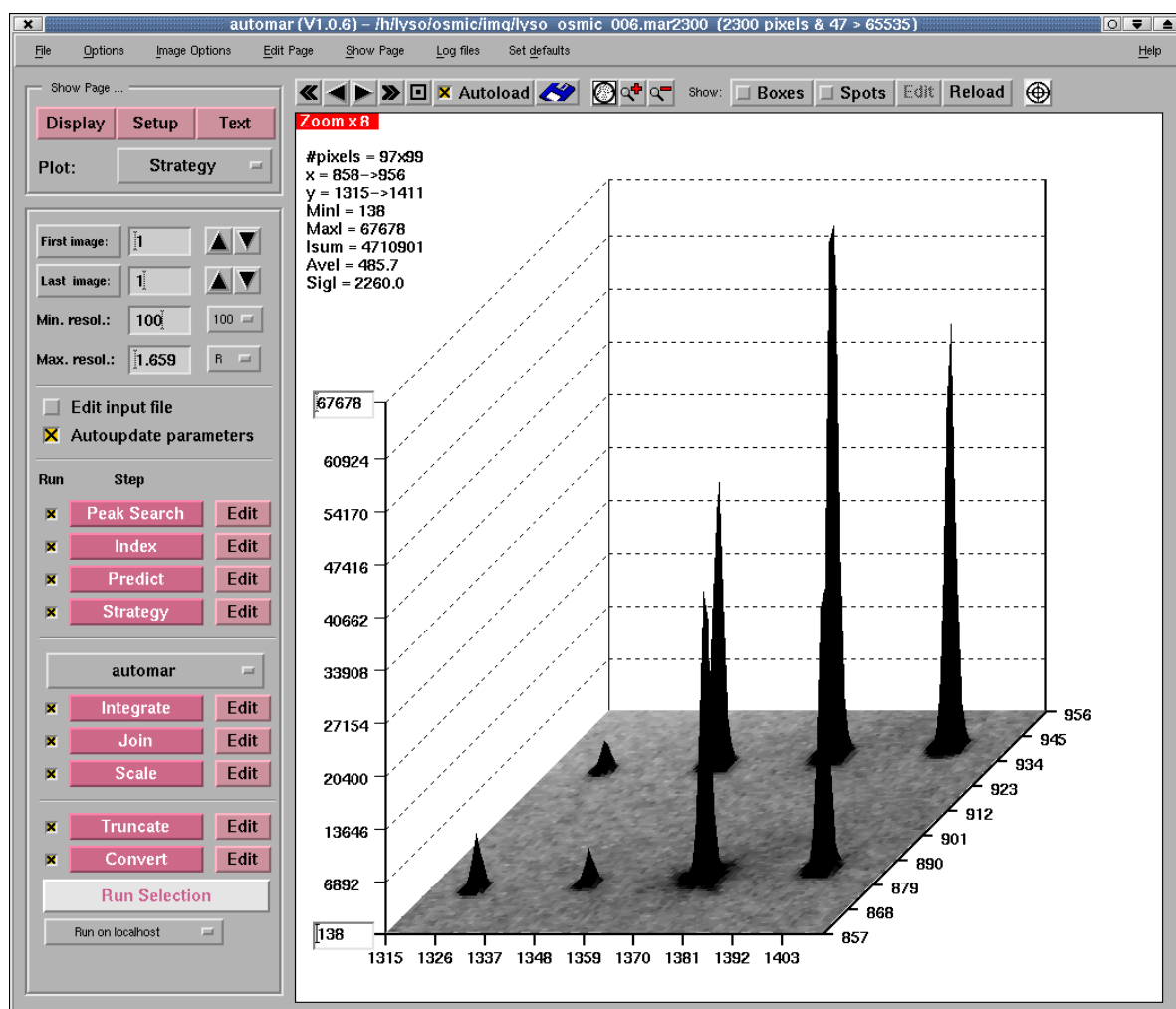


The *automar* Manual

By Dr. Klaus S. Bartels and Dr. Claudio Klein



The *automar* Manual

Version 1.5

May 2005

(*automar* 1.8 - June 2005)

Written by Dr. Klaus S. Bartels and Dr. Claudio Klein

Copyright 2002-2005 Marresearch G.m.b.H.

All rights reserved.

This document contains proprietary and confidential information of Marresearch G.m.b.H.. The contents of this document may not be disclosed to third parties, copied or duplicated in any form, in whole or in part, without the prior written consent of Marresearch G.m.b.H..

Possession, use, or copying of the software described in this publication is authorized only pursuant to a valid written license from Marresearch G.m.b.H.

marresearch, *automar*, *desktop beamline (dtb)* and the *marresearch* logo are registered trademarks of Marresearch G.m.b.H..

Table of contents:

1. Introduction	5
1.1 Preface	5
1.2 Overview	5
1.2.1 <i>mar</i> Peaks	5
1.2.2 <i>mar</i> Index	6
1.2.3 <i>mar</i> Predict	6
1.2.4 <i>mar</i> Strategy	6
1.2.5 <i>mar</i> Process	6
1.2.6 <i>mar</i> Post	7
1.2.7 <i>mar</i> Scale	7
1.2.8 <i>mar2mtz</i>	8
1.3 Definitions	8
1.3.1 Coordinate systems	8
1.3.2 Syntax	8
2. Auto-Indexing: <i>mar</i> Peaks and <i>mar</i> Index	8
3. Spot Prediction: <i>mar</i> Predict	9
3.1 Usage	9
3.2 Parameter File	9
3.2.1 Valid keywords and parameters	9
3.2.1.1 Crystal	10
3.2.1.2 Beamline and goniometer	10
3.2.1.3 Detector	10
3.2.1.4 File names	10
3.2.1.5 Output control	10
3.3 Command Line Options	10
3.4 Image Sequences	11
3.5 Output files	11
3.5.1 The log-file	11
3.5.2 The .prd-file	11
4. Data Collection Strategy: <i>mar</i> Strategy	12
5. Image Processing: <i>mar</i> Process	12
5.1 Usage	12
5.2 Methods	12
5.2.1 References	14
5.3 Parameter File	14
5.3.1 Valid keywords and parameters	14
5.3.1.1 Crystal parameters	14
5.3.1.2 Beamline and goniometer	15
5.3.1.3 Detector	15
5.3.1.4 File names	15
5.3.1.5 Options	15
5.4 Command Line Options	15
5.4.1 "-v" (verbose)	15
5.4.2 "-s" (service)	15
5.5 Image Sequences	16
5.6 Output Files	16
5.6.1 The .i-file	16
5.6.2 The log-file	16
5.6.3 The history-file	22
5.6.4 The summary-file	24
5.6.5 The background log-file	26
6. Summation of Partial and Post-refinement: <i>mar</i> Post	28
6.1 Usage	29
6.1.1 Example	29
6.2 Parameter File	30
6.2.1 Valid keywords and parameters	30
6.2.1.1 Crystal parameters	30
6.2.1.2 File names	31
6.2.1.3 Image numbers	31

6.2.1.4 Options	31
6.3 Command Line Options	31
6.3.1 -s (service)	31
6.3.2 -v (verbose)	31
6.4 Image Sequences	31
6.5 Input and Output Files	32
6.5.1 The optional background-file (input)	32
6.5.2 The summary-file	32
6.5.3 The .Fsq-file	32
6.5.4 The optional .F-file	33
7. Scaling: <i>mar</i> Scale	34
7.1 Introduction	34
7.2 Method	34
7.3 Usage	35
7.4 Parameter File	35
7.4.1 Valid keywords and parameters	35
7.4.1.1 Crystal parameters	35
7.4.1.2 File names	35
7.4.1.3 Input/output format	35
7.4.1.4 Options	36
7.4.2 Description of keywords, parameters and options	36
7.5 Command Line Options	41
7.5.1 -vp ("verbose")	41
7.5.2 -s ("service")	41
7.6 Input and Output files	41
7.6.1 The reflexion input	41
7.6.2 The <i>mar</i> Scale listing file	42
7.6.3 The output intensity files	45
7.6.3.1 The mean intensity file (.mean)	45
7.6.3.2 The asymmetric unit file (.asu)	46
7.6.3.3 The scaled measurements file (.unmerged)	46
7.6.4 The optional output files	46
7.6.4.1 service PROTEIN	46
7.6.4.2 service SCALEPACK	46
7.6.4.3 service SHELX	46
8. Data Conversion	47
8.1 mar2mtz	47
8.1.1 Usage	47
8.1.2 Valid keywords and parameters	47
8.1.3 Command line options	48
APPENDIX: Keywords in Parameter Files	49
A1.1 Syntax	49
A1.2 Comments in the Parameter File	49
A1.3 Basic Parameters	49
A1.4 Note on Eulerian angles and cradles & non- <i>mar</i> goniostats:	50
A2.1 Input Parameters in Alphabetical Order	50
A2.2 Keywords in Groups and Examples	61

1. Introduction

1.1 Preface

automar is a suite of programs designed to render crystallographic image data reduction as automatic as conceivable; manual intervention should be needed at very few points only, if any at all, except for the assessment of the definite space group (which is not needed until scaling or even thereafter).

At every time, all known meaningful parameters are saved, either in file header records or in standard parameter files, from the very data collection on any *mar* detector onwards, right down to the scaled mean intensities. All this information is ASCII-coded and can therefore be read, and possibly be edited (except for compressed *mar* images, where special *mar Tools* are available).

Alternatively, each program can read input from some control file, e.g. "xxx.ctrl" (same syntax as standard parameter files).

automar is most conveniently run under the control of its versatile *graphics user interface (GUI)* which includes:

- setup of directories, file names, image and detector parameters, and display of any refinement results (marked by colour);
- display of images and spot positions (predicted or integrated);
- soft buttons to call any individual program of the *automar* suite, or a sequence of them, generating appropriate control files.
- graphics display of results.

