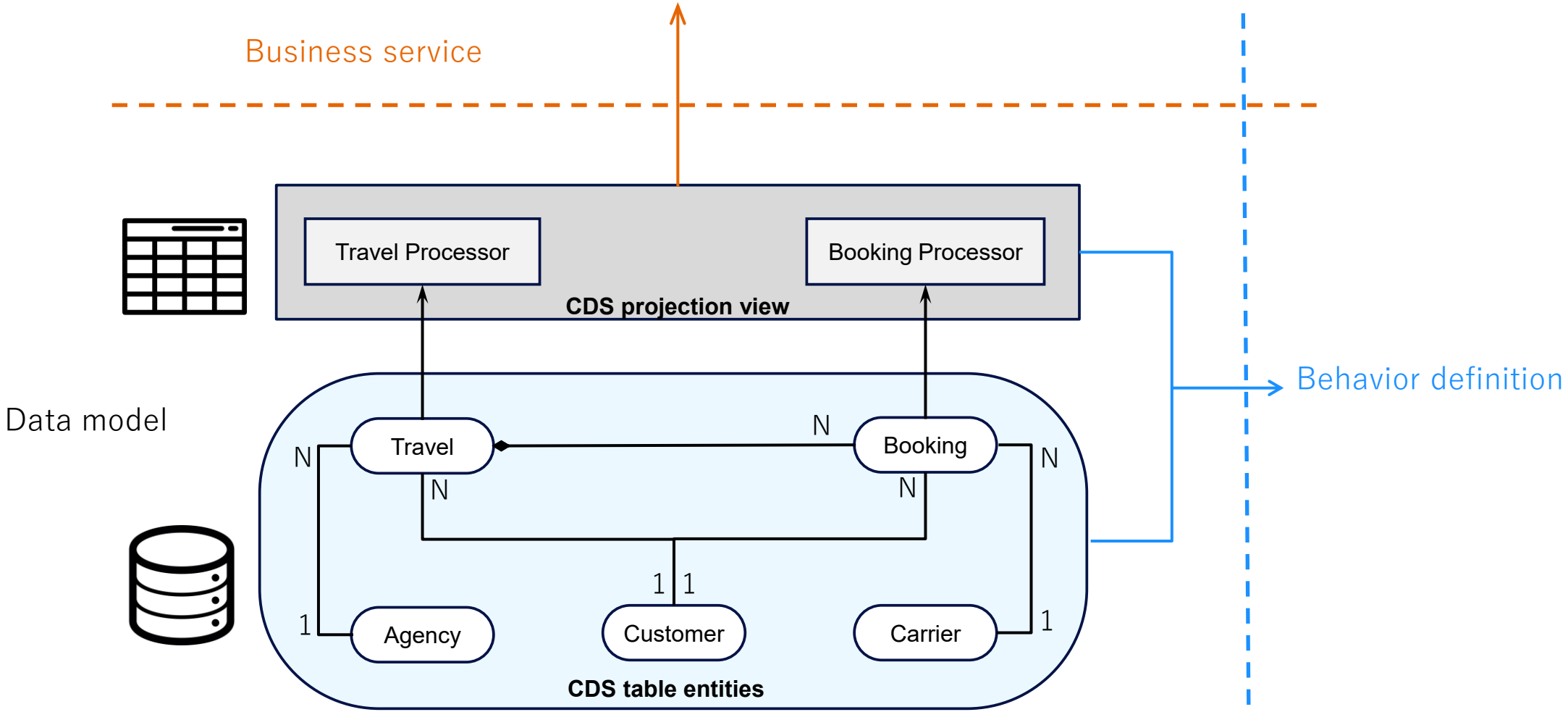


Example data model: SFLIGHT (simplified)



Behavior definitions

Example data model: SFLIGHT (simplified)



