Towards an Interoperability Manifesto

Franz-Josef Behr, ICA Commission on SDI and Standards

Commission Meeting, Helsinki, 11/06/2016

Outline

- Some thoughts and observations regarding interoperability
- Lessons learned from the OO Database Manifesto
- Towards an Interoperability Manifest to foster international scientific and technical interoperability

Interoperability: Definition

The ability to collaborate and to exchange information seemlessly and without barriers.

- often restricted to technical aspects.

But: Interoperability should be extended.

Interoperability: OGC‘s Definition

„…software components operating reciprocally (working with each other) to overcome

tedious batch conversion tasks,

import/export obstacles, and

distributed resource access barriers

imposed by heterogeneous processing environments and heterogeneous data.

[McKee and Buehler, 1998 ; Sondheim, Gardels and Buehler 1999]

Interoperability: IEEE‘s Definition

„Ability of two or more systems or components to exchange information and to use the information that has been exchanged.”

[IEEE 1990]

Interoperability: IEEE‘s Definition

"The capability to communicate, execute programs, or transfer data among various functional units in a manner that requires the user to have little or no knowledge of the unique characteristics of those units".

[ISO/IEC 2382-36:2008]

Interoperability: a stack of means and contributions

- Technical Interoperabilty
- Personal Interoperabilty
- Semantical Interoperabilty
- Institutional Interoperabilty
- Political Interoperabilty

- Standards, scientific work
- Challenges & Activities
- Common Scientific Activities

