Enhancing the detector control system of the CMS experiment with object oriented modelling

R. Jiménez Estupiñán, A. Andronidis, T. Bawej, O. Chaze, C. Deldicque, M. Dobson, A. Dupont, D. Gigi, F. Glege, J. Hegeman, M. Janulis, L. Masetti, F. Meijers, E. Meschi, S. Morovic, C. Nunez-Barranco-Fernandez, L. Orsini, A. Petrucci, A. Racz, P. Roberts, H. Sakulin, C. Schwick, B. Stieger, S. Zaza, P. Zejdl (CERN, Geneva, Switzerland), U. Behrens (DESY, Hamburg, Germany), O. Holme (ETH Zurich, Switzerland), J. Andre, R. K. Mommsen, V. O'Dell (Fermilab, Batavia, Illinois, USA), G. Darlea, G. Gomez-Ceballos, C. Paus, K. Sumorok, J. Veverka (MIT, Cambridge, Massachusetts, USA), S. Erhan (UCLA, Los Angeles, California, USA), J. Branson, S. Cittolin, A. Holzner, M. Pieri (UCSD, La Jolla, California, USA)

SIMATIC WinCC Open Architecture

The control and supervisory systems of the CMS experiment have a distributed, redundant architecture based on SIMATIC WinCC OA.

Runtime database

The hardware interfacing and persistence layer is implemented with a runtime database. It stores process variables using the concept of structured Datapoints.
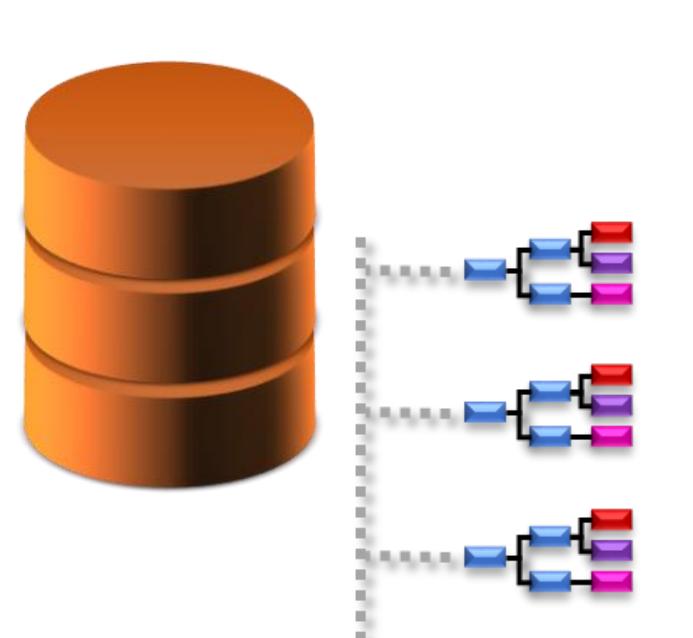
Limited data manipulation language
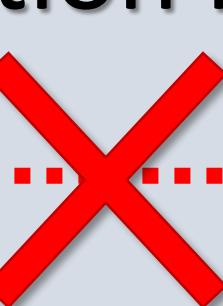
CTRL language

CTRL language is an interpreted, procedural, C-like scripting language to program control scripts and user interfaces. It includes basic library functions to manipulate process variables.
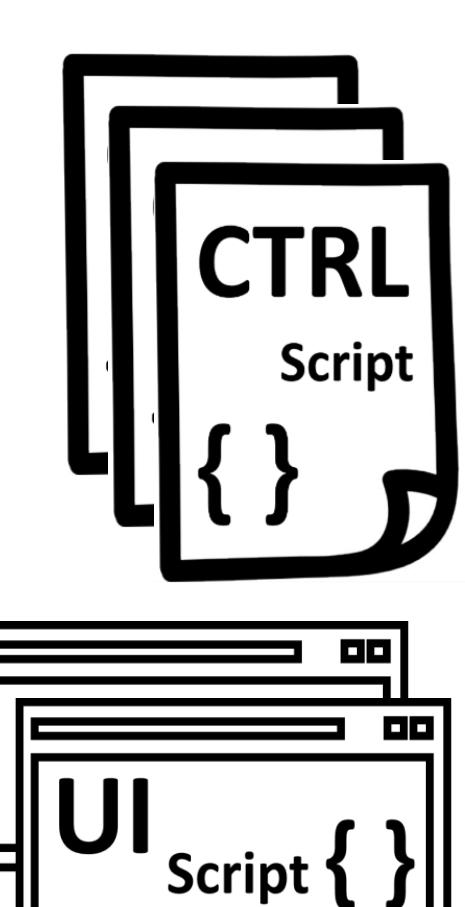
CMSfwClass

