HANDLING INORGANIC AND ORGANOMETALLIC STEREOCHEMISTRY IN RDKIT

Roger Sayle and John Mayfield
NextMove Software, Cambridge, UK
Greg Landrum
ETH Zurich, Zurich, Switzerland

10th RDKit UGM, Virtual Online, Friday 15th October 2021

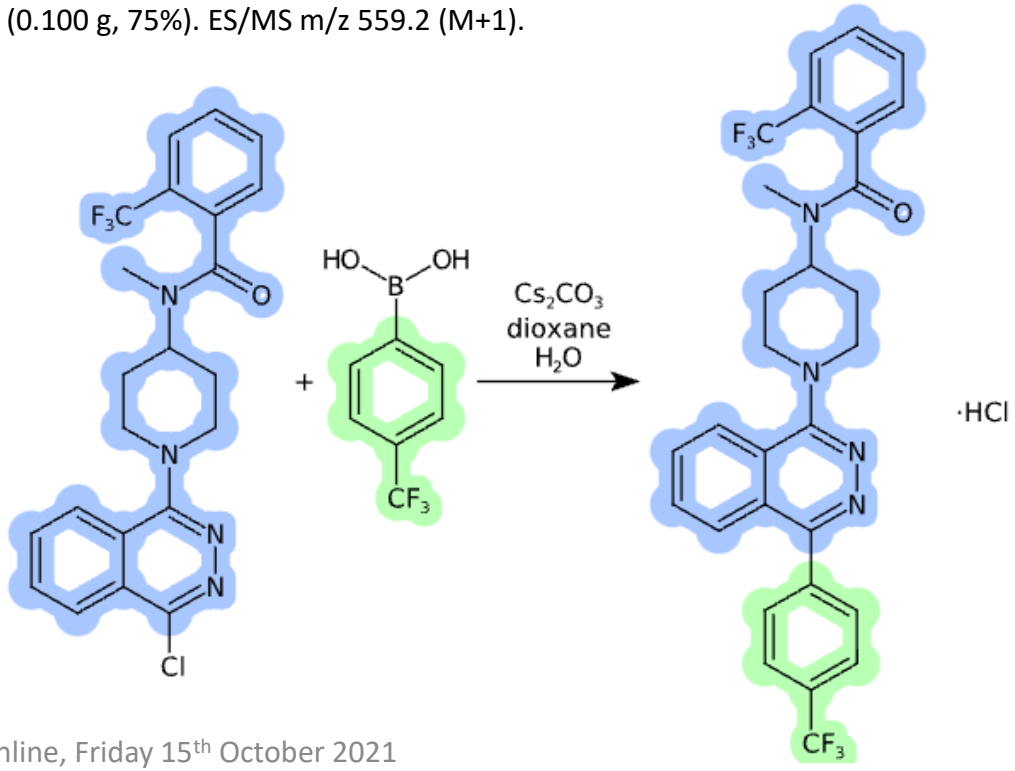
MOTIVATIONS

- Representing reagents and catalysts in reactions.
- Lack of support/normalization in ChEMBL/PubChem.
- Slow progress from InChI “inorganic” working group.
- Influence RDKit’s design decisions.

US20110046143A1 [0102]

N-Methyl-2-(trifluoromethyl)-N-(1-(4-(4-(trifluoromethyl)phenyl)phthalazin-1-yl) piperidin-4-yl)benzamide hydrochloride

Charge a microwave vessel with N-(1-(4-chlorophthalazin-1-yl)piperidin-4-yl)-N-methyl-2-(trifluoromethyl) benzamide (0.101 g, 0.23 mmol), 4-(trifluoromethyl)phenylboronic acid (0.171 g, 0,9 mmol), cesium carbonate (0.295 g, 0.91 mmol), 1,4-dioxane (3 mL), and water (1 mL). Purge the reaction vial two times with nitrogen. Add **(SP-4-1)-bis[bis(1,1-dimethylethyl)(4-methoxyphenyl)phosphine-κP]dichloro-palladium** (J. Org. Chem. 2007, 72, 5104-5112) (0.002 g; 0.003 mmol) and heat the reaction at 90° C. for 16 h. After cooling, separate the two layers and remove the water. Evaporate the organic solvent with a stream of nitrogen. Purify the residue from the organic layer using silica gel chromatography (0-10% methanol in dichloromethane). Add 4 N HCl in dioxane to a solution of the isolated product in methanol and remove the solvents in vacuo to obtain the title compound (0.100 g, 75%). ES/MS m/z 559.2 (M+1).



RDKIT PERSPECTIVE

- Internal representation
- SMILES: reading, writing and canonicalization.
- 3D Coordinates: reading and writing.
- 2D Depictions: reading and writing.
- IUPAC names/CIP: reading and writing.
- SMARTS: patterns and matching.

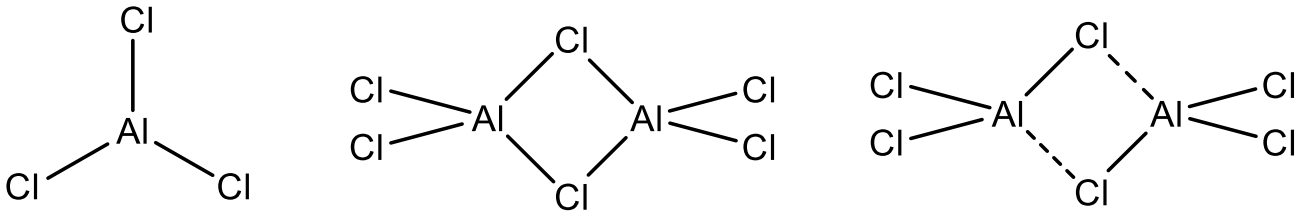
3D DATA SETS

- COD (Crystallography Open Database)
  - http://www.crystallography.net/cod/
- CSD (Cambridge Structural Database)
  - https://www.ccdc.cam.ac.uk/structures/?
- PDB (Protein Data Bank)
  - https://www.rcsb.org/

- Peter Schmidtke from Discngine's DataWarrior COD to 3D SDF protocol:
  https://pschmidtke.github.io/blog/rdkit/crystallography/small%20molecule%20xray/xray/database/2021/01/25/cod-and-torsion-angles.html

THE FIRST HURDLE

- Bonds to metal elements exist, get over it.
- Alas between solid state (crystal) effects and dilute aqueous phenomena the situation can be confusing.



- Indeed OPSIN's (ab)use of the van Arkel diagram was a motivating factor in developing inorganic name to structure (and structure to name): cupric chloride, cuprous chloride and vanadium oxychloride.