Technical Interoperability: between IT systems

Personal Interoperabilty

WMS, WFS, WCS, WPS,
Shape, GML, KML, …
HTTP, XML, JavaScript

Semantical Interoperabilty

Institutional Interoperabilty

Political Interoperabilty

1. Awareness,
2. Capacity building,
3. Education
4. Contribution

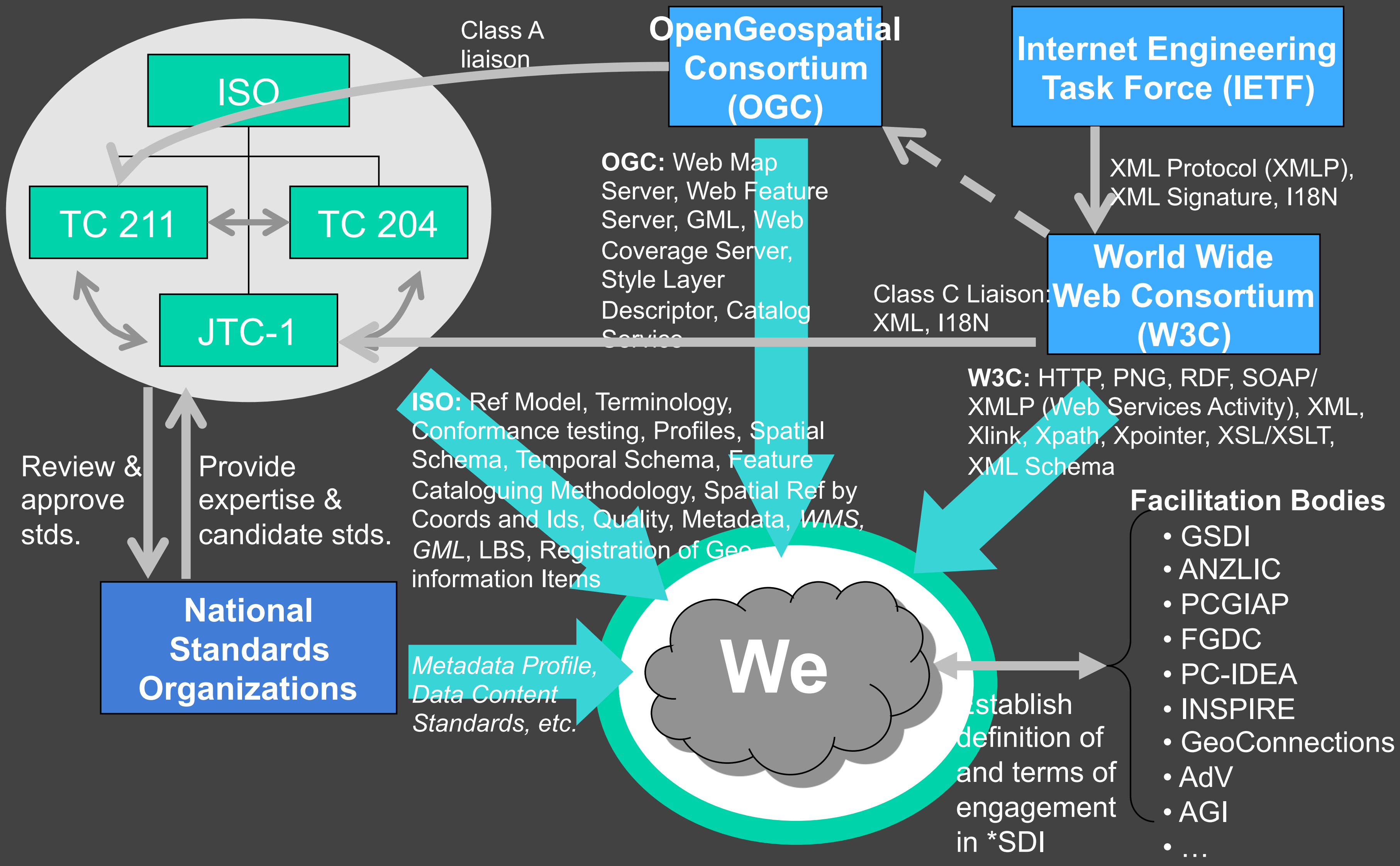
Technical Interoperabilty

Internatioal Standards

OGC, W3C, ISO & others

- Standards evolved in many years, often in consensus driven processes
- well established

Keynotes, presentations, workshops



Source: Doug Nebert (2002): Focus on Standards. https://www.fgdc.gov/library/presentations/documents/2002-presentations/Focus_on_Standards.ppt [20013-03-16]

Personal Interoperability: between people

Technical Interoperabilty

Communication skills,
personal relationships,
social networking,
cultural understanding,
foreign languages

Semantical Interoperabilty

Institutional Interoperability

Sharing our data
sharing through web services
student / staff / academic exchange

Political Interoperability

common research
common publications

Personal Interoperabilty

Relationships

Interpersonal Standards

- evolving over years,
- Based on trust
- and on scientific ethics

Presentations,
Tea breaks,
Meetings
…

Semantical Interoperability: between knowledge islands

Technical Interoperabilty

Personal Interoperabilty

Vocabularies, Ontologies

Political Interoperabilty

Keynotes, presentations, workshops, tea breaks, sharing time

Semantical Interoperabilty

Collaboration between groups and **extreme different** disciplines

Collaboration between groups and disciplines

- Transcending the barriers
- communication
- evolving over years,
- based on knowledge
- still part of scientific research

Semantics

- Part of the linguistics (study of language) and semiotics (study of signs)
- According to Wikipedia:
  - From the ancient Greek: σημαντικός sēmantikós; important),
  - the study of meaning
  - focuses on the relation between signifiers, like words, phrases, signs, and symbols, and what they stand for, their denotation.

Ontology

- In philosphy: "metaphysical science or study of being" [1] study of the nature of being, becoming, existence, or reality, as well as the basic categories of being and their relations
- In computer science and information science, an ontology formally represents knowledge as a hierarchy of concepts within a domain, using a shared vocabulary to denote the types, properties and interrelationships of those concepts
- "The hierarchical structuring of knowledge about things by subcategorising them according to their essential (or at least relevant and/or cognitive) qualities." [2]

[1] http://www.etymonline.com/index.php?search=ontology&searchmode=none [2014-04-11]
[2] http://foldoc.org/ontology [2014-04-11]

Institutional Interoperability: between organisations

Technical Interoperabilty

Personal Interoperabilty

Semantical Interoperabilty

Political Interoperabilty

Institutional Interoperabilty

Personal Interoperabilty

Common publications
common undertakings
Memorandum of Understand (MoU),
dual education

- dependencies
- evolving over years,
- based on decision, shaped by communicated convictions

How can we learn to extend collaboration?

