Directed Acyclic Graphs for Modeling Integration, and Interoperability between CMS data sources (*The Integration Database*)

Michael Case, case@ucdhep.ucdavis.edu (UC Davis)
Florida Estrella, Florida.Estrella@cern.ch (University of the West of England)
Zsolt Kovacs, Zsolt.Kovacs@cern.ch (CERN)
Frank van Lingen, frank.van.lingen@cern.ch (CalTech)
(Worked for CERN during this project)
Martin Liendl, martin.liendl@cern.ch (CERN)
Richard McClatchey,richard.mcclatchey@uwe.ac.uk (University of the West of England)
Ian Willers, ian.willers@cern.ch (CERN)

Current situation:

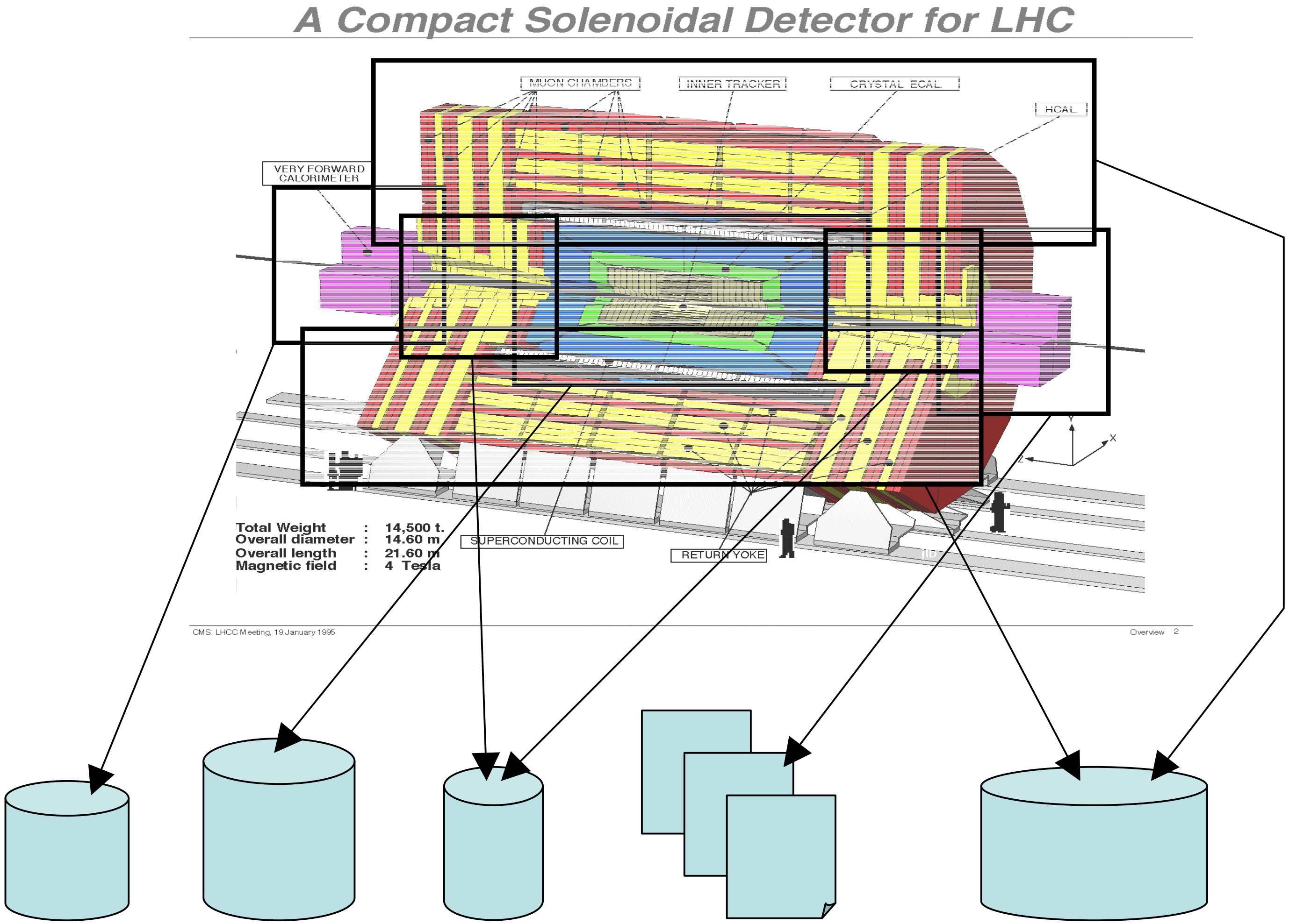
Different databases in CMS contain data that describe the detector:

- Construction db’s: measurements, tests, constants;
- Design db: geometry information;
- Calibration db’s: calibration constants;
- Detector Description db: geometry, calibration constants.
- Etc…

Within CMS approximately between 10 to 15 databases that contain detector description data.

- The “union” of these different detector description sources describe the status of the detector in a time dependent manner
- Different descriptions within the db’s can be overlapping, complementary or have a different granularity level
- Different sources can have different versions of the detector descriptions
- Examples of description data sources are:
  - DDD (simulation, and visualization)
  - ECAL (CRISTAL)
  - TrackerDB (Oracle based)
  - EDMS (Some CMS data stored here)
  - Etc..

