Programming the Science of Crystallography

PLATON, a Multipurpose Crystallographic Tool

Ton Spek, Utrecht University

Programming Languages

- Current choices are Fortran-(xx), C(++) or one of the many scripting languages (e.g. Python).
- My choice for scientific software over the last 30 years was and still is Fortran.

  I have seen many (scripting) languages come and go … algol, pascal, ratfor … and changed only once …
- I might consider a conversion to C++ after my official retirement in 2009 (assuming that C++ is still mainstream by that time and not superseded by Fortran2xxx …..)

Pro's and Con's of Fortran

- Fortran Pro's:

  - Designed for scientific computing, readily available and still evolving to include additional useful constructs.

  - Relatively easy to learn and port to other platforms.
- Fortran Con's:

  - No longer mainstream in the current software development community.

  - Interface to C libraries (e.g. Xlib) needed for graphics functionality.

PLATON AS AN EXAMPLE

- PLATON is focused mainly on small-molecule applications.
- The development of PLATON is essentially evolutionary, science driven and based on the needs of a national single crystal structure facility.
- Following is an overview of the IDEAS and TOOLS that have been implemented over the past 25 years in the program suite PLATON.

PLATON IMPLEMENTATION

- The development of PLATON started on various CDC mainframe platforms and migrated via VAX/VMS and DEC-UNIX to the PC/LINUX platform.
- Implementations are also available for MS-WINDOWS (thanks to Louis Farrugia, Glasgow) and Mac-OSX.
- PLATON tries to be compatible and complementing to the SHELX software suite.
- PLATON is currently used as the major structure validation engine in the IUCr CHECKCIF facility.

PLATON ORGANISATION

- Single FORTRAN source + a small C routine as an interface to X11 graphics.
- Separate group of routines for the handling of the Space Group Symmetry.
- Separate group of routines for handling the Graphics (X11/PostScript/HPGL).
- Separate group of reusable global routines (SORT, INVERT, etc.)

Input Files

Input files are:

1. A parameter/coordinate file of type *res, cif, fdat, spf.* The file type is guessed from the content, not from the file extension.
2. A reflection file of type *hkl* or *fcf.*
3. Command line input for instructions.

Output Files

- A full listing file (.lis).
- The PostScript version of .lis (.lps) for printing on a laserprinter or viewing with GhostScript.
- A summary listing on the console.
- Optionally a new parameter file
- Optionally a new reflection file
- Optionally a validation report (.chk, .fck)

Graphics Output

- Graphics output is implemented via calls to a single routine.
- This routine implements graphics instructions for the various types of graphics hardware.
- Currently, only X11, PostScript and HPGL are supported.
- In the past there was similar support for Tektronix etc.
- X11 library calls are implemented in a single C routine.
- The Windows version substitutes its own library calls.
- PLATON implements its own character set.

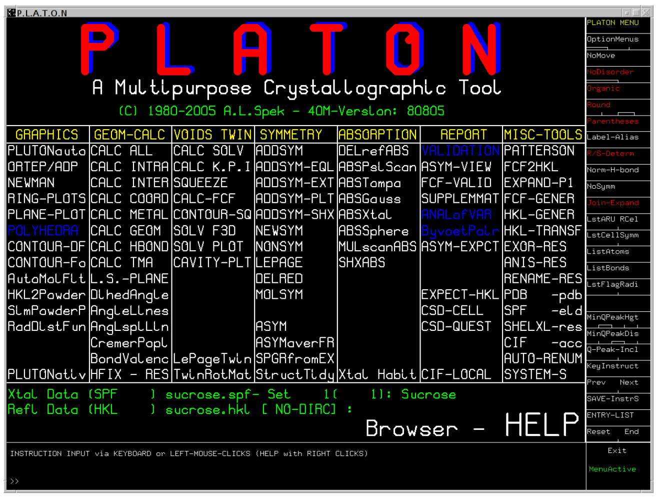
Features

- PLATON includes a number of unique tools such as ADDSYM, VOIDS, SQUEEZE, TwinRotMat, CIF, FCF Validation, BijvoetPairs, and SYSTEM-S.
- Provides a 'research framework' for the convenient implementation and testing of new ideas.
- Few outside dependencies (single source) (libX11 or equivalent for graphics).
- Non-standard language features are avoided.
- Up-to-Date HTML-HELP (via right mouse click on item) with a browser over the Internet or locally installable.