Political Interoperability: between communities

Technical Interoperabilty

Personal Interoperabilty

Semantical Interoperabilty

Institutional Interoperabilty

- political interests, economical interests
- Long term process

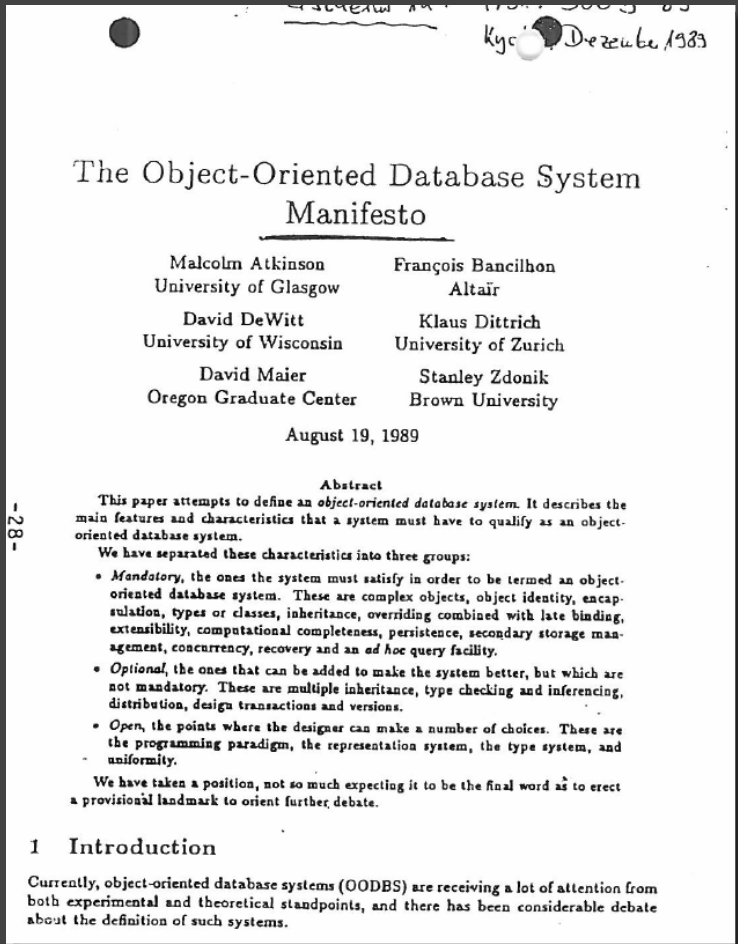
"We all learn
if we share what we know."

Robert R. DiBlasi

Lessons learned from the OO Database Manifesto

Malcolm Atkinson et al.: The Object-Oriented Database System Manifesto (in: Proceedings of the DOOD 89) Kyoto 1989

https://www.cl.cam.ac.uk/teaching/2003/Databases/oo-manifesto.pdf [2016-06-21]



The Object-Oriented Database System Manifesto

Malcolm Atkinson
University of Glasgow

François Bancilhon
Altaïr

David DeWitt
University of Wisconsin

Klaus Dittrich
University of Zurich

David Maier
Oregon Graduate Center

Stanley Zdonik
Brown University

August 19, 1989

Abstract

This paper attempts to define an *object-oriented database system*. It describes the main features and characteristics that a system must have to qualify as an object-oriented database system.

We have separated these characteristics into three groups:

- *Mandatory*, the ones the system must satisfy in order to be termed an object-oriented database system. These are complex objects, object identity, encapsulation, types or classes, inheritance, overriding combined with late binding, extensibility, computational completeness, persistence, secondary storage management, concurrency, recovery and an *ad hoc* query facility.
- *Optional*, the ones that can be added to make the system better, but which are not mandatory. These are multiple inheritance, type checking and inferencing, distribution, design transactions and versions.
- *Open*, the points where the designer can make a number of choices. These are the programming paradigm, the representation system, the type system, and uniformity.

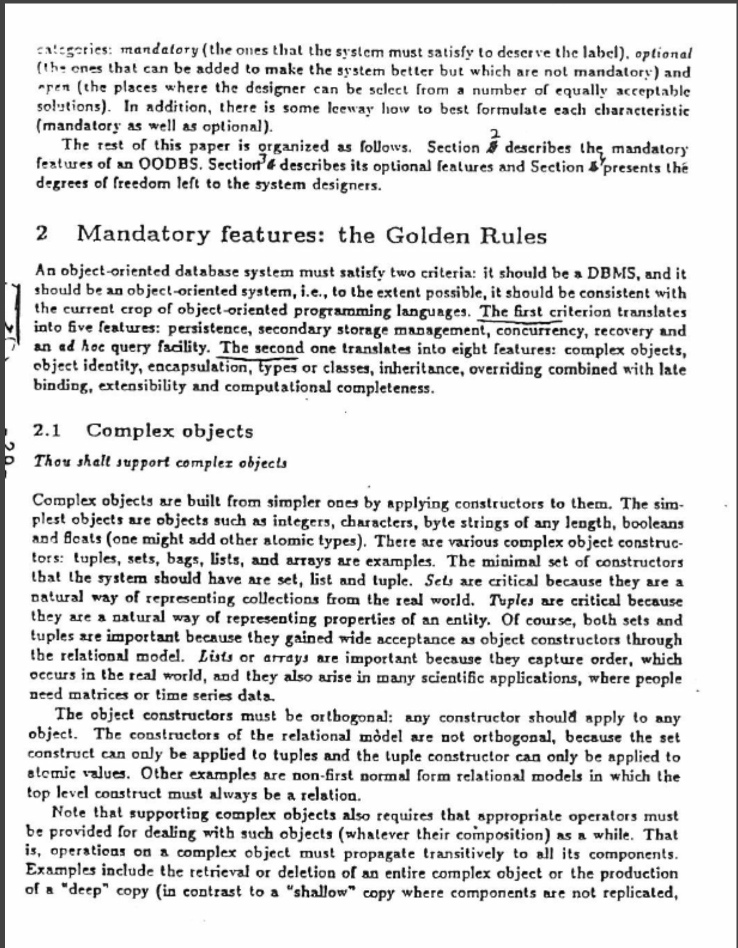
We have taken a position, not so much expecting it to be the final word as to erect a provisional landmark to orient further debate.