Overlapping and complementary descriptions stored in different data sources



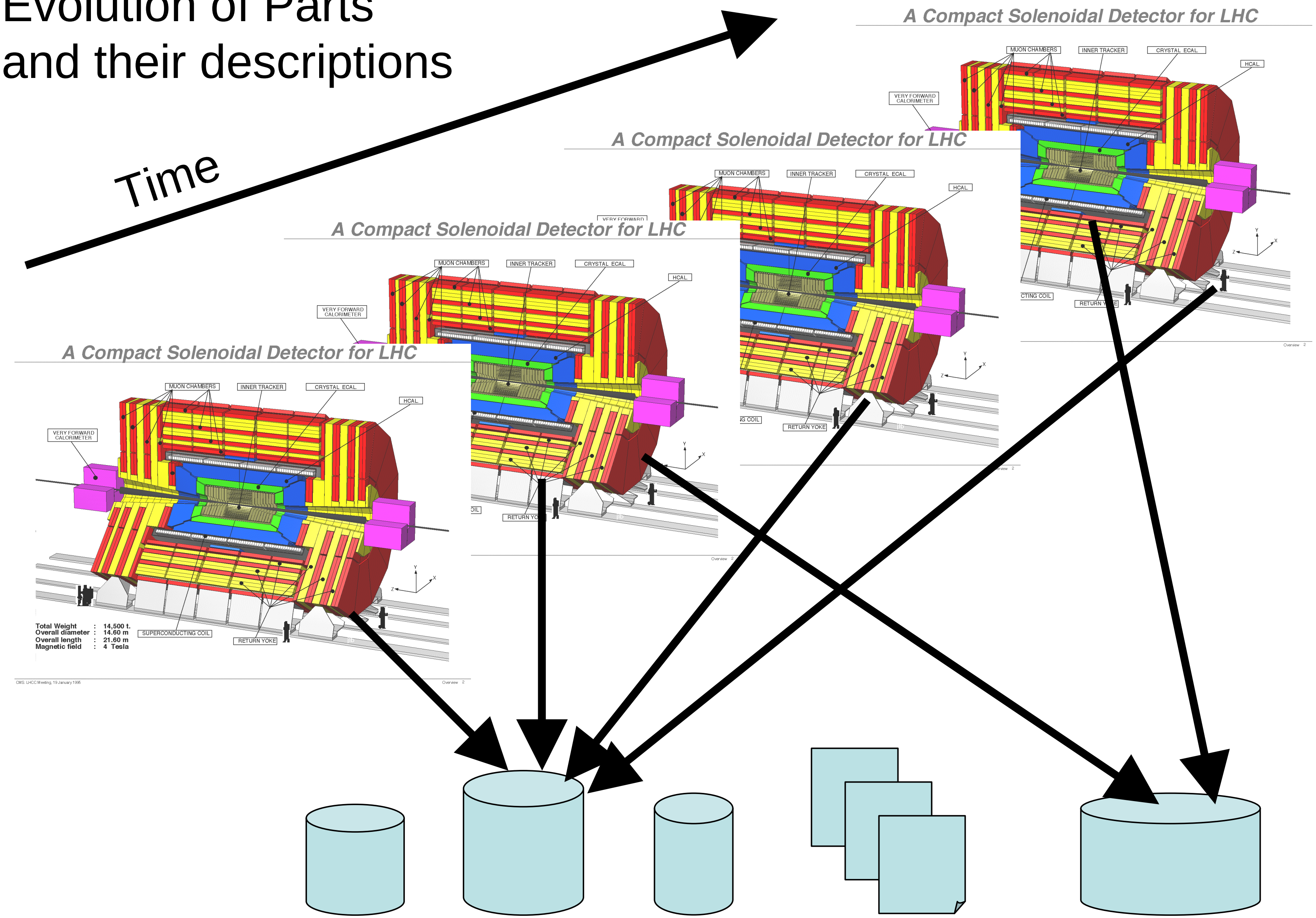
A Compact Solenoidal Detector for LHC

Total Weight	:	14,500 t.
Overall diameter	:	14.60 m
Overall length	:	21.60 m
Magnetic field	:	4 Tesla

