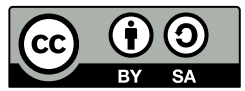
20 years of the COD: making it possible

Saulius Gražulis

Grenoble, 2024

Vilnius University Institute of Biotechnology

Id: slides1-techniques.tex 2860 2024-09-23 08:58:34Z saulius September 23, 2024



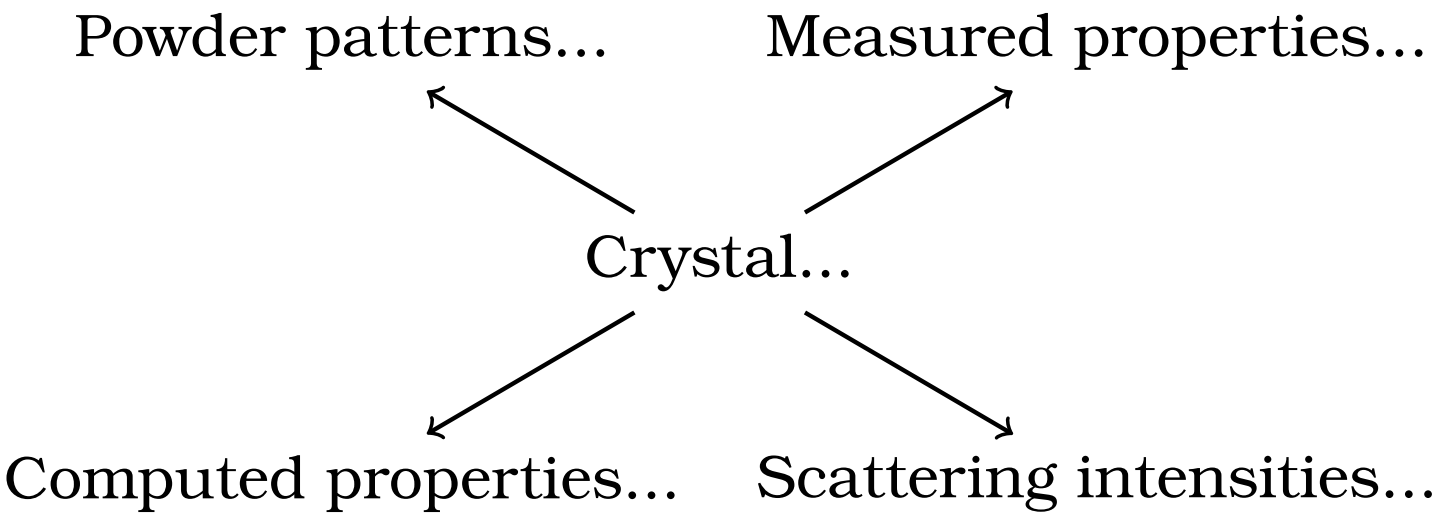
Overview of the talk

- The COD founding principles (openness, scientific rigour, FAIRness, availability on-line)
- COD history and development;
- COD data collection, its scope, data definitions;
- COD curation practices;
- COD links with other databases;
- derived data from the COD;
- data deposition to the COD;
- data search and retrieval from the COD;
- data extraction and processing tools in the COD

https://www.crystallography.net/archives/2024/slides/NOBUGS-Database-Workshop/slides1-techniques.pdf

The importance of crystallographic data

All observations *must* be compatible with crystallographic models.



Things to do with Open Data

Crystal structures
- Drug design
- Material property prediction
- Teaching
- Citizen science
- Machine learning models

The COD project

But what if crystallographers work together to establish a public domain database with all relevant crystallographic data? This would not only overcome the current situation with 'fragmented' databases, it would also prevent for becoming dependent from monopolists.

What would be needed?

1. A small team of engaged scientists with some experience in database and software design to coordinate the project.

2. The authors (i.e. the scientific community = YOU) who provides the project with database entries (note, that if you have'nt sold your experimental results exclusively, you are free to distribute the data to such a database, even if they have already been part of a publication - and a lot of good data have never been published).

3. Free software a) for maintaining the database, b) for data evaluation and calculation of derived data (e.g. calculated powder pattern from crystal structures for search-match purposes), c) for browsing and retrieval.

gemstonede (Dr. Michael BERNDT) Fri Feb 14, 2003 1:26 pm

The Crystallography Open Database (COD)

https://www.crystallography.net/cod

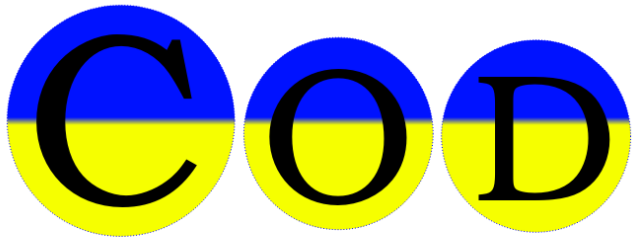
Crystallography Open Database

COD Home
- Home
- What's new?

Accessing COD Data
- Browse
- Search
- Search by structural formula

Add Your Data
- Deposit your data
- Manage depositions
- Manage/release prepublications

Documentation
- COD Wiki
- Obtaining COD
- License
- Privacy and GDPR
- Querying COD
- Citing COD
- COD Mirrors
- Advice to donators
- Useful links



Open-access collection of crystal structures of organic, inorganic, metal-organic compounds and minerals, excluding biopolymers.

Including data and software from CrystalEye, developed by Nick Day at the department of Chemistry, the University of Cambridge under supervision of Peter Murray-Rust.

All data on this site have been placed in the public domain by the contributors.

Currently there are **509888** entries in the COD.
Latest deposited structure: 7159763 on **2024-01-11** at **01:32:14 UTC**

CIFs Donators

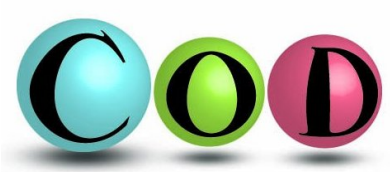
Advisory Board

Daniel Chateigner, Xiaolong Chen, Marco Ciriotti,
Robert T. Downs, Saulius Gražulis, Werner Kaminsky, Armel Le Bail, Luca Lutterotti,
Yoshitaka Matsushita, Andrius Merkys, Peter Moeck, Peter Murray-Rust, Miguel Quirós Olozábal,
Hareesh Rajan, Antanas Vaitkus, Alexandre F.T. Yokochi

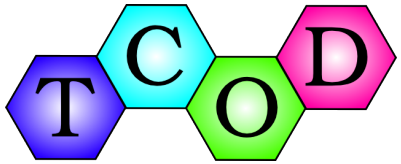
If you find bugs in the COD or have any feedback, please contact us at
cod-bugs@ibt.lt

Top of the page

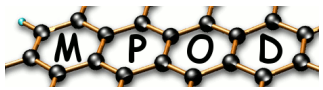
COD “sisters”



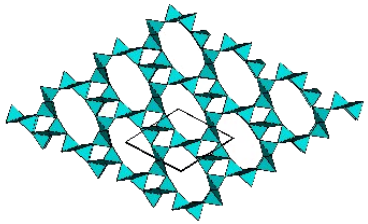
http://www.crystallography.net/cod
$>$ 500 000 entries



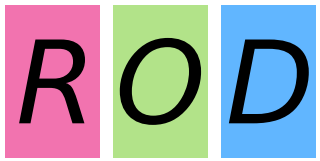
http://www.crystallography.net/tcod
$>$ 7400 entries (ready to grow to $>$ 10^7?)



http://mpod.cimav.edu.mx/
$>$ 300 entries



http://www.crystallography.net/pcod
$>$ 10^6 entries (ready to grow to $>$ 10^8?)

