

Managing the Euclid Data Model

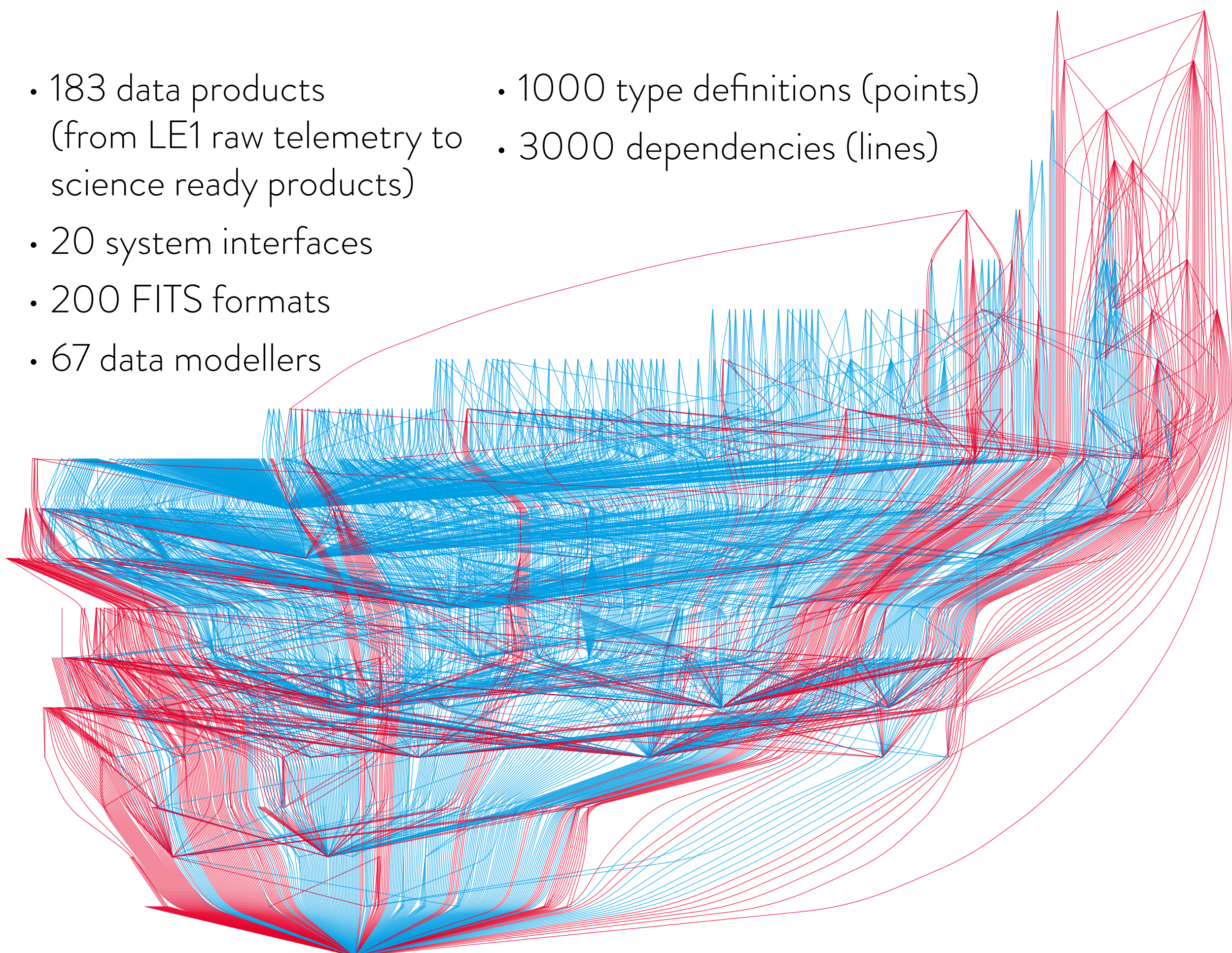


Christophe Dabin, Edwin A. Valentijn

Teake Nutma

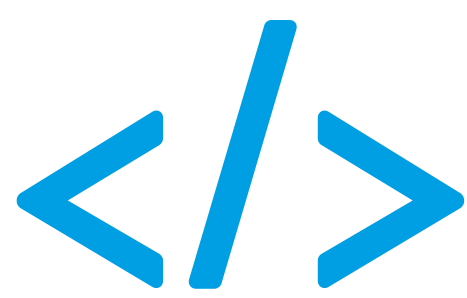
The Euclid Data Model

- 183 data products (from LE1 raw telemetry to science ready products)
- 20 system interfaces
- 200 FITS formats
- 67 data modellers
- 1000 type definitions (points)
- 3000 dependencies (lines)



● Data Product definitions ● Other (interfaces etc)

Restrictions on modelling



- Metadata database schema is derived from the Data Model.
- Numeric types in the database are always bounded in size.
- But e.g. the XSD type `xs:integer` is unbounded!
- So impose XML modelling rule: always use bounded XSD types.