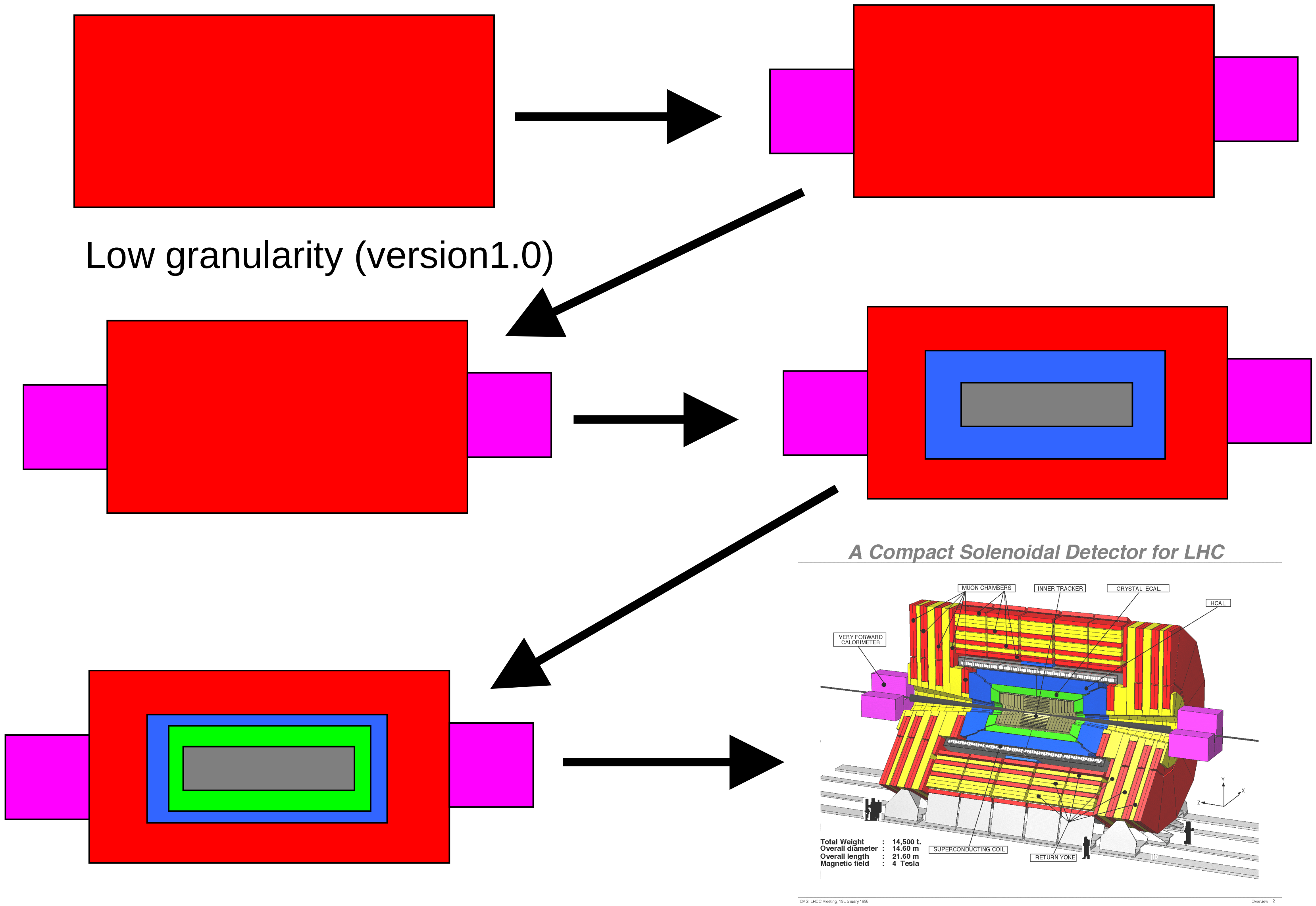
CMS LHCC Meeting, 19 January 1995

Overview 2

Evolution of Parts
and their descriptions



Granularity and Versions



Low granularity (version1.0)

A Compact Solenoidal Detector for LHC

Total Weight	:	14,500 t.
Overall diameter	:	14.60 m
Overall length	:	21.60 m
Magnetic field	:	4 Tesla

CMS. LHCC Meeting, 19 January 1995

Overview 2

High granularity (version 15.1)

The problem:

Different domains need different views on the detector description data which are stored within different heterogeneous data sources. These views are needed for calibration of the different sub detectors, physics analysis, detector simulation, detector visualization, etc…

- Currently it is difficult to create and manage views:
- No uniform mechanism to access description data

Commonalities between different description sources:

- Use of directed a-cyclic graphs to describe part hierarchies
  - Used in a generic form in: CRISTAL DB, DDD, EDMS.
- Versioning of part hierarchies:
  - Used in a generic form in: CRISTAL DB, DDD, EDMS
- Workflow management:
  - Used in a generic form in: CRISTAL DB

Differences between description sources:

- Attach different types of data to the part hierarchies
- There is no relation between sources that contain descriptions related to the same “physical” CMS detector
- Different granularities of description
- No version management over multiple description sources

(minimum) requirements for an integration database (based on existing description sources):

1. **Support for hierarchical part structures**
2. **Support for navigation through these hierarchies**
3. **Ability to attach data to the nodes in the hierarchy**
4. **Version management of the different hierarchies**
5. **Ability to correlate descriptions of different granularities**
6. **Ability to “merge” complementary descriptions**
7. **Evolution of parts (is necessary to track the status of different parts (INB rules))**
8. **Version and view management of multiple sources with multiple versions (Multi-versions)**
9. **Exploded view support**