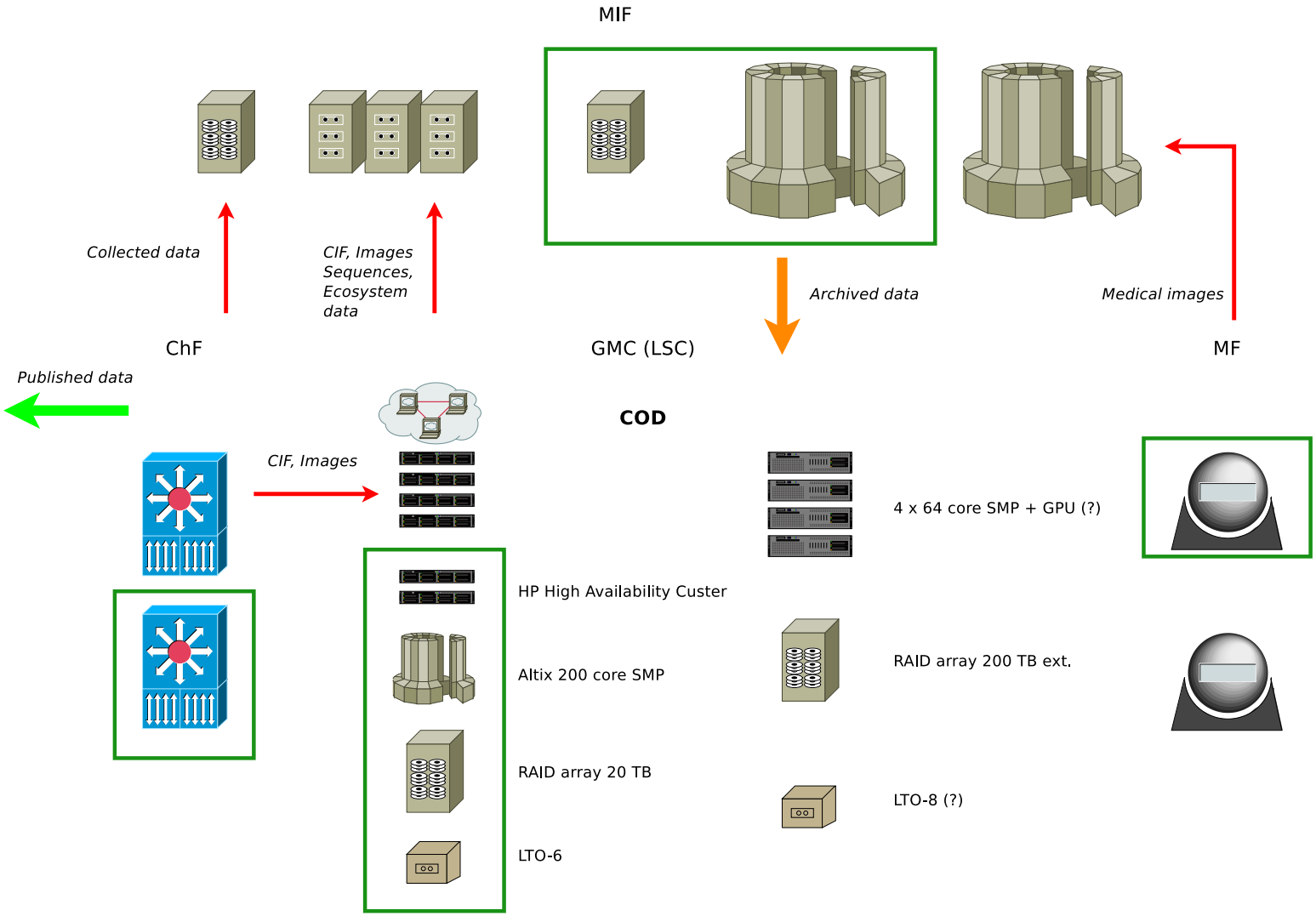


http://solsa.crystallography.net/rod/
$>$ 1100 entries

(Gražulis et al. 2009; Gražulis et al. 2012; Pepponi et al. 2012; Fuentes-Cobas et al. 2017; Mendili et al. 2019)

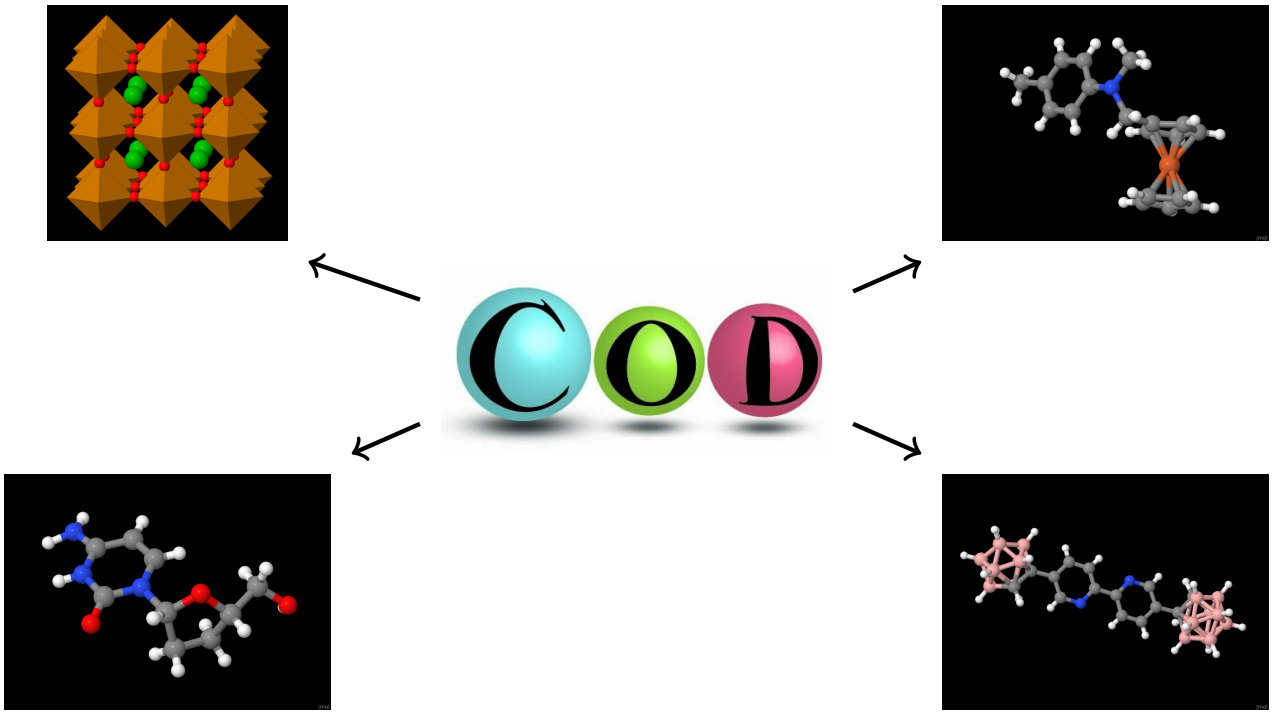
Vilnius University Data Centre of Excellence

Data Center for Machine Learning and Quantum Computing in Natural and Biomedical Sciences



COD contents

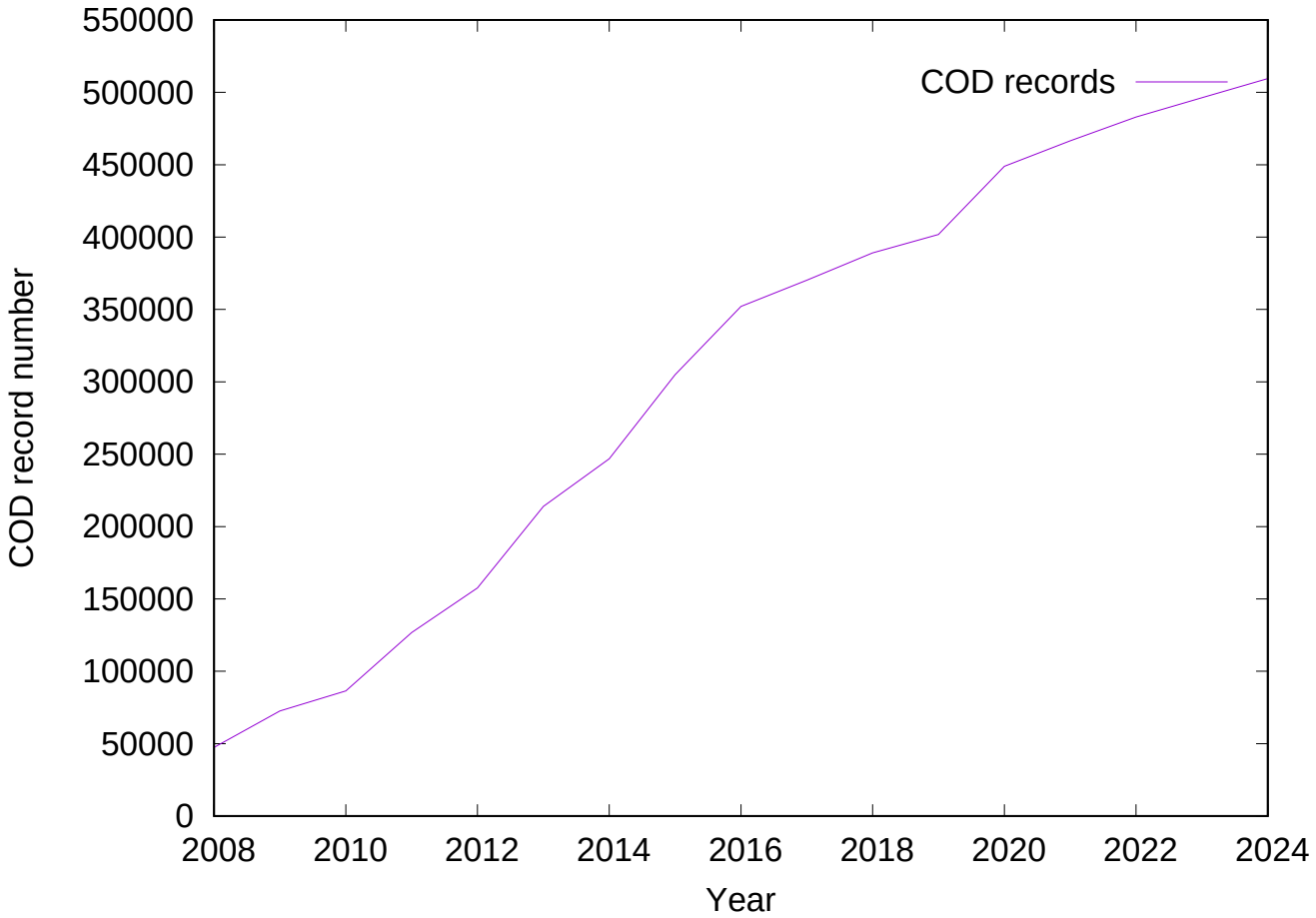
https://www.crystallography.net/cod



509 888 records as of 2024-01-11, available under **CC0 License**

All data are presented in a standardised, machine-readable form (Gražulis et al. 2009; Gražulis et al. 2012).