CMSfwClass is a control systems development toolkit for WinCC OA to add Object Orientation and encapsulation down to the datapoint level.

The toolkit enables modern software engineering techniques:
- System entities modelling in classes.
- Design complex software architectures
- Rapid implementation with a computer aided engineering tool.

UML design
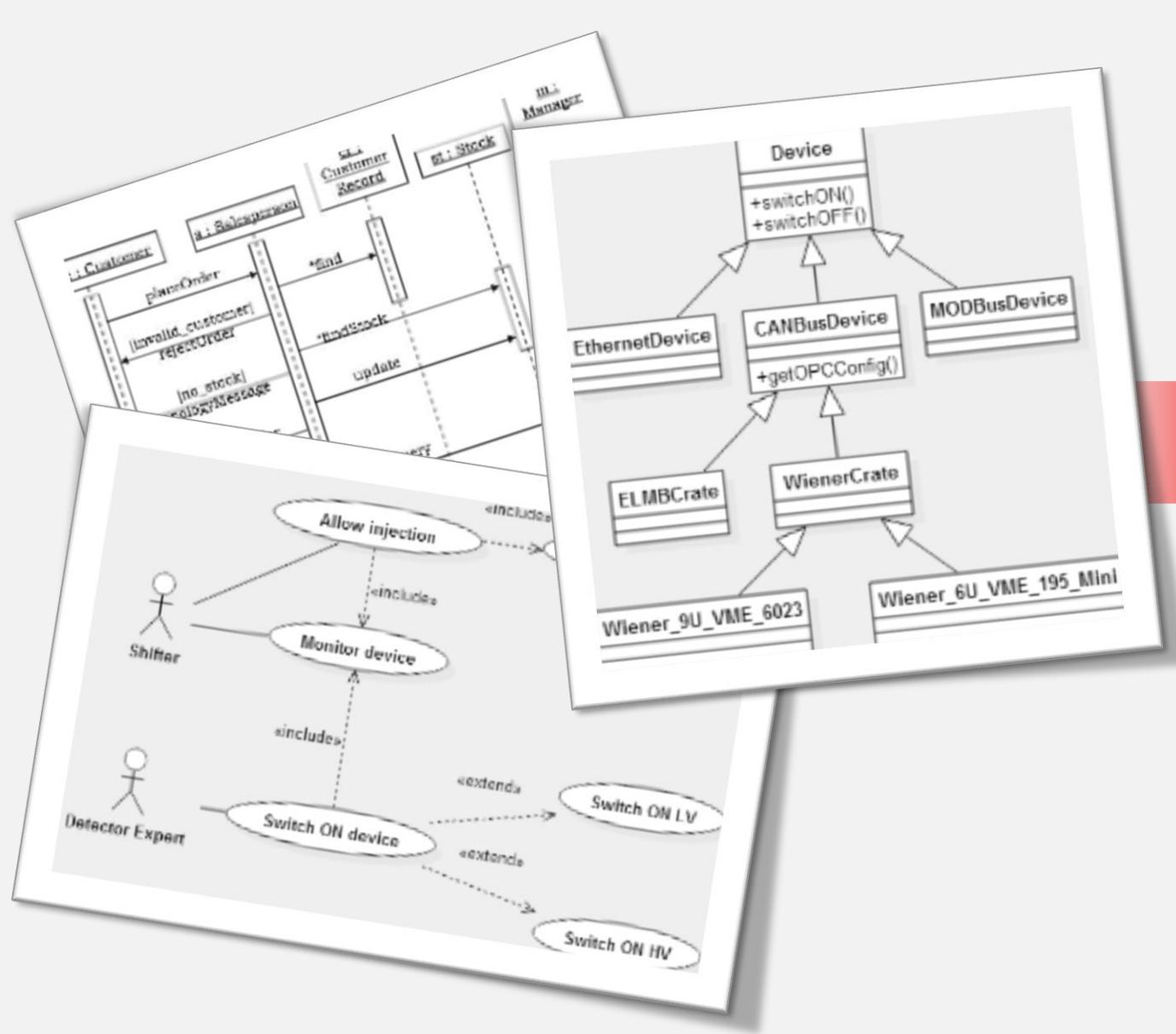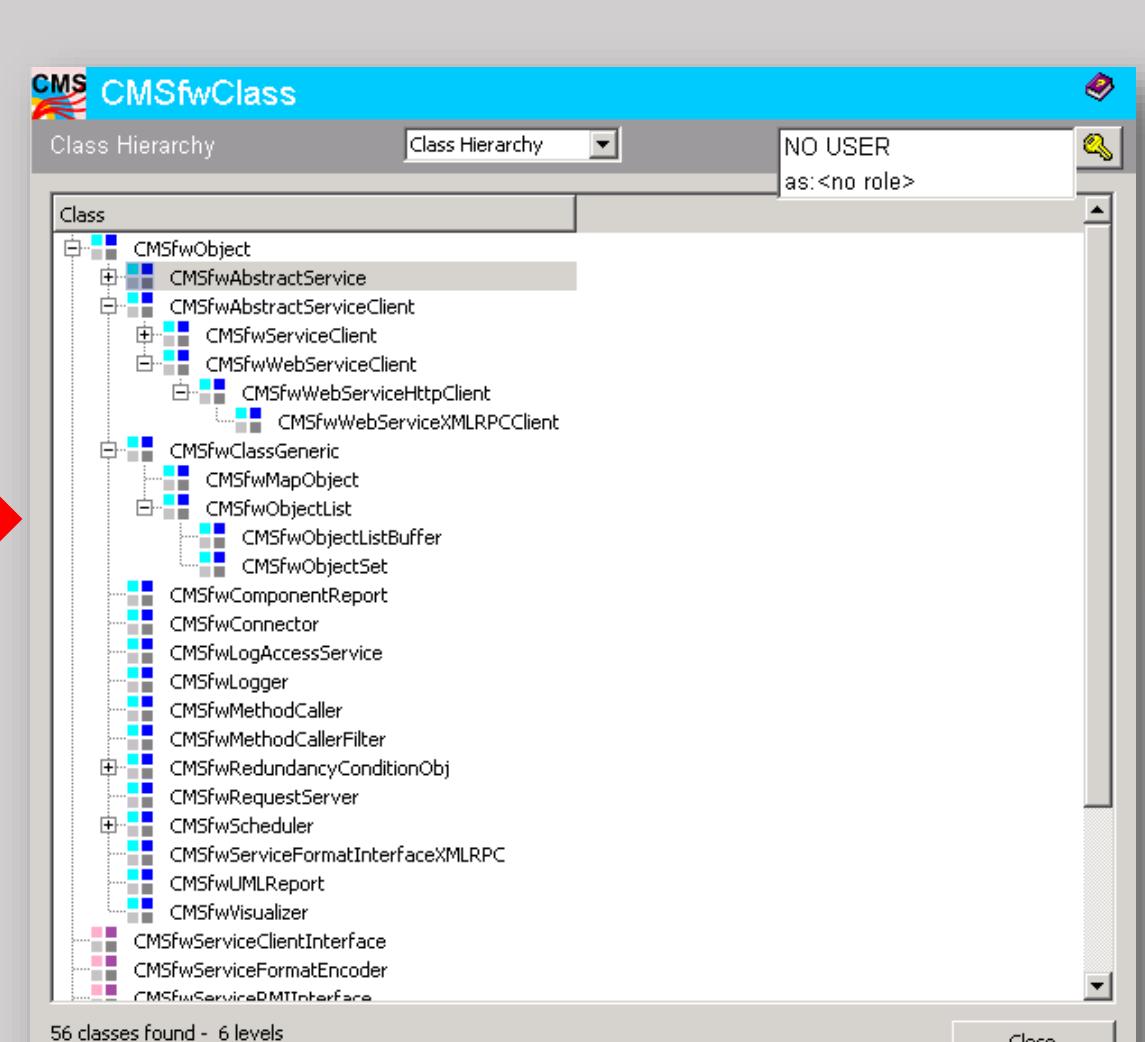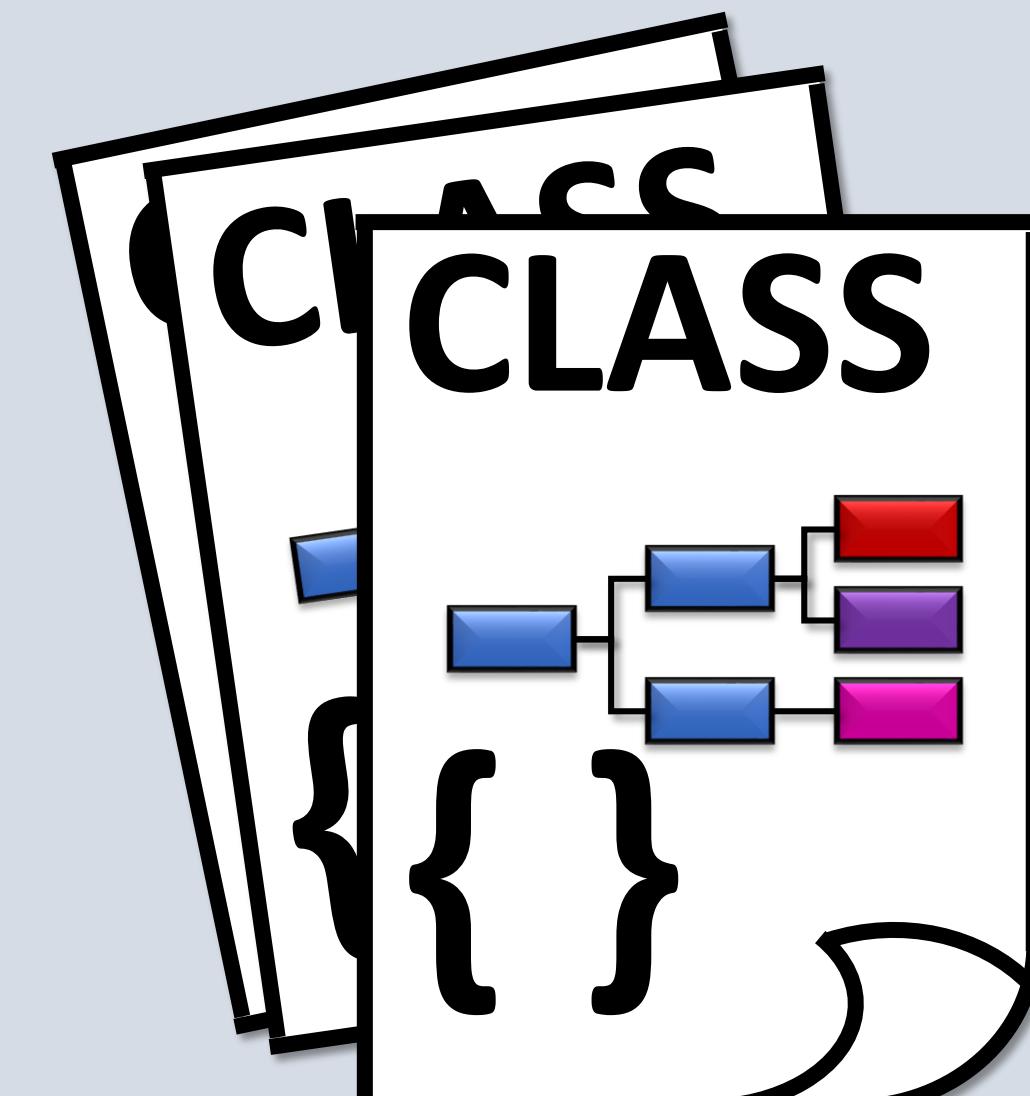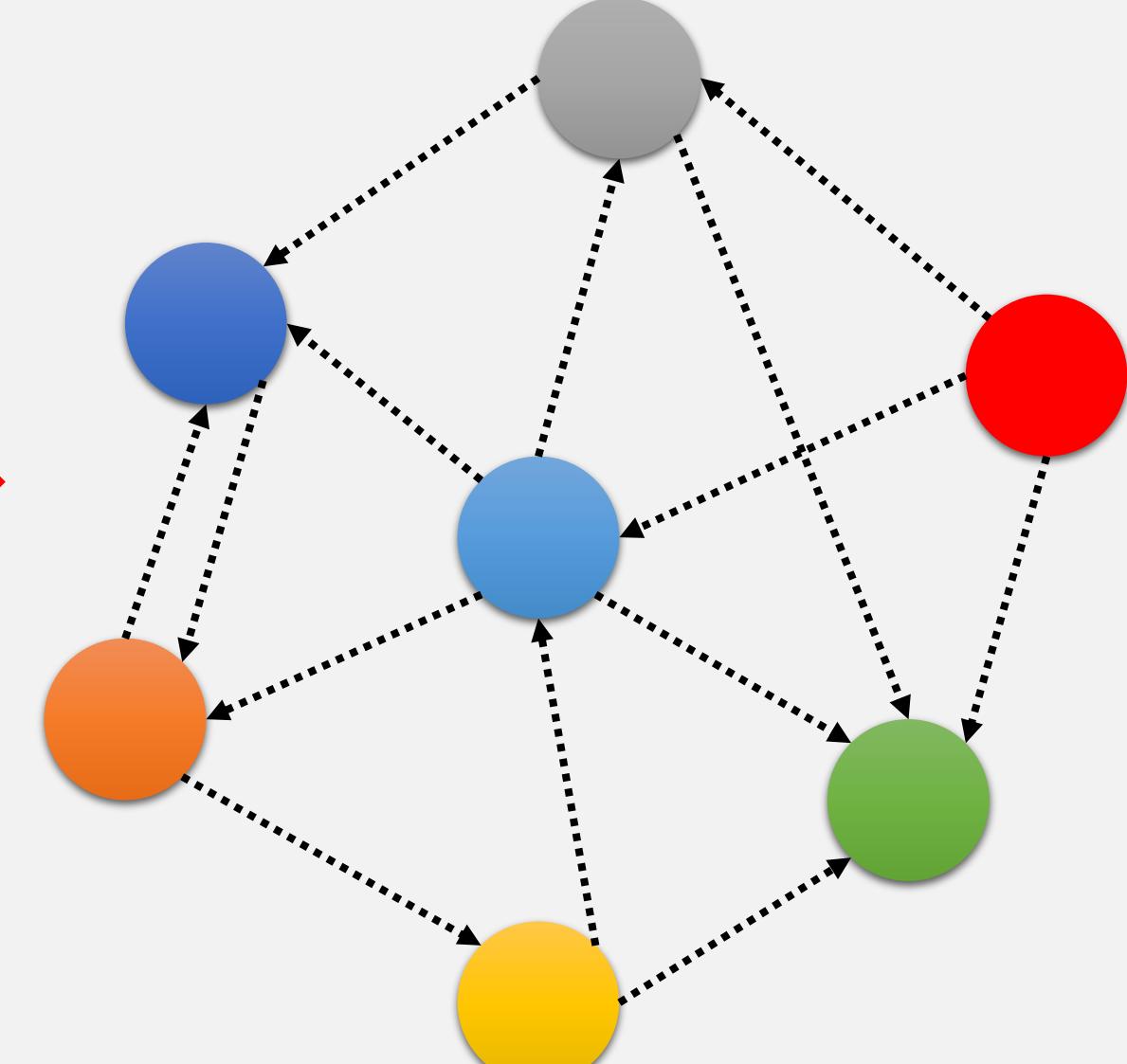
The toolkit permits more direct translation of software design into algorithms and data models. It can also generate documentation from the implementation in the form of UML class diagrams.