Tutorial introductions are found throughout the *automar* manuals.
For details, see the *automar GUI* manual.

Each program can alternatively be run from the command line, this way enabling additional features - mostly for better overview or for debugging purposes.

1.2 Overview

Note: The input/output file names given in this overview are in accordance with the *mar* IP scanner file conventions; *mar* CCD - image files (and their derived *automar* file-names) have the running number in the extension.

1.2.1 *mar* Peaks

- locates spot coordinates on one or more images.
- *input:* image-file "root_#.mar_type" (with # = running image number; and "mar_type" = the file extension, e.g. mar2000)
- *output:* file-name "root_#.pks"

1.2.2 *mar Index*

- selects reflexion coordinates from one or more images (e.g. from **mar Peaks** output); less than 100 spots are usually sufficient; as few as a dozen may do in favourable situations.
 - constructs and refines numerous conceivable "zone axes", based on sums of difference vectors in reciprocal space; finds the smallest reduced cell from the best zone axis triples. The Bravais lattice is chosen to have the highest symmetry consistent with the standard deviations; this symmetry is possible solely in geometrical terms, without reference to reflexion intensities.
 - refines the beam centre on the basis of intersecting lunes rather than spot coordinates; in most cases its radius of convergence is considerably larger than half the spot distance.
 - deduces starting values for mosaicity/beam divergence and detector parameters;
 - outputs the results in a format suitable for **mar Predict** etc.
- *input:* peaks file "root_#.pks"
- *output:* file-name "root_#.amp" (**automar-p**parameters)

1.2.3 *mar Predict*

- compiles expected spot coordinates for one or more images, based on input parameters that are usually generated by **mar Index**;
 - includes alpha1/alpha2 - split for conventional X-ray sources;
 - can predict two superimposed lattices (and their mutual overlaps) caused by double wavelength ($K\alpha/K\beta$; S.R. 3rd order) or by twinned or split crystals;
 - computes approximate width of rocking curve for each reflexion and concomitant movement of reflexion centre for partials.
- *input:* **automar**-parameters "root_#.amp" (or e.g. "marPredict.ctl")
- *output:* direct-access file "root_#.prd"
log-file "root_#.marPredict"

1.2.4 *mar Strategy*

- based on cell parameters and orientation (e.g. from **mar Index**) and spot prediction, several rotation ranges are suggested that promise a most economic data collection (i.e. smallest PHI-ranges with high degrees of completeness), along with measurement multiplicities.

1.2.5 *mar Process*

- integrates reflexion intensities of one or more blocks of images, and refines/updates current parameters; restart if input values differ considerably and consistently from observations;
 - determines reflexion boundaries by a modified [seed-skewness method](#), and builds average profiles on this basis, taking into account oblique incidence and divergence;
 - profile-fitting includes possible intruders and saturated pixels; otherwise it is restricted to weak and medium intensities of non-split spots;
 - sigma values are purely Poissonian, assuming a gain factor of 1 ADC unit per X-ray quantum ... see **mar Scale** for updated experimental sigmas.
- *input:* parameters "root_#.amp" (or e.g. "marProcess.ctl")
image-files "root_#.mar_type"
- *output:* (1) common files for all images of the sequence(s):
- | | |
|------------------|---|
| all refinements | "root_##.history" (incl. restarts) |
| last pass | "root_##.summary" |
| average profiles | "root_##.prf" (binary working file, but kept for display) |
| average backgrd. | "root_##-bgr" (for pre-scaling in mar Post) |

(2) for each image separately:

binary	"root_#.prd" (<i>mar Predict</i> step, kept for display)
log-file	"root_#.marProcess"
hkl-intensities	"root_#.i"
optional listing	"root_#.int_list"

1.2.6 *mar Post*

- one or more blocks of processed image reflexions are sorted;
- reflexions are optionally pre-scaled (e.g. based on background);
- fully recorded reflexions are passed, unmodified except for pre-scaling;
- partials are added, their centroid and angular widths are fitted, those too close to any rotation boundary, or non-contiguous sequences are rejected; the summed intensity is assigned a batch number closest to the fitted centroid (cf. *mar Scale* B-factors !)
- widths and centroids are used to post-refine :
 - o crystal mosaicity and beam divergence
 - o setting angles
 - o misalignment angles of rotation axis
 - o cell parameters
- *input:* sequence(s) of hklI files "root_#.i"
optional control file to override header parameters
- *output:* marPost.Fsq
marPost.summary

1.2.7 *mar Scale*

- allows scaling of $I/+/$ and $I/-/$ as separate mean values ("ano");
- allows scaling with respect to some external reference data set;
- restricts B-determination to continuous functions; selectable are functions of time (i.e. exposure number) and/or *PHI* angle; functions of detector coordinates might be added in future.
- image scale factors may be determined individually or, like B, "smoothed" by functions, or by digital filter.
- allows scaling of data from multiple or multi-facetted detectors - up to nine (9) simultaneously exposed images - which share the B value and possibly S-functions of *PHI* (not presently for *mar* bases).
- summed partials receive their own scale factors
- scaled intensity variances of multiple measurements are used to adapt output sigmas S as:

$$S^2 = (g \cdot \sigma)^2 + (f \cdot I)^2,$$
 where g (gain) and f (systematic error fraction) are fitted from all data.
- *input:* xxx.Fsq
optional control file
optional reference data
- *output:* xxx.marScale
xxx.mean
xxx.unmerged + optional .dF, .asu

1.2.8 mar2mtz

Converts one or more sequences of .i-files to CCP4 mtz-format for use by

—► mtzsort —► scala —► etc.

1.3 Definitions

1.3.1 Coordinate systems

Detector coordinates are usually denoted by capital (X,Y);
reciprocal lattice vectors are (x,y,z), written in lower case.

The detector is viewed from the crystal (downstream with the X-ray beam) with X pointing from left to right, and Y up (as is usual for 2d-axes).

The origin depends on the units: pixel coordinates are from the lower left corner (as read from the **mar View** display) ranging from 1 to n(scan);

mm coordinates are relocatable from the beam centre, from 0 to R_{\max} .

The reciprocal space coordinate axes are chosen to be as little confusing as possible: $x \parallel X$, $y \parallel Y$, which makes the z-axis point against the X-ray beam.

The detector may, however, incorrectly be set not perpendicular to the beam (or vice versa). The sphere of reflection ("Ewald-sphere") should remain centred on the z-axis, with beam vector and z-axis antiparallel.

This implies that the (x,y)-plane continue to be perpendicular to the beam. It is chosen in such a way that X be the projection of x onto the detector plane. The latter is defined by two tilt angles around X_0 and Y_0 .

This definition may no longer be applicable if the detector is offset by 2θ (tth) and if the tth-axis is not in the plane of x and X; an extra angle (AD_rot) may be needed then; this generalization is described for the [tth keyword](#) in [APPENDIX](#).

See also section 1.2 in the **mar Index** manual on the orientation of crystal axes.

1.3.2 Syntax

[p..]	optional parameters or characters are enclosed in square brackets
[deg]	ditto units ([mm], [pixel]) that apply to more than one value
a b	alternatives are separated by a vertical bar ("or")
x \ y	backslash followed by y indicates the default value(s) for x

The (British) orthography reflexion is used to denote what is measured as hkl-indexed spot, whereas reflection is to denote the physical process of X-rays being reflected.

2. Auto-Indexing: *mar* Peaks and *mar* Index

See separate manuals.

3. Spot Prediction: *mar* Predict

mar Predict is a stand-alone version of **mar Process**'s theoretical reflexion spot coordinate compilation, based on experimental parameters for crystal, beam line / goniometer and detector, as specified in the input parameter file, e.g. a **mar Index** .amp output file. For details [see the keyword sections](#).

> No use of image data, no refinement of parameters.

3.1 Usage

```
marPredict Parameter.file [options] [f1 l1 f2 l2 f5 l5]
```

```
    <---3.2---> <---3.3---> <---3.4--->
```

Compute one or more patterns, depending on the "image[s]" keyword (cf. the [keyword section 3.2](#) and [APPENDIX](#)) and/or on an optional image sequence definition as given by [f1 l1 ...] —▶ [see section 3.4](#). One (image_#).prd output file per image/prediction.

The output file .prd is accepted by **automar** display window and by **mar View**.

This way **mar Predict** / **automar** display (or **mar View**) can be a tool, both for interactive indexing and for crystallographic teaching.

If started from the **automar GUI**, parameters are inferred from the SETUP window; standard name for its parameter file is "marPredict.ctl".

3.2 Parameter File

Mandatory parameters:

- crystal cell and its orientation
- start & end of PHI rotation/oscillation (END - START < 180)
- wave-length(s)
- sample-to-detector distance
- detector size, pixel size, beam centre
- spot size (for overlap test)

Optional parameters:

- inner and outer radius, or resolution limit, for calculated pattern
- run or crystal number (id. for later scaling, immaterial for prediction)
- lattice and/or space group (for extinctions)
- crystal mosaicity] [for extension of
- beam divergence] [partiality range
- polarization (monochromator, mirrors, S.R.)
- alignment of PHI axis (CHI, CHI_{offset}, OMEGA_{offset})
- alignment of detector (2Theta; tilt_X / _Y, relative rotation = turn_Z)
- useless detector areas (beam stop, ice or powder rings, other shadows) or window limits
- title for log-file
- list (additional listing with all hkl along with coordinates, Lorentz-polarisation factor, predicted centroid of rocking curve, and lattice-#)

All of these optional parameters have reasonable default values.

3.2.1 Valid keywords and parameters

Mandatory keywords are * starred.

For description see —▶ [APPENDIX](#).