CDS Table Entities

```
@ClientHandling.type: #CLIENT_DEPENDENT
@AbapCatalog.deliveryClass: #APPLICATION_DATA
@AccessControl.authorizationCheck: #NOT_REQUIRED
@EndUserText.label: 'Flight Ref Scenario: Managing Travels'
define table entity T_DD_TSM_TRAVEL_MANAGED
{
  key TravelID : abap.numc(8);
      AgencyID : abap.numc(6);

  @Semantics.amount.CurrencyCode : 'CurrencyCode'
  TotalPrice : DD_TSM_TOTAL_PRICE;
  CurrencyCode : abap.cuky;
  Description : abap.string null;


  ...
  //Associations
  _Booking: composition exact one to many T_DD_TSM_BOOKING_MANAGED;
  _Agency:  association of many to exact one
                              T_DD_TSM_AGENCY on $projection.AgencyID = _Agency.AgencyID;
}
```

Header **annotations** for delivery class and client handling

Support for **CDS simple types**

Improved **nullability handling**

Modeling of (to-parent) **associations** and **compositions**

CDS table entities are the CDS-based successor of classic dictionary tables!

CDS Exact Cardinalities

```
{

...
//_Agency:  association of [1...1] to T_DD_TSM_AGENCY on $projection.AgencyID =
_Agency.AgencyID


_Agency:  association of many to exact one
                          T_DD_TSM_AGENCY on $projection.AgencyID = _Agency.AgencyID

...

}
```

Numeric notation replaced by **exact notation**

Specified **exact cardinalities** can lead to significant **performance improvements on HANA!**

CDS Simple Types

```
@EndUserText.label: 'Flight Ref Scenario: Price'
define type DD_TSM_PRICE: abap.curr(16,2)
```

Header **annotation** support for semantic enrichment

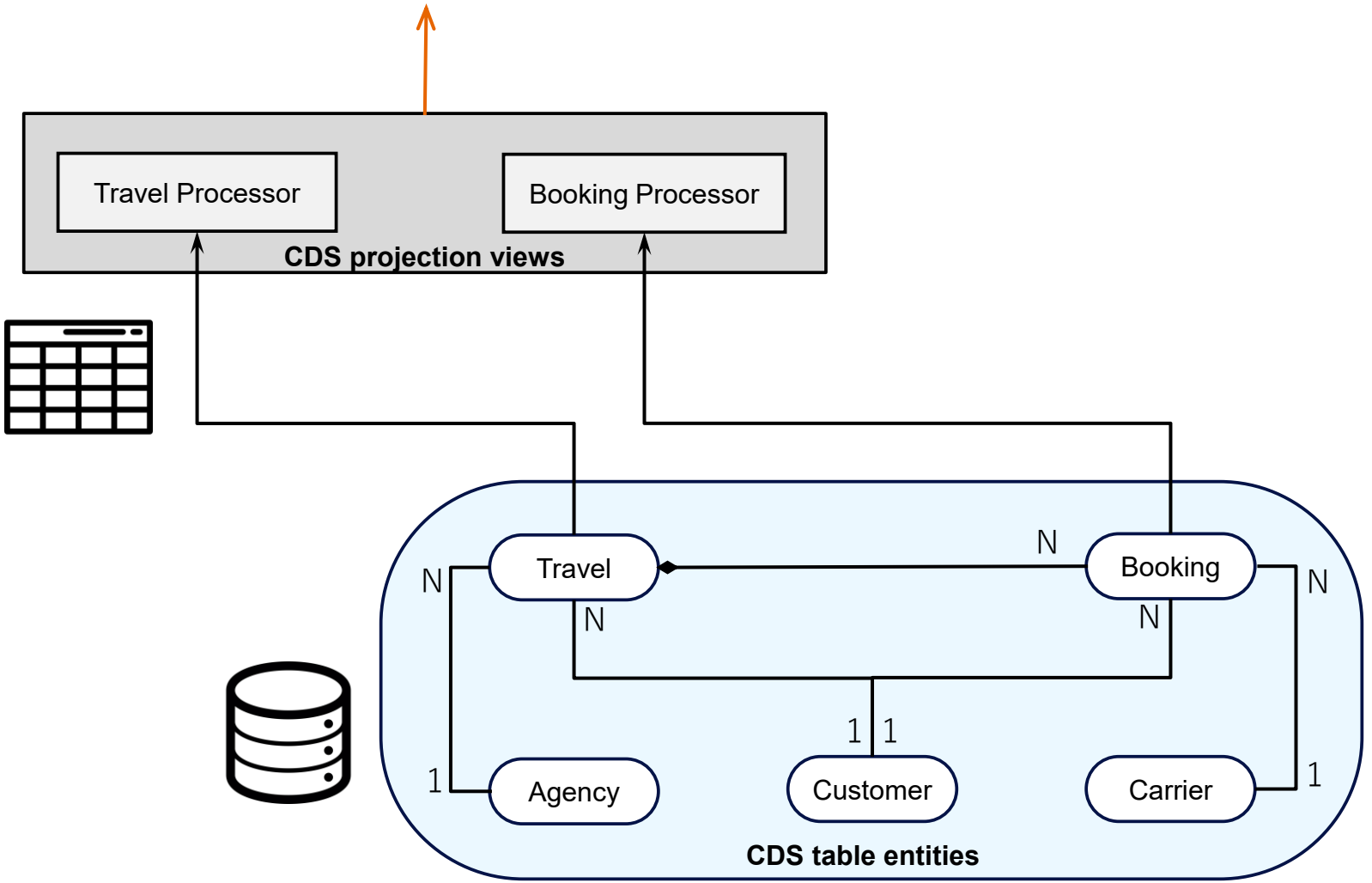
Type declaration
(wrapping of data elements possible)

```
@EndUserText.label: 'Flight Ref Scenario: Total Price'
define type DD_TSM_TOTAL_PRICE: DD_TSM_PRICE
```