Entry points

- Via command line options allowing for use in scripts:

  e.g. 'platon –u shelxl.cif' will produce as the only output a file 'shelxl.chk' with a validation report.
- The clickable PLATON main menu gives an overview of the available functions.



```
Xtal Data (SPF      ) sucrose.spf- Set    1(     1): Sucrose
Refl Data (HKL      ) sucrose.hkl [ NO-DIRC] :
INSTRUCTION INPUT via KEYBOARD or LEFT-MOUSE-CLICKS (HELP with RIGHT CLICKS)
>>
```

Space Group Symmetry

- 230 Unique Space Groups, multiple settings, synonyms, specification.
- Explicit symmetry operator, H-M or Hall Symbol input
- Space Group Routine: Multiple callable functions:
  - Expansion of the set of symmetry generators
  - Explicit symmetry ➔ H-M and Hall Symbol
  - Symmetry operations on coordinates or reflection h,k,l
  - Multiplication of two supplied symmetry operators (R'|t') = (R1|t1)(R2|t2) → Network Analysis
  - Return inverted symmetry operation (including transl.)

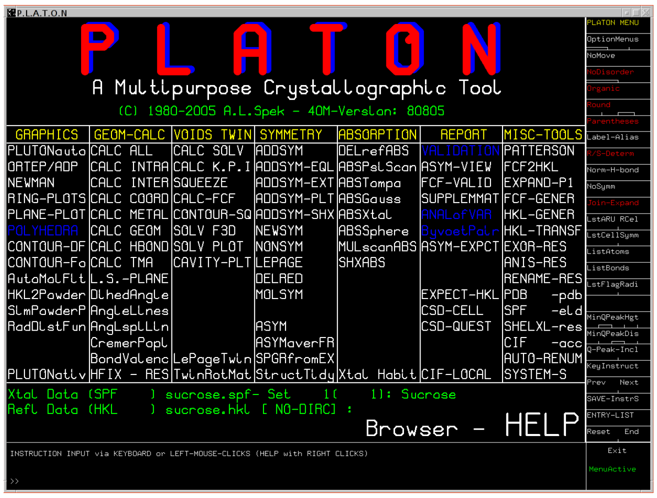
Geometry Analysis

- **Intra-molecular geometry**

  bonds,angles,torsions,rings,planes etc.
- **Inter-molecular geometry**

  Short contacts, H-bonds, networks
- **Coordination geometry**

Default: CALC ALL


P.L.A.T.O.N
P L A T O N
A Multipurpose Crystallographic Tool
(C) 1980-2005 A.L.Spek - 40M-Version: 80805
GRAPHICS | GEOM-CALC | VOIDS TWIN | SYMMETRY | ABSORPTION | REPORT | MISC-TOOLS
PLUTONauto | CALC ALL | CALC SOLV | ADDSYM | DELrefABS | VALIDATION | PATTERSON
ORTEP/ADP | CALC INTRA | CALC K.P.I | ADDSYM-EQL | ABSPsiScan | ASYM-VIEW | FCF2HKL
NEWMAN | CALC INTER | SQUEEZE | ADDSYM-EXT | ABSTompa | FCF-VALID | EXPAND-P1
RING-PLOTS | CALC COORD | CALC-FCF | ADDSYM-PLT | ABSGauss | SUPPLEMMAT | FCF-GENER
PLANE-PLOT | CALC METAL | CONTOUR-SQ | ADDSYM-SHX | ABSXtal | ANALofVAR | HKL-GENER
POLYHEDRA | CALC GEOM | SOLV F3D | NEWSYM | ABSSphere | BijvoetPair | HKL-TRANSF
CONTOUR-DF | CALC HBOND | SOLV PLOT | NONSYM | MULscanABS | ASYM-EXPCT | EXOR-RES
CONTOUR-Fo | CALC TMA | CAVITY-PLT | LEPAGE | SHXABS | | ANIS-RES
AutoMolFit | L.S.-PLANE | | DELRED | | | RENAME-RES
HKL2Powder | DihedAngle | | MOLSYM | | EXPECT-HKL | PDB -pdb
SimPowderP | AngleLines | | | | CSD-CELL | SPF -eld
RadDistFun | AngLspLin | | ASYM | | CSD-QUEST | SHELXL-res
| CremerPopl | | ASYMaverFR | | | CIF -acc
| BondValenc | LePageTwin | SPGRfromEX | | | AUTO-RENUM
PLUTONativ | HFIX - RES | TwinRotMat | StructTidy | Xtal Habit | CIF-LOCAL | SYSTEM-S
Browser - HELP
PLATON MENU
OptionMenus
NoMove
NoDisorder
Organic
Round
Parentheses
Label-Alias
R/S-Determ
Norm-H-bond
NoSymm
Join-Expand
LstARU RCel
LstCellSymm
ListAtoms
ListBonds
LstFlagRadi
MinQPeakHgt
MinQPeakDis
Q-Peak-Incl
KeyInstruct
Prev Next
SAVE-InstrS
ENTRY-LIST
Reset End
Exit
MenuActive

```
Xtal Data (SPF     ) sucrose.spf- Set     1(     1): Sucrose
Refl Data (HKL     ) sucrose.hkl [ NO-DIRC] :
INSTRUCTION INPUT via KEYBOARD or LEFT-MOUSE-CLICKS (HELP with RIGHT CLICKS)
>>
```