3.2.1.1 Crystal

- * cell
- * axes
- * orientation
- lattice
- space-group
- run | xtal | crystal
- mosaicity
- resolution

3.2.1.2 Beamline and goniometer

- * image[s] (including START and OSC sub-keywords)
- * distance (sample-to-detector)
- * lambda (wave-length)
- polarization
- divergence
- chi
- omega | omg
- alignment

3.2.1.3 Detector

- scan size
- * pixel size
- * spot size
- * beam centre
- beamstop
- twotheta | tth
- radius
- tilt
- rotate
- ice
- ring
- shadow
- window

3.2.1.4 File names

- data | template
- file | name

Comment:

Although **mar Predict** does not use any image data, one image file name definition is needed for construction of the appropriate output .prd file name.

3.2.1.5 Output control

(see also 3.3 "Command line options")

- title
- list

3.3 Command Line Options

The only user-friendly option is "**-vp**" (verbose protocol) which causes a bit more intermediate information to be written.

3.4 Image Sequences

Optionally introduced by `-r (= range[s])`.

While the "image" keyword may specify one range: first & last image number, the command line option extends this concept to more than one ranges:

`{f1 l1 ...}` = first and last image number each, for up to 5 sequences, e.g.

"1 30 32 90" if image #031 is nonsense or unreadable, or

"1 1 180 180" for inspection of first and last image of 180 exposures.

Purpose: Mainly for inspection and survey:

- Before the experiment, run **mar Predict** for the whole *PHI* range to check whether overlap is significant in certain regions. If so, reduce image rotation range $\Delta\text{PHI}=\text{END}-\text{START}$ (throughout or only in these regions).
- To check the **mar Index** result, predict a few patterns throughout the total *PHI* range of the experiment and check the superposition of image and prediction.
- *Phi*-axis misalignment can be found immediately by comparison of 2 images 180 deg. apart in *PHI*
- Useful for crystallographic teaching.

3.5 Output files

3.5.1 The log-file

The first page reflects the input parameter file, then gives three tables with all parameters in use, whether specified or default; titled:

► Beam-line, ► Crystal, and ► Image Parameters.

The "Listing of point reflexions" is by default limited to the reciprocal lattice [A]-matrix and the index range that is being scanned for acceptable spot coordinates.

Comment: The hkl scan is restricted to a reciprocal space volume which approximates the relative movement of the Ewald sphere by spherical boundaries on the "inside and outside".

The full listing of all predicted and all accepted hkl, including coordinates and attributes, can be generated as (image_#).prd_list by means of "list" parameter. Its content should be self-explanatory.

The final "Elimination of reflexions" tabulates the prediction result in groups of full / partial reflexions vs. expected overlap or incompleteness.

Comment: The flagged hkl, either overlapped or cusp, are not really eliminated from the list at this stage. This is rather a warning for the integration step which may still be able to rescue overlapped spots by profile fitting, depending on the size of the experimental profile; or it may use cusp spots for the fit of horizontal mosaicity and divergence.

3.5.2 The .prd-file

This is a direct access file that holds binary data for each predicted reflexion, except for the header which is ASCII - but still unreadable (except on VMS).

These data include: h,k,l
X,Y
Lorentz-pol.-factor
calculated fraction of rocking curve
possibly split $K\alpha_1 / K\alpha_2$
nearest neighbour X,Y

and are read by marIntegrate (**mar Process**).

All floating point values are INTEGER encoded and so scaled that rounding errors are not a problem.

4. Data Collection Strategy: *mar* Strategy

See separate manual.

5. Image Processing: *mar* Process

5.1 Usage

```
marProcess Parameter.file [options] [f1 I1 f2 I2 .. f5 I5]
```

◀---5.3---▶ ◀-5.4-▶ ◀-----5.5-----▶

mar Process does the "real job" of number crunching. One or more images or even (up to 5) sequences of images are evaluated.

5.2 Methods

For each image, the final integration step is preceded by one or more refinement passes. Even if all parameters be fixed by the user, still one pass is needed for building the updated average reflexion profiles.

Each spot coordinate (X,Y) is dependent on a number of experimental parameters that are cyclically adapted (unless fixed) by least-squares procedures, such as to make the predicted pattern fit the image. Within each refinement pass, the shifts are "learned": Working from the centre outwards, the centroid shifts of the strongest spots are saved and applied "a priori" to the subsequent spots, which in turn may update the saved shifts (if their intensity is among the strong ones).

As a safety means, neighbouring spots are centred together, with their predicted distance kept constant. This will prevent a weak spot from moving into its strong neighbour, because this neighbour will dominate, keeping its position.

The very first image out of a sequence is more critical than the following ones. Preliminary refinement proceeds, therefore, in shells of increasing radius, centring on each spot by means of the classical integration box. Mind that all shifts are somehow proportional to the spot's distance from the beam centre, such that the danger of getting hooked by the wrong spot increases with the radius.

Refined parameters are mainly detector position and orientation (distance, beam centre, and rotation around the beam).

If the radius is big enough for that, estimates are calculated for detector tilt and - based on the displacement of partials - crystal mosaicity and beam divergence in the horizontal plane (i.e. parallel to the PHI axis).

These estimates are not immediately applied, but collected and checked for consistency; they are tabulated in the history file, and possibly updated before a restart.

A routine refinement pass locates each spot not only by the integration box but also by the "seed-skewness-method" (adapted after R.Bolotovskiy et al.)

Intuitively, the argument goes like this: X-ray background obeys Poissonian statistics, which in turn can be approximated by a Gaussian (if its mean is big enough), i.e. a symmetric distribution. The deviation from symmetry around its mean value $\langle x \rangle$ of a distribution of numbers x is defined by its

$$\text{skewness} = \left\{ \sum [(x - \langle x \rangle) / \text{sigma}(x)]^3 \right\} / N$$

which is positive if the distribution has a tail towards $x \gg \langle x \rangle$.

In crystallographic patterns, any "tail pixels" (significantly stronger than average background $\langle B \rangle$) do obviously belong to a reflexion spot, or they are outliers like cosmics or CCD-zingers.

Such spots are located by way of collecting contiguous dark pixels into "islands" until the skewness of the remaining background pixels approaches zero. This procedure yields a more reliable average background than any method that must "know" before hand, where to find the spot(s).

Care must be taken that adjacent spots are not mistaken as one, or that closely neighbouring weak and small islands may represent one reflexion.

Not only do significant islands tell us where they are but also what the shape of their circumference and profile. This is the basis for collecting average profiles in 8 octants, centred at $n \cdot 45 \text{deg}$. each, plus one central area (tabulated as "octant 0"). Each individual profile has been corrected for oblique incidence and ray divergence.

If the "main" island reached out to one or even 2 borders of the bounding area, it is checked whether tails are parallel or perpendicular to the polar radius vector; if so, streaked spots, or tentative powder rings are monitored and summarized in the end of the history file.

The grand-total average spot profile is analyzed such as to possibly update spot size and asymmetry of spot profile (in other words, excentricity of the intensity's centre of gravity: Such spot shape may cause confusion when checking the spot coordinates in the image display; the coordinates are the geometrical centre, not necessarily the peak position.).

Profile-fit is not applied to strong peaks unless they enclose saturated pixels, or are merging with overlapping spots.

The corresponding output column in the [.i-file](#) is, therefore, given the heading "Adaptive fit" rather than "Profile fit": (1) adapted method(s) for integration; (2) both methods have adaptive features: The background threshold in the seed-skewness method, and the relative scaling factor in profile-fitting.

Significant partials common to adjacent images are used for refinement of orientation angles around rotation axes other than the beam. Simultaneously, the [ratio Q of average partial intensity before/after the break](#) is monitored. It is expected to be close to 1.0 - otherwise a shutter problem may exist.

Also, the partials with an expected centre beyond the oscillation range are analyzed (like in **mar Index**) to control the range of partiality. Here remains an uncertainty, though, because spots cannot participate unless they had been predicted. But still, far reaching significant partials can point to a necessary increase of the partiality range. (See also the "mean average intensity profile as a function of PHI" in the [log-file](#)).

In intervals of at least 10 deg. or 5 images, a preliminary post-refinement yields not only crystal missetting angles but also 2 misalignment angles of the PHI-axis. They become quite consistent, not surprisingly, after 30 deg. or more.

Optionally the free crystal parameters are refined simultaneously. Because of their high correlation (in particular of cell angles) with setting and alignment, their refinement is only enabled if the total rotation range is big enough. How much is "big enough" depends on the number of free cell parameters: In small rotation ranges, the least well defined axis is the one pointing along the beam, unless in a hexagonal, tetragonal, or cubic cell it has its equivalent axis more or less parallel to the detector.

The same mode of refinement - restricted to setting angles - is applied in place of the preceding one, image by image, if two adjacent exposures are not sufficient to make partials add up to 100%.

Significant changes in Phi axis alignment and/or cell constants cause restart of the image sequence, ... up to 3 times after increasing PHI intervals - unless restricted by the [rest\[art\] keyword](#).

5.2.1 References

- A.Jones, K.Bartels & P.Schwager, Chapter 8 in "The Rotation Method in Crystallography"
(Ed. U.W.Arndt & A.J.Wonacott; North-Holland, 1977)
R.Bolotovskiy et al., J.Appl.Cryst.(1995) **28**, 86-95
R.Bolotovskiy & Ph.Coppens, J.Appl.Cryst.(1997) **30**, 244-253
A.G.W.Leslie, Acta Cryst.(1999) **D55**, 1696-1702

5.3 Parameter File

For a description, see APPENDIX.