COD growth

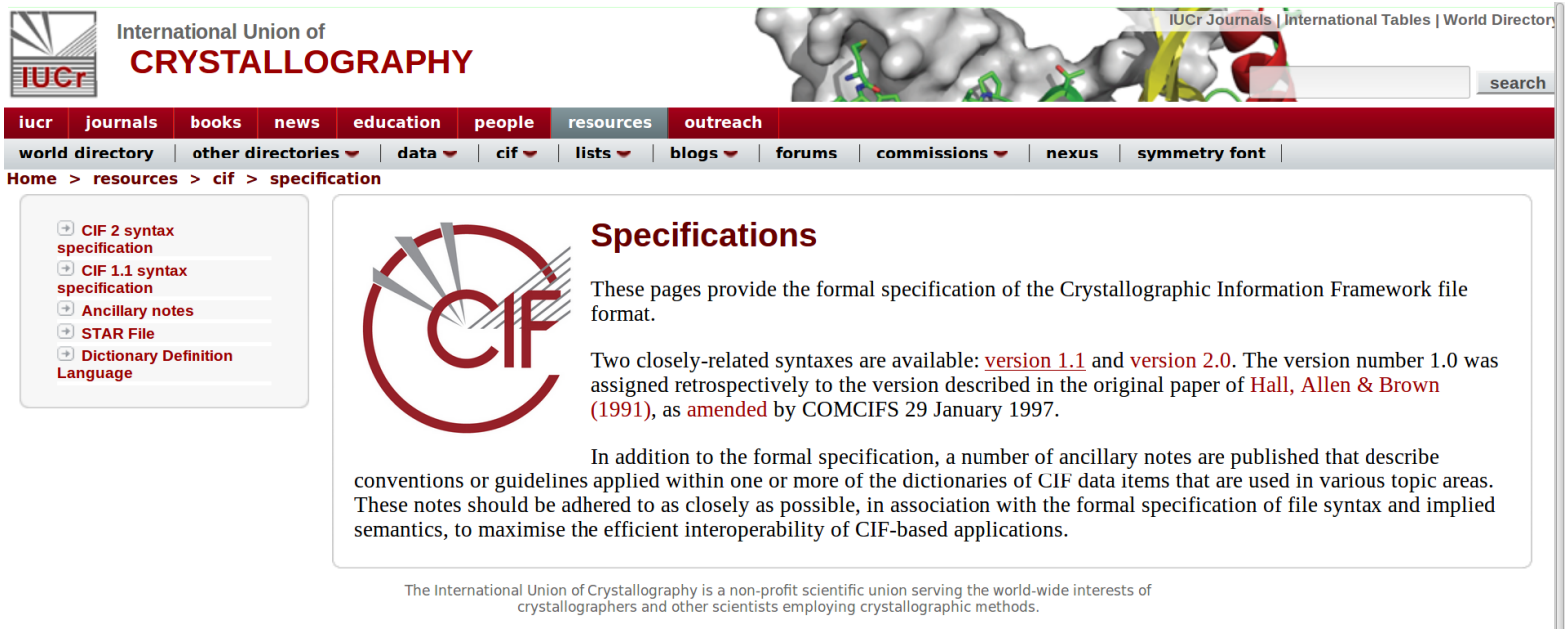

COD records
COD record number
Year
0
50000
100000
150000
200000
250000
300000
350000
400000
450000
500000
550000
2008
2010
2012
2014
2016
2018
2020
2022
2024

COD data sources

- Peer-reviewed publications;
- Preprints, dissertations;
- Depositions by crystallographers (pers. comm., pre-publ.);
- Other databases; notably **AMCSD**, maintained by the group of Robert Downs (Downs et al. 2003; Rajan et al. 2006)

https://rruff.geo.arizona.edu/AMS/amcsd.php

CIF for the COD



(Hall et al. 1991; Bernstein et al. 2016)

The Crystallographic Interchange File/Framework (CIF):

- Provides standard means for data publishing and exchange;
- Is suitable for archiving;
- Is maintained by the IUCr;

Three levels of data validation

- Check of file syntax;
- Validation against dictionaries;
- Domain-specific checks:
  - internal consistency;
  - coherence with raw data;
  - scientific plausibility;

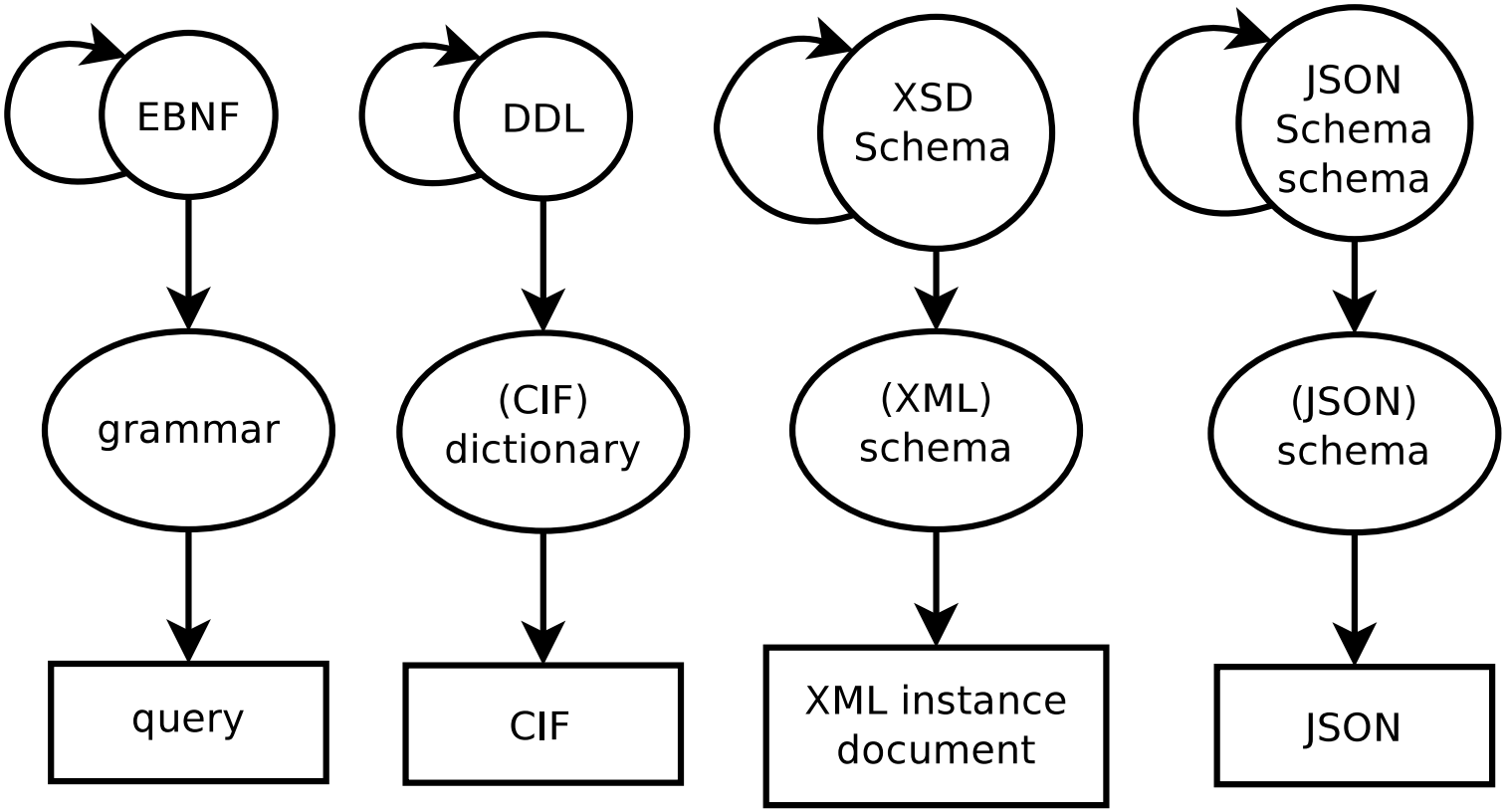
COD data validation

COD data validation policies:

1. Syntactic checks:
   ```
   $ cifparse 7234818.cif
   ```
   Syntax recently expanded to CIF2 (Bernstein et al. 2016; Merkys et al. 2016)
2. Semantic validation (against dictionaries)
   ```
   $ cif_validate -D cif_core.dic 7234818.cif
   ```
   Validation capabilities recently expanded to DDLm (Vaitkus et al. 2021).
3. Database-specific checks
   ```
   $ cif_cod_check 7234818.cif
   ```

Commands from the `cod-tools` package:
svn://cod.ibt.lt/cod-tools
https://github.com/cod-developers/cod-tools

Common pattern of self-describing data definitions



COD data curation principles

- Strictly stick to IUCr standards (CIF syntax, dictionaries);
- Do not invent data;
- Better to have no data than wrong data;
- Consult original papers or authors themselves if in doubt;
- Document: record and explain (justify) all changes;
- Keep track of all changes in a version control system;
- Keep data provenance (original file names);

COD data curation example

Data curation in the COD:

```
svn log -r283960 --diff svn://www.crystallography.net/cod/cif/9
```

```
--- 00/15/9001556.cif (revision 283959)
+++ 00/15/9001556.cif (revision 283960)
@@ -68,8 +68,24 @@
_atom_site_fract_y
_atom_site_fract_z
_atom_site_U_iso_or_equiv
{+_atom_site_type_symbol+}
{+_atom_site_attached_hydrogens+}
Fe 0.25000 0.25000 0.25000 0.00490 {+Fe 0+}
O-H1 0.50000 0.17800 0.30800 0.00100 {+O 1+}
O-H2 0.19500 0.19000 0.50000 0.00100 {+O 1+}
O-H3 0.31800 0.50000 0.32300 0.00100 {+O 1+}
Wat 0.00000 0.50000 0.50000 0.00640 {+O 2+}
/.../
```

Syntax errors in *published* CIFs

Among 3 most prolific publishers in 2021–2022:

- ≈ 12 000 files harvested,
- ≈ 43 000 structures deposited to the COD,
- **52** correctable syntax errors detected in **14** files.