Derived Geometry and Standard Uncertainties

- Standard uncertainties for derived quantities *f* (p) can be derived in principle using the Least-Squares Covariance Matrix and the expression for the propagation of error:

  σ^2(*f*) = Σij (δ*f* / δp(i))(d*f* / δp(j)) cov (p(i),p(j))
- Or in case only variances are available:

  σ^2(*f*) = Σi (δ*f* / δp(i))^2 σ^2(p(i))
- Analytical Evaluation (clumsy for torsion angles and up)
- Numerical: approximate δ*f* / δp(i) ~ (*f*(p + Δi) – *f*(p)) / Δi

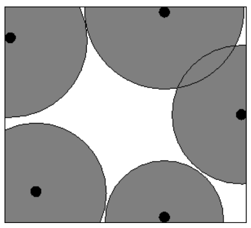
  Take: Δi = σ(p(i)), then

  σ^2(*f*) ~ Σi (*f*(p + σ(p(i)) – *f*(p)) ^2

Solvent Accessible Voids

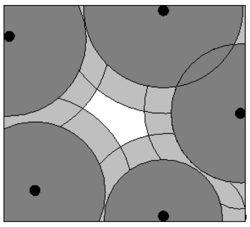
- A typical crystal structure has only 65% of the available space filled.
- The remainder volume is in voids (cusps) in-between atoms (too small to accommodate an H-atom)
- Solvent accessible voids can be defined as regions in the structure that can accommodate at least a sphere with radius 1.2 Angstrom without intersecting with any of the van der Waals spheres assigned to each atom in the structure.
- Algorithm: Graphical and Computational