Mandatory parameters:

- crystal cell and its orientation
- start & end of PHI rotation/oscillation
- wave-length(s) & polarization (monochromator, mirrors, S.R.)
- sample-to-detector distance
- detector size, pixel size, beam centre
- spot size

Optional parameters:

- inner and outer radius, or resolution limit, for pattern
- lattice and/or space group (for extinctions only)
- crystal mosaicity
- run or crystal number (id. for later scaling)
- asymmetric spot profile (i.e. skew, excentric distribution)
- beam divergence
- alignment of PHI axis (CHI, CHI_{offset}, OMEGA_{offset})
- alignment of detector (2Theta; tilt_X / _Y, rel.rot.=turn_Z)
- useless detector areas (beam stop, ice or powder rings, other shadows) or window limits
- limit for detector saturation
- fix → exclude parameters from refinement
- title for log-file
- list all hkl with coordinates, Lorentz-polarisation factor,
predicted centroid of rocking curve, lattice-#

All of these parameters default to their image file header values (if applicable); optional parameters have reasonable (programmed) default values.

Note that some mandatory parameters may still have programmed default values, because they are optional for stand-alone **mar Predict** – make sure that these are correct for your setup (e.g. polarization)

5.3.1 Valid keywords and parameters

Mandatory keywords are * starred.

[For description see → APPENDIX.](#)

5.3.1.1 Crystal parameters

- * cell
- * axes
- * orientation
- lattice
- space-group
- run | xtal | crystal
- mosaicity
- resolution

5.3.1.2 Beamline and goniometer

- * image[s] (including START and OSC sub-keywords)
- * distance
- * lambda | wave[-]length
- * polarization
- * divergence
- chi
- omega | omg

5.3.1.3 Detector

- * scan size
- * pixel size
- * spot size
- * beam centre
- beamstop
- twotheta | tth
- rad (= radius or radii)
- ice
- ring
- shadow
- saturation

5.3.1.4 File names

- * data | template
- file | name

5.3.1.5 Options

- fix
- title
- list
- service

5.4 Command Line Options

5.4.1 "-v" (verbose)

-vp (verbose protocol) causes more intermediate information to be written, both on screen and log-file

-vtrf (any combination, where

- t** = preliminary refinement passes,
- r** = refinement pass(es),
- f** = final integration pass) produces •very• extensive listings, named (image_#).int_list, for every individual spot, with integration box / seed-skewness islands / local profile.

5.4.2 "-s" (service)

-sMOSFLM causes **mar Process** to write an updated AUTOMAR.UMAT file at the end of the run.

-sDENZO causes **mar Process** to write, not the .i-file, but a .mar_x file which is largely compatible with DENZO .x-files.

Comment: 'service' is also available as input keyword.

5.5 Image Sequences

Optionally introduced by `-r` (= ranges).

{f1 l1 ...} = first and last image number each for up to 5 sequences, e.g.

"1 30 32 67 69 90" if images #031 and #068 are nonsense or unreadable, or

"1 60 181 240" for anomalous hexagonal data taken @ PHI=0-60 & 180-240deg.

This option extends the image keyword which specifies first and last image number. It is now also supported by the *automar* GUI [[Integrate — Edit](#)] by way of multiple 'image' keywords.

The advantage of processing such blocks of images in one go is that the continuous update of refined parameters will be consistent for the whole lot.

5.6 Output Files

5.6.1 The .i-file

The file is ASCII readable.

(1) header information (title and amp-style parameter records)

(2) reflexion records, 1 record per hkl, 16 numbers :

h,k,l, lattice-# (=1 unless superimposed lattices)

partiality flag (0=full;

1=partial, predicted centroid before the mid PHI

2=partial, predicted centroid after the mid PHI

9= cusp, never passing through the sphere entirely)

I,sigma(I) ("adaptive fit")

I,sigma(I) (box)

Lorentz-polarization-factor

X,Y (spot centre in [pixel])

PHI (predicted centre of rocking curve)

w=estimated width of rocking curve

f=estimated fraction of total integral

m=MOSFLM counter of partiality (0; 21,22; 31,32,33; 41,42,43,44; ..)

(FORTRAN) FORMAT (3I4,I3,I2,2(F10.1,F7.1),F6.3,2F7.1,F7.2,F5.2,F6.3,I3)

5.6.2 The log-file

One log-file "(image_#).marProcess" is written for each image processed.

Part 1 is identical to the *mar Predict* [log-file, cf. 3.5.1](#), except for the reflection of the input parameter file, which *mar Process* writes as header to its [history file \(see 5.6.3\)](#)

► Each pass may write two tables that summarize all centroid shifts of significant spots, i.e. from predicted to found centre: This output is forced by the `-vp` (protocol) option, or otherwise by non-negligible shifts (as defined by more than half the spot size).

□ The first table divides the image into a grid of equal squares; each grid displays the average shifts (X,Y, in [pixels]), and - below X,Y - the average $I/\sigma(I)$ and the number of contributing spots, which will allow some judgement of the significance of the average shifts.

This is a section from such a table, taken from a first preliminary refinement pass to $R = 36.0$ mm (= 240 pixel) :

```

MEAN SHIFTS in 1.5 x 1.5 cm wide fields (X across, Y up)
+ average INTENSITY/SIGMA & NUMBER of contributing reflexions:

```

	...	-2	-1	0	1	2	...				
3	+	+	+	+	+	+	+	+			
2	+	+	+	+	+	+	+	+			
			-0.7	2.7	-0.3	2.8	1.7	2.7	3.0	3.0	
			160.	7	316.	6	183.	3	232.	1	
1	+	+	+	+	+	+	+	+	+	+	
		-2.4	1.2	-1.3	1.5	0.0	2.3	3.0	2.0	3.0	1.3
		113.	5	193.	6	472.	4	222.	2	225.	4
0	+	+	+	+	+	+	+	+	+	+	+
		-2.3	0.2	-1.0	0.3	-1.0	-2.0	2.5	1.5	3.2	-0.2
		196.	6	486.	4	644.	2	719.	2	68.0	5
-1	+	+	+	+	+	+	+	+	+	+	+
		-1.7	-1.3	-1.0	-2.0	1.0	-2.0	2.6	-1.2	2.5	-1.5
		123.	3	168.	4	261.	2	495.	5	95.5	4
-2	+	+	+	+	+	+	+	+	+	+	+
		-2.0	-2.0	-0.5	-3.0	0.8	-2.5	2.5	-2.3	4.0	-2.0
		80.6	1	193.	2	84.4	4	251.	4	55.4	1
-3	+	+	+	+	+	+	+	+	+	+	+
-4	+	+	+	+	+	+	+	+	+	+	+

□ The second table may be used as a quick guide to the shifts table above, although it presents different information: not average shifts but the final state of the "learned shifts" when moving from the centre to the left and the right hand image boundaries (as described in [section 5.2, Methods](#)). In other words, they are the last entries each to the left- and rightmost squares of the previous table:

```

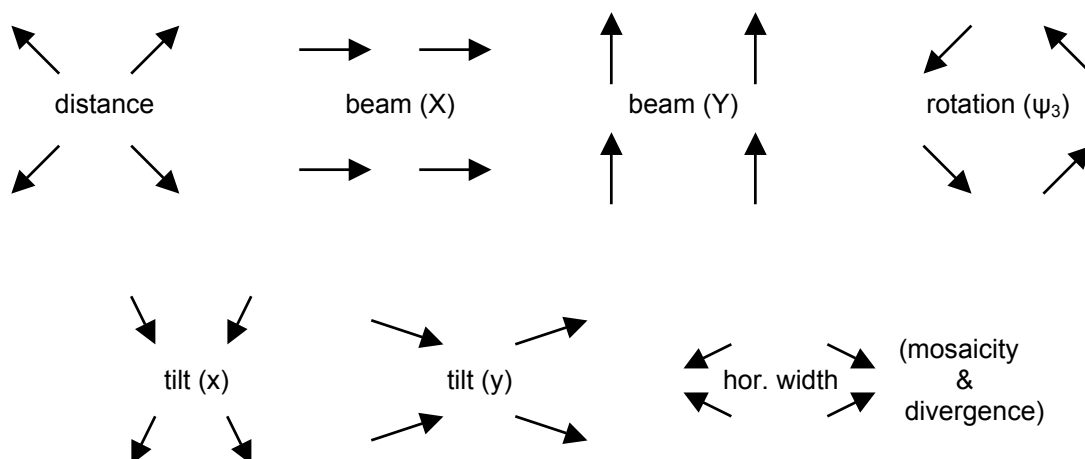
Last applied shifts, in zones of Y, 1.5 cm wide (Y up; units = pixel),

```

	LEFT hand side		RIGHT hand side	
	dX	dY	dX	dY
2.	-1	2	3	3
1.	-3	1	3	1
0.	-3	0	3	0
-1.	-2	-1	2	-2
-2.	-2	-2	4	-2

These shifts indicate a correction of the sample-to-detector distance (increase of D).

Basically, shift patterns like these indicate positive corrections of:



► A statistical table summarizes the integration in terms of significance and rejections:

```
SUMMARY of this pass :
=====
TOTAL REFLEXIONS treated:  781
Thereof found:
  GOOD (I > 3.0*Sig):      580
  0 <= I <= 3.0*Sig.:     153
  Fsq NEGATIVE . . . :     23
  Reflexions TOO DARK :    0  ( >249999 )
  too close OVERLAP :     0
  Refl. out of bounds :    25
  Average I/sigma(I) :    55.7      dito 50% high res.:  23.3
  .. contributed :        748      362
```

► Each refinement pass finishes with a table of resulting parameters, and, unless preliminary, the average profile(s):

```
Summary of positional spot refinement :
=====
Effective detector tilt . . X:  0.010  Y: -0.033 [deg.]
                               ...ignored, but saved.
Fit horizontal divergence . . :  0.439 +/- 0.015 [deg.]
Fit horizontal mosaicity . . :  0.053 +/- 0.029 [deg.]
                               ...ignored, but saved.
Correction for BEAM CENTRE  X:  0.10   Y: -0.08 [pixel]
=> new BEAM CENTRE . . . :  596.8   597.6 [pixel]
DISTANCE correction 0.00 % :  0.00 mm
                               new DELTA : -0.95 mm
=> effective DISTANCE :  90.26 mm
New setting angles phi1|-2|-3 : 36.307 -11.940 -10.223
```

► The protocol (**-vp**) option causes each of the 9 profiles and the grand total to be printed (like reflexion images, both average and normalized), otherwise a summary of contributing spots and the overall-average profile only are printed:

```

Sum of all profiles :
    Try modified spot size : 0.82 x 0.82 --> 1.10 x 1.10 [mm]
-> "user-mask" size : 7 x 7
=> unweighted average: <Fsq> = 3201.723      <Bg> = 58.35018
    region half widths :      7              7
** weighted average profile :
    
```

	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	13.	14.	15.
14.
13.	1
12.	1	1	2	2	1	1
11.	.	.	.	1	2	2	4	5	4	2	1
10.	.	.	.	1	3	7	20	35	18	6	2	1	.	.	.
9.	.	.	.	1	5	22	152	316	134	16	3	1	.	.	.
8.	.	.	1	2	5	36	327	689	291	26	4	1	.	.	.
7.	.	.	.	1	4	22	157	332	157	18	4	1	.	.	.
6.	.	.	.	1	3	7	22	39	23	8	3	1	.	.	.
5.	.	.	.	1	2	3	5	7	5	3	1
4.	1	1	2	2	2	1	1
3.	1	1
2.