E.g.:

```
cifparse: example1.cif(15,39) data_block_1: ERROR, incorrect CIF syntax:
 _exptl_crystal_description  structure obtained
                                          ^
```

Syntax errors in *published* CIFs

Among 3 most prolific publishers in 2021–2022:

- ≈ 12 000 files harvested,
- ≈ 43 000 structures deposited to the COD,
- **52** correctable syntax errors detected in **14** files.

E.g.:

```
cifparse: example1.cif(15,39) data_block_1: ERROR, incorrect CIF syntax:
 _exptl_crystal_description  structure obtained
                                          ^
```

Most of these errors are fixed automatically by the COD CIF parser (Merkys et al. 2016), but ...

Syntax errors in *published* CIFs

Among 3 most prolific publishers in 2021–2022:

- ≈ 12 000 files harvested,
- ≈ 43 000 structures deposited to the COD,
- **52** correctable syntax errors detected in **14** files.

E.g.:

```
cifparse: example1.cif(15,39) data_block_1: ERROR, incorrect CIF syntax:
 _exptl_crystal_description  structure obtained
                                                 ^
```

Most of these errors are fixed automatically by the COD CIF parser (Merkys et al. 2016), but ...

> Data do not get the same attention from reviewers as the main text.

Syntax formally right, but ...

```
_publ_contact_author
;
    Name, Surname
    Department of Chemistry
    University of ...
    ;
_publ_contact_letter This is the CIF file for ...
_publ_contact_author_phone                ;
;
_publ_section_title
;
  The correct title follows ...
;
```

[Boerrigter 2023, pers. comm.]

Syntax formally right, but ...

```
_publ_contact_author
;
    Name, Surname
    Department of Chemistry
    University of ...
    ;
_publ_contact_letter This is the CIF file for ...
_publ_contact_author_phone                ;
;
_publ_section_title
;
  The correct title follows ...
;
```

[Boerrigter 2023, pers. comm.]

Syntax formally right, but ...

```
_publ_contact_author
;
    Name, Surname
    Department of Chemistry
    University of ...
    ;
_publ_contact_letter This is the CIF file for ...
_publ_contact_author_phone                ;
;
_publ_section_title
;
  The correct title follows ...
;
```

[Boerrigter 2023, pers. comm.]

Data review and the use of proper authoring tools could help...

Description of semantics

CIF dictionaries

IUCr: https://github.com/COMCIFS

COD: https://www.crystallography.net/cod/cif/dictionaries

Example from the `cif_core` DDL1 dictionary:

```
data_cell_length_
    loop_ _name                 '_cell_length_a'
                                '_cell_length_b'
                                '_cell_length_c'
    _category                    cell
    _type                        numb
    _type_conditions             esd
    _enumeration_range           0.0:
    _units                       A
    _units_detail               'angstroms'
    _definition
;               Unit-cell lengths in angstroms corresponding to the structure
                reported. The values of _refln_index_h, *_k, *_l must
                correspond to the cell defined by these values and _cell_angle_
                values. The values of _diffrn_refln_index_h, *_k, *_l may not
                correspond to these values if a cell transformation took place
                following the measurement of the diffraction intensities. See
                also _diffrn_reflns_transf_matrix_.
;
```

COD data curation – validation against dictionaries

- Several types of dictionaries (DDL1, DDL2, DDLm);
- COD validation tools in CIF1 and CIF2 frameworks (`cif_validate`, `ddlm_validate`[1]);

(Vaitkus et al. 2021)

[1] Available in the `cod-tools` package on Debian and Ubuntu systems.
[2] https://sql.crystallography.net/db/cod_validation/validation_issue

COD data curation – validation against dictionaries

- Several types of dictionaries (DDL1, DDL2, DDLm);
- COD validation tools in CIF1 and CIF2 frameworks (`cif_validate`, `ddlm_validate`[1]);

(Vaitkus et al. 2021)

Running validation on all COD yields over **11 mln.** validation messages...[2]

[1] Available in the `cod-tools` package on Debian and Ubuntu systems.

[2] https://sql.crystallography.net/db/cod_validation/validation_issue

COD validation examples

```
/usr/bin/cif_validate: 1506432.cif data_1506432:
NOTE, data item '_atom_site_aniso_label' contains value 'F40'
that was not found among the values of the parent data item
'_atom_site_label'.
```

COD validation examples

```
/usr/bin/cif_validate: 1506432.cif data_1506432:
NOTE, data item '_atom_site_aniso_label' contains value 'F40'
that was not found among the values of the parent data item
'_atom_site_label'.
```

```
loop_
_atom_site_label
_atom_site_type_symbol
_atom_site_fract_x
_atom_site_fract_y
_atom_site_fract_z
_atom_site_U_iso_or_equiv
# ... some data names omitted for brevity
>F40 F 0.21810(11) -1.5061(4) 0.7984(2) 0.0684(9) # ...
F41 F 0.29902(11) -1.4446(4) 0.8587(2) 0.0724(9)  # ...
```

COD validation examples

```
/usr/bin/cif_validate: 1506432.cif data_1506432:
NOTE, data item '_atom_site_aniso_label' contains value 'F40'
that was not found among the values of the parent data item
'_atom_site_label'.
```

```
loop_
_atom_site_label
_atom_site_type_symbol
_atom_site_fract_x
_atom_site_fract_y
_atom_site_fract_z
_atom_site_U_iso_or_equiv
# ... some data names omitted for brevity
>F40 F 0.21810(11) -1.5061(4) 0.7984(2) 0.0684(9) # ...
F41 F 0.29902(11) -1.4446(4) 0.8587(2) 0.0724(9)  # ...
```

Validation *might* help to catch data errors if applied consistently during the publication.

COD entry validation examples

- Example: wrong coordinates;
- Example: missing/wrong keys;
- Example: mistyped enumerator values;
- Example: typos in data/OCR errors?

COD entry validation examples

- Example: wrong coordinates;
- Example: missing/wrong keys;
- Example: mistyped enumerator values;
- Example: typos in data/OCR errors?

Ideally, validation should be applied during the data peer review process

Corrupted data in a text field

```
_iucr_refine_reflections_details
;
   0    0    2    -0.20     0.30  99-0.77969 0.78029 0.62494-0.62494 0.03182 0.03182
   0    0    2    -0.30     0.30 209 0.78190-0.78130-0.62292 0.62292 0.03182 0.03182

# ... lines omitted for brevity

 -15   -3   -5    -4.60     8.40 316-0.62905-0.26313 0.12897-0.27170 0.76065-0.92814
  15   -3    5    -7.40     8.00 166 0.27655 0.61563-0.16429 0.02155 0$1 0$1$0$1(0(2?
"10 0$0(0$0(0$0 0(0 0(0$0"4 0"2 0(2%0(6%0"2 0"0 0 0?   ?    4   0
0$0$0$5$0&7 0&0 0&8??  ?    4   0                      00 0(4 0"2 0"2 0(2%0(4$0" ...
0                  "10 0$4 0 4 0 4$0(0$0(0$0&4 0&2 0"0%0"5(0(4 0(0 0 0??  ?   ..."

# ... lines omitted for brevity
;
```

[Boerrigter 2023, pers. comm.]

Corrupted data in a text field

```
_iucr_refine_reflections_details
;
   0    0    2    -0.20     0.30  99-0.77969 0.78029 0.62494-0.62494 0.03182 0.03182
   0    0    2    -0.30     0.30 209 0.78190-0.78130-0.62292 0.62292 0.03182 0.03182

# ... lines omitted for brevity

 -15   -3   -5    -4.60     8.40 316-0.62905-0.26313 0.12897-0.27170 0.76065-0.92814
  15   -3    5    -7.40     8.00 166 0.27655 0.61563-0.16429 0.02155 0$1 0$1$0$1(0(2?
"10 0$0(0$0(0$0 0(0 0(0$0"4 0"2 0(2%0(6%0"2 0"0 0 0?   ?    4   0
0$0$0$5$0&7 0&0 0&8??  ?    4   0                      00 0(4 0"2 0"2 0(2%0(4$0" ...
0                  "10 0$4 0 4 0 4$0(0$0(0$0&4 0&2 0"0%0"5(0(4 0(0 0 0??  ?   ..."

# ... lines omitted for brevity
;
```

[Boerrigter 2023, pers. comm.]

It would be better to use CIF `loop_` constructs and *avoid* text fields with internal structure.