DEFINE SOLVENT ACCESSIBLE VOID



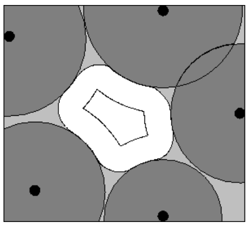
STEP #1 – EXCLUDE VOLUME INSIDE THE VAN DER WAALS SPHERE

DEFINE SOLVENT ACCESSIBLE VOID



STEP # 2 – EXCLUDE AN ACCESS RADIAL VOLUME TO FIND THE LOCATION OF ATOMS WITH THEIR CENTRE AT LEAST 1.2 ANGSTROM AWAY

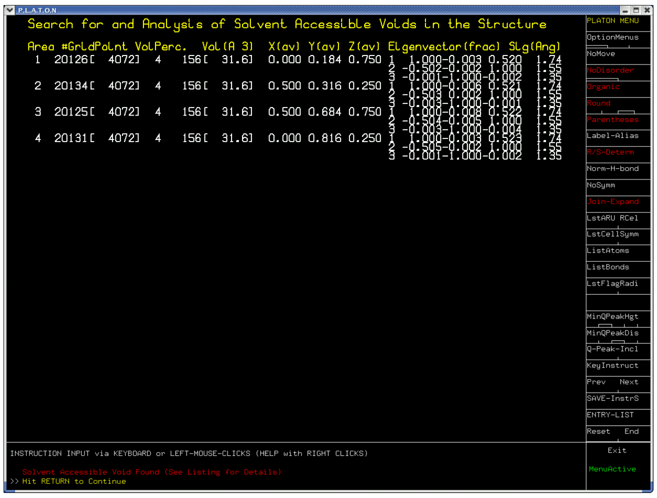
DEFINE SOLVENT ACCESSIBLE VOID



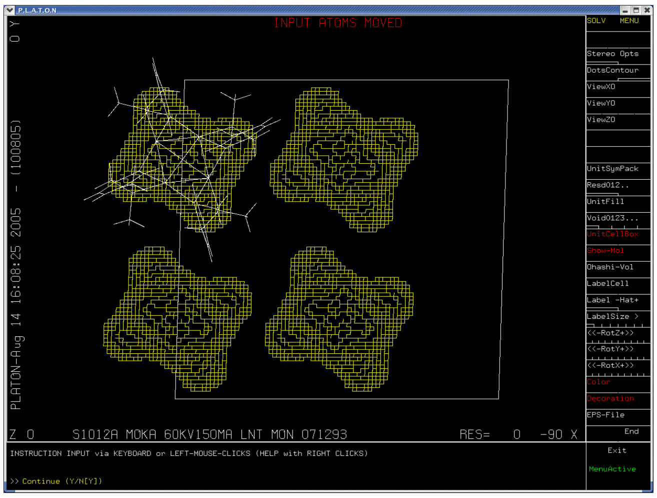
STEP # 3 – EXTEND INNER VOLUME WITH POINTS WITHIN 1.2 ANGSTROM FROM ITS OUTER BOUNDS

Voids: Algorithm

1. Expand the unitcell contents to P1
2. Define a 3D grid with gridstep ~ 0.2 Angstrom and with the number of gridpoints in each direction a multiple of 12 (for exact symmetry mapping)
3. Scan through all gridpoints in search of gridpoints that have a distance greater than the probe radius to the nearest van der Waals sphere.
4. Join gridpoints into connected sets (S).
5. Expand this set with gridpoints within the probe radius from the surface of S.



```
Search for and Analysis of Solvent Accessible Voids in the Structure

Area #GridPoint VolPerc.  Vol(A 3)  X(av) Y(av) Z(av) Eigenvector(frac) Sig(Ang)
  1  20126[  4072]  4   156[  31.6]  0.000 0.184 0.750 1  1.000-0.003 0.520  1.74
                                                        2 -0.502-0.002 1.000  1.55
                                                        3 -0.001-1.000-0.002  1.35
  2  20134[  4072]  4   156[  31.6]  0.500 0.316 0.250 1  1.000-0.006 0.521  1.74
                                                        2 -0.503 0.002 1.000  1.55
                                                        3 -0.003-1.000-0.001  1.35
  3  20125[  4072]  4   156[  31.6]  0.500 0.684 0.750 1  1.000-0.008 0.522  1.74
                                                        2 -0.504-0.005 1.000  1.55
                                                        3 -0.003-1.000-0.004  1.35
  4  20131[  4072]  4   156[  31.6]  0.000 0.816 0.250 1  1.000-0.003 0.523  1.74
                                                        2 -0.505-0.002 1.000  1.55
                                                        3 -0.001-1.000-0.002  1.35

INSTRUCTION INPUT via KEYBOARD or LEFT-MOUSE-CLICKS (HELP with RIGHT CLICKS)

   Solvent Accessible Void Found (See Listing for Details)
>> Hit RETURN to Continue
```



```
Z 0     S1012A MOKA 60KV150MA LNT MON 071293          RES=   0   -90 X
INSTRUCTION INPUT via KEYBOARD or LEFT-MOUSE-CLICKS (HELP with RIGHT CLICKS)

>> Continue (Y/N[Y])
```

VOID APPLICATIONS

- Calculation of Kitaigorodskii Packing Index
- As part of the SQUEEZE routine to handle the contribution of disordered solvents in crystal structure refinement
- Determination of the available space in solid state reactions (Ohashi)
- Determination of pore volumes, pore shapes and migration paths in microporous crystals

SQUEEZE

- Takes the contribution of disordered solvents to the calculated structure factors into account by back-Fourier transformation of density found in the ‘solvent accessible volume’ outside the ordered part of the structure.
- Filter: Input shelxl.res & shelxl.hkl

  Output: ‘solvent free’ shelxl.hkl
- Refine with SHELXL or Crystals

SQUEEZE Algorithm

1. Calculate difference map (FFT)
2. Use the VOID-map as a mask on the FFT-map to set all density outside the VOID’s to zero.
3. FFT-1 this masked Difference map -> contribution of the disordered solvent to the structure factors
4. Calculate an improved difference map with F(obs) phases based on F(calc) including the recovered solvent contribution and F(calc) without the solvent contribution.
5. Recycle to 2 until convergence.