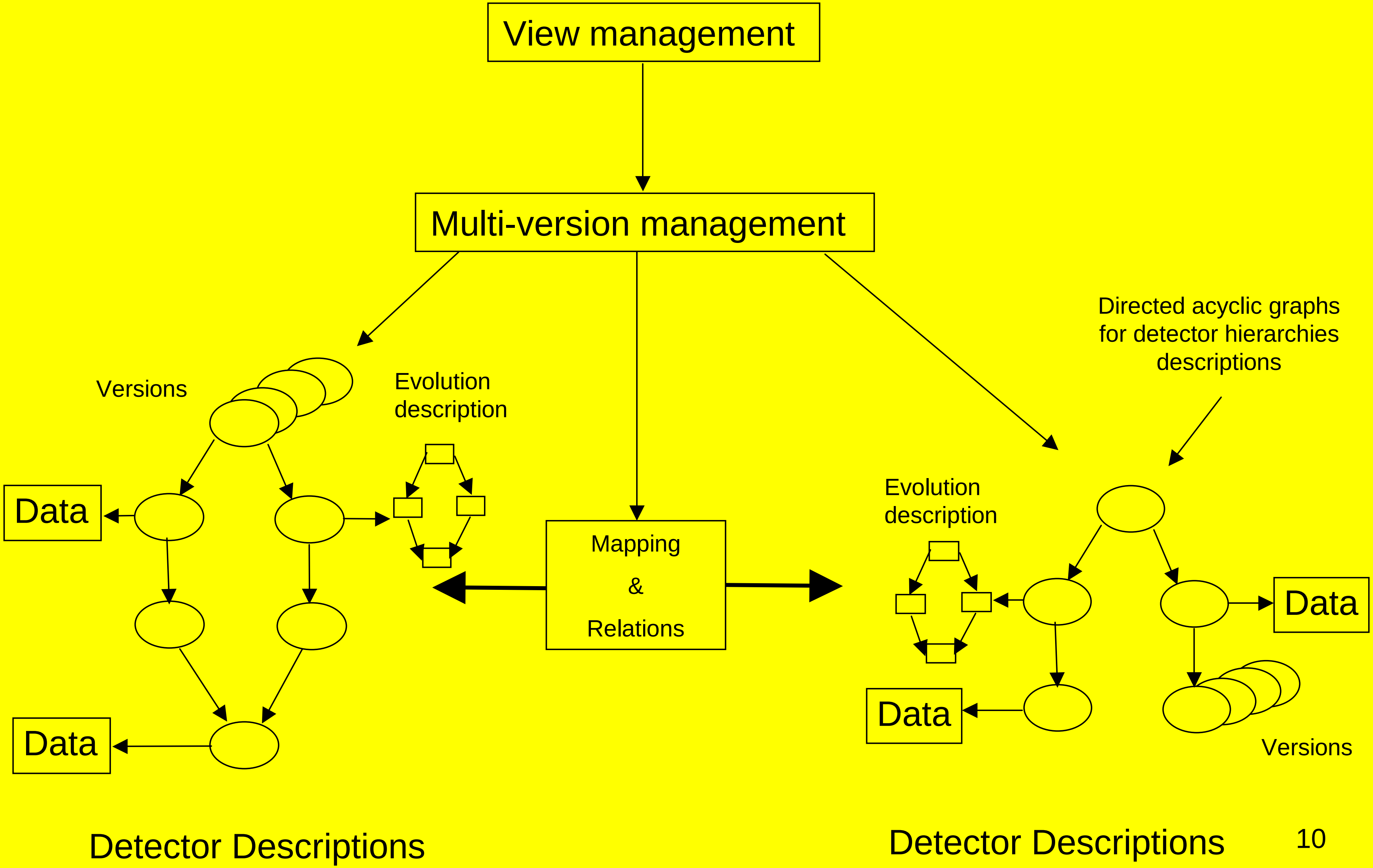
1,2,3,4, are supported by CRISTAL

(7 is partly supported by workflow functionality in CRISTAL)

1,2,3,4,9 are supported by DDD

5,6,8 were implemented within prototypes discussed on this poster

Generic data model for integration of heterogeneous description sources:



Two small prototypes have been constructed:

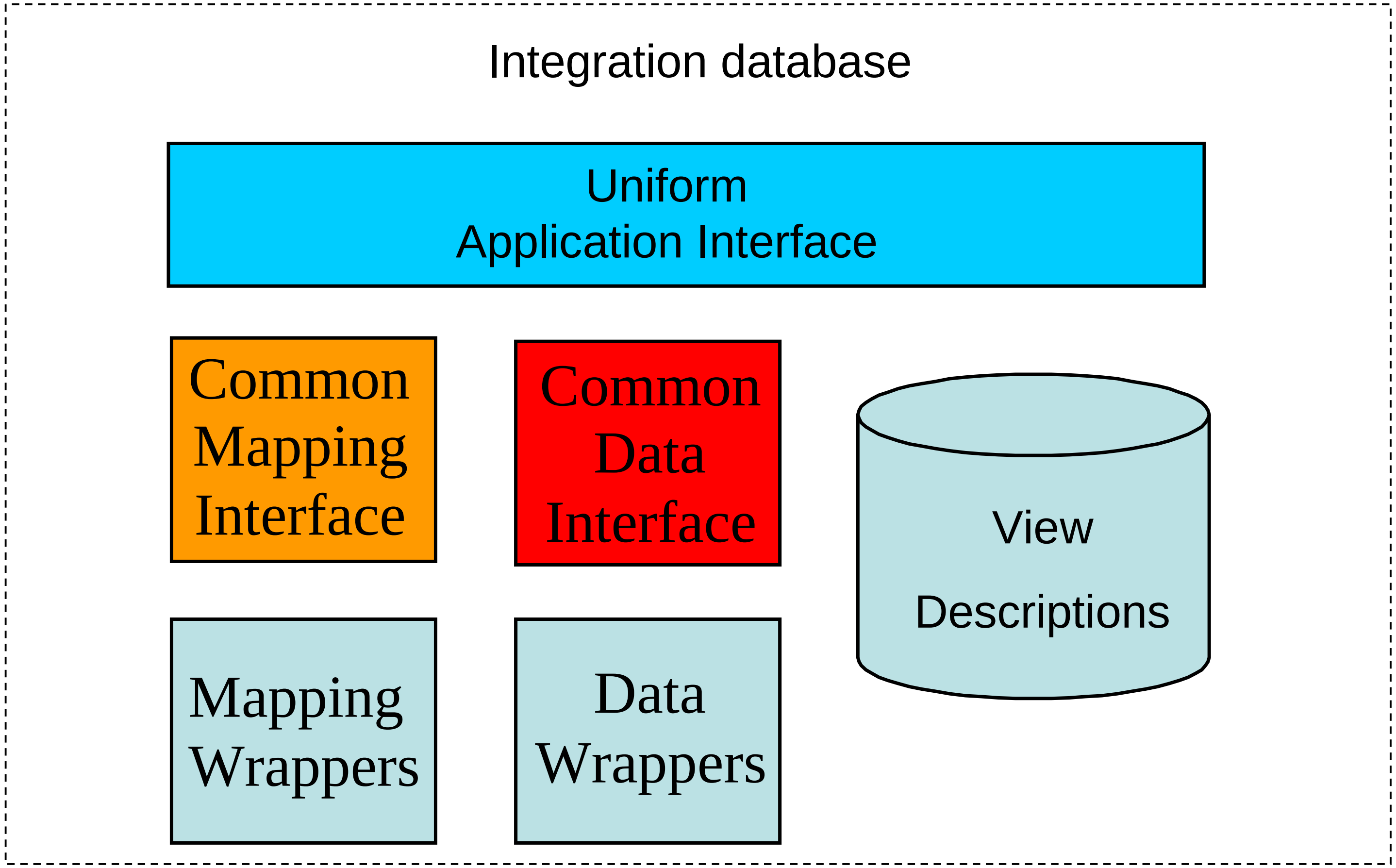
Decentralized: Preserve the autonomy of the different description sources. Focus on integration of data, and multiple version management