Corrupted numeric tables

```
/usr/bin/cif_validate: 2009384.cif data_2009384:
NOTE, data item '_atom_site_aniso_U_11' value
'H91' violates type constraints -- the value
should be a numerically interpretable string,
e.g. '42', '42.00', '4200E-2'.
```

Corrupted numeric tables

```
/usr/bin/cif_validate: 2009384.cif data_2009384:
NOTE, data item '_atom_site_aniso_U_11' value
'H91' violates type constraints -- the value
should be a numerically interpretable string,
e.g. '42', '42.00', '4200E-2'.
```

```
loop_
_atom_site_aniso_label
_atom_site_aniso_U_11
_atom_site_aniso_U_22
_atom_site_aniso_U_33
_atom_site_aniso_U_12
_atom_site_aniso_U_13
_atom_site_aniso_U_23
# ... some atoms omitted for brevity
C9 0.086(10) 0.061(8) 0.053(8) -0.003(7) -0.025(7) 0.008(7)
H5 0.062 H81 0.111 H82 0.111 H83
0.111 H91 0.081 H92 0.081 H93 0.081
```

COD entry checks – IUCr criteria checks

- Checks on prepublications and Personal communications;
- Checks on published structures;
- *Statistics of structures in the database*

IUCr data validation criteria (Version: 2000.06.09,
ftp://ftp.iucr.ac.uk/pub/dvntests or
ftp://ftp.iucr.org/pub/dvntests)

COD entry checks – IUCr criteria checks

- Checks on prepublications and Personal communications;
- Checks on published structures;
- *Statistics of structures in the database*

IUCr data validation criteria (Version: 2000.06.09, ftp://ftp.iucr.ac.uk/pub/dvntests or ftp://ftp.iucr.org/pub/dvntests)

```
cif_cod_check 3000424.cif
```

```
/usr/bin/cif_cod_check: 3000424.cif data_3000424: NOTE, data item
    '_refine_ls_R_factor_gt' value '0.1120' is > 0.1.
/usr/bin/cif_cod_check: 3000424.cif data_3000424: NOTE, data item
    '_refine_ls_wR_factor_ref' value '0.3195' is > 0.25.
/usr/bin/cif_cod_check: 3000424.cif: NOTE, 2 NOTE(s) encountered.
```

COD entry checks – IUCr criteria checks

- Checks on prepublications and Personal communications;
- Checks on published structures;
- *Statistics of structures in the database*

```
cif_cod_check 3000424.cif
```

```
/usr/bin/cif_cod_check: 3000424.cif data_3000424: NOTE, data item
    '_refine_ls_R_factor_gt' value '0.1120' is > 0.1.
/usr/bin/cif_cod_check: 3000424.cif data_3000424: NOTE, data item
    '_refine_ls_wR_factor_ref' value '0.3195' is > 0.25.
/usr/bin/cif_cod_check: 3000424.cif: NOTE, 2 NOTE(s) encountered.
```

COD entry checks – IUCr criteria checks

- Checks on prepublications and Personal communications;
- Checks on published structures;
- *Statistics of structures in the database*

`cif_cod_check 3000424.cif`

```
/usr/bin/cif_cod_check: 3000424.cif data_3000424: NOTE, data item
    '_refine_ls_R_factor_gt' value '0.1120' is > 0.1.
/usr/bin/cif_cod_check: 3000424.cif data_3000424: NOTE, data item
    '_refine_ls_wR_factor_ref' value '0.3195' is > 0.25.
/usr/bin/cif_cod_check: 3000424.cif: NOTE, 2 NOTE(s) encountered.
```

- Area specific quality criteria are helpful;
- Data need to be reviewed by intelligent reviewers;

COD internal consistency checks

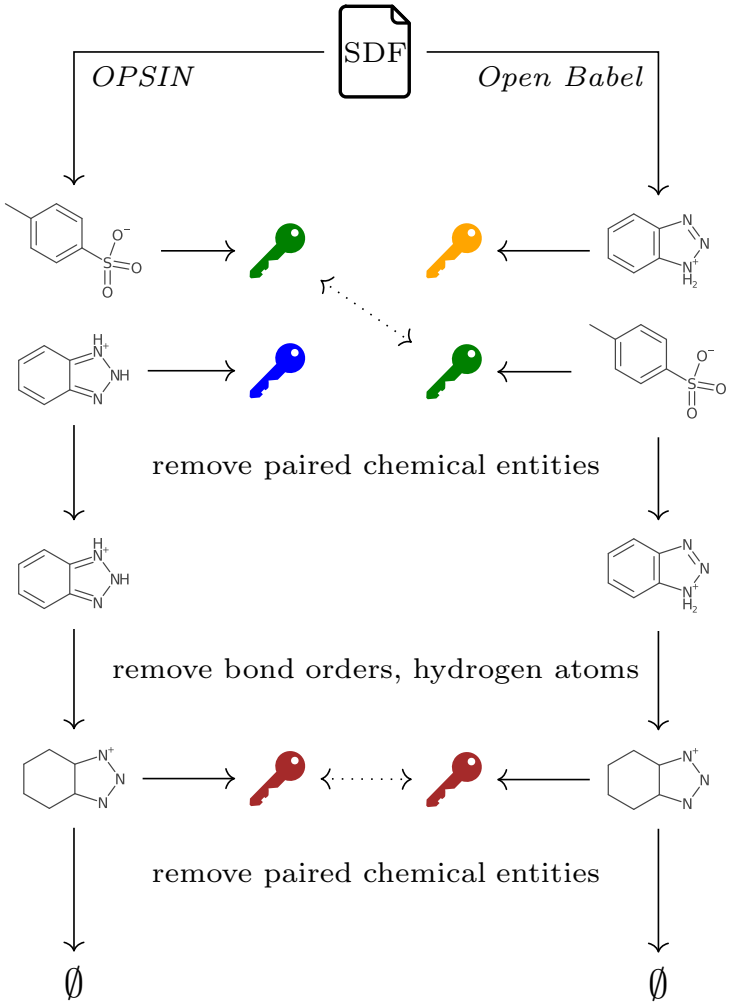
- Checks of/against deposited F_{obs} data;

(Henn 2019)

COD has over **58 000** Fobs files; most recent COD files contain SHELX HKL data as a text field...

- Checks using **QM relaxation** with F/LOSS DFT and QM codes; work in progress ...
- Checks against **QM-derived ML potentials** (fast and fairly accurate); work in progress (Linas Vilčiauskas, Vytautas Žalandauskas, Justinas Šlepavičius)

Matching the chemical structure graphs



(Merkys et al. 2023)

Fraudulent structures...

- more than 100 published structures were falsified;
- looked “OK” based on usual criteria;
- detected by crystallographers in the IUCr-led effort; based on implausible chemistry

Can we limit data fraud and honest mistakes?

- data *must* be reviewed as the main text, and possibly even more thoroughly;
- collaborative tools are necessary (a-la GitLab or GitHub); work in progress;
- reviewers for data as well as reviewers for paper text?

Documenting COD changes

In the version control system

Subversion logs document all changes:

`svn log -l 2 \$(codid2file 2105675)`

```
------------------------------------------------------------------------
r281832 | saulius | 2023-03-13 09:35:08 +0200 (Mon, 13 Mar 2023) | 12 lines

cod/ (saulius@tasmanijos-velnias)
Updating 15 COD entries:
- Adding symmetry operator fixes by Steef Boerrigter;
- Marking '_geom_...._symmetry_...' data items as unknown (value '?');
- converting _cod_depositor_comments to the structured
  _cod_changelog_... loop;
- fixing several issues reported by CIF validators ('cif_validate',
  'ddlm_validate') and by 'cif_cod_check.

All details of the changes are given in the _cod_changelog_... and
_cod_entry_issue_... loops.

------------------------------------------------------------------------
r176768 | antanas | 2016-02-20 02:53:41 +0200 (Sat, 20 Feb 2016) | 4 lines

cif/2/ (antanas@kurmis)
Replacing _[local]_cod_* tags with their equivalents from the COD CIF
dictionary in multiple entries in subrange 2/10.

------------------------------------------------------------------------
```

Documenting COD changes

In the CIF

Previously, reports about issues and structure curation were given in the `_cod_depositor_comment` data item (example from COD 1516168):

```
_cod_depositor_comments
;
The following automatic conversions were performed:

'_diffrn_ambient_temperature' value 'room temperature' was changed to
'295(2)' - the room/ambient temperature average [293;298] in
Kelvins(K) was taken.

Automatic conversion script
Id: cif_fix_values 2281 2013-09-26 08:29:07Z andrius
;
```

Documenting COD changes

A structured way

Within CIFs, changes are documented in the category:

`'_cod_changelog_entry_[]'`

(example from COD 2105675):

```
loop_
_cod_changelog_entry_id
_cod_changelog_entry_author
_cod_changelog_entry_date
_cod_changelog_entry_text
5 'Saulius Gra\<zulis' 2023-02-24T15:04:53+02:00
;Converting _cod_depositor_comments to the _cod_changelog_entry_... loop.
;

6 'Steef Boerrigter' 2023-02-22T10:07:00+02:00
;Corrected space group operators:
added a missing a minus symbol
x,y-1/2,z+1/2 --> x,-y+1/2,z+1/2
;
```

Describing COD issues

Structured reports are in the '_cod_entry_issue_[]' category (example from COD 1516168):

```
loop_
_cod_entry_issue_id
_cod_entry_issue_origin
_cod_entry_issue_severity
_cod_entry_issue_description
_cod_entry_issue_author
_cod_entry_issue_date
1 upstream warning
;
  Crystal data of this structure was keyed in from a scanned copy of the
  original publication:

  Amit, A.; Mester, L.; Klewe, B. & Furberg, S.

  Some atom coordinates and/or anisotropic factors may be incorrect as
  the digits in the scanned copy could not be read clearly.
;
'Andrius Merkys' 2014-05-21
```

COD duplicates

Definition (Duplicate)

is a structure published with identical unit cell in the same publication and measured from the same sample under the same conditions.

There are currently 5413 marked duplicates in the COD.

```
codslave 'select file, duplicateof, a, b, c, sg, formula, doi from data where file in (7030506,7002340)'
```
[3]

```
+---------+-------------+--------+--------+--------+------------+-------------------+------------------+
| file    | duplicateof | a      | b      | c      | sg         | formula           | doi              |
+---------+-------------+--------+--------+--------+------------+-------------------+------------------+
| 7002340 |        NULL | 8.3079 | 19.091 | 16.587 | P 1 21/c 1 | - C25 H37 N7 Zn - | 10.1039/b910048b |
| 7030506 |     7002340 | 8.3079 | 19.091 | 16.587 | P 1 21/c 1 | - C25 H37 N7 Zn - | 10.1039/b910048b |
+---------+-------------+--------+--------+--------+------------+-------------------+------------------+
```

```
_cod_duplicate_entry                    7002340
```

[3] alias codslave='mysql -u cod_reader -h sql.crystallography.net cod -e'

COD suboptimal structures

Definition (Suboptimal structure)

A structure that was deliberately solved and published with worse-than-ideal parameters to provide the evidence that the alternative structure is better.