HOW TETRAHEDRAL STEREO WORKS #1

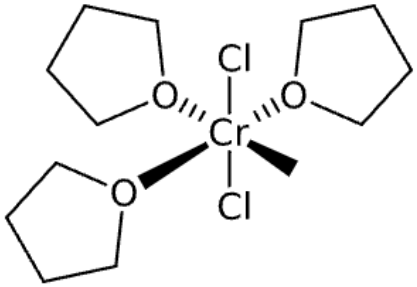
- RDKit stores an Atom::ChiralType on each atom:
  - CHI_TETRAHEDRAL_CW, CHI_TETRAHEDRAL_CCW, CHI_UNSPECIFIED and CHI_OTHER.
- This records the “parity” (or “handedness”) of the neighbors around a centre in getAtomBondsOrder.

NON-TETRAHEDRAL STEREOCHEMISTRY

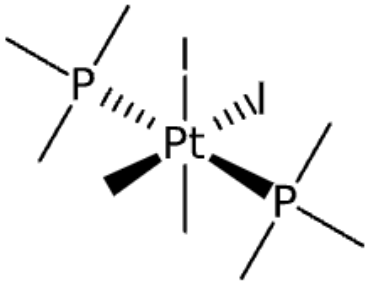
- Beyond tetrahedral stereochemistry and its two permutations (parity), are a variety of geometries with different (larger) numbers of permutations.

Class	Degree	Pairs	IUPAC	SMILES	Perm	Parent
Octahedral	6	3	(OC-6)	@OH	30	
Trigonal bipyramidal	5	1	(TBPY-5)	@TB	20	
Square planar	4	2	(SP-4)	@SP	3	@OH
T-shape	3	1	(TS-3)		3	@SP/@OH
Square pyramid	5	2	(SPY-5)		30	@OH
See-saw (major)	4	1	(SS-4)		20	@TB
See-saw (minor)	4	1	(SS-4)			@OH

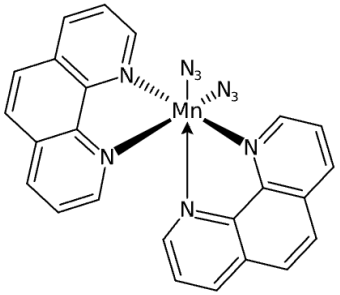
OCTAHEDRAL EXAMPLES



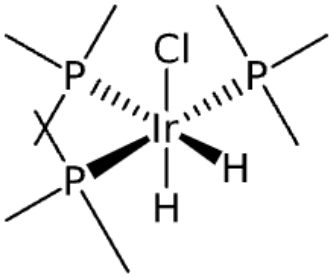
COD4073210



COD4130638



COD2008644



COD4078254

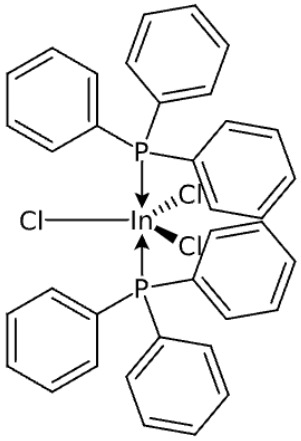
[Cr@OH25](Cl)(Cl)(O1CCCC1)(O1CCCC1)(O1CCCC1)C COD4073210

SMILES examples (test suite) available at
https://github.com/greglandrum/rdkit/commit/46c57c99a149b1083a1e9c69c823fe626ae05af6

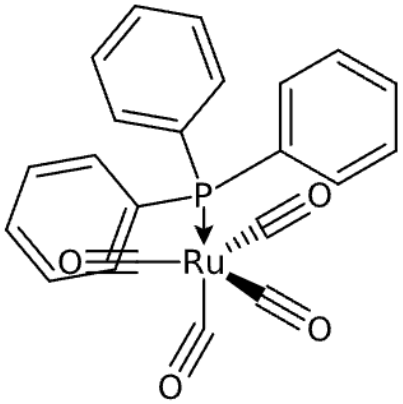
3D Structures at e.g. http://www.crystallography.net/cod/4073210.html

CDKDepict at https://www.simolecule.com/cdkdepict/depict.html

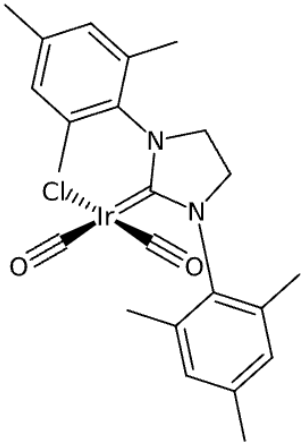
NON-OCTAHEDRAL EXAMPLES



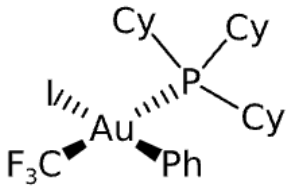
COD4102282



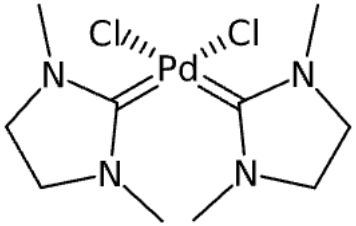
COD4334746



COD4070781



COD4121836



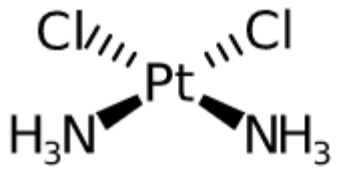
COD1514811

SMILES PERSPECTIVE

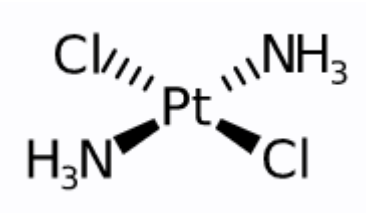
- SMILES supports a number of chiral classes, each allowing a permutation number to be specified on atom.
- Chiral classes names are optional, with tetrahedral being the default, hence “@” is an abbreviation (short form) of “@TH1” and “@@” short for “@TH2”.
- In this work, this chiral class is stored in RDKit’s ChiralType enumeration, and the permutation is stored in an unsigned integer property called common_properties::_chiralPermutation.