Centralized: Migrate all descriptions from the different sources to one database. Focus on data model.

Within both approaches it is not necessary to migrate all the data stored in the detector description sources to the integration database, since certain data is only used in the domains that manage a particular database

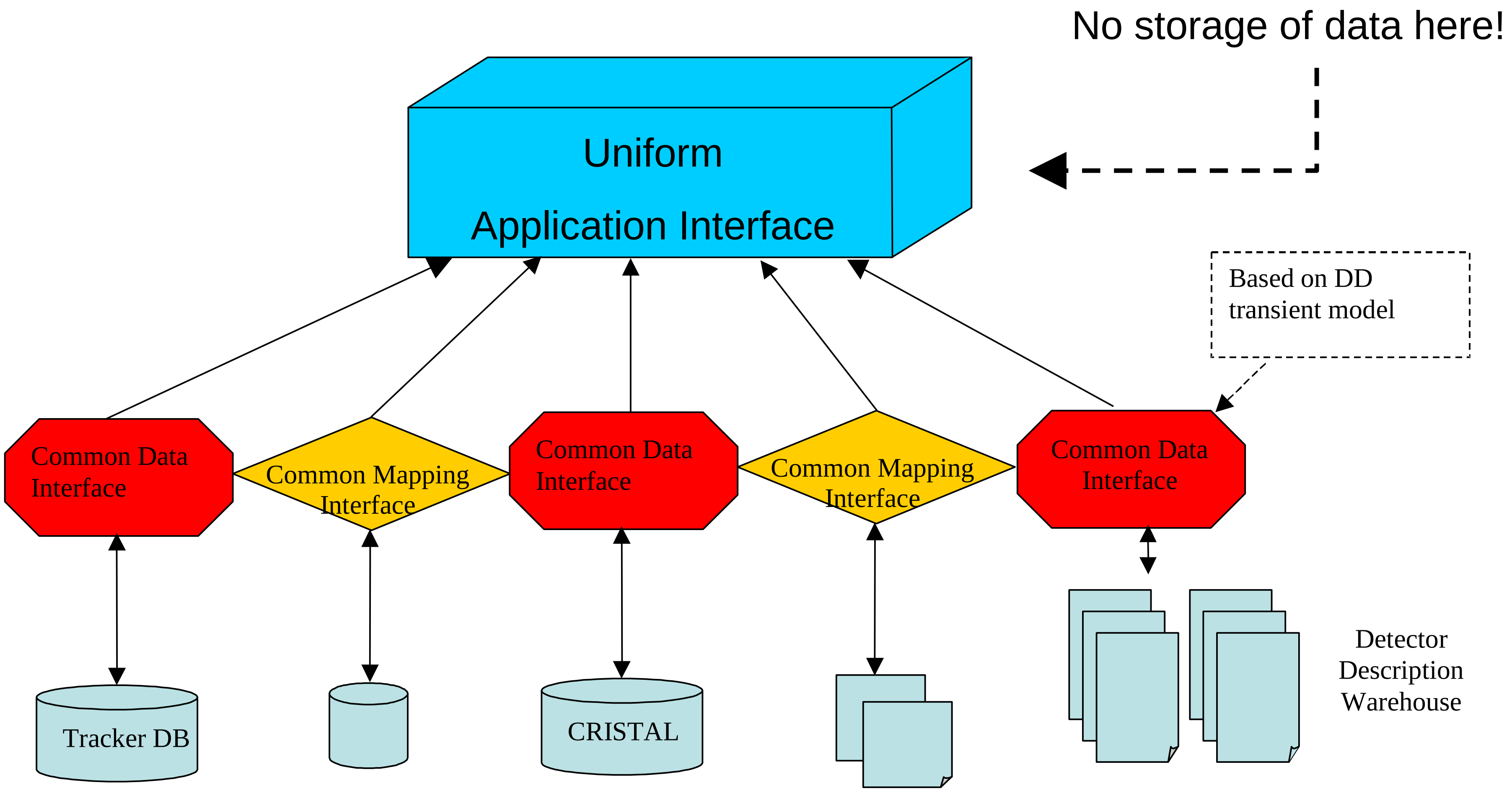
Prototypes based on the ideas of CRISTAL and DDD
(DDD is discussed on a poster elsewhere in this room)