There are currently 38 structures marked as 'suboptimal' in the COD.

```
codslave 'select file, optimal, a, b, c, sg, sgHall, formula from data where file in (2100860,2100858)'
```
[4]

```
+---------+---------+--------+--------+-------+-----------+--------+--------------+
| file    | optimal | a      | b      | c     | sg        | sgHall | formula      |
+---------+---------+--------+--------+-------+-----------+--------+--------------+
| 2100858 |    NULL | 3.9998 | 3.9998 | 4.018 | P 4 m m   | P 4 -2 | - Ba O3 Ti - |
| 2100860 | 2100858 | 3.9998 | 3.9998 | 4.018 | P 4/m m m | -P 4 2 | - Ba O3 Ti - |
+---------+---------+--------+--------+-------+-----------+--------+--------------+
```

```
_cod_related_optimal_struct       2100858
```

[4] alias codslave='mysql -u cod_reader -h sql.crystallography.net cod -e'

COD structures without coordinates

Some old structures (1600) do not have coordinates coordinates (example from COD):

```
_publ_section_title
;
 Magnetic and electric properties of RCo6Ge6 (R = Y, Dy, Er-Lu)
;
loop_
_atom_site_label
_atom_site_fract_x
_atom_site_fract_y
_atom_site_fract_z
? ? ? ?
```

COD structures without coordinates

Some old structures (1600) do not have coordinates coordinates (example from COD):

```
_publ_section_title
;
 Magnetic and electric properties of RCo6Ge6 (R = Y, Dy, Er-Lu)
;
loop_
_atom_site_label
_atom_site_fract_x
_atom_site_fract_y
 atom_site_fract_z
? ? ? ?
```

COD structures without coordinates

Some old structures (1600) do not have coordinates coordinates (example from COD):

```
_publ_section_title
;
 Magnetic and electric properties of RCo6Ge6 (R = Y, Dy, Er-Lu)
;
loop_
_atom_site_label
_atom_site_fract_x
_atom_site_fract_y
 atom_site_fract_z
? ? ? ?
```

Programs should be prepared to handle `'?'` data values.

COD structures with missing coordinates

A few structures are missing coordinates (example: COD 5900030):

```
loop_
_atom_site_label
_atom_site_type_symbol
_atom_site_symmetry_multiplicity
_atom_site_Wyckoff_symbol
_atom_site_fract_x
_atom_site_fract_y
_atom_site_fract_z
Br1 Br 4 e . 0.228 0.216
Br2 Br 4 e . 0.461 0.387
```

COD structures with missing coordinates

A few structures are missing coordinates (example: COD 5900030):

```
loop_
_atom_site_label
_atom_site_type_symbol
_atom_site_symmetry_multiplicity
_atom_site_Wyckoff_symbol
_atom_site_fract_x
_atom_site_fract_y
_atom_site_fract_z
Br1 Br 4 e . 0.228 0.216
Br2 Br 4 e . 0.461 0.387
```

COD structures with missing coordinates

A few structures are missing coordinates (example: COD 5900030):

```
loop_
_atom_site_label
_atom_site_type_symbol
_atom_site_symmetry_multiplicity
_atom_site_Wyckoff_symbol
_atom_site_fract_x
_atom_site_fract_y
_atom_site_fract_z
Br1 Br 4 e . 0.228 0.216
Br2 Br 4 e . 0.461 0.387
```

Programs should be prepared to handle `'.'` data values.

Fraudulent and retracted structures

```
codslave 'select file, doi, status, sg, formula from data
    where status like "%retracted%" order by file limit 3'
```

```
+---------+--------------------------------+-----------+------------+--------------------------+
| file    | doi                            | status    | sg         | formula                  |
+---------+--------------------------------+-----------+------------+--------------------------+
| 1558625 | 10.1103/PhysRevLett.125.016001 | retracted | F m -3 m   | - C -                    |
| 2015946 | 10.1107/S0108270107001977      | retracted | C 1 2/c 1  | - C4 H18 N2 Na2 O15 -    |
| 2204639 | 10.1107/S1600536804028296      | retracted | P 1 21/c 1 | - C40 H56 Cl2 Cu N4 O4 - |
+---------+--------------------------------+-----------+------------+--------------------------+
```

```
loop_
_atom_site_type_symbol
_atom_site_label
_atom_site_fract_x
_atom_site_fract_y
_atom_site_fract_z
_atom_site_U_iso_or_equiv
_atom_site_adp_type
_atom_site_calc_flag
_atom_site_refinement_flags
_atom_site_occupancy
. . . . . .
```

Fraudulent and retracted structures

```
codslave 'select file, doi, status, sg, formula from data
    where status like "%retracted%" order by file limit 3'
```

```
+---------+--------------------------------+-----------+------------+--------------------------+
| file    | doi                            | status    | sg         | formula                  |
+---------+--------------------------------+-----------+------------+--------------------------+
| 1558625 | 10.1103/PhysRevLett.125.016001 | retracted | F m -3 m   | - C -                    |
| 2015946 | 10.1107/S0108270107001977      | retracted | C 1 2/c 1  | - C4 H18 N2 Na2 O15 -    |
| 2204639 | 10.1107/S1600536804028296      | retracted | P 1 21/c 1 | - C40 H56 Cl2 Cu N4 O4 - |
+---------+--------------------------------+-----------+------------+--------------------------+
```

```
loop_
_cod_entry_issue_id
_cod_entry_issue_origin
_cod_entry_issue_severity
_cod_entry_issue_description
1 original retraction
;
    This file does not describe any real compound and was retracted by
    the authors. The previously deposited coordinates can be reviewed
    in older revisions of this file or in the COD Subversion
    repository, svn://www.crystallography.net/cod/retracted/.

    The precise reasons for the retraction and other circumstances are
    extensively discussed in the Acta Cryst. E editorial:

    Acta Cryst. E 68 (2012) e14--e15
;
```

Selecting desired structures

SQL statement example:

```
SELECT file AS codid
FROM `data` WHERE flags LIKE "%has coordinates%" AND
(status IS NULL OR
 NOT (status LIKE "%retracted%" OR
       status LIKE "%error%")) AND
duplicateof IS NULL AND
optimal IS NULL AND
(method IS NULL OR method != "theoretical");
```

Accessing the COD

COD data can be accessed:

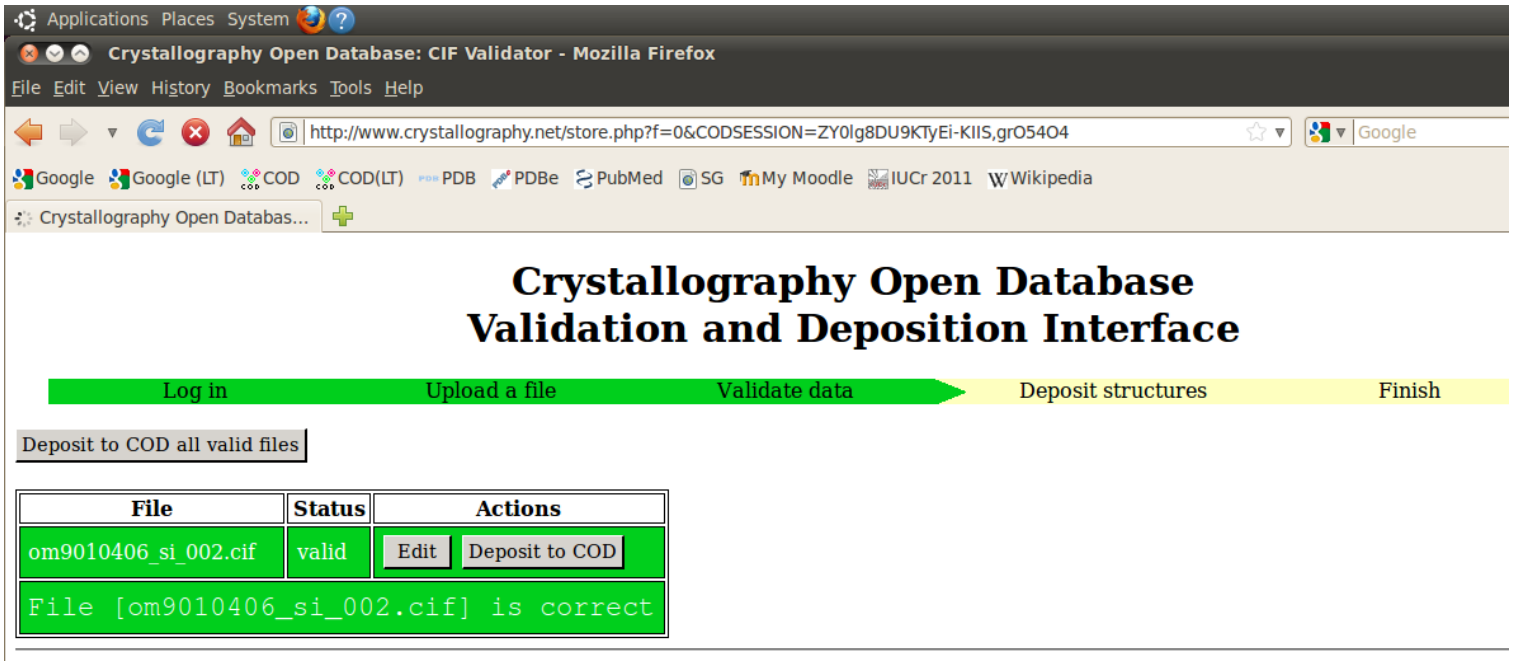
1. Via the Web page:
   https://www.crystallography.net/cod/7159763.html
2. Via the COD REST API:
   https://www.crystallography.net/cod/7159763.cif
   https://www.crystallography.net/cod/result?text=perovskite
3. Via the OPTIMADE API (Andersen et al. 2021):
   https://www.crystallography.net/cod/optimade/structures?filter=elements+HAS+"U"
4. Via SQL:
   ```
   mysql -u cod_reader -h sql.crystallography.net cod -e \
   'select file from data where formula = "- H2 O -"'
   ```
5. By downloading to your computer using Subversion, `rsync` or simple Web download:
   https://wiki.crystallography.net/howtoobtaincod