SP-4 EXAMPLES

- Let's start with a classic example:



CHEBI27899
cisplatin
(SP-4-2)-diamminedichloroplatinum
Cl[Pt@SP1](Cl)([NH3])[NH3]



CHEBI35852
transplatin
(SP-4-1)-diamminedichloroplatinum
Cl[Pt@SP2](Cl)([NH3])[NH3]

WORKED EXAMPLE(S)

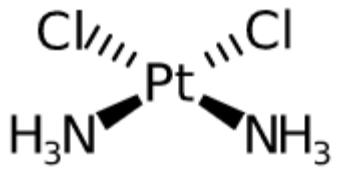
P[M@SP1](Q)(R)S means P is across from R, Q across from S. “U”

P[M@SP2](Q)(R)S means P is across from Q, R across from S. “4”

P[M@SP3](Q)(R)S means P is across from S, Q across from R. “Z”

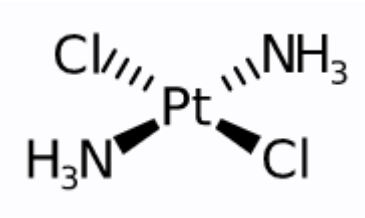
Cl[Pt@SP1](Cl)([NH3])[NH3]

Cl[Pt@SP3](Cl)([NH3])[NH3]



cisplatin

Cl[Pt@SP2](Cl)([NH3])[NH3]



transplatin

See the excellent documentation at http://opensmiles.org

THE IMPORTANCE OF LINEAR PAIRS

- A useful conceptual abstraction are pairs of neighbours that are linearly across from one another.
- The number of pairs present is a defining characteristic of a non-tetrahedral chirality class.
- The permutation index encodes the identities of these pairs and (typically) the parity/handedness.
  - Permutation = Pairs + Parity.
- These pairs (and parity) need to be preserved when interconverting between 3D, 2D and 1D [i.e. SMILES].

PROPOSED "HELPER" API

- To simplify working with permutation numbers, we propose the following set of "helper" functions:

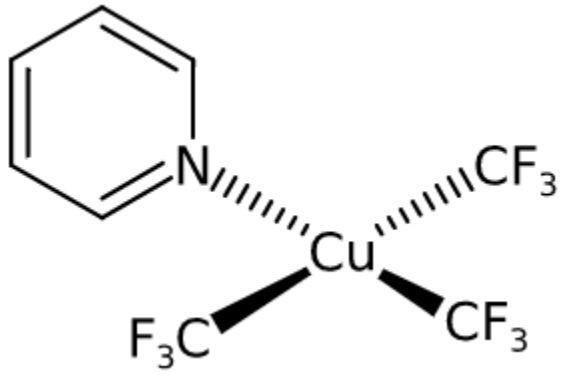
```
Bond *getChiralAcrossBond(Atom *cen, Bond *qry);
Bond *getChiralAcrossBond(Atom *cen, Atom *qry);
Atom *getChiralAcrossAtom(Atom *cen, Bond *qry);
Atom *getChiralAcrossAtom(Atom*cen, Atom *qry);
```

3D DISTANCE GEOMETRY/FORCEFIELDS

```
// Return ideal bond angle in degrees
double GetIdealAngle(Atom *cen, Atom *nbr1, Atom *nbr2) {
    chiralType tag = cen->getChiralTag();
    if (tag == ChiralType::CHI_OCTAHEDRAL ||
        tag == ChiralType::CHI_SQUAREPLANAR)
            return getChiralAcrossAtom(cen,nbr1) == nbr2 ? 180 : 90;
    if (tag == CHI_TRIGONALBIPYRAMIDAL) {
            if (getChiralAcrossAtom(cen,nbr1) == nbr2)
                    return 180;
            if (getChiralAcrossAtom(cen,nbr1) != 0 ||
                getChiralAcrossAtom(cen,nbr2) != 0)
                    return 90;
            return 120;
    }
```

SP-4 EXAMPLES

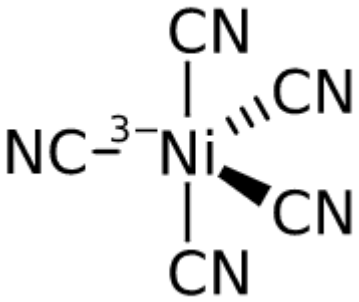
- It's useful to annotate symmetric centres as @SP.



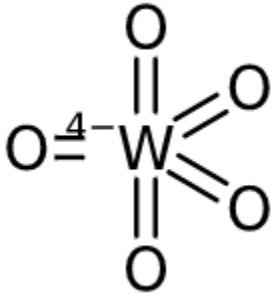
COD4128444
(SP-4)-(pyridine)tris(trifluoromethyl)copper
[Cu@SP1]([N]1=CC=CC=C1)(C(F)(F)F)(C(F)(F)F)C(F)(F)F

TBPY-5 EXAMPLES

- It's useful to annotate symmetric centres as @TB.



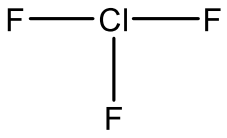
CHEBI30370
(TBPY-5)-pentacyanonickelate(3-)
N#C[Ni@TB1-3](C#N)(C#N)(C#N)C#N



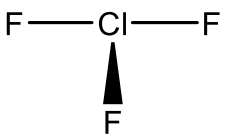
CHEBI30370
(TBPY-5)-pentaoxotungstate(4-)
O=[W@TB1-4](=O)(=O)(=O)=O

PERCEPTION HEURISTICS

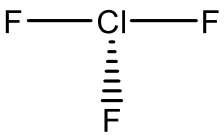
- Limit non-tetrahedral stereochemistry to elements with atomic numbers 15 or above.
- Non-tetrahedral stereochemistry requires at least one pair of bonds to be (close to) linear.
- From 2D, stereochemistry is only perceived on atoms with explicit wedge/hash bonds*.



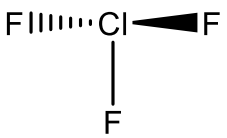
Unspecified



T-Shape



T-Shape

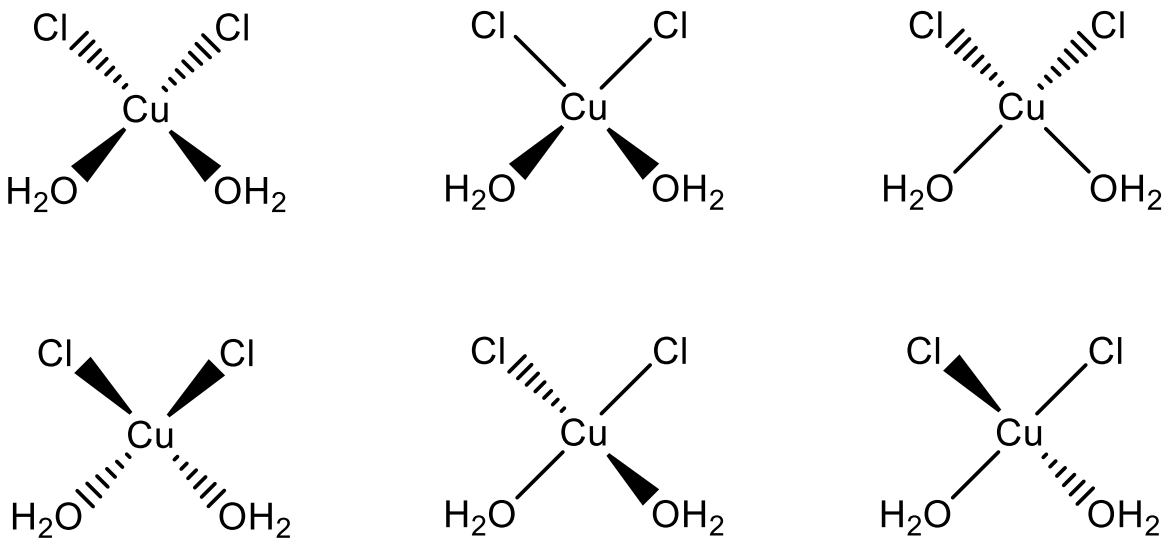


T-Shape

* This conflicts with ST-2.3 of IUPAC's Graphical Representation of Stereochemical Configuration 2006