- Histogram-bar like projections of the latter are printed for a quick control of the spot centring procedure (profile normalized):

```

Central profile constructed from 509 spots :
> 99%      ...
> 90%      XXX
> 80%      XXX
> 70%      XXX
> 60%      XXX
> 50%      .: XXX
> 40%      XXX XXX %+:
> 30%      XXX XXX XXX
> 20%      XXX XXX XXX
> 10%      XXX XXX XXX
> 0%      _ % _ :+&_XXX_XXX_XXX_ %+:_ + _

      -3._-2._-1._ 0._ 1._ 2._ 3._
3.:  5  8 14 18 14  8  3  |+
2.:  9 23 67 117 59 21  7  |XX:
1.: 15 74 5071051 445 54 11  |XXXXXXXXXXXXXXXXX+
0.: 18 12110892293 969 87 14  |XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
-1.: 15 73 5241104 523 61 12  |XXXXXXXXXXXXXXXXX+
-2.: 10 24 75 128 76 25  9  |XX+
-3.:  5 10 16 22 17  9  4  |*
    
```

Comment: This profile projection graph may also be written to the screen window for a continuous check by input keyword "list 1" (*automar* GUI text window: select "extended output" in the Integrate —► Edit menu), at the expense of less concise screen listings.

Precaution: The above theory assumes that the individual background pixels be statistically independent. This assumption may hold for direct radiation conversion, where each X-ray quantum contributes in exactly one pixel. Both CCD and IP detectors measure fluorescence light in one or the other mode. For the assumption to hold is equivalent with the claim that all light quanta emerging from one X-ray quantum be selected in one pixel only, or in other words, the "point spread function" of the readout system be considerably smaller than the pixel size, which is probably not true. Therefore, these gain (and bias) values are not printed by default, and are not now applied to output sigma(I) values. Rather the scaling (**marScale**) will find variances of multiple measurements and deduce both gain (g) and systematic error fraction (f) of intensity, and apply those to the sigmas of mean intensities.

Following the statistical summary table of the evaluation pass, two graphs are printed. These show the distribution of significantly measured spots. Each reflexion is placed according to its (predicted) PHI position relative to the current oscillation range, in shells of 1/10 of this range. Both the number and their mean intensity are displayed:

```

Mean significant spot intensities :
-----
          significant spots:                mean intensities:
<-1.5*OSC 26                               99.
-0.75  2 | *                               :        62. <I>                :
-0.70  7 | | *                             :        206. | <I>                :
-0.65  5 | | *                             :        193. | <I>                :
-0.60  5 | | *                             :         92. | <I>                :
-0.55  2 | | *                             :        151. | <I>                :
-0.50  8 | | *                             :        135. | <I>                :
-0.45 15 | | *                             :        305. | <I>                :
-0.40 15 | | *                             :        311. | <I>                :
-0.35 29 | | *                             :        413. | <I>                :
-0.30 18 | | *                             :        677. | <I>                :
-0.25 28 |+- - - - - - - - - - - - - - + 412. +- - - - <I> - - - - - - - - -
-0.20 31 | |                               :         874. | <I>                :
-0.15 29 | |                               :         848. | <I>                :
-0.10 23 | | *                             :         878. | <I>                :
-0.05 40 | |                               :        1336. | <I>                :
 0.00 31 | |= = = = = = = = = = = = = = | 1160. |= = = = = = = = = = = <I> |
 0.05 33 | |                               :         991. | <I>                :
 0.10 32 | |                               :         987. | <I>                :
 0.15 45 | |                               :        1165. | <I>                :
 0.20 37 | |                               :        1111. | <I>                :
 0.25 35 |+- - - - - - - - - - - - - - + 756. +- - - - - - - - <I> - - - - -
 0.30 32 | |                               :         633. | <I>                :
 0.35 27 | |                               :         626. | <I>                :
 0.40 25 | | *                             :         413. | <I>                :
 0.45 18 | | *                             :         396. | <I>                :
 0.50 12 | | *                             :         316. | <I>                :
 0.55 10 | | *                             :         359. | <I>                :
 0.60  6 | | *                             :         191. | <I>                :
 0.65  8 | | *                             :         143. | <I>                :
 0.70 10 | | *                             :         161. | <I>                :
 0.75  5 | | *                             :         258. | <I>                :
> 1.5*OSC 30                               107.
    
```

- A step function in primary beam intensity (e.g. synchrotron) may be visible here, though not as a step but as an asymmetry, due to the widths of the rocking curves => length of time that the hkl reflects.
 - Both the number and the mean intensity of significant partials outside the rotation range will be indicative of the range of partiality (= width of rocking curve). As mentioned above, there remains an uncertainty here: reflexions that have not been predicted cannot contribute (cf. 5.2. Methods: Refinement of common partial centroids). But still the range of significant mean intensities, and the drop-off of the graph tails are a good estimator.
- Also, an erratic distribution here will most probably be due to a wrong crystal setting (e.g. caused by a discrepancy for the defined PHI between **mar Index** and **mar Process**). This is then paired with low values for average $\langle I/\sigma(I) \rangle$ and small fractions of significant vs. predicted spots.

- The usually last paragraph will summarize the positional refinement of partials common to adjacent exposures. It is only present if sufficient preceding images have been processed to make partials add up to 100%.

One of two methods is applied depending on whether two or more successive images are required for any full reflexion profile (see [section 5.2. "Methods"](#), and also next [section 5.6.3.\(3\)](#) on the history-file identifiers "R" and "S").

```
Refinement of setting angles from partials :
-----
==>  868 acceptable reflexion records sorted;
      139 significant pairs of partials.

psi1,2 =  0.026  0.018 [deg.]
+|-      0.004  0.007

average of 147 r.l.points (I > 3.0 sigma):
psi1 = -0.0243 +|-  0.1150
psi2 =  0.0036 +|-  0.0534

New setting angles phi1|-2|-3 :  -0.880  4.528  30.390
-----
```

5.6.3 The history-file

(1) repeat of input parameters

Thereafter alternating listings (for each restart):

(2) Integration results:

- (A) step, and
 - (B) image numbers with rotation range.
- Then hkl-statistics with 9 columns:
- (C) total predicted;
 - (D) significant,
 - (E) insignificant (< 3sigma)
 - (F) negative I;
 - (G) saturated (with threshold for current detector);
 - (H) overlap (unresolved => rejected);
 - (I) out of bounds;
 - (J) <I/sigma(I)> ratio for all spots, and
 - (K) .. dito for the 50% high resolution spots

➔ In particular the last 2 columns are worth watching, because they reveal any decay in crystal quality.

This is an abbreviated example listing:

(A)	(B)	(C)	(D)	(E)	(F)	(G)	(H)	(I)	(J)	(K)	
step	image	rotation range	total	signif/insign/neg/satur./ovl/out	<I/s>						
#		[deg.]	predicted	(3.0*sig)	(> 65534)				total	50%high	
										resol.	
....											
15	1473	16.00 .. 16.50	1164	696	369	93	0	0	0	11.4	5.9
16	1474	16.50 .. 17.00	1145	704	359	74	0	0	0	11.5	5.9
17	1475	17.00 .. 17.50	1171	701	359	104	0	0	0	10.8	4.7
18	1476	17.50 .. 18.00	1195	708	401	82	0	2	0	9.8	4.7
19	1477	18.00 .. 18.50	1180	681	399	94	0	0	0	10.3	4.6

(3) History of refined parameters:

- (A) step & (B) image numbers as before;
- then refined values for:
- (L) setting angles phi1,2,3 [deg.]

- (M) distance [mm]
 (N) detector tilt(X,Y) [deg.]
 (O) width of rocking curve in the vertical plane
 (P) mosaicity & divergence in the horizontal plane
 (Q) <I>-ratio Q of partials before/after the break
 (R) PHI-misalignment ($\delta_{1,2} = \chi_{\text{off}} - \omega_{\text{off}}$) [deg.]
 (S) percentage of significant intensities

Setting angles and distance (and beam centre) are actually updated at each step (unless fixed), the remainder is saved and checked for significance and consistency and may be applied before a next restart.

Any line marked "R" (column between step and image-#) results from the refinement of partials located on the border between two adjacent images.

"S" in this column marks the marPost-like refinement which can either replace "R" in cases where partials are extended over more than 2 images (and will only include as many of the most recent images as are needed to yield 100% sums); otherwise "S" is the intermediate post-refinement that may lead to a restart due to significant changes in misalignment angles or cell constants.

Spot size and shape and/or crystal cell parameters have no dedicated columns but are monitored on separate lines where appropriate.