CASE UI

A GUI guides developers during the process of creating classes and objects while encouraging consistency and best-practices with:
- Code generation.
- Syntax checking.
- Object management.

Encapsulation

Provides full encapsulation by putting together entity definitions and behaviour in a single source code file, describing how the model interacts with other classes and libraries.

OO Architecture

It provides many common OO programming features:
- Single inheritance.
- Method overriding.
- Interface definition.
- Subtyping polymorphism.
- Object serialization.

Implementation details
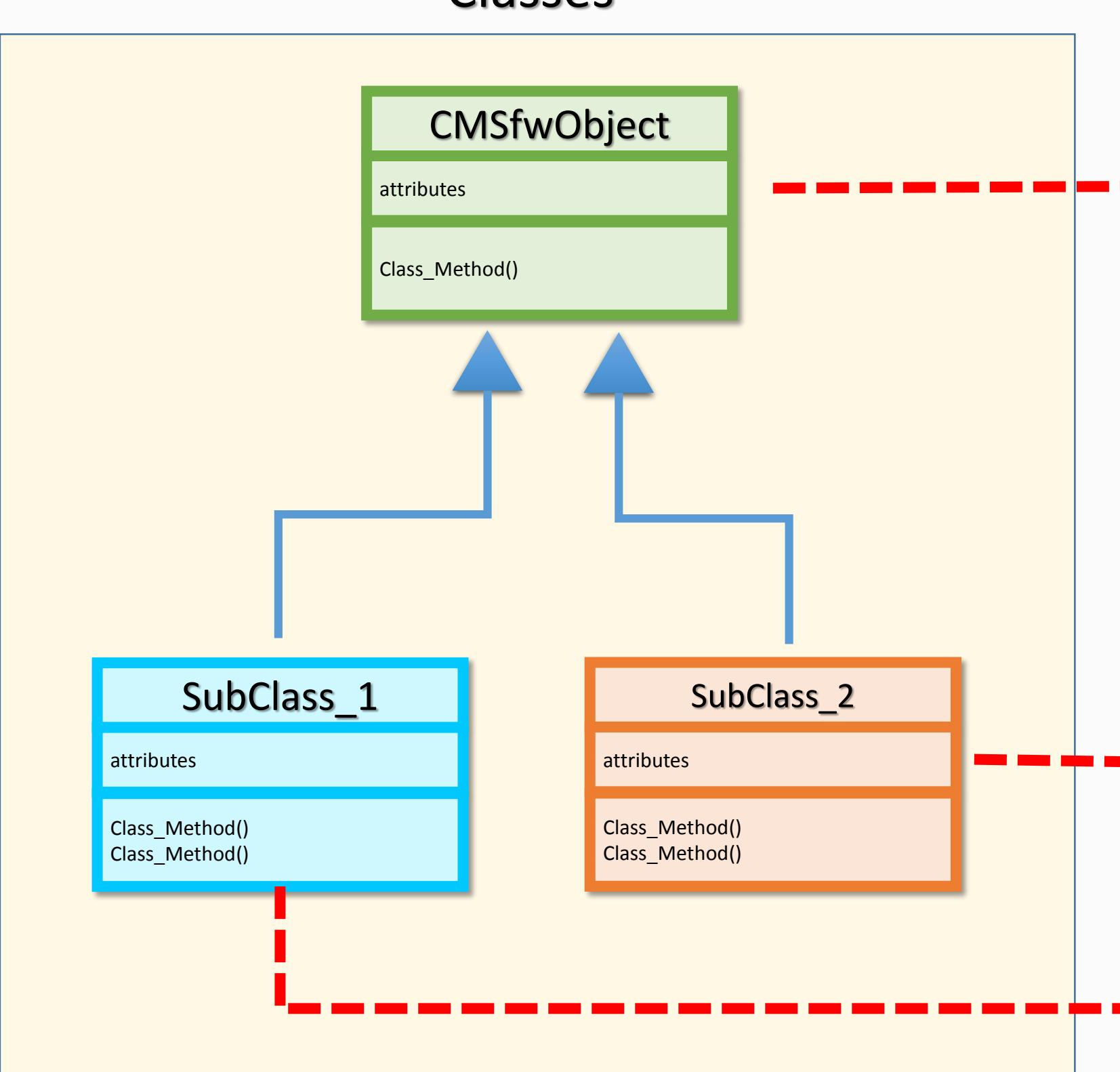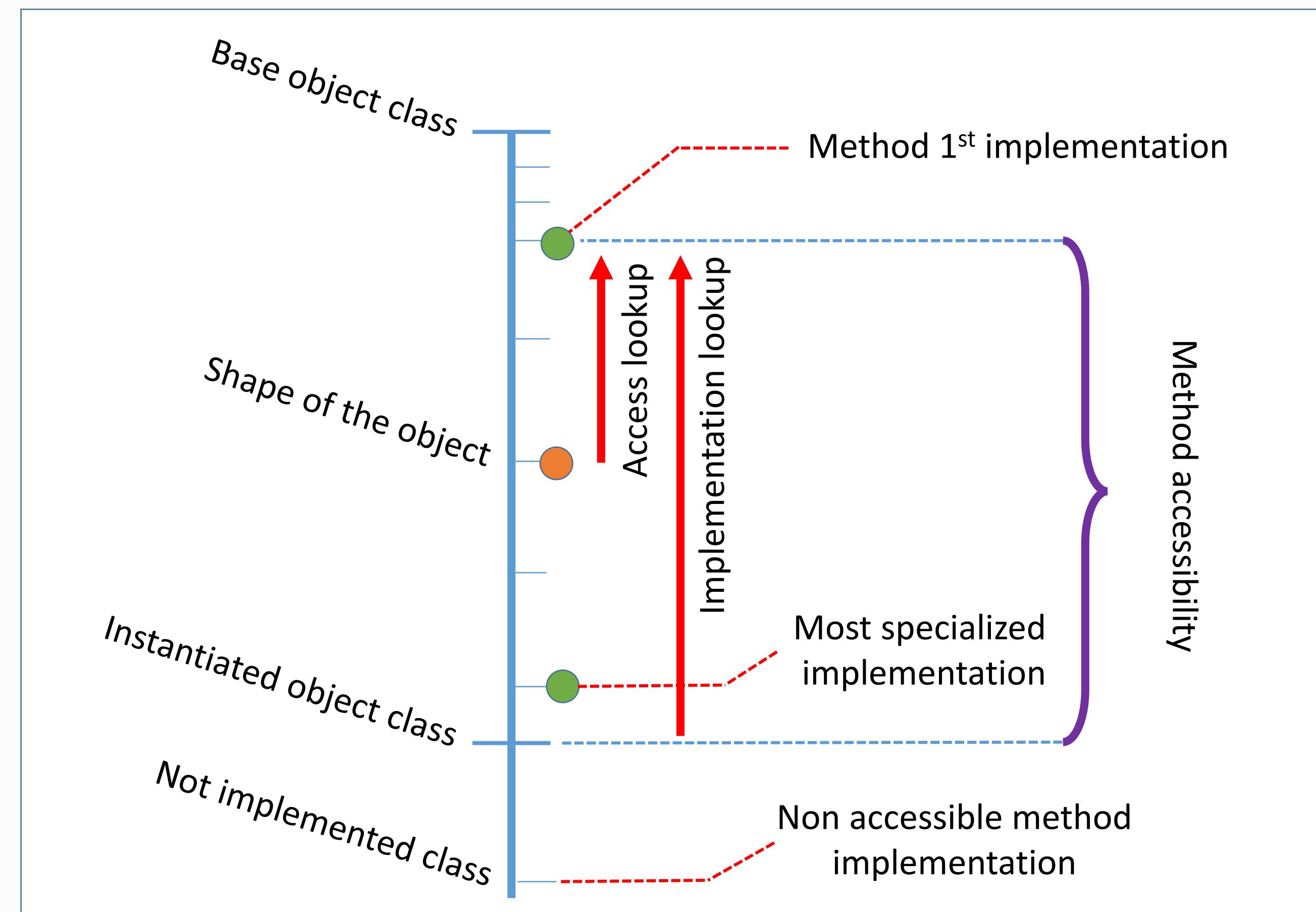
For every object there will be one datapoint per implemented class. The parent-child class hierarchy determines where to find a particular attribute.

For a method to be executed from a certain object, it has to exist in the class used to reference the object, or in a less specialized class of the hierarchy.