Decentralized approach



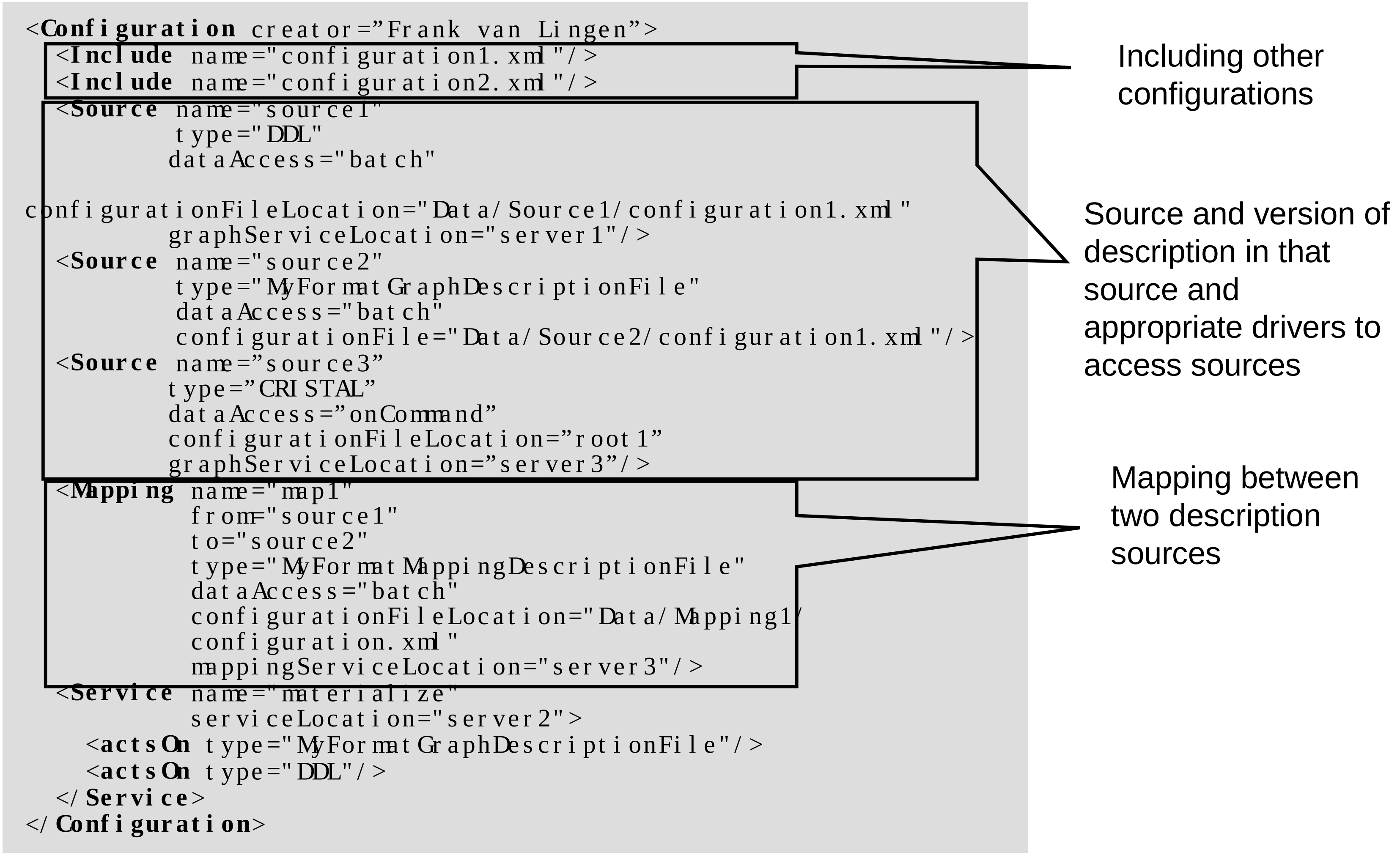
Mapping and Data interface are based on the description data model

Decentralized approach (example)



No storage of data here!

Example of a multi configuration (version) file used within decentralized prototype



```
<Configuration creator=”Frank van Lingen”>
  <Include name="configuration1.xml"/>
  <Include name="configuration2.xml"/>
  <Source name="source1"
          type="DDL"
         dataAccess="batch"

configurationFileLocation="Data/Source1/configuration1.xml"
         graphServiceLocation="server1"/>
  <Source name="source2"
          type="MyFormatGraphDescriptionFile"
          dataAccess="batch"
          configurationFile="Data/Source2/configuration1.xml"/>
  <Source name="source3"
         type=”CRISTAL”
         dataAccess=”onCommand”
         configurationFileLocation=”root1”
         graphServiceLocation=”server3”/>
  <Mapping name="map1"
           from="source1"
           to="source2"
           type="MyFormatMappingDescriptionFile"
           dataAccess="batch"
           configurationFileLocation="Data/Mapping1/
           configuration.xml"
           mappingServiceLocation="server3"/>
  <Service name="materialize"
           serviceLocation="server2">
    <actsOn type="MyFormatGraphDescriptionFile"/>
    <actsOn type="DDL"/>
  </Service>
</Configuration>
```