The last column will reveal if the PHI axis failed to move. Depending on the actual size of divergence & mosaicity, the percentage will drop sharply and possibly recover if refinement be able to "lock in" again. Anyway such a region should be eliminated or - if reasonable - reprocessed.

The Q column should oscillate around 1.0 - any average significantly above or below may be indicative of shutter problems that may either lead to a still exposure superimposed on the rotation (in the beginning or in the end) or to one end under-exposed.

This is again an abbreviated example listing:

(A)	(B)	(L)			(M)	(N)	(O)	(P)	(Q)	(R)	(S)	
step	image	setting angles			distance	tilt	vert.	hor.width	misalignment		sign.	
#		[deg.]			[mm]	[deg.]	width	(mos. div.)	(chi;-omega)		I	
		(1)	(2)	(3)			[deg.]	Q	[deg.]		[%]	
initial:		-1.021	4.639	30.203	49.20	0.16	0.06	0.550	0.300	0.350	0.220	-0.110
	spot-size	5 x 5										
	cell:	53.945	57.330	66.230	90.00	90.00	90.00					
1.	526	-1.024	4.637	30.335	49.21	0.11	0.26		0.299	0.346		62.9
2.	527	-1.023	4.637	30.332	49.23	0.16	0.27		0.301	0.336		61.7
	R	-1.006	4.614	30.331				0.412	0.211	1.32		
adapted >	spot-size	7 x 7			.	.	.	0.26
3.	526	-1.006	4.614	30.331	49.25	0.11	0.44		0.301	0.328		62.7
4.	527	-1.006	4.614	30.330	49.25	0.16	0.25		0.312	0.294		60.4
	R	-0.999	4.611	30.330				0.418	0.205	1.18		
5.	528	-0.999	4.611	30.327	49.25	0.17	0.21		0.296	0.342		63.1
	R	-0.999	4.609	30.327				0.440	0.218	1.06		
...												
285.	808	-0.879	4.612	30.355	49.28	0.16	0.23		0.261	0.296		57.8
	R	-0.850	4.622	30.347				0.406	0.222	1.67		
286.	809	-0.850	4.623	30.347	49.28	0.22	0.24		0.266	0.301		57.5
	R							0.615	0.278	0.66		
	S	-0.977	4.491	30.306					0.290	0.397	0.176	-0.163
+ -	<	0.000	0.001	0.001					0.013	0.010	0.000	0.002 >
post-refined	cell:	53.947	57.316	66.237	90.00	90.00	90.00					

283 sets of partials on image boundaries:

ratio Q of average intensities 1.03
 (= geometric mean; average = 1.06 +/- 0.26)

(continued)

(continued)

```

?? Superimposed 1 powder rings with d-spacings [A] of:
                                4.301
-> contributing spots: 10
!! 7 streaked spots diagnosed, 4 rejected.

```

>>>> *Comment:* 10 spots on nearly 300 images are obviously insufficient to prove the existence of a powder ring, and one might argue whether this comment should not be omitted in such a case. Still it is interesting to note that the d-spacing coincides with the diffuse scatter, which is described in section 5.6.5 below, based on the general background analysis. Seemingly most spots have been able to treat this scatter as local background, few exceptions have not.

5.6.4 The summary-file

This is a mixture of the history-tables of the final pass only, designed to give a quick overview over the parameters that have determined the content of the .i-files, plus the overall background analysis that may hint experimental problems.

(1) Title and parameters unmodified throughout the final pass.

Example:

```

marProcess parameters for "trp_ham2_01":
cell      53.945  57.330  66.230  90.00  90.00  90.00
mosaicity 0.300
divergence 0.250 0.350
align     0.220 -0.110
tilt_X,Y  0.158 0.262

```

(2) Table with updated parameters and statistical results:

image #	setting (1)	angles (2)	distance (3)	sign.I %	satur. (> 65534)	ovl.	<I/s> total	50%high resol.
....								
1473	-13.188	4.597	30.036	49.25	59.8	0	0	11.4 5.9
1474	-13.189	4.601	30.048	49.26	61.5	0	0	11.5 5.9
1475	-13.201	4.590	30.063	49.25	59.9	0	0	10.8 4.7
1476	-13.200	4.592	30.063	49.25	59.2	0	2	9.8 4.7
1477	-13.190	4.594	30.063	49.25	57.7	0	0	10.3 4.6

(3) Overall background analysis:

Each analyzed spot throughout the *mar Process*-run contributes to this analysis with its local background.

This is the idea: Except for variations in absorption, background is independent of the particular crystal setting. It ought to be fairly symmetric around the beam, too. Of course, there may be shadows. If a shadow is caused by an obstacle for diffuse air scatter originating between collimator and crystal, that is fine; it is helpful for the precision of the intensities. If, however, this shadow affects the path of reflexions, it must be excluded from integration - four input keywords exist for this problem.

The rotational-symmetry analysis will point out whether there are shadows; the user must decide whether they are detrimental.

The overall-background analysis will try to fit one or more slopes to the images' average backgrounds (linear regression over). These may reflect decaying synchrotron beam intensity and refills, or long-range variations in absorption by the crystal (mind that the X-ray transmission of the crystal determines the amount of air scatter originating between crystal and beam-stop).

Individual images may be outliers to (any of) these slopes. This might be the image that experienced the synchrotron refill or a temporary closing of the main shutter (if exposure was time-based).

The following example was caused by glitches in the X-ray generator's high voltage:

		Difference in oct.#		1	2	3	4	5	6	7	8
		centred at [deg.]		45.	90.	135	180	225	270	315	360
R-shell	Grand-total	average	profile:	---	---	---	---	---	---	---	---
1	33	179.2 +- 14.0	
2	684	199.7 +- 35.9		[+]	.	(-)	(-)	(-)	.	[+]	.
3	2147	204.5 +- 18.3		[+]	.	.	(-)	.	.	[+]	.
4	3840	191.3 +- 13.5		.	.	.	(-)
5	5880	172.8 +- 13.1		(-)
6	7791	165.6 +- 14.3		(-)
7	9948	156.8 +- 16.4		[+]	(-)
8	12207	137.1 +- 13.6		(-)
9	14321	108.6 +- 10.7	
10	16420	85.0 +- 7.8		(-)
11	17792	69.6 +- 5.8		(-)
12	18506	60.2 +- 4.5		(-)
13	19966	53.9 +- 3.7		(-)
14	20522	48.7 +- 3.6	
15	21338	43.9 +- 2.9		.	.	(-)
16	22358	39.5 +- 2.7		.	.	(-)
17	21978	35.4 +- 2.7		.	.	(-)
18	20759	32.5 +- 2.4		.	.	(-)
19	20147	30.7 +- 2.0		.	.	(-)	.	.	[+]	.	.
20	17984	29.4 +- 1.8		.	.	(-)	.	.	[+]	.	.
R-max. = 65.0 mm				---	---	---	---	---	---	---	---
affected resol.-shells :				2	0	7	3	1	2	3	8

>>>>> Comment:

If the detector is placed other than centric (→ [twotheta keyword](#)) then radius shells are not comparable; instead TT-shells [for the reflexion's Bragg angle 2Theta in deg.] are used.

!!! There seem to be shadowed areas not excluded from integration :

```

-----
shadow in octant #3: R-shell 15 .. 20
shadow in octant #4: R-shell 2 .. 4
shadow in octant #8: R-shell 5 .. 13
and 2 local deviations
--> where octant #n is centred at polar angle n*45deg.

```

--> The central beam-stop shadow may be shifted
into polar direction 225. deg.

!!! You may want to check the following images ...

```

-> reason (B) : average background deviates from <<B>>
    or (b) : local background deviates from average;
and/or (I) : mean intensity/PHI deviates from <I>
    ... more than 3 standard deviations !!!

```

-> Expected = 58.4 + [-0.02 * (img.# - 526)] from image # 526 to 809

```

# 550 (B) : <B> = 53.4 << expected 58.0 +- 1.5
# 579 (B) : <B> = 50.9 << expected 57.6 +- 1.5
# 594 (B) : <B> = 46.2 << expected 57.3 +- 1.5
# 595 (B) : <B> = 52.0 << expected 57.3 +- 1.5
...
# 656 (b) : patch in octant #1, R-shell 14 .. 14
...
# 730 (B) : <B> = 48.0 << expected 55.2 +- 1.5
# 808 (B) : <B> = 43.0 << expected 54.0 +- 1.5

```

>>>>> Comment: Indeed all features can be verified in the images :

- octant #3: R-shell 15 .. 20 : the upper left is shaded due to cryo-cooler;
- octant #4 and #8: weaker background due to beam stop holder;
- central beam stop shifted left (though already excluded)
- The outliers are verified in the [background log-file below](#), and also by immediate comparison of images in **mar View** (>10% show up without doubt).

5.6.5 The background log-file

This is, in principle, a scratch or debug file, but ASCII readable, and kept for the possible [pre-scaling in mar Post](#). Any problem monitored in the [summary file \(cf. above\)](#) can thus be verified here.