-28-

1 Introduction

Currently, object-oriented database systems (OODBS) are receiving a lot of attention from both experimental and theoretical standpoints, and there has been considerable debate about the definition of such systems.

International team of authors from academia and business



categories: *mandatory* (the ones that the system must satisfy to deserve the label), *optional* (the ones that can be added to make the system better but which are not mandatory) and *open* (the places where the designer can be select from a number of equally acceptable solutions). In addition, there is some leeway how to best formulate each characteristic (mandatory as well as optional).

The rest of this paper is organized as follows. Section 2 describes the mandatory features of an OODBS. Section 3 describes its optional features and Section 4 presents the degrees of freedom left to the system designers.

2 Mandatory features: the Golden Rules

An object-oriented database system must satisfy two criteria: it should be a DBMS, and it should be an object-oriented system, i.e., to the extent possible, it should be consistent with the current crop of object-oriented programming languages. The first criterion translates into five features: persistence, secondary storage management, concurrency, recovery and an *ad hoc* query facility. The second one translates into eight features: complex objects, object identity, encapsulation, types or classes, inheritance, overriding combined with late binding, extensibility and computational completeness.

2.1 Complex objects

Thou shalt support complex objects

Complex objects are built from simpler ones by applying constructors to them. The simplest objects are objects such as integers, characters, byte strings of any length, booleans and floats (one might add other atomic types). There are various complex object constructors: tuples, sets, bags, lists, and arrays are examples. The minimal set of constructors that the system should have are set, list and tuple. *Sets* are critical because they are a natural way of representing collections from the real world. *Tuples* are critical because they are a natural way of representing properties of an entity. Of course, both sets and tuples are important because they gained wide acceptance as object constructors through the relational model. *Lists* or *arrays* are important because they capture order, which occurs in the real world, and they also arise in many scientific applications, where people need matrices or time series data.

The object constructors must be orthogonal: any constructor should apply to any object. The constructors of the relational model are not orthogonal, because the set construct can only be applied to tuples and the tuple constructor can only be applied to atomic values. Other examples are non-first normal form relational models in which the top level construct must always be a relation.

Note that supporting complex objects also requires that appropriate operators must be provided for dealing with such objects (whatever their composition) as a while. That is, operations on a complex object must propagate transitively to all its components. Examples include the retrieval or deletion of an entire complex object or the production of a "deep" copy (in contrast to a "shallow" copy where components are not replicated,

THE basic recognized paper in this field of Computer Science

Core set of rules – “Commandments”

- Mandatory, "golden" rules
- optional characteristics
- open characteristics

Influencing further IT developments

Steps towards Interoperability Manifesto

Our manifesto…

- can assemble research results from various fields together
- can bundle our forces and disseminate them,
- can be attractive to the scientific community worldwide (for support, for collaboration and dissemination),
- can be used to influence stakeholders in the political and technical arena
- can extend our impact as ICA commission
- …

Some basic rules…

- Just as a starter - to be discussed and extended:
  - Thou shalt support open standards!
  - Thou shalt provide appropriate licences!
  - Thou shalt support a reasonable amount of meta data!
  - Thou shalt not charge for public data.
  - Thine data shalt be reliable and complete
  - …

Some steps

1. Setting up a working group with mailing list (30/09/2016)
2. Thorough scientific work on the different fields of interoperability
3. Searching für liaisons to other bodies
4. Development of a draft (until February 2017)
5. Further discussion and refinement (ICA 2017, Washington)

Contact

- Prof. Dr. Franz-Josef Behr,
- Co-chair ICA Commission on SDI and Standards

- http://sdistandards.icaci.org/, http://gis-management.de

- franz-josef.behr@hft-stuttgart.de

References

- Yaser Bishr (1998). Overcoming the semantic and other barriers to GIS interoperability, International Journal of Geographical Information Science, 12:4, 299-314, DOI: 10.1080/136588198241806
- Sondheim, M., K. Gardels, and K. Buehler. 1999. GIS interoperability. In Longley et al. (1999): Geographical Information Systems: Principles, Techniques, Applications and Management. John Wiley & Sons, New York