Comment

- The Void-map can also be used to count the number of electrons in the masked volume.
- A complete dataset is required for this feature.
- Ideally, the solvent contribution is taken into account as a fixed contribution in the Structure Factor calculation (CRYSTALS) otherwise it is substracted temporarily from F(obs)^2 (SHELXL) and reinstated afterwards for the final Fo/Fc list.

(Pseudo)Merohedral Twinning

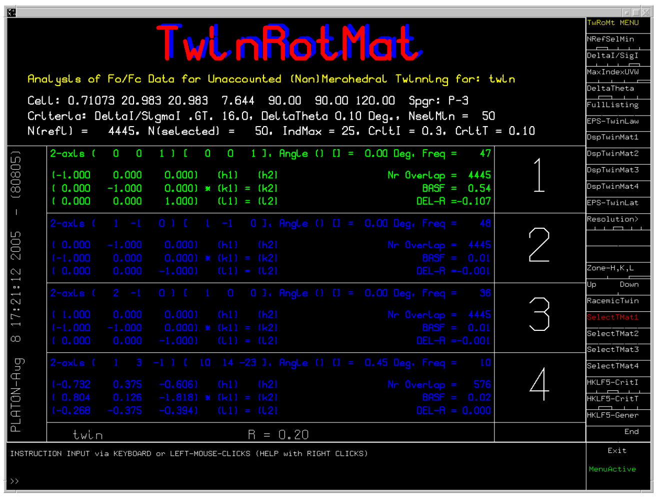
- Options to handle twinning in L.S. refinement available in SHELXL, CRYSTALS etc.
- Problem: Determination of the Twin Law that is in effect.
- Partial solution: coset decomposition, try all possibilities

  (I.e. all symmetry operations of the lattice but not of the structure)
- ROTAX (S.Parson et al. (2002) J. Appl. Cryst., 35, 168.

  (Based on the analysis of poorly fitting reflections of the type F(obs) >> F(calc))
- TwinRotMat Automatic Twinning Analysis as implemented in PLATON (Based on a similar analysis but implemented differently)

Example

- Structure refined to R= 20% in P-3
- Run TwinRotMat on CIF/FCF
- Result: Twinlaw with estimate of the twinning fraction and drop in R-value

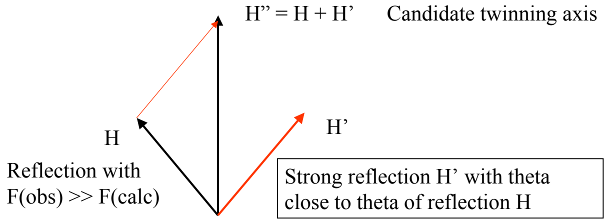


```
Analysis of Fo/Fc Data for Unaccounted (Non)Merohedral Twinning for: twin

Cell: 0.71073 20.983 20.983  7.644  90.00  90.00 120.00  Spgr: P-3
Criteria: DeltaI/SigmaI .GT. 16.0, DeltaTheta 0.10 Deg., NselMin =  50
N(refl) =   4445, N(selected) =   50, IndMax = 25, CritI = 0.3, CritT = 0.10

2-axis (   0   0   1 ) [   0   0   1 ], Angle () [] =  0.00 Deg, Freq =    47
(-1.000    0.000    0.000)   (h1)   (h2)                   Nr Overlap =  4445
( 0.000   -1.000    0.000) * (k1) = (k2)                         BASF =  0.54
( 0.000    0.000    1.000)   (l1) = (l2)                        DEL-R =-0.107

2-axis (   1  -1   0 ) [   1  -1   0 ], Angle () [] =  0.00 Deg, Freq =    48
( 0.000   -1.000    0.000)   (h1)   (h2)                   Nr Overlap =  4445
(-1.000    0.000    0.000) * (k1) = (k2)                         BASF =  0.01
( 0.000    0.000   -1.000)   (l1) = (l2)                        DEL-R =-0.001

2-axis (   2  -1   0 ) [   1   0   0 ], Angle () [] =  0.00 Deg, Freq =    36
( 1.000    0.000    0.000)   (h1)   (h2)                   Nr Overlap =  4445
(-1.000   -1.000    0.000) * (k1) = (k2)                         BASF =  0.01
( 0.000    0.000   -1.000)   (l1) = (l2)                        DEL-R =-0.001

2-axis (   1   3  -1 ) [  10  14 -23 ], Angle () [] =  0.45 Deg, Freq =    10
(-0.732    0.375   -0.606)   (h1)   (h2)                   Nr Overlap =   576
( 0.804    0.126   -1.818) * (k1) = (k2)                         BASF =  0.02
(-0.268   -0.375   -0.394)   (l1) = (l2)                        DEL-R = 0.000

    twin                        R = 0.20

INSTRUCTION INPUT via KEYBOARD or LEFT-MOUSE-CLICKS (HELP with RIGHT CLICKS)

>>
```