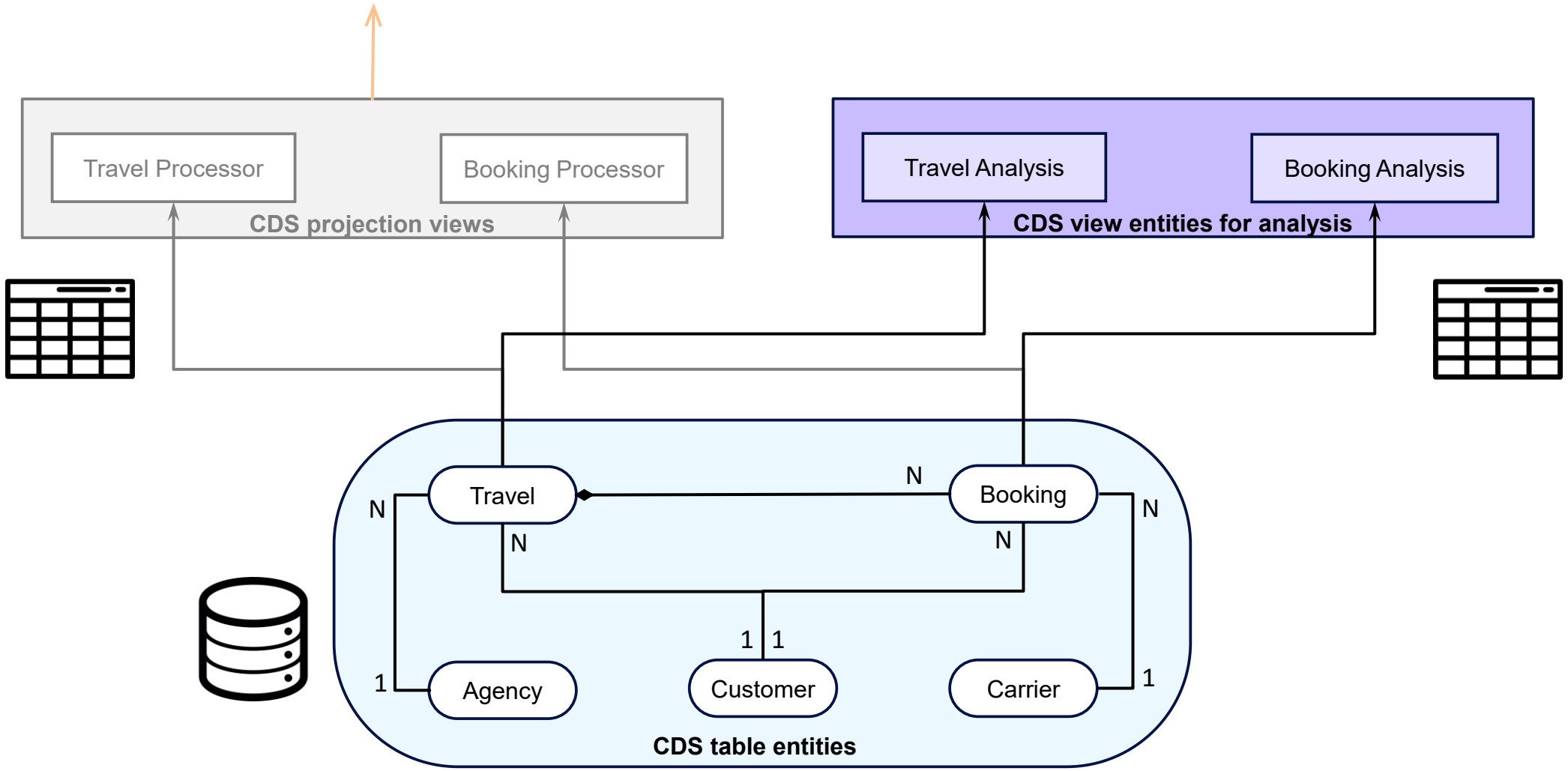
Support of **nested simple types**

CDS simple types are the CDS-based successor of data elements!

Expanding the data model



Expanding the data model



CDS Scalar Functions

```
define scalar function TSM_GET_WEEKDAY
  with parameters

    input_date : abap.datn

returns abap.char(3)
```

Scalar function **name**

Parameter definition
parameter names, datatypes (optionally also reference type support)

Definition of **result type** *(optionally also reference type support)*

A **CDS Scalar Function** consists of a **function definition** and engine-specific **function implementation reference!**

CDS Aspects

```
@EndUserText.label: 'Flight Ref Scenario: calculate update'
define aspect A_DD_TSM_CALCULATE_UPDATE
{
    BeginDate : abap.datn;
    EndDate   : abap.datn;
    LastChangedAt : DD_TSM_LAST_CHANGED_AT;

    BeginWeekday = TSM_GET_WEEKDAY( input_date => BeginDate );
    EndWeekday = TSM_GET_WEEKDAY( input_date => EndDate );
    ...

    HoursSinceLastUpdate = cast(
        utcl_seconds_between (LastChangedAt, utcl_current()) / 3600
        as abap.dec(16,0));

    ...
}
```

Header **annotation** support for semantic enrichment

Support of **various datatypes** *(Data elements, built-in and CDS simple types, reference types, (typed) literals)*

Support of **functions** (builtin, scalar) for calculations

CDS aspects are CDS-based reuse artefacts!