COD validation and deposition Web site

https://www.crystallography.net/cod/deposit
https://www.crystallography.net/cod/initiate_deposition.php

COD validation and deposition Web site

https://www.crystallography.net/cod/deposit
https://www.crystallography.net/cod/initiate_deposition.php



Programmatic data deposition

Structures can be deposited programmatically with a user name and password:

```
$ curl
        --silent
        --show-error
        --user-agent '<your program ID + version, pls.  :)>'
        -F strict=1
        -F output_mode=html
        -F username=</tmp/tmp-cif_cod_deposit-7183/username
        -F password=</tmp/tmp-cif_cod_deposit-7183/password
        -F cif=@input.cif;filename=input.cif
        -F deposition_type=published
https://www.crystallography.net/cod/cgi-bin/cif-deposit.pl
```

COD data curation

Inputs from COD users

Thomas Dortmann (2013), PANalytical, “`COD-minerals.xlsx`”:

> *The entries that now have the mineral name are minerals, the rest are not.*

$> 3\,500$ unique mineral names assigned 104 “atypical” names[5].

Update (2024):

$> 4\,257$ unique mineral names, 566 “atypical” names

[5] Not matching the RE `/^[A-Z][-a-zA-Z ()]+$/`

COD versioning

Essential for reproducibility

The COD Subversion revisions can be accessed on the Web:

▾ Version history

Revision	Date	Message	Files
277834 (current)	2022-09-14	cif/ Added space group information derived from the space group operation list using the 'cif_filter' program.	2000000.cif
199748	2017-08-14	cif/2/00/00/ (antanas@echidna.ibt.lt) Removing 43 symmetrically equivalent atoms in entry 2000000.	2000000.cif

- The latest revision has a stable URI:
  https://www.crystallography.net/cod/2000000.cif
- A URI with a specific revision allows to reconstruct the *specific byte stream*:
  https://www.crystallography.net/cod/2000000.cif@199748

CIF management tools:

1. CIF pretty-printer and autocorrector:
   `$ cifparse --fix --print 7234818.cif`

```
echo 'data_ _tag some value _tag "some value" _tag2 "quote?' | cifparse --fix --print
```

```
cifparse: -(1): WARNING, zero-length data block name detected -- ignored
cifparse: -(1): WARNING, string with spaces without quotes -- fixed
cifparse: -(1): WARNING, tag _tag appears more than once with the same value 'some value'
cifparse: -(1): WARNING, double-quoted string is missing a closing quote -- fixed
```

```
data_
_tag                                    'some value'
_tag2                                   "quote?"
```

Commands from the cod-tools package:
svn://cod.ibt.lt/cod-tools
https://github.com/cod-developers/cod-tools
^/trunk revision r10308.

CIF data extraction – cifvalues:

1. Obtain CIF values:

   ```
   $ cifvalues --header --tsv --tags ...
   ```

```
cifvalues --header --tsv --tags _cell_formula_units_Z,_chemical_formula_sum 2200010.cif 4031234.cif
```

```
dblname _cell_formula_units_z   _chemical_formula_sum
2200010 8        C12 H12 N2 O S
4031234 2        Cl5 Cs2 H8 O4 V
```

Commands from the cod-tools package:
svn://cod.ibt.lt/cod-tools
https://github.com/cod-developers/cod-tools
^/trunk revision r10308.

CIF data extraction – cif2csv, cif2tsv, cif2adt:

1. Obtain CIF values:

   ```
   $ cif2csv --header --tags ...
   ```

```
cif2csv --header --tags _cell_formula_units_Z,_chemical_formula_sum 2200010.cif 4031234.cif
```

```
dblname,tag,index,loopnr,value,filename
2200010,_chemical_formula_sum,0,-1,"C12 H12 N2 O S",2200010.cif
2200010,_cell_formula_units_z,0,-1,8,2200010.cif
4031234,_chemical_formula_sum,0,-1,"Cl5 Cs2 H8 O4 V",4031234.cif
4031234,_cell_formula_units_z,0,-1,2,4031234.cif
```

Supports the following output formats: CSV, TSV and ADT (ASCII Delimited Text – uses ASCII GS, RS and US control characters as delimiters).

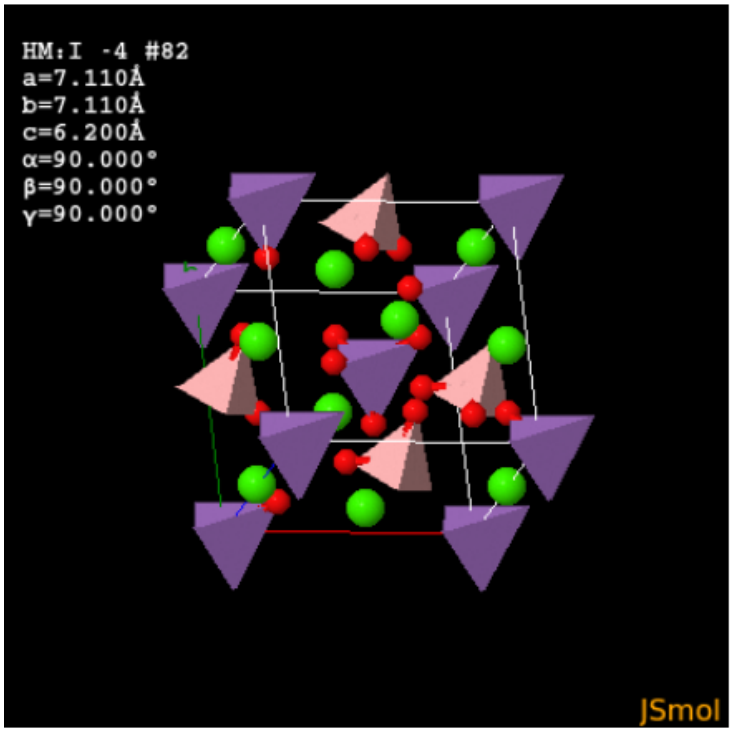
Commands from the cod-tools package:
svn://cod.ibt.lt/cod-tools
https://github.com/cod-developers/cod-tools
^/trunk revision r10308.

Data cross-referencing

External links

Links to external databases are implemented and populated:

- Implemented: AMCSD, Wikidata, Wikipedia, MPOD, ChemSpider;
- Planned: PubChem, **raw diffraction data**;



Coordinates 9016740.cif
External links AMCSD; Wikidata; Wikipedia

Interlinked data in COD

Crystallography Open Database

COD Home
Home
What's new?

Accessing COD Data
Browse
Search
Search by structural formula

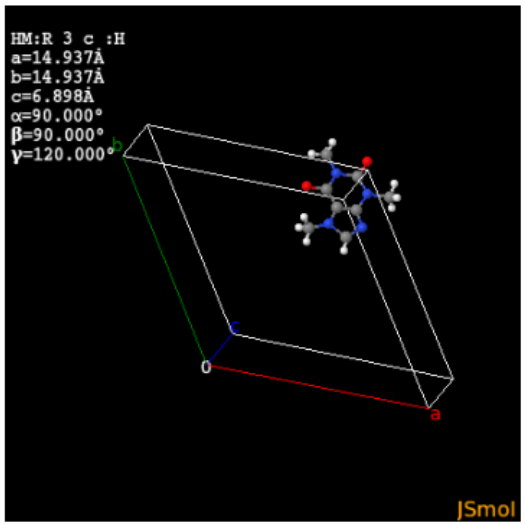
Add Your Data
Deposit your data
Manage depositions
Manage/release prepublications

Documentation
COD Wiki
Obtaining COD
Querying COD
Citing COD
COD Mirrors
Advices to donators
Useful links

Information card for 2100202

2100201 << **2100202** >> 2100203

Preview



Display in Jmol

Coordinates 2100202.cif
Original IUCr paper HTML
External links ChemSpider; DrugBank; PubChem; Wikipedia

▾ **Structure parameters**

```
select * from wikipedia_x_cod
+----+---------------+---------+-------------+
| id | ext_id        | cod_id  | relation_id |
+----+---------------+---------+-------------+
|  1 | Ibuprofen     | 2006278 |           1 |
|  2 | Caffeine      | 2100202 |           1 |
|  3 | Serotonin     | 2019147 |           1 |
|  4 | Pristinamycin | 1000001 |           1 |
|  5 | Cucurbituril  | 1516465 |           1 |
|  6 | Rubrene       | 1516682 |           1 |
+----+---------------+---------+-------------+
```