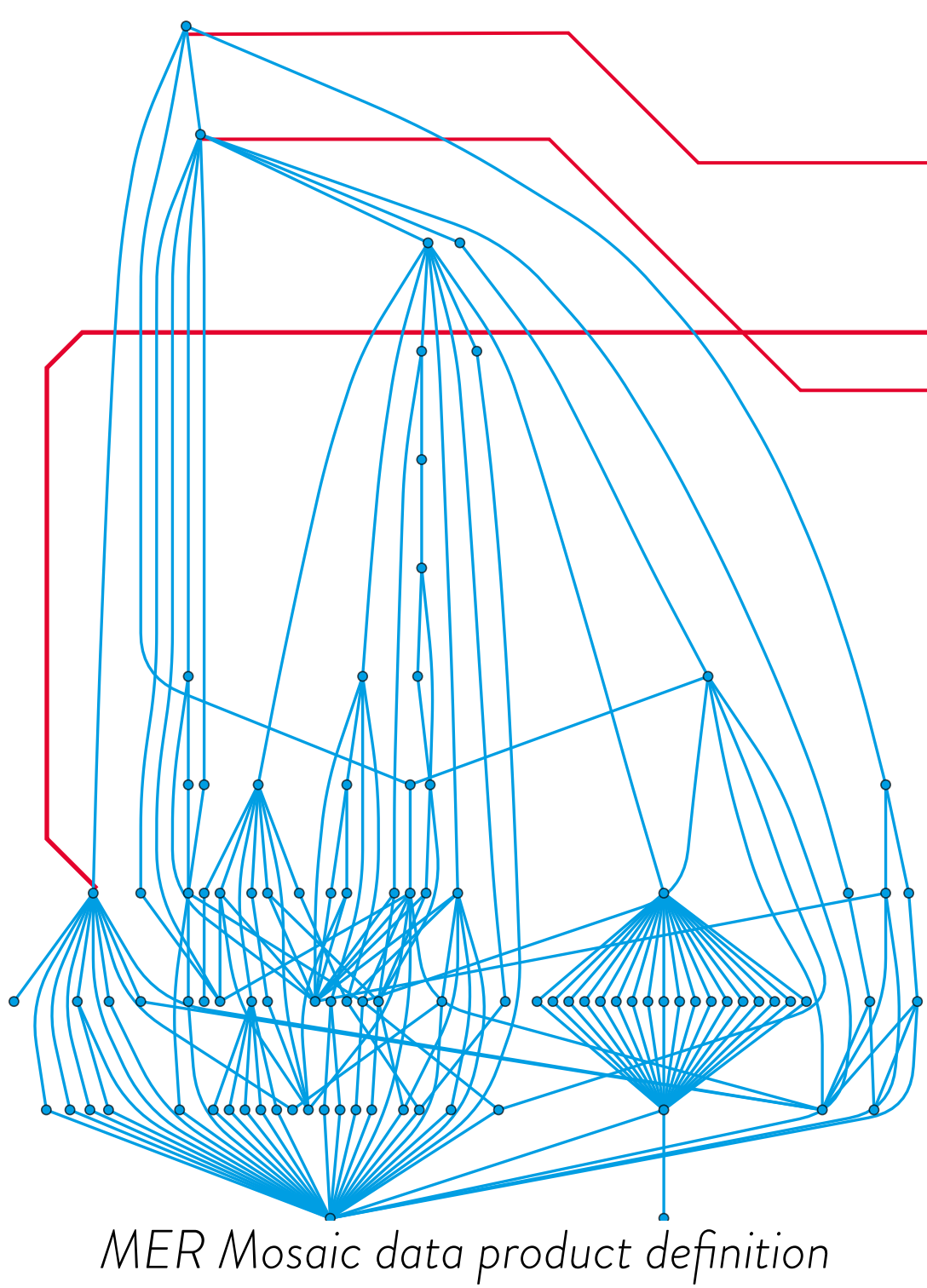
- All rules are collected in a Word document.
- Simplifies readability, reusability, and implementation of the DM.
- 74 pages, 60 rules.
- No-one reads it, let alone adheres to it.



- I wrote a CLI program that validates the Data Model w.r.t. all rules.
- Ensures database schema can be build.
- Data modellers can run it locally.
- But nobody does that!