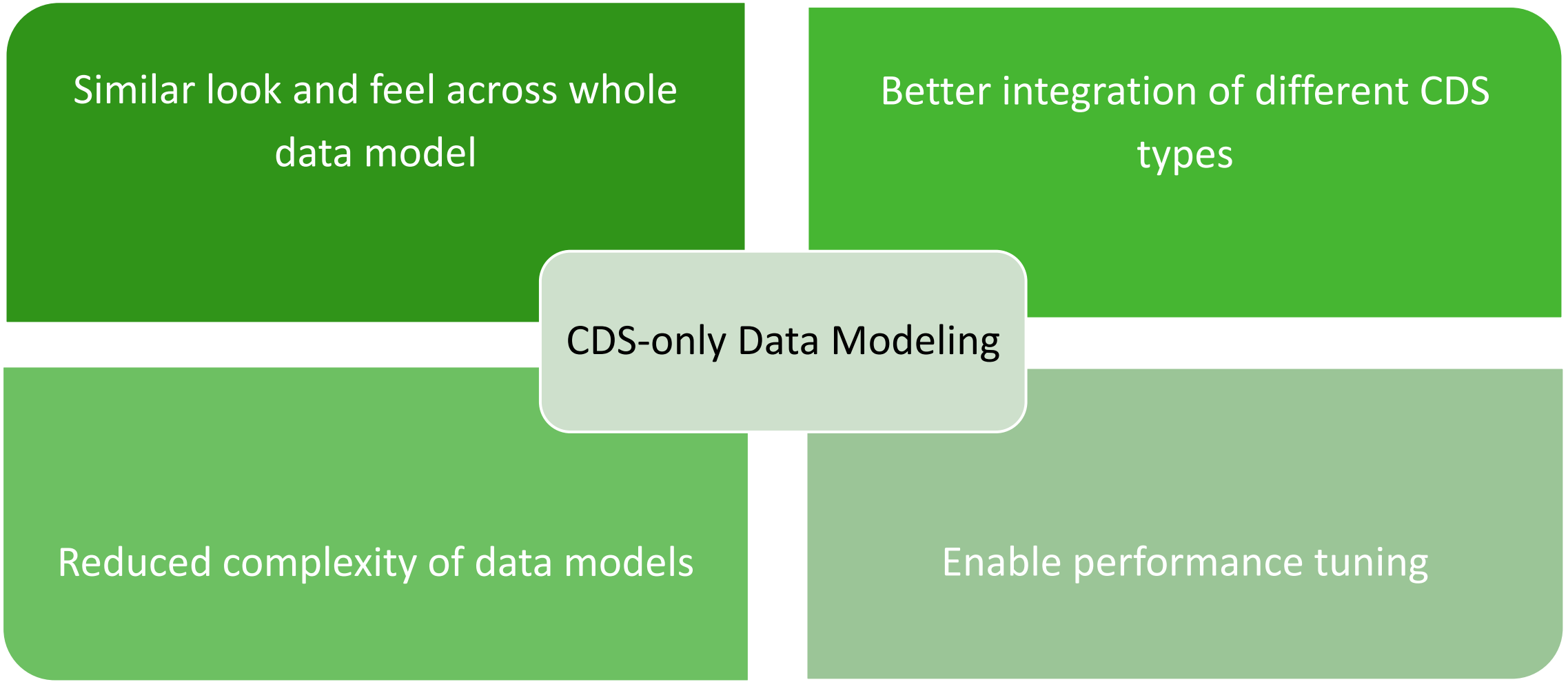
CDS Buffer Propagation

```
propagate view entity buffer on I_DD_TSM_BOOKING_ANALYSIS
{
  FirstName, LastName
}
```