SQUARE PLANAR PERCEPTION

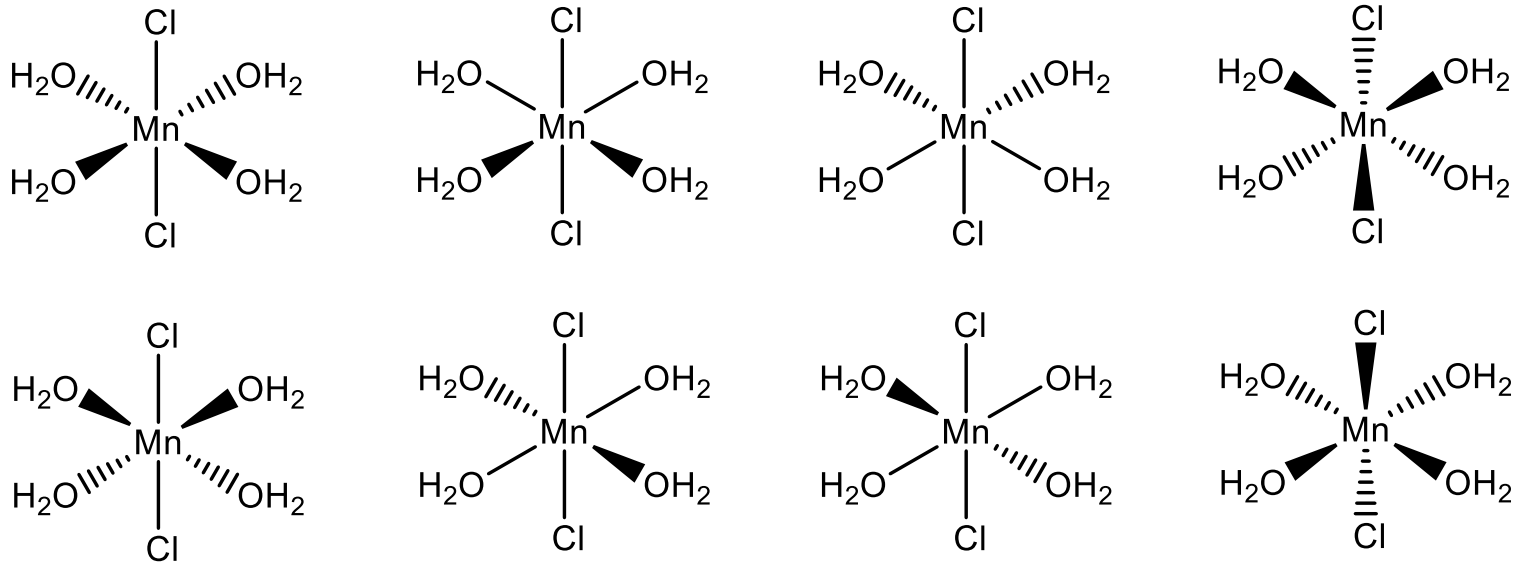
- Square planar centres require two pairs of linear bonds, in any of the following configurations.



* This conflicts with ST-2.6 of IUPAC's Graphical Representation of Stereochemical Configuration 2006

OCTAHEDRAL PERCEPTION

- Octahedral centres require three pairs of linear bonds, in any of the following configurations.



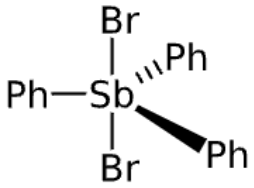
* This conflicts with ST-2.11 of IUPAC's Graphical Representation of Stereochemical Configuration 2006

WHERE AND WHY DO WE DIFFER?

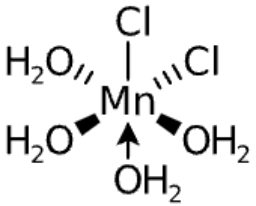
- All “preferred” IUPAC configurations are understood.
- These rules are affine (i.e. rotationally) invariant.
- The mirror image of the 2D depiction represents the mirror image of the 3D structure.
- This typically provides interpretation for IUPAC’s “unacceptable” sketch representations, for which no semantic interpretation is given.
- Existing 2D depictions retain their meaning, and non-tetrahedral geometry is only interpreted in cases where tetrahedral chemistry would be invalid.

TRADITIONAL NAMING

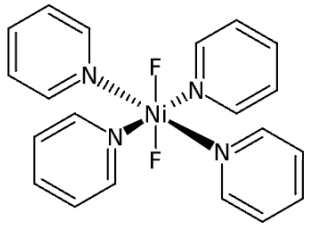
- Many non-tetrahedral centres may be named using cis/trans.



COD2007221
trans-dibromidotriphenyl antimony

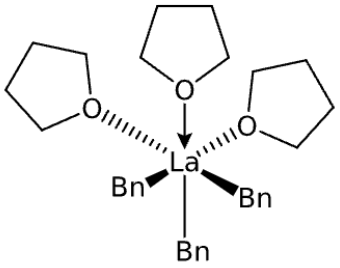


COD8103034
cis-tetraaquadichlorido manganese

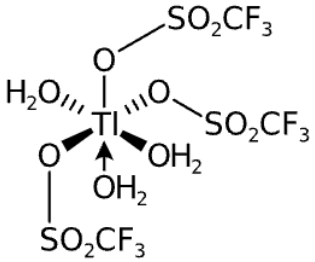


COD7035539
trans-diflurotetrakis (pyridine)nickel

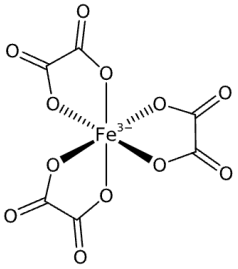
- Others using fac-, mer-, Delta- (Δ-) and Lambda (Λ-).



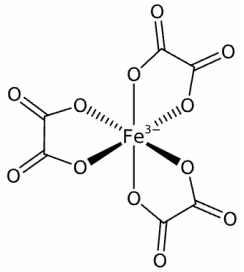
COD4075390
fac-tribenzyltris(tetrahydrofuran) lanthanum



COD7011937
mer-triaquatris(triflato-κO) thallium



Δ-tris(oxalate) ferrate(3-)



Λ-tris(oxalate) ferrate(3-)

IUPAC NAMING

- IUPAC's blue book contains recommendations for stereochemical prefixes based on CIP priority rules.
- See "Algorithmic Analysis of Cahn-Ingold-Prelog Rules of Stereochemistry: Proposals for Revised Rules and a Guide for Machine Implementation", JCIM 2018.