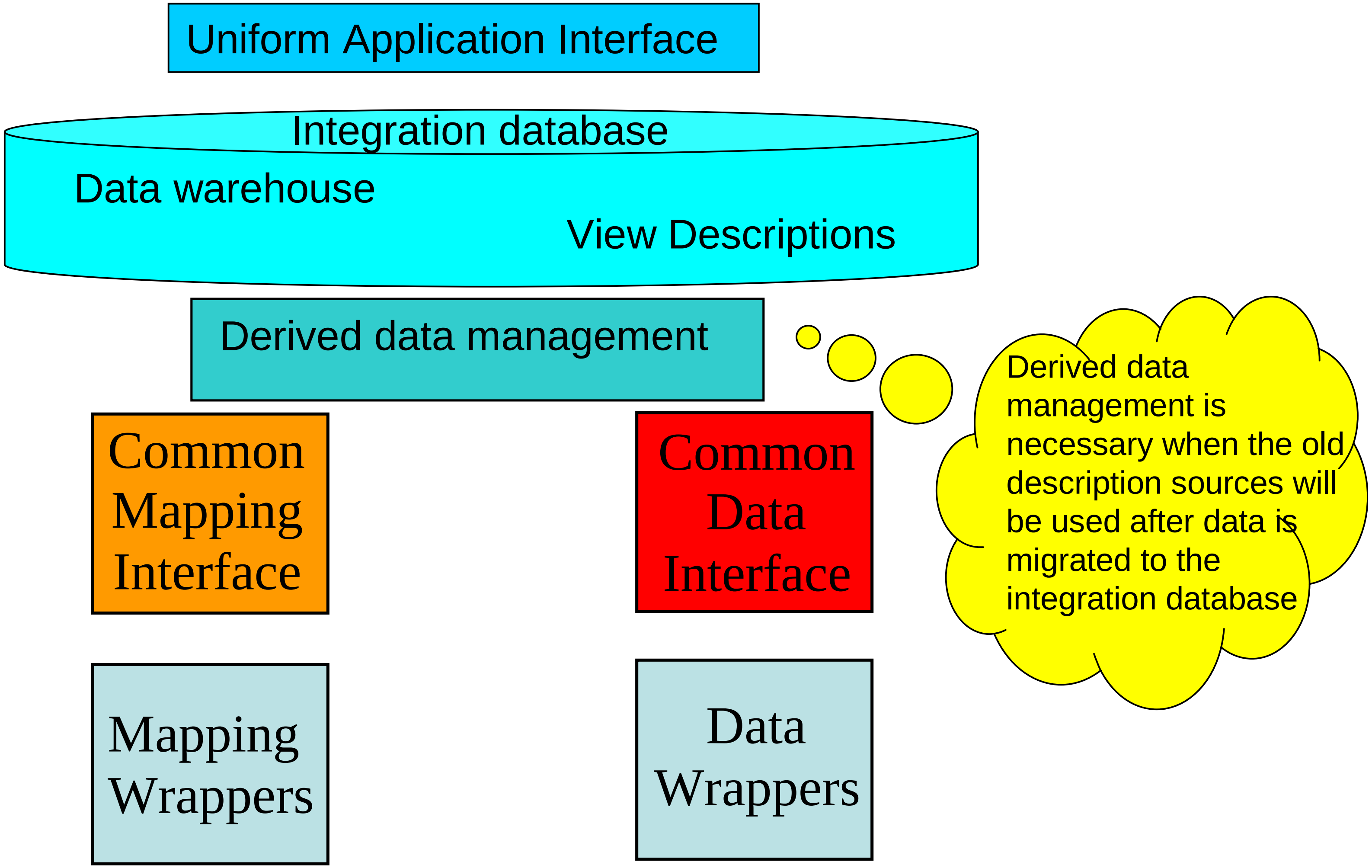
Including other configurations

Source and version of description in that source and appropriate drivers to access sources

Mapping between two description sources

Configuration files can be queried and combined, such that different domains can create different views based on other domains