Buffered **entity**

Names of **propagated fields**

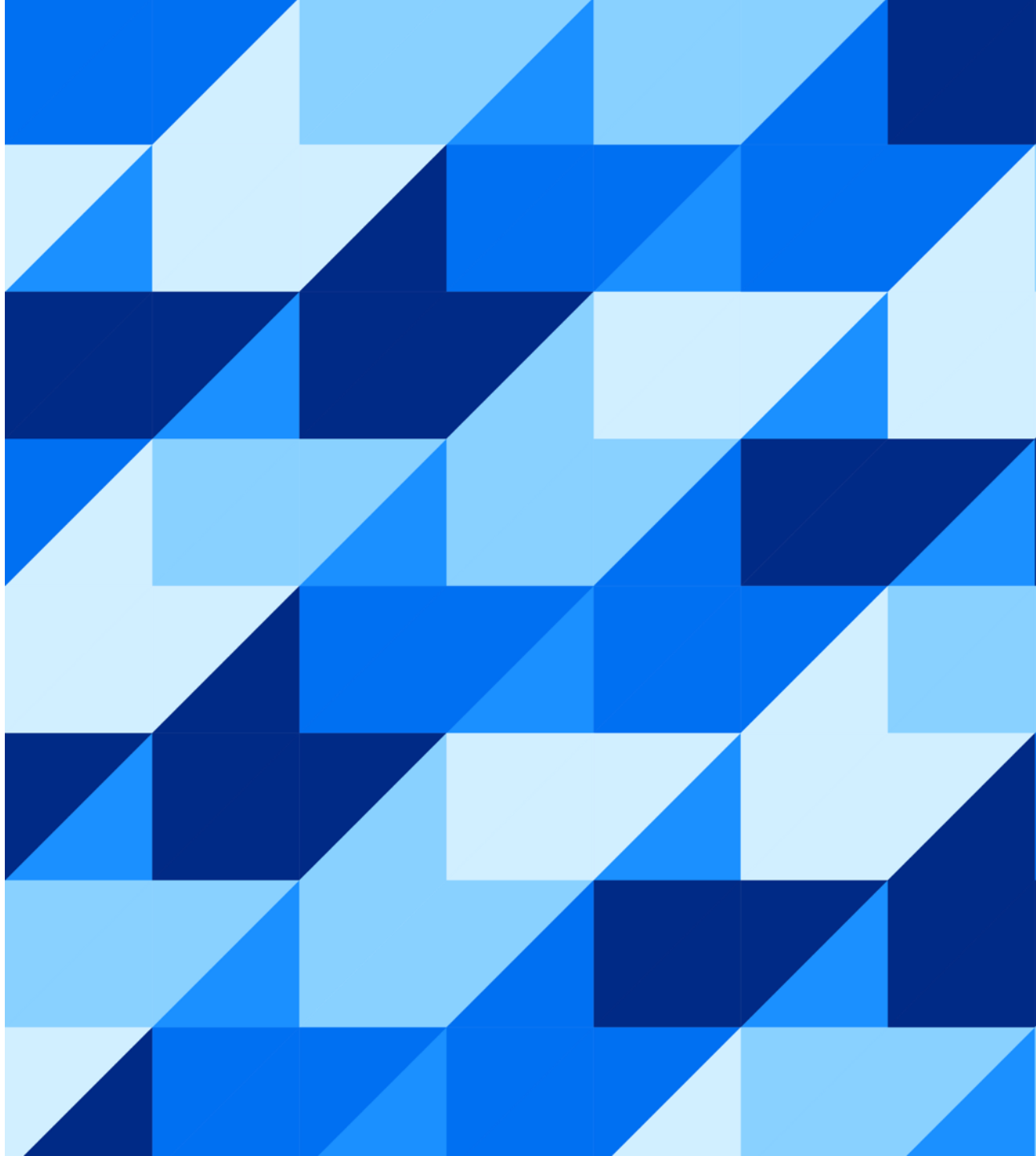
Buffer propagation allows to leverage buffers being defined in the underlying data sources of the base entity!


Similar look and feel across whole data model
Better integration of different CDS types
CDS-only Data Modeling
Reduced complexity of data models
Enable performance tuning

Release info and documentation

- CDS simple types and CDS scalar functions already available with ABAP release 7.58 (e.g. S/4 OP 2023)
- Everything else shown is available in current cloud releases and will be available in On Premise with the next ABAP release 8.16 (i.e. S/4 OP 2025)
- Links to latest cloud docu:
  - CDS table entities
  - CDS simple types
  - CDS aspects
  - CDS scalar functions
  - CDS entity buffers
  - Exact cardinalities

  Further information can also be found in the ABAP data modeling guide



Thank you.

Contact:
Katharina Altrichter (Katharina.Altrichter@sap.com)
Matthias Herchenroether (Matthias.Herchenroether@sap.com)