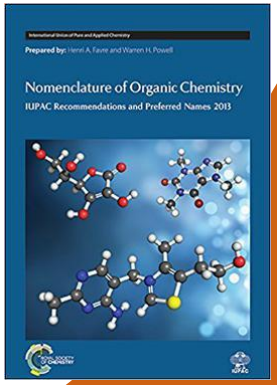
(OC-6-12)-dichlorobis(4-amino-1,2,4-triazole-κN1)dimethyltin

(TBPY-5-12)-(tricyclohexylphosphine)tetracarbonyliron

(OC-6-43)-bis(triphenylphosphine)dichloridohydridooxidorhenium

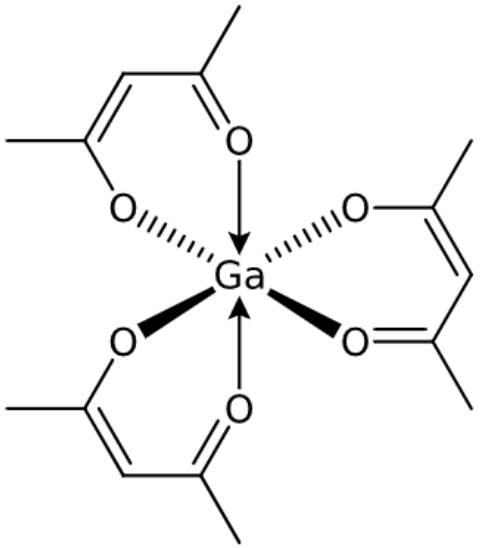
(SP-4-3)-(tricyclohexylphosphine)carbonylchloridoiodidopalladium

(SPY-5-35-C)-5-phenyl-1-oxa-thia-5λ^5-phosphaspiro[4.4]nonane



PUSHING THE LIMITS...

- Unfortunately, CIP's priority rules can't (yet) handle resonant ligands in difficult cases.



- COD1548616 Δ-tris(acetoacetato)gallium

SMARTS MATCHING

- [*@OH] can be used to identify octahedral centers.
- *[@OH25]* can be used to identify “across” atoms.

- SMARTS matchers traditionally loop over all unvisited neighbors to extend the match by the next atom.
- At non-tetrahedral stereo centers, the matcher can efficiently use the proposed getChiralAcrossAtom.
- Like regular tetrahedral matching, once three axes (neighbours) have be identified, the parity can be checked.

HOW TETRAHEDRAL STEREO WORKS #2

- RDKit stores an Atom::ChiralType on each atom:
  - CHI_TETRAHEDRAL_CW, CHI_TETRAHEDRAL_CCW, CHI_UNSPECIFIED and CHI_OTHER.
- This records the “parity” of the neighbors around a centre in getAtomBondsOrder.
- When querying a centre, RDKit counts the number of swaps used to permute the query (controlling) order into the reference order.
- As each swap inverts the “parity”, the result is based on whether the swap count is even or odd.

GENERALIZING TO NON-TETRAHEDRAL

- The current functionality (returning nswaps)

```
int Atom::getPerturbationOrder(const INT_LIST &probe);
```

can be generalized to return the permutation index.

```
unsigned int getChiralPermutation(const Atom *cen,
                                  const  INT_LIST &probe);
```

- Internally tetrahedral can continue to count swaps, but other chirality classes can use alternate (table) based implementations.

SQUARE PLANAR SWAP TABLE

- The following tables encode how exchanging neighbors updates the @TH and @SP permutations.

Exchange	@TH1	@TH2
1 ↔ 2	2	1
1 ↔ 3	2	1
1 ↔ 4	2	1
2 ↔ 3	2	1
2 ↔ 4	2	1
3 ↔ 4	2	1

Exchange	@SP1	@SP2	@SP3
1 ↔ 2	3	2	1
1 ↔ 3	1	3	2
1 ↔ 4	2	1	3
2 ↔ 3	2	1	3
2 ↔ 4	1	3	2
3 ↔ 4	3	2	1

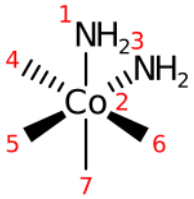
- Tracking the swaps during the sort gives the result.

THREE CLEVER TRICKS

1. When deleting atoms or suppressing hydrogens the permutation can be updated as though the deleted neighbor has been moved to the end. This is the representation used to support “partial” chirality classes, such as T-shape and See-saw.
2. The table can be shrunk even further to only encode swaps with the first position, as for any pair X ↔ Y is equivalent to 1 ↔ X; 1 ↔ Y; 1 ↔ X.
3. These tables are also useful for identifying the canonical (lowest) permutation from symmetries.

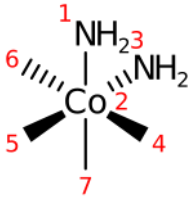
CANONICAL CONFIGURATION

Canonical configurations are more tricky than **tetrahedral** due to symmetry. These all mean the same**:**

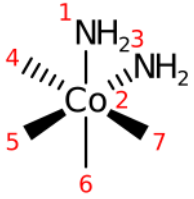


`N[Co@OH1](N)(C)(C)(C)C`

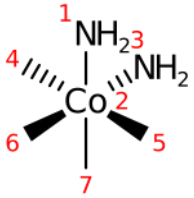
Notice symmetric atoms are moving around in layout for different config



`N[Co@OH2](N)(C)(C)(C)C`



`N[Co@OH3](N)(C)(C)(C)C`

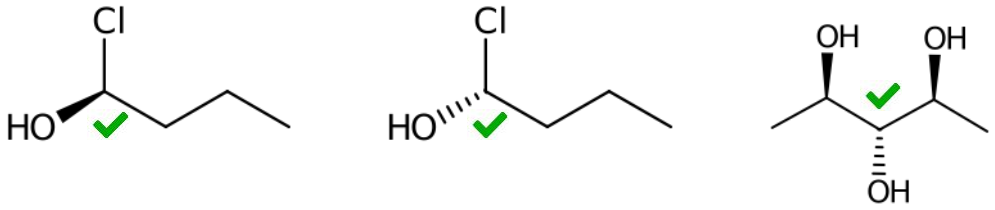


`N[Co@OH4](N)(C)(C)(C)C`

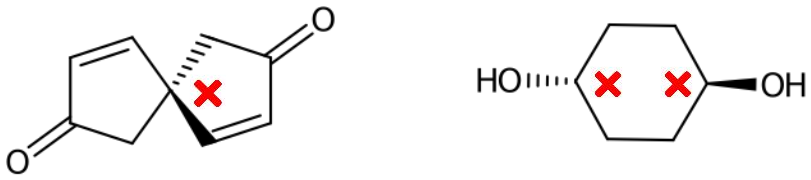
*Select the **lowest number** one as canonical, how?*