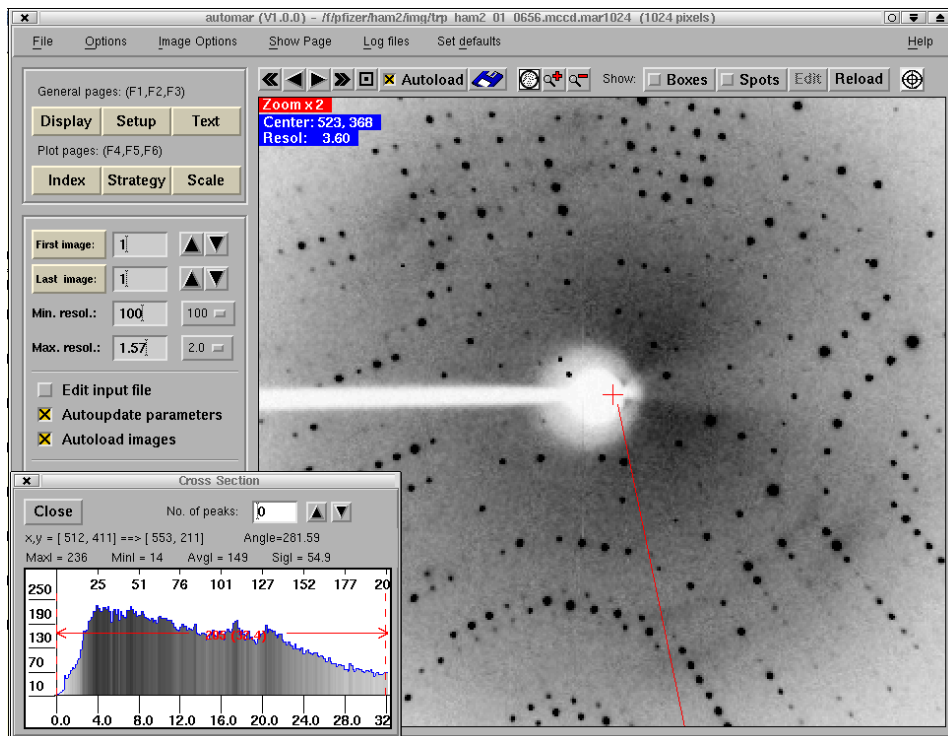
Each spot contributes its local background value to the radial background profile of its octant (mind: octant n is centred around n*45deg; e.g. octant 8 around the positive X-axis). These 8 profiles are written to the background log-file (and also to the log-file if option **-vp** is specified), both as numbers and horizontally displaced printer graphics:

```

\ Octant background profile (intensity to the right)
R-shell \ (Profiles are horizontally displaced to render them discernable.)
 1 |
 2 |           1  2  :  :  :           7  8
 3 |           1  2  3  4  5           6  7  8
 4 |           1  2  3  4  5  6  7  8
 5 |           1  2  3  4  5  6  7  8
 6 |           1  2  3  4  5  6  7  8
 7 |           1  2  3  4  5  6  7  8
 8 |           1  2  3  4  5  6  7  8
 9 |           1  2  3  4  5  6  7  8
10 |           1  2  3  4  5  6  7  8
11 |           1  2  3  4  5  6  7  8
12 |           1  2  3  4  5  6  7  8
13 |           1  2  3  4  5  6  7  8
14 |           1  2  3  4  5  6  7  8
15 |           1  2  3  4  5  6  7  8
16 |           1  2  3  4  5  6  7  8
17 |           1  2  3  4  5  6  7  8
18 |           1  2  3  4  5  6  7  8
19 |           1  2  3  4  5  6  7  8
20 |           1  2  3  4  5  6  7  8

```

Concentrate on "5" first - this is a fair, smooth profile as you would expect it. The other profiles - most prominently #2,3 and #6,7 "bulge" in the radius region around shell 7. It is exactly in these areas that the patterns display strong diffuse scatter of two d-spacings:



The summary computes the overall average for each octant profile, and also the radial average (grand total of all octants).

Comparison of each overall average octant profile with the grand total reveals general patches or shadows.

Comparison of each image octant profile with its overall average may show local problems of individual exposures.

Searching the background log-file for string " =" (with *grep* in unix or linux, or *SEARCH* in VMS) produces this summary, where the outliers pointed out in the summary, e.g. #550, #727, #730, or #808, can be verified:

```

IMAGE # 547: <B> = 57.77 +- 46.51      1132
IMAGE # 548: <B> = 59.50 +- 46.98      1108
IMAGE # 549: <B> = 58.99 +- 47.33      1109
IMAGE # 550: <B> = 53.38 +- 41.95      1118 ← !
IMAGE # 551: <B> = 58.15 +- 45.61      1125
IMAGE # 552: <B> = 58.48 +- 46.42      1136
IMAGE # 553: <B> = 58.44 +- 46.38      1141
...
IMAGE # 720: <B> = 53.30 +- 43.55      861
IMAGE # 721: <B> = 52.81 +- 43.04      864
IMAGE # 722: <B> = 53.21 +- 43.76      850
IMAGE # 723: <B> = 53.72 +- 43.77      855
IMAGE # 724: <B> = 48.95 +- 39.66      870 ← !
IMAGE # 725: <B> = 51.26 +- 42.23      861
IMAGE # 726: <B> = 52.77 +- 42.96      860
IMAGE # 727: <B> = 52.32 +- 43.70      846
IMAGE # 728: <B> = 51.17 +- 42.14      863
IMAGE # 729: <B> = 53.36 +- 44.59      854
IMAGE # 730: <B> = 48.05 +- 39.45      851 ← !
IMAGE # 731: <B> = 53.02 +- 44.00      875
IMAGE # 732: <B> = 53.93 +- 45.26      858
...
IMAGE # 803: <B> = 57.75 +- 48.44      984
IMAGE # 804: <B> = 57.59 +- 48.50     1007
IMAGE # 805: <B> = 56.39 +- 47.51      985
IMAGE # 806: <B> = 51.42 +- 42.57      990
IMAGE # 807: <B> = 56.97 +- 48.08      974
IMAGE # 808: <B> = 42.97 +- 33.66      972 ← !
IMAGE # 809: <B> = 55.24 +- 46.81     1032
ALL IMAGES : <B> = 56.12 +- 46.25    274621
=====> from # 526 to # 809;
          from # 526: <B> = 58.39 + incr * -0.016

```

6. Summation of Partial and Post-refinement: *mar Post*

- ***mar Post*** reads one or more blocks of .i-files, or alternatively .x-files or .mar_x-files, selects BOX- or FIT-intensities (cf. **-s** option) and sorts in hkl, with the image-number, and possibly run|lattice numbers as sub-keys to make partials contiguous.
- Optionally pre-scale partials with one scale factor for an exposure each, which may be read from the optional background-file.
- In general, each reciprocal lattice point (reflexion) will pass through the sphere of reflection more than once; the relative rotation angle and the expected width of reflection are calculated and compared with the measurements present in the data.
- Partial sequences with non-contiguous profiles are rejected (contributions may have been lost due to overlap). Profiles are omitted whose centroid is too close to an image-block boundary (start or end of sequence), or if the number of partial measurements exceeds 99.
- Write summed partials with compound sigma, profile centroid and width; pass fully recorded reflexions, possibly pre-scaled, else unmodified, with calculated profile parameters.
- Centroid and width of rocking curve are fitted for all profiles with at least 3 parts. There are usually sufficient long-ranging partials to fit from their widths the horizontal divergence and mosaicity. If the vertical angular spread has enough contributions, it is calculated as well. The latter condition needs an input angular spread of somewhat more than one image's rotation range, or 3 contributions will never have been predicted.
- Each centroid contributes to the post-refinement.

Here is a bit of theory:

Bragg's reflection condition can be reformulated in the following way:

$$2 \cdot \sin\theta = d^* / \lambda$$

where left and right side describe the length of a normalized reciprocal lattice vector in reflecting condition, i.e. with the reciprocal lattice point located on the sphere of reflection with radius 1 (not $1/\lambda$!) as is the case for those centroids.

Each of these lattice vectors d^* is defined by the integer coefficients hkl of the basis vectors a^*, b^*, c^* (i.e. the crystal cell):

$$d^* = h \cdot a^* + k \cdot b^* + l \cdot c^*$$

(unrotated, in standard orientation);

or in matrix/vector notation:

$$d^* = \begin{bmatrix} a^* & b^* & c^* \end{bmatrix} \cdot \begin{pmatrix} h \\ k \\ l \end{pmatrix}$$

Let

$$[A] = (a^* \ b^* \ c^*) = \begin{bmatrix} (a_x) & (b_x) & (c_x) \\ (a_y) & (b_y) & (c_y) \\ (a_z) & (b_z) & (c_z) \end{bmatrix} = \begin{bmatrix} a_x & b_x & c_x \\ a_y & b_y & c_y \\ a_z & b_z & c_z \end{bmatrix}$$

$$d^* / \lambda = \begin{pmatrix} (x_0) \\ (y_0) \\ (z_0) \end{pmatrix} = [A_0] \cdot \begin{pmatrix} h \\ k \\ l \end{pmatrix} \quad \text{with } [A_0] = [A] / \lambda$$

After application of the setting angles, and rotation about some axis, say PHI (spindle axis with its direction defined by $\delta_1 = \text{CHI}_{\text{off}}$ and $\delta_2 = -\text{OMG}_{\text{off}}$):

$$\begin{pmatrix} (x) \\ (y) \\ (z) \end{pmatrix} = [R] \cdot \begin{pmatrix} (x_0) \\ (y_0) \\ (z_0) \end{pmatrix} = [\Phi] \cdot [\varphi_1] \cdot [\varphi_2] \cdot [\varphi_3] \cdot [A_0] \cdot \begin{pmatrix} (h) \\ (k) \\ (l) \end{pmatrix}$$

with

$$|\mathbf{d}^*| / \lambda = |(x_0, y_0, z_0)| = |(x, y, z)| = r$$

In "mar"-notation (see section 1.2.1: Coordinate systems) the centre of the Ewald sphere is at $x=y=0, z=1$. The condition for (xyz) to fall exactly onto the sphere of reflection reads then

$$\begin{array}{l} (x) - (0) \\ | (y) - (0) | \\ (z) - (1) \end{array} = \begin{array}{l} (h) (0) \\ | [\mathbf{R}] \cdot [\mathbf{A}_0] \cdot (k) - (0) | \\ (l) (1) \end{array} = 1 \quad (\text{normalized sphere})$$

$$x^2 + y^2 + (1-z)^2 = 1$$

$$\Rightarrow r^2 = x^2 + y^2 + z^2 = 2z$$

$$\Rightarrow z = r^2 / 2$$

All centroids (in different Φ orientations) satisfy these equations and are thus observations to define corrections of $[\mathbf{R}]$ and optionally of $[\mathbf{A}_0]$, such as to refine δ_1, δ_2 and $\varphi_1, \varphi_2, \varphi_3$ and possibly the cell constants.