Ideas behind the Algorithm

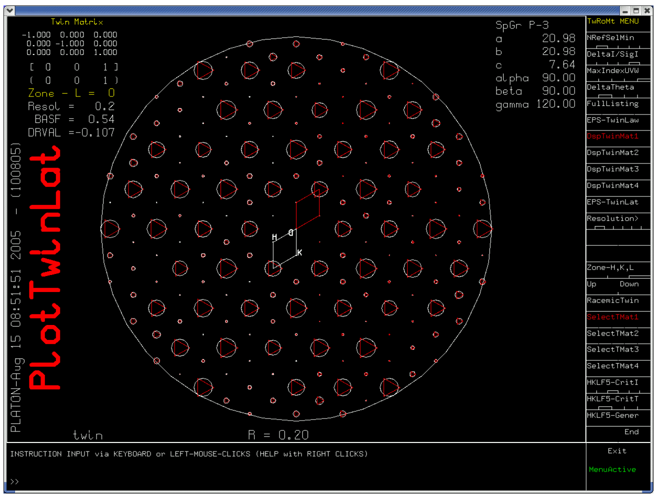
- Reflections effected by twinning show-up in the least-squares refinement with F(obs) >> F(calc)
- Overlapping reflections necessarily have the same theta within a tolerance.
- The more interesting cases of twinning in the current context are those with layers of overlapping reflections that can be described with a rotation about a reciprocal axis.

Possible Twin Axis



TwinRotMat Algorithm

- Select the set of reflections H with F(obs) >> F(calc)
- Loop over all reflections H' (including symmetry related ones) for which F(H') > F(H) and Θ(H') ~ Θ(H).
- Register H" = H + H' (reduced to co-prime integers) as a possible twinning axis (I.e. count the frequency of occurrence)
- Eliminate symmetry directions and H" that are related by symmetry.
- Determine the twinning factor that gives the lowest R-factor (simple gridsearch).



```
        Twin Matrix
 -1.000  0.000  0.000
  0.000 -1.000  0.000
  0.000  0.000  1.000
  [  0    0    1 ]
  (  0    0    1 )
  Zone - L =  0
  Resol =    0.2
   BASF =   0.54
  DRVAL =-0.107
```

```
SpGr P-3
a        20.98
b        20.98
c         7.64
alpha    90.00
beta     90.00
gamma   120.00
```

```
PLATON-Aug 15 08:51:51 2005 - (100805)
twin                         R = 0.20
INSTRUCTION INPUT via KEYBOARD or LEFT-MOUSE-CLICKS (HELP with RIGHT CLICKS)
>>
```

Special Implementations

Older programs with dated input.

StructureTidy (standardisation of Inorganic Structures). CIF interface generates the proper input in original input format.

Bond Valence Analysis

System S

- Automatic structure determination
  (Space group determination, solution, refinement, analysis)
- Build-in in PLATON (Unix only)
- Calls external programs including itself for various functions.
- Program runs in either *guided* or *no-questions-asked* mode

Concluding Remarks

- The ‘Single Source’ approach of PLATON makes it easy (for me) to implement new tools within the existing framework of already available tools.
- Only one program has to be maintained.
- A one-person project, so no internal discussions.
- Of-course, the above is controversial …

Thanks

- Thanks to the users for their:
- Complaints
- Bug reports (‘undocumented features ..)
- Suggestions