Group theory in Ada/SPARK

examples/group_theory.ads

```
pragma Spark_Mode (On);

generic
   type Element is private;
   Identity : Element;
   with function "*" (E, F: Element) return Element is <>;
```

```
   function Is_Closed_On_Multiplication (G : Group) return Boolean
   is (for all E of G =>
         (for all F of G => (Belongs_To (E*F, G))))
     with Ghost;

   function All_Elements_Have_Inverses (G : Group) return Boolean
   is (for all E of G => Has_Inverse (E, G))
     with Ghost;

   function Is_Group (G : Group) return Boolean
   is (Has_Identity (G) and then
         All_Elements_Have_Inverses (G) and then
         Is_Closed_On_Multiplication (G)
      )
     with Ghost;
```

(Petrauskas et al. 2022)

Automatic compilation of proven code

Ada and SPARK

examples/make_group.ads

```
8     type Ring_Element is mod 37;
```

```
29    function Build_Group (E : Ring_Element) return Group
30      with
31      Post => Is_Group (Build_Group'Result);
```

gnatprove -P main.gpr --report=all make_group.adb

```
make_group.ads:23:14: info: postcondition proved
make_group.ads:27:14: info: postcondition proved
make_group.ads:31:14: info: postcondition proved
group_theory.ads:16:15: info: postcondition proved, in instantiation at make_group.ads:16
```

```
saulius@tasmanijos-velnias spacegroups/ $ ./run_make_group 8
(1, 8, 27, 31, 26, 23, 36, 29, 10, 6, 11, 14)

saulius@tasmanijos-velnias spacegroups/ $ ./run_make_group 7
(1, 7, 12, 10, 33, 9, 26, 34, 16)
```

Where to go further?

- Derive chemical names;
- Collect more structures;
- Find all papers with crystal structures;
- Apply machine learning;
- Expand the community – **your contributions are invaluable!**

Acknowledgements

VU LSC IBT (KICIS)

Andrius Merkys
Antanas Vaitkus
Algirdas Grybauskas
Yaroslav Rozdobudko

QM community

Audrius Alkauskas
Vytautas Žalandauskas
Lukas Razinkovas
Nicola Marzari
Giovanni Pizzi
Lubomir Smrcok
Linas Vilčiauskas
Rickard Armiento

VU MIF II (FMG)

Linas Laibinis
Karolis Petrauskas

COD Advisory board

Daniel Chateigner
Robert T. Downs
Werner Kaminsky
Armel Le Bail
Luca Lutterotti
Peter Moeck
Peter Murray-Rust
Miguel Quirós

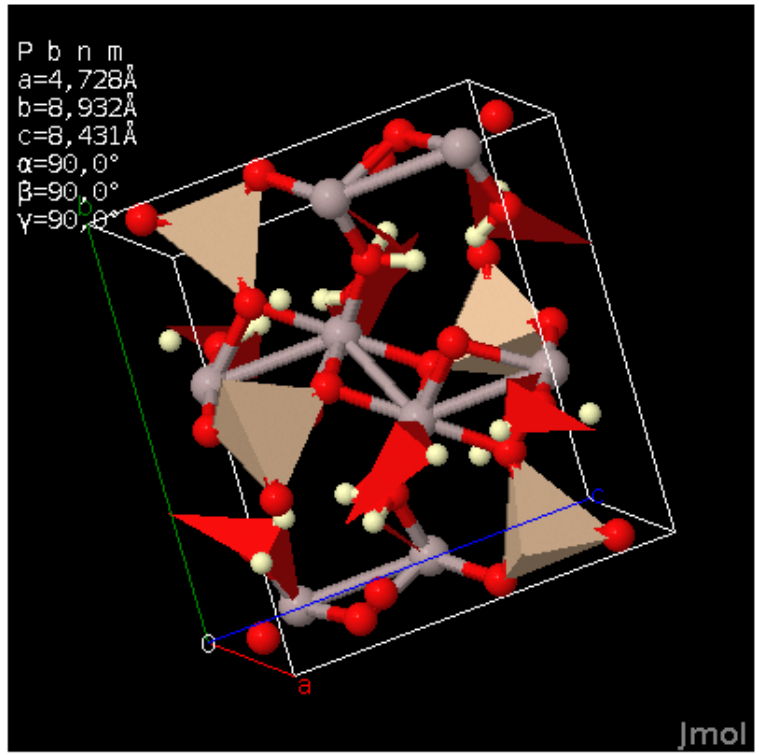
Cheminf community

Evan Bolton
Paul Thiessen
Thomas Sander

Enormous thanks for our commercial users and supporters: PANalytical, Rigaku, Bruker

Funding:
Lithuanian-French Program “Gilibert” (RCoL grant S-LZ-23-3); CECAM; RCoL grants S-MIP-20-21, S-MIP-23-87, VU Intramural funding.
Data Centre: **Research Council of Lithuania Project No.: S-A-UEI-23-11**

Thank you!



Coordinates	2207377.cif
Original IUCr paper	HTML

http://en.wikipedia.org/wiki/Topaz

http://www.crystallography.net/2207377.html

https://www.crystallography.net/archives/2024/slides/NOBUGS-Database-Workshop/slides1-techniques.pdf

References I

Andersen, Casper W. et al. (Aug. 2021). "OPTIMADE, an API for exchanging materials data". In: *Scientific Data* 8.1, pp. 1–10. DOI: 10.1038/s41597-021-00974-z.

Bernstein, Herbert J. et al. (Feb. 2016). "Specification of the Crystallographic Information File format, version 2.0". In: *Journal of Applied Crystallography* 49.1, pp. 277–284. ISSN: 1600-5767. DOI: 10.1107/s1600576715021871. URL: http://dx.doi.org/10.1107/S1600576715021871.

Downs, Robert T. et al. (2003). "The American Mineralogist crystal structure database". In: *American Mineralogist* 88, pp. 247–250. URL: http://geo.arizona.edu/xtal/group/pdf/am88_247.pdf.

Fuentes-Cobas, Luis E. et al. (Aug. 2017). "The representation of coupling interactions in the Material Properties Open Database (MPOD)". In: *Advances in Applied Ceramics* 116.8, pp. 428–433. DOI: 10.1080/17436753.2017.1343782.

Gražulis, Saulius et al. (2009). "Crystallography Open Database – an open-access collection of crystal structures". In: *Journal of Applied Crystallography* 42, pp. 726–729. DOI: 10.1107/S0021889809016690. URL: http://dx.doi.org/10.1107/S0021889809016690.

Gražulis, Saulius et al. (2012). "Crystallography Open Database (COD): an open-access collection of crystal structures and platform for world-wide collaboration". In: *Nucleic Acids Research* 40, pp. D420–D427. DOI: 10.1093/nar/gkr900. URL: http://nar.oxfordjournals.org/content/40/D1/D420.abstract.

References II

Hall, S. R. et al. (1991). "The crystallographic information file (CIF): a new standard archive file for crystallography". In: *Acta Crystallographica Section A* 47, pp. 655–685. DOI: 10.1107/S010876739101067X. URL: http://dx.doi.org/10.1107/S010876739101067X.

Henn, Julian (Apr. 2019). "Metrics for crystallographic diffraction- and fit-data: a review of existing ones and the need for new ones". In: *Crystallography Reviews* 25.2, pp. 83–156. ISSN: 1476-3508. DOI: 10.1080/0889311x.2019.1607845. URL: http://dx.doi.org/10.1080/0889311X.2019.1607845.

Mendili, Yassine El et al. (May 2019). "Raman Open Database: first interconnected Raman–X-ray diffraction open-access resource for material identification". In: *Journal of Applied Crystallography* 52.3, pp. 618–625. DOI: 10.1107/s1600576719004229.

Merkys, Andrius et al. (Feb. 2016). "*COD::CIF::Parser*: an error-correcting CIF parser for the Perl language". In: *Journal of Applied Crystallography* 49.1, pp. 292–301. DOI: 10.1107/S1600576715022396. URL: http://dx.doi.org/10.1107/S1600576715022396.

Merkys, Andrius et al. (Feb. 2023). "Graph isomorphism-based algorithm for cross-checking chemical and crystallographic descriptions". In: *Journal of Cheminformatics* 15.1. ISSN: 1758-2946. DOI: 10.1186/s13321-023-00692-1. URL: https://jcheminf.biomedcentral.com/articles/10.1186/s13321-023-00692-1.

References III

Pepponi, Giancarlo et al. (2012). "MPOD: A Material Property Open Database linked to structural information". In: *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms* 284.0. E-MRS 2011 Spring Meeting, Symposium M: X-ray techniques for materials research-from laboratory sources to free electron lasers, pp. 10–14. ISSN: 0168-583X. DOI: 10.1016/j.nimb.2011.08.070. URL: http://www.sciencedirect.com/science/article/pii/S0168583X11008639.

Petrauskas, Karolis et al. (May 2022). "Proving the correctness of the algorithm for building a crystallographic space group". In: *Journal of Applied Crystallography* 55.3, pp. 515–525. DOI: 10.1107/s1600576722003107.

Rajan, H. et al. (2006). "Building the American Mineralogist Crystal Structure Database: A recipe for construction of a small Internet database". In: *Geoinformatics: Data to Knowledge*. Ed. by A.K. Sinha. Vol. 397. Geological Society of America Special Papers. Boulder, CO, United States: Geological Society of America, pp. 73–80. DOI: 10.1130/2006.2397(06).

Vaitkus, Antanas et al. (Feb. 2021). "Validation of the Crystallography Open Database using the Crystallographic Information Framework". In: *Journal of Applied Crystallography* 54.2, pp. 1–12. ISSN: 1600-5767. DOI: 10.1107/s1600576720016532. URL: https://doi.org/10.1107/S1600576720016532.