Data Model implementation

- The Euclid Data Model is written in XSD (XML Schema Definition).
- Common data types rely on IVOA standards.
- The Data Model defines the structure of data passed between ground segment components and pipelines.
- It dictates the schema of the metadata database.



```
<?xml version='1.0' encoding='UTF-8'>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="DpdMerMosaic" type="dpdMerMosaic"/>
  <xs:complexType name="dpdMerMosaic">
    <xs:sequence>
      <xs:element name="Header" type="genericHeader"/>
      <xs:element name="Data" type="merMosaic"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="merMosaic">
    <xs:complexContent>
      <xs:extension base="stackedImage">
        <xs:sequence>
          <xs:element name="TileIndex" type="xs:integer"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:schema>
```

XSD Data Model file

Validation

Python database bindings generation

```
<DpdMerMosaic>
  <Header>
    <!-- ... -->
  </Header>
  <Data>
    <!-- ... -->
    <TileIndex>741773</TileIndex>
    <!-- ... -->
  </Data>
  <!-- ... -->
</DpdMerMosaic>
```

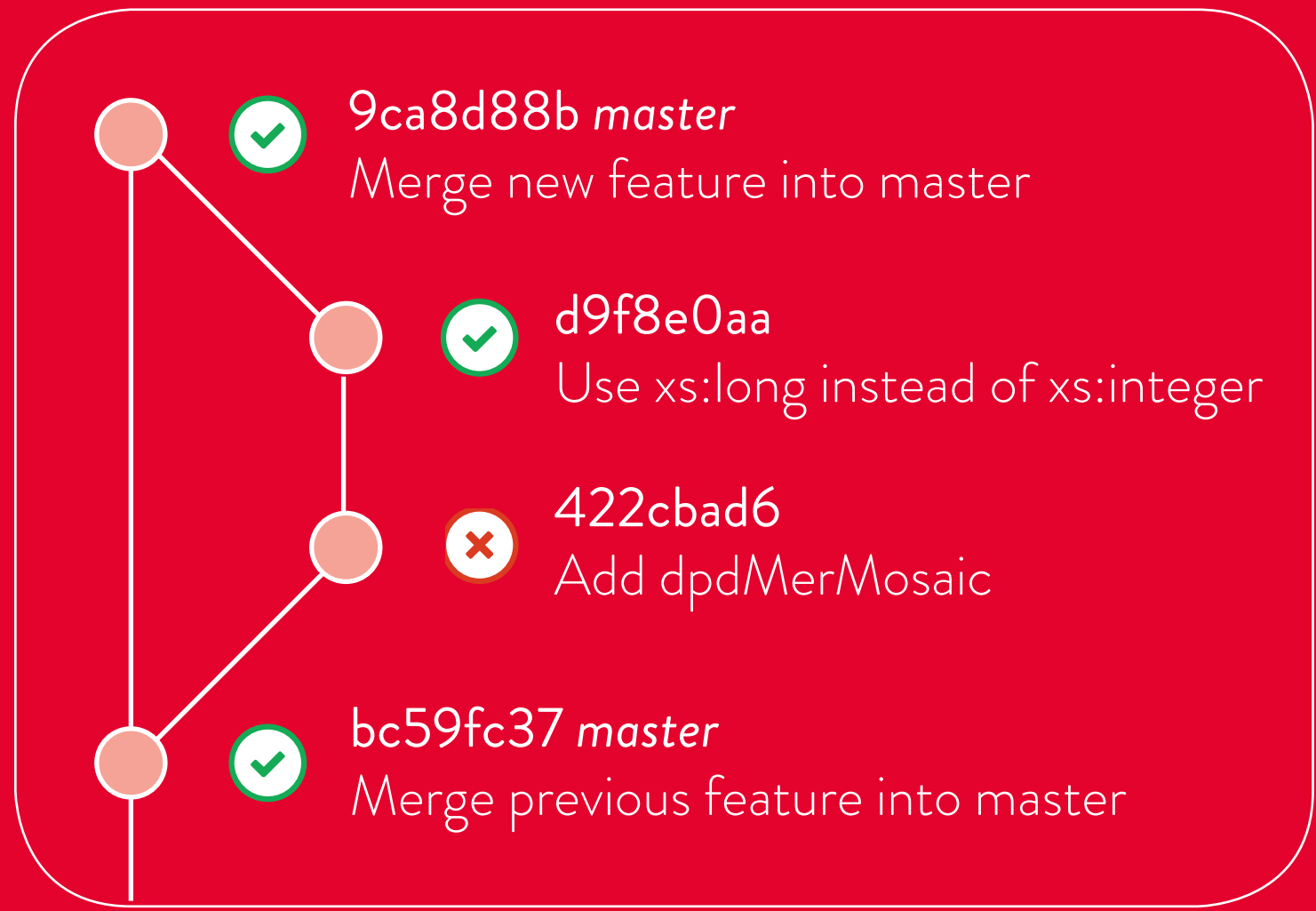
XML metadata file

```
from wise_orm import DBOObject, persistent
# ...
class MerMosaic(StackedImage):
    tile_index = persistent("int, None")
    # ...
class DpdMerMosaic(DBObject):
    header = persistent("GenericHeader, None")
    data = persistent("MerMosaic, None")
    # ...
```