CANONICAL CONFIGURATION

~98% tetrahedral's are asymmetric[1] and easily handled in the partition refinement using the ranking to split ties



Comprehensive solution is more complex but the solution helps us with octahedral



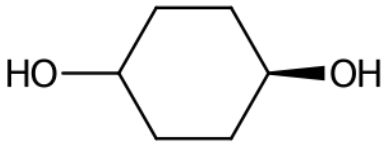
It involves inspecting the automorphisms (self-match) during canonical labelling

1. John Mayfield *et al.* Comparing Cahn-Ingold-Prelog rule implementations: The need for an open CIP. ***ACS Fall 2017***, Washington, D.C. https://nextmovesoftware.com/talks/Mayfield_ChanIngoldPrelog_ACS_201708.pdf

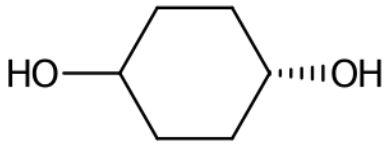
CANONICAL CONFIGURATION

Tetrahedral specification is **redundant** if the mirror image is super imposable

Flip and check if we match ourselves (automorphism):



OC1CC**[C@H]**(O)CC1

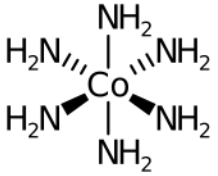


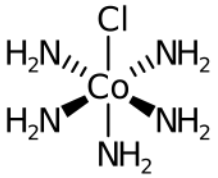
OC1CC**[C@@H]**(O)CC1

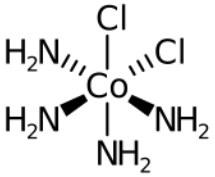
```
>>> Chem.MolFromSmiles('OC1CC[C@H](O)CC1')
        .HasSubstructMatch(Chem.MolFromSmiles('OC1CC[C@@H](O)CC1'))
True # therefore this atom is non-stereogenic
>>> Chem.MolFromSmiles('O[C@H]1CC[C@H](O)CC1')
        .HasSubstructMatch(Chem.MolFromSmiles('O[C@H]1CC[C@@H](O)CC1'))
False # therefore this atom is stereogenic
```

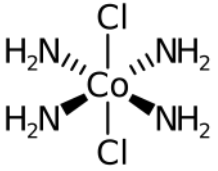
CANONICAL CONFIGURATION

Octahedral specification is **redundant** if all possible configurations @OH1…@OH30 are automorphic

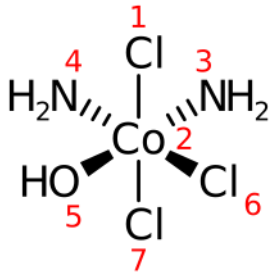
`N[Co@OH1](N)(N)(N)(N)N`  ✗

 `Cl[Co@OH1](N)(N)(N)(N)N` ✗

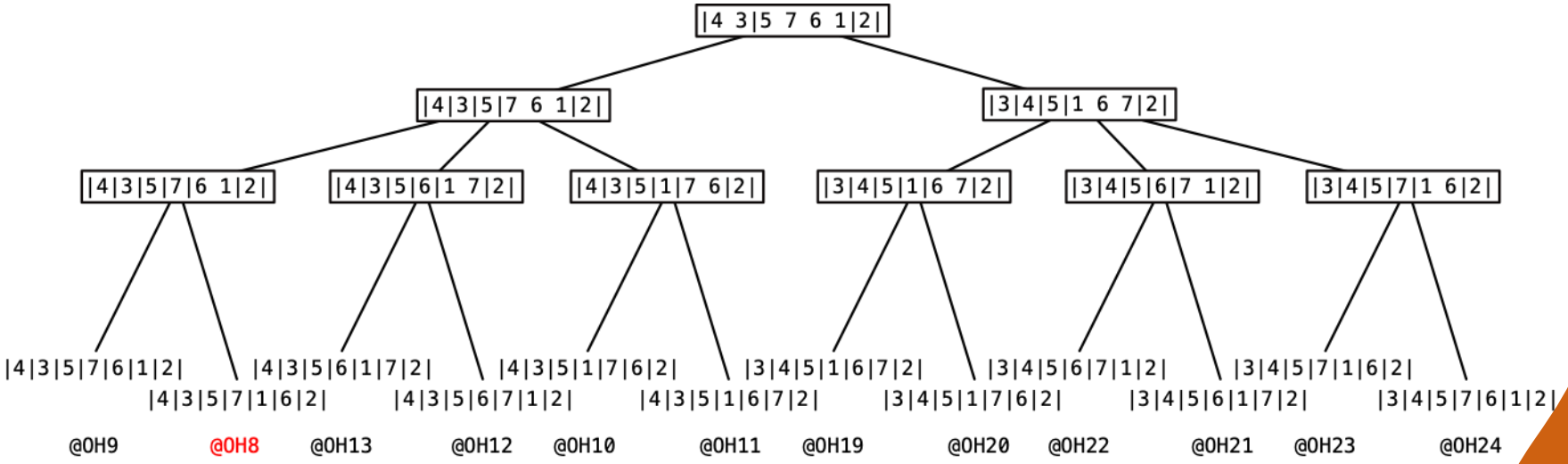
`Cl[Co@OH1](Cl)(N)(N)(N)N`  ✓ *cis-*

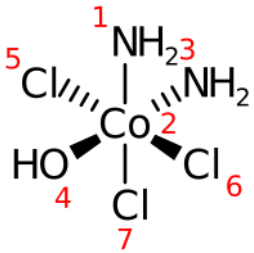
 `Cl[Co@OH25](Cl)(N)(N)(N)N` ✓ *trans-*

CANONICAL CONFIGURATION




```
Cl[Co@](N)(N)(O)(Cl)Cl
(input non-cansmi)
```





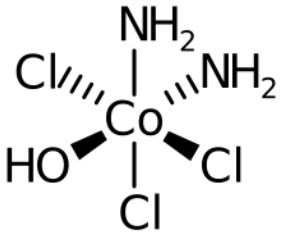
Maintain the set of configs we see, and select the lowest - **@OH8**

```
N[Co@OH8](N)(O)(Cl)(Cl)Cl
```

