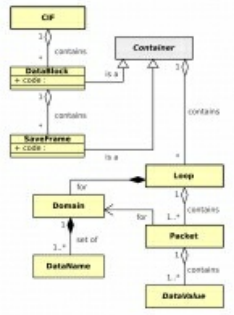
MS107.O05

Fast and flexible CIF processing with the CIF API

John C. Bollinger[1]
[1]*St. Jude Children's Research Hospital, Department Of Structural Biology, Memphis, United States*
E-mail: John.Bollinger@StJude.org

Software wanting to consume or produce CIF data has several challenges to overcome. On one hand, there is a diversity of both official and unofficial variations on the format itself, culminating in CIF 2.0, that the most flexible CIF consumers will want to support. On the other hand, the volume and complexity of crystallographic data continue to increase, whereas CPU speed has reached a plateau, and focus in some quarters is shifting toward smaller, less powerful devices. These trends make both processing efficiency and memory efficiency ongoing concerns for CIF software. Furthermore, CIF carries the legacy of having been promoted as a human readable and writable format, in the form of a comparative prevalence of malformed instances. These challenges and others were addressed during the design and implementation of the open-source CIF API, a powerful, flexible, and portable C library for reading and writing CIF data in any of its variations. We discuss some of the lessons learned through designing and implementing the CIF API, and we present some of the features it has to offer for supporting robust CIF programs.

[1] Bollinger, J. C. (2016). Journal of Applied Crystallography, 49, 285-291.



Keywords: cif software programming