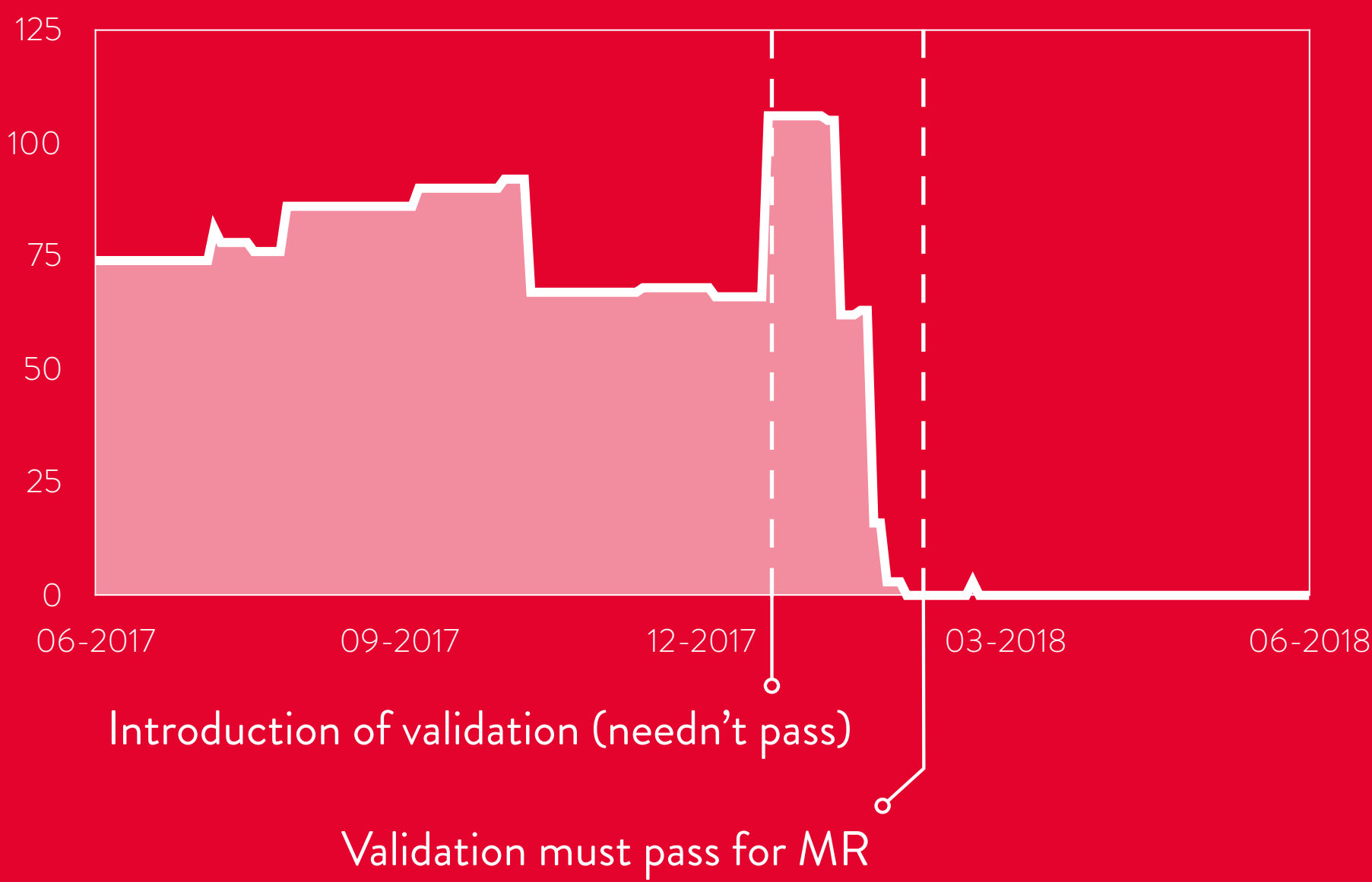
Archive Python XML-DB binding

Validation via Continuous Integration

- CLI tool is automatically run on every commit.
- Validation must have passed before merging.



Data Model errors over time



- Only Data Model custodians can merge feature branches into master
- No more errors in the Data Model!

Want to know more?

- Scan the QR code for an animated and narrated version of this poster.

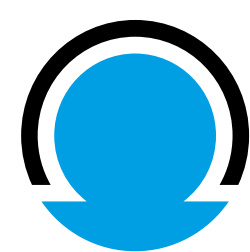


euclid



university of
 groningen

kapteyn astronomical
 institute



OmegaCEN