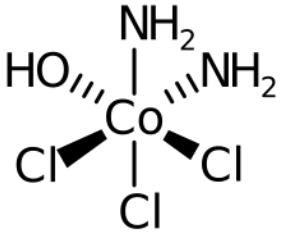
CANONICAL CONFIGURATION

```
Cl[Co@OH1](N)(N)(O)(Cl)Cl => N[Co@OH8](N)(O)(Cl)(Cl)Cl
Cl[Co@OH2](N)(N)(O)(Cl)Cl => N[Co@OH8](N)(O)(Cl)(Cl)Cl
Cl[Co@OH3](N)(N)(O)(Cl)Cl => N[Co@OH8](N)(O)(Cl)(Cl)Cl
Cl[Co@OH4](N)(N)(O)(Cl)Cl => N[Co@OH8](N)(O)(Cl)(Cl)Cl
Cl[Co@OH5](N)(N)(O)(Cl)Cl => N[Co@OH8](N)(O)(Cl)(Cl)Cl
Cl[Co@OH6](N)(N)(O)(Cl)Cl => N[Co@](N)(O)(Cl)(Cl)Cl
Cl[Co@OH7](N)(N)(O)(Cl)Cl => N[Co@](N)(O)(Cl)(Cl)Cl
Cl[Co@OH8](N)(N)(O)(Cl)Cl => N[Co@OH25](N)(O)(Cl)(Cl)Cl
Cl[Co@OH9](N)(N)(O)(Cl)Cl => N[Co@OH25](N)(O)(Cl)(Cl)Cl
Cl[Co@OH10](N)(N)(O)(Cl)Cl => N[Co@OH25](N)(O)(Cl)(Cl)Cl
Cl[Co@OH11](N)(N)(O)(Cl)Cl => N[Co@OH25](N)(O)(Cl)(Cl)Cl
Cl[Co@OH12](N)(N)(O)(Cl)Cl => N[Co@OH25](N)(O)(Cl)(Cl)Cl
Cl[Co@OH13](N)(N)(O)(Cl)Cl => N[Co@OH25](N)(O)(Cl)(Cl)Cl
Cl[Co@OH14](N)(N)(O)(Cl)Cl => N[Co@OH8](N)(O)(Cl)(Cl)Cl
Cl[Co@OH15](N)(N)(O)(Cl)Cl => N[Co@OH8](N)(O)(Cl)(Cl)Cl
Cl[Co@OH16](N)(N)(O)(Cl)Cl => N[Co@OH8](N)(O)(Cl)(Cl)Cl
Cl[Co@OH17](N)(N)(O)(Cl)Cl => N[Co@](N)(O)(Cl)(Cl)Cl
Cl[Co@OH18](N)(N)(O)(Cl)Cl => N[Co@](N)(O)(Cl)(Cl)Cl
Cl[Co@OH19](N)(N)(O)(Cl)Cl => N[Co@](N)(O)(Cl)(Cl)Cl
Cl[Co@OH20](N)(N)(O)(Cl)Cl => N[Co@](N)(O)(Cl)(Cl)Cl
Cl[Co@OH21](N)(N)(O)(Cl)Cl => N[Co@OH8](N)(O)(Cl)(Cl)Cl
Cl[Co@OH22](N)(N)(O)(Cl)Cl => N[Co@OH8](N)(O)(Cl)(Cl)Cl
Cl[Co@OH23](N)(N)(O)(Cl)Cl => N[Co@](N)(O)(Cl)(Cl)Cl
Cl[Co@OH24](N)(N)(O)(Cl)Cl => N[Co@](N)(O)(Cl)(Cl)Cl
Cl[Co@OH25](N)(N)(O)(Cl)Cl => N[Co@](N)(O)(Cl)(Cl)Cl
Cl[Co@OH26](N)(N)(O)(Cl)Cl => N[Co@](N)(O)(Cl)(Cl)Cl
Cl[Co@OH27](N)(N)(O)(Cl)Cl => N[Co@OH8](N)(O)(Cl)(Cl)Cl
Cl[Co@OH28](N)(N)(O)(Cl)Cl => N[Co@OH8](N)(O)(Cl)(Cl)Cl
Cl[Co@OH29](N)(N)(O)(Cl)Cl => N[Co@](N)(O)(Cl)(Cl)Cl
Cl[Co@OH30](N)(N)(O)(Cl)Cl => N[Co@](N)(O)(Cl)(Cl)Cl
```

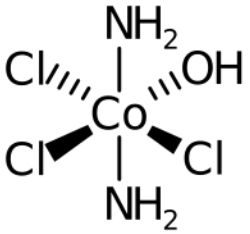
3 configurations



@OH8



@OH1

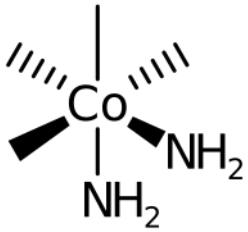


@OH25

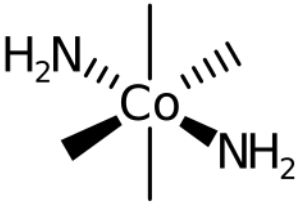
CANONICAL CONFIGURATION

```
N[Co@OH1](N)(C)(C)(C)C => C[Co@](C)(C)(C)(N)N
N[Co@OH2](N)(C)(C)(C)C => C[Co@](C)(C)(C)(N)N
N[Co@OH3](N)(C)(C)(C)C => C[Co@](C)(C)(C)(N)N
N[Co@OH4](N)(C)(C)(C)C => C[Co@](C)(C)(C)(N)N
N[Co@OH5](N)(C)(C)(C)C => C[Co@](C)(C)(C)(N)N
N[Co@OH6](N)(C)(C)(C)C => C[Co@](C)(C)(C)(N)N
N[Co@OH7](N)(C)(C)(C)C => C[Co@](C)(C)(C)(N)N
N[Co@OH8](N)(C)(C)(C)C => C[Co@](C)(C)(C)(N)N
N[Co@OH9](N)(C)(C)(C)C => C[Co@](C)(C)(C)(N)N
N[Co@OH10](N)(C)(C)(C)C => C[Co@](C)(C)(C)(N)N
N[Co@OH11](N)(C)(C)(C)C => C[Co@](C)(C)(C)(N)N
N[Co@OH12](N)(C)(C)(C)C => C[Co@](C)(C)(C)(N)N
N[Co@OH13](N)(C)(C)(C)C => C[Co@](C)(C)(C)(N)N
N[Co@OH14](N)(C)(C)(C)C => C[Co@](C)(C)(C)(N)N
N[Co@OH15](N)(C)(C)(C)C => C[Co@](C)(C)(C)(N)N
N[Co@OH16](N)(C)(C)(C)C => C[Co@](C)(C)(C)(N)N
N[Co@OH17](N)(C)(C)(C)C => C[Co@](C)(C)(C)(N)N
N[Co@OH18](N)(C)(C)(C)C => C[Co@](C)(C)(C)(N)N
N[Co@OH19](N)(C)(C)(C)C => C[Co@](C)(C)(C)(N)N
N[Co@OH20](N)(C)(C)(C)C => C[Co@](C)(C)(C)(N)N
N[Co@OH21](N)(C)(C)(C)C => C[Co@](C)(C)(C)(N)N
N[Co@OH22](N)(C)(C)(C)C => C[Co@](C)(C)(C)(N)N
N[Co@OH23](N)(C)(C)(C)C => C[Co@](C)(C)(C)(N)N
N[Co@OH24](N)(C)(C)(C)C => C[Co@](C)(C)(C)(N)N
N[Co@OH25](N)(C)(C)(C)C => C[Co@OH12](C)(C)(C)(N)N
N[Co@OH26](N)(C)(C)(C)C => C[Co@OH12](C)(C)(C)(N)N
N[Co@OH27](N)(C)(C)(C)C => C[Co@OH12](C)(C)(C)(N)N
N[Co@OH28](N)(C)(C)(C)C => C[Co@OH12](C)(C)(C)(N)N
N[Co@OH29](N)(C)(C)(C)C => C[Co@OH12](C)(C)(C)(N)N
N[Co@OH30](N)(C)(C)(C)C => C[Co@OH12](C)(C)(C)(N)N
```