Centralized approach



Uniform Application Interface

Integration database

Data warehouse

View Descriptions

Derived data management

Common Mapping Interface

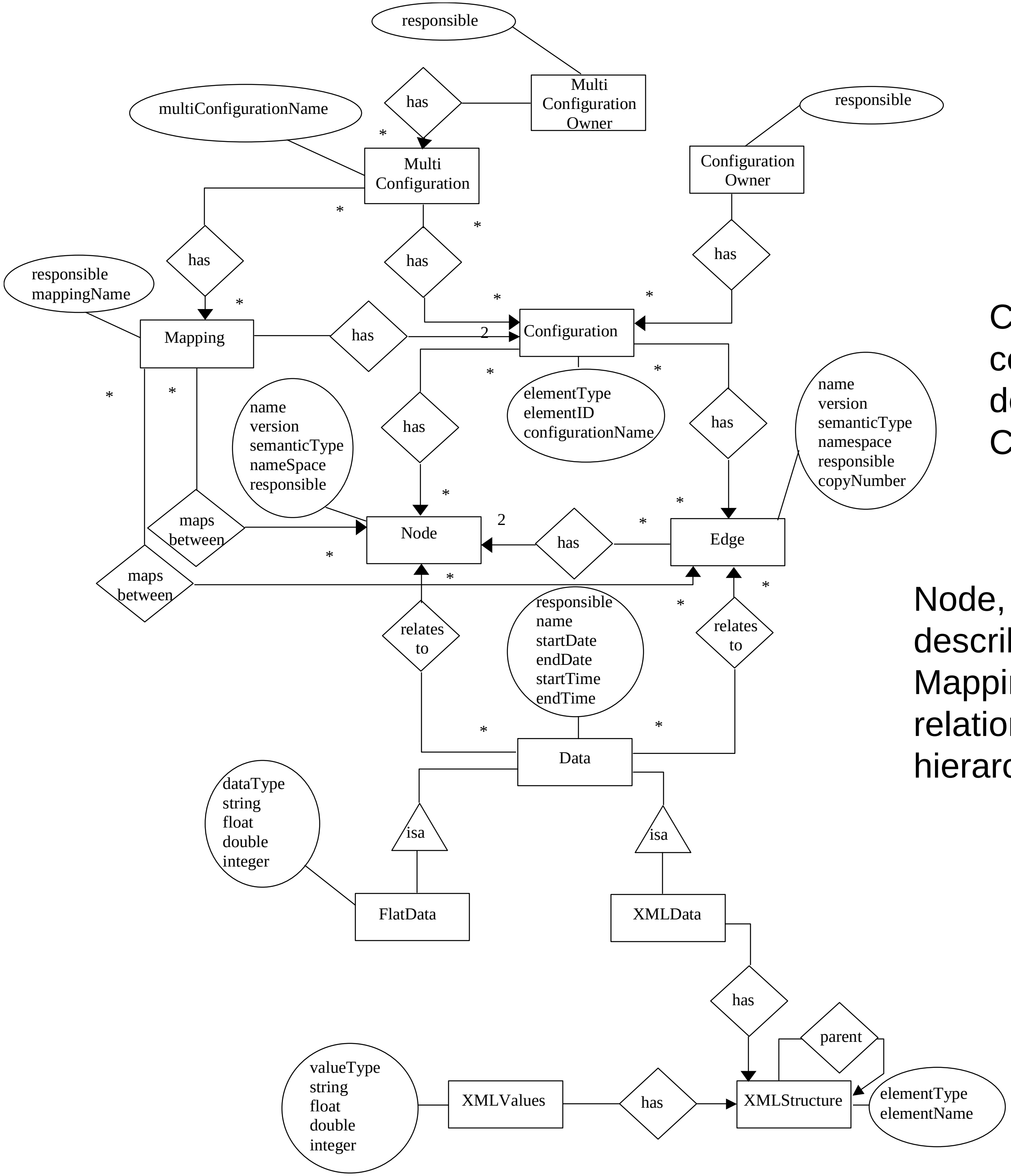
Common Data Interface

Mapping Wrappers

Data Wrappers

Derived data management is necessary when the old description sources will be used after data is migrated to the integration database

Mapping and Data interface are based on the description data model



Initial (relational) schema

For centralized approach based on data model

Configuration files of de centralized model are described within the Multi Configuration entities.

Node, and Edge entities describe hierarchy. Mapping entity describes relation between different hierarchies.

Model caters for XML data, arbitrary string, float or integer data.

Conclusions

- Both centralized and de-centralized approach are feasible for implementation of the integration data base.

- A centralized approach =de-centralized approach+data warehouse+derived data management

- Implementation of the integration database is independent of the CMS environment. Migrating data, describing the CMS hierarchy, and keeping track of part evolution, is CMS specific. As such implementation of such an integration database could be a common project for the four experiments.

- Any implementation of the integration database will need to satisfy the basic requirements stated within this poster. Several of these requirements were investigated within the CRISTAL and DDD project.
- W.r.t the centralized approach, CERN has already 2 systems in place that meets part of the requirements: EDMS and CRISTAL. EDMS or CRISTAL could be used as starting point for a centralized approach of the integration database.
- The EDMS and CRISTAL project give a realistic estimate of the manpower needed to build an integration database from scratch.
- The requirements and implementation can be based on well known results in the area of data modeling w.r.t. part hierarchies (see references).

References

- M. Case, M. Liendl, F. van Lingen, “**Detector Description Domain Architecture & Data Model”** , CMS NOTE-2001/057
- J.-M. Le Goff, G. Chevenier, A Bazan, T. Le Flour, S. Lieunard, S. Murray, J.-P.Vialle, N. Baker, F. Estrella, Z. Kovacs, R. McClatchey, G. Organtini, S. Bityukov, “**From Design to Production Control Through the Integration of Engineering Data Management and Workflow Management Systems”,** CMS NOTE-1997/104
- **EDMS**, http://cedar.web.cern.ch/CEDAR/
- Randy H. Katz. “**Toward a Unified Framework for Version Modelling in Engineering Databases”**. ACM Computing Surveys, 22(4):375--408, 1990.
- Yu, L., and Rosenkrantz, D.J., “**Ancestor Controlled Sub module Inclusion in Design Databases**”, IEEE Transactions on Knowledge and Data Engineering, 5, 2, April 1993, 352-362.
- Blostein, D. and Schrr, A., "**Computing with Graphs and Graph Transformation,**" Software - Practice and Experience, Vol. 29, No. 3, 1999, pp. 197-217
- Michael Case, Martin Liendl, Frank van Lingen, “**CMS XML-based Detector Description Database System**” poster@CHEP 2003