2 configurations, cis/trans-



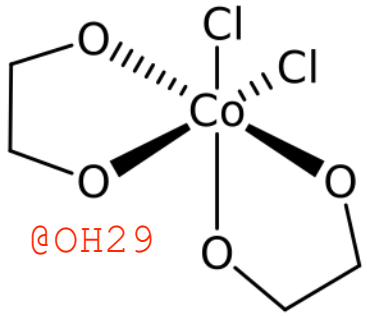
@OH1



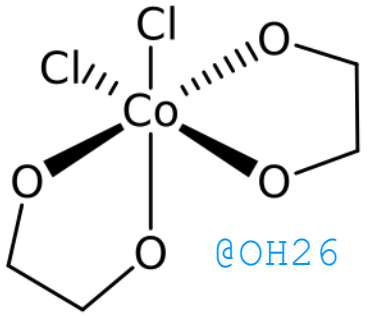
@OH12

CANONICAL CONFIGURATION

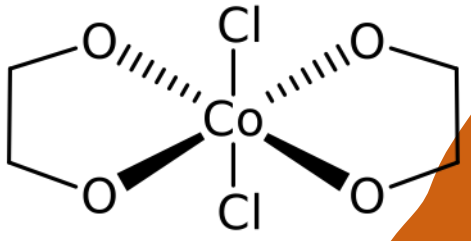
```
Cl[Co@OH1]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH29]12(OCCO2)(OCCO1)Cl
Cl[Co@OH2]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH26]12(OCCO2)(OCCO1)Cl
Cl[Co@OH3]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH28]12(OCCO2)(OCCO1)Cl
Cl[Co@OH4]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH28]12(OCCO2)(OCCO1)Cl
Cl[Co@OH5]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH26]12(OCCO2)(OCCO1)Cl
Cl[Co@OH6]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH14]12(OCCO2)(OCCO1)Cl
Cl[Co@OH7]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH8]12(OCCO2)(OCCO1)Cl
Cl[Co@OH8]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH26]12(OCCO2)(OCCO1)Cl
Cl[Co@OH9]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH29]12(OCCO2)(OCCO1)Cl
Cl[Co@OH10]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH29]12(OCCO2)(OCCO1)Cl
Cl[Co@OH11]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH26]12(OCCO2)(OCCO1)Cl
Cl[Co@OH12]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH8]12(OCCO2)(OCCO1)Cl
Cl[Co@OH13]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH8]12(OCCO2)(OCCO1)Cl
Cl[Co@OH14]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH28]12(OCCO2)(OCCO1)Cl
Cl[Co@OH15]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH29]12(OCCO2)(OCCO1)Cl
Cl[Co@OH16]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH28]12(OCCO2)(OCCO1)Cl
Cl[Co@OH17]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH8]12(OCCO2)(OCCO1)Cl
Cl[Co@OH18]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH14]12(OCCO2)(OCCO1)Cl
Cl[Co@OH19]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH28]12(OCCO2)(OCCO1)Cl
Cl[Co@OH20]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH26]12(OCCO2)(OCCO1)Cl
Cl[Co@OH21]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH26]12(OCCO2)(OCCO1)Cl
Cl[Co@OH22]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH29]12(OCCO2)(OCCO1)Cl
Cl[Co@OH23]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH29]12(OCCO2)(OCCO1)Cl
Cl[Co@OH24]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH28]12(OCCO2)(OCCO1)Cl
Cl[Co@OH25]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH26]12(OCCO2)(OCCO1)Cl
Cl[Co@OH26]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH28]12(OCCO2)(OCCO1)Cl
Cl[Co@OH27]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH29]12(OCCO2)(OCCO1)Cl
Cl[Co@OH28]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH26]12(OCCO2)(OCCO1)Cl
Cl[Co@OH29]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH28]12(OCCO2)(OCCO1)Cl
Cl[Co@OH30]12(Cl)(OCCO1)OCCO2 => Cl[Co@OH29]12(OCCO2)(OCCO1)Cl
```



@OH29



@OH26



@OH8

@OH14/@OH28
hindered

ENHANCED NON-TETRAHEDRAL STEREO

A similar idea can be applied to handle Relative/Racemic stereo groups (enhanced stereo) as well as removing them from achiral molecules

Handling stereochemistry and configuration like this allows robust handling of complex cases

Stereo/configuration can be efficiently stored as a packed 32-bit int on each atom (and bond):

```
struct StereoConfig {
  unsigned char index;     // @,@@,@3,@4,@5
  unsigned char type;      // @TH,@SP,@OH
  unsigned char grp_type; // 0=abs,1=rac,2=rel
  unsigned char grp_num;   // rac1,rac2,rel1,etc
};
```

PROPERTY CALCULATION

- Polar surface area and volume differ between cis-platin and trans-platin, affecting LogP and other properties.
- Peter Ertl's TPSA method could be updated to handle organometallic structures (and cis/trans differences).
- Hypothesis: Across atoms have no/less influence.
- This suggests alternate definitions of radius for path-based, topological torsion and Morgan-like radial fingerprints for inorganic 2D chemical similarity.

SUMMARY & CONCLUSIONS

- Some progress is being made in representation of organometallic and inorganic compounds in cheminformatics.

ACKNOWLEDGEMENTS

- John Mayfield, NextMove Software
- Greg Landrum, ETH Zurich/T5 informatics

- David Weininger
- Daniel Lowe
- Ian Bruno
- Alex Clark
- Colin Batchelor
- Hinnerk Rey
- Jonathan Brecher
- Jonathan Goodman
- Evan Bolton
- Matt Swain