As it happens, φ_3 and χ and/or χ_{off} at $\omega=0$ are both rotations around the beam in the initial orientation and only their difference can be found by this expansion of the Bragg condition. However we have a set of measurements at hand that defines the arbitrary $\chi=0$ orientation:

All rotational pattern refinements are based on the x-pixel direction. So the $\varphi_1, \varphi_2, \varphi_3$ setting angles as refined image by image are complementary information.

- It is possible to deal with superimposed lattices; or with more than one run or crystal; but the post-refinement of the crystal cell in the end will work on one lattice of one run only (the last one, at least in the current version).

6.1 Usage

```

▶ marPost @Parameter.file [options] [f1 l1 f2 l2 .. f5 l5]
      ◀----6.1----▶ ◀--6.2--▶ ◀-----6.3-----▶
or : ▶ marPost 'first.i-file' [options] [f1 l1 f2 l2 .. f5 l5]
Example: ▶ marPost trp_ham2_01_0526.i 526 809 -sBOX

```

6.1.1 Example

The simplest and most direct mode of usage is indicated by the "Example".

- o It is sufficient to specify just the first .i-file in the sequence. If the file name is standard (either IP-names with numbers before the dot, or CCD-convention with numbers after the dot), **mar Post** constructs the template and increments the image number; by default it finishes if the next incremented file name does not exist (or cannot be opened).
- o If in doubt about the standard, enclose the image number in pointed brackets rendering it a template; if `unix / linux`, this whole construct must be enclosed in quotes because otherwise the shell will interpret the `<` and `>` differently.
- o If **mar Post** is to stop at an earlier image number, specify it on the command line.
- o Here are equivalent calls to sum images 526 through 809 (where 809 is the last processed image on this directory):

```

> marPost trp_ham2_01_0526.i
> marPost trp_ham2_01_0526.i 809
> marPost trp_ham2_01_0526.i 526 809
> marPost 'trp_ham2_01_<0526>.i' 809
> marPost 'trp_ham2_01_<0001>.i' 526 809
> marPost trp_ham2_01_####.i 526 809

```

- o Assume that in the same example image 0623 be corrupted or non-existent:
 - > *marPost* trp_ham2_01_0526.i 622 624 809
 - > *marPost* trp_ham2_01_0001.i 526 622 624 809
 - > *marPost* 'trp_ham2_01_<0526>.i' 622 624 809
 - > *marPost* 'trp_ham2_01_<0001>.i' 526 622 624 809
 - > *marPost* trp_ham2_01_####.i 526 622 624 809
- o The same result could be achieved with a parameter file like this:
 - > *marPost* marPost.ctrl
 - with marPost.ctrl :
 - | data trp_ham2_01_####.i
 - | image 526 622
 - | image 624 809

mar Post with control parameter file may also have options and image sequences on the command line, but the latter should not be in conflict with any "image[-range]" parameter definition, otherwise the result is unpredictable.

- o In contrast to the previous programs, different file-names on different directories, of different runs or even crystals are allowed, each one with up to 5 blocks of images as defined by repeated "image[-range]" with one sequence each, given by first and last number.
- o Previous processing parameters may be replaced (cell, space group)
- o For each definition of up to 20 file name templates, crystal or run number may be replaced, or the numbering be altered by an offset ("add").
- o Note the difference between command-line sequences and "image" keyword if "add" is specified: "image" refers to the input numbering. In the following example, anomalous data had been collected, with name A_001 through A_070 starting from PHI=0, and B_001 through B_070 starting from PHI=180. To avoid confusion and/or duplication of numbers, command-line sequences must give the output numbers:

<pre> > marPost S.ctrl template A_###.i images 1 70 template B_###.i images 1 70 add 180</pre>	<pre> > marPost S.ctrl 1 70 181 250 with S.ctrl reading: template A_###.i template B_###.i add 180</pre>
--	--

6.2 Parameter File

The file-name on the command-line is interpreted as control parameter file if either its file extension is '.ctrl' or if it is preceded by an @ sign.

6.2.1 Valid keywords and parameters

Up to 20 file names may be specified, each with an optional 'add' number and up to 5 image ranges. This makes a total of 100 sequence sections. If 'image-range' is not specified for any one file-name, the input will start from the given file-name, then increment its image number and continue until the incremented file does not exist.

Any crystal parameters specified will replace the corresponding header values of the input files.

Only one crystal parameter set of cell and space group is accepted. The crystal|run number may, however, optionally be given for each file name to replace the header value.

6.2.1.1 Crystal parameters

```
unit cell
space group
crystal | xtal | run
```


As long as a single template file name can describe all partial sequences, this option may substitute the image keyword which specifies the first and last image number of one image section per line (also up to 5 per template).

This option is also supported by the *Join* – Edit menu of the *automar* GUI.

6.5 Input and Output Files

6.5.1 The optional background-file (input)

mar Post tries to open a file named 'background'. If it exists and suits one of the acceptable formats, it is used for pre-scaling of reflexion intensities and sigmas, in particular of partials before they are summed.

The basic idea is that variations in primary beam intensity are visible as variations of the average image background. If these variations are smooth enough to affect all reflexions of one exposure in (nearly) the same way, the inverse is a suitable pre-scale factor.

This condition of smooth variation may, however, not be fulfilled, e.g. in the case of X-ray generator glitches that affect only part of the image's rotation range.

mar Post does not, therefore, read the background log-file by default. It is up to the user to define 'background' - either by a logical name in VMS, or as an environment parameter in unix/linux or (better) by a soft link to the background log-file, or (better still) by the `back[ground]]bgr` keyword.

Cf. also description of the [background log-file in section 5.6.5](#) above.

6.5.2 The summary-file

Apart from self-explanatory tables that summarize all input and output, two sections of output print the results of independent refinements.

- (1) The fit of mosaicity and divergence from extensive profiles is described in the [introduction to *mar Post* above](#).
- (2) Post-refinement of crystal cell and orientation. As described above, alternating refinements are based on the set of [phi1,2,3 refinements \(to define the CHI=0 direction\) and on the Taylor- expanded reflecting condition](#) for all partial centroids.

6.5.3 The .Fsq-file

Summation of partials is independent of subsequent post-refinement. It is for this reason that the historical default output file name has been maintained as "SumParts.Fsq".

The header is similar to the .i-files, with values updated by the post-refinement. The data columns are headed

```
h k l source-id I sigma(I) PHI X Y
```

where the source-id has its own description in the header; it is made up from crystal/run-number, lattice-number, partials flag, detector number, and running exposure number.

The lattice-number equals 1 unless superimposed lattices had been defined by double-wavelength, etc.

The partials flag is 0 for fully recorded reflexions; summed partials have usually flag 5.

If more than 1 and up to 4 detectors are exposed simultaneously, the partials flag is incremented by the detector number. (A different code is envisaged if there be more than 4 detectors; it may be implemented upon request.)

The running exposure number is possibly modified by the add keyword.
Note that partials are assigned the exposure number whose rotation range contains the fitted centroid. The partials flag makes the difference.

The example "12.1_5_0184" denotes a reflexion from crystal (or run) #12

- . which has only 1 lattice (no split, etc.);
- it is a summed partial (5, + standard detector 0)
- with its PHI centroid in the rotation range of exposure number 0004,
which was modified by the "add 180" keyword to become **0184**.

▶ I & sigma(I) are either the original values passed from the .i-file for fully recorded reflexions (possibly pre-scaled) or the sum of partials and their root-mean-square sigma.

▶ PHI (centroid in deg.; = result of fit for partials, or predicted for full)

▶ X,Y detector coordinates (in [mm] from the beam centre)

```
FORMAT (3I4, 1X,A12, F12.1,F10.1, F8.3, 2X,2F7.1)
```

6.5.4 The optional .F-file

▶ h,k,l, F, DERI,SOURCE, sigma(F), PHI(found), PHI(predicted), X,Y

```
FORMAT (3I4, F9.2, 1X,A4,A9, F8.2, 2F8.3, 2X,2F7.1)
```

This file is written as a scratch file for post-refinement, too, both because of its sort order and the double PHI centroid. It is deleted unless service PROTEIN has been specified.

7. Scaling: *mar Scale*

7.1 Introduction

- This program is still under development.
- At present it does not by itself scale and add partials. Partial must rather be added before hand by *mar Post*, the output of which flags summed partials. *mar Scale* assigns separate scale factors to fully and to partially recorded reflexions.
- The *default* workspace can store 1 million reflexions measured on up to 900 exposures in up to 50 runs / from 50 different crystals.
- These figures are dynamically expanded according to the needs of the input data, to more than 4 million reflexions from 2000 exposures with fully and/or partially recorded reflexions on up to 5 detectors. The 2000 x 2000 matrix inversion will take its time, though. (Just by different program parameters, limits needed for the sorting, 5000 exposures on 2 detectors, etc., could be realized.)
- (*mar Scale* can cope with data collected on up to 5 detectors simultaneously. This feature is not, presently, supported by *mar* hardware, but will be of interest with the development of *spherical marCCD-detectors*.)
- *mar Scale* reads and evaluates the header information of the *marPost.Fsq* file; additional input may be specified by a keyworded parameter file which would take priority over the header.
- If a non-*marPost* input file without header information is used, and if no parameter file is given or vital numbers are amiss (cell, spacegroup), then these are prompted for.
- All 230 spacegroups and 28 non-standard settings are encoded within the program in CCP4 compatible form. No separate library file is needed.
- B-factors are not assigned to each source, nor even to a range of sources, individually since the decay of crystals is a monotonous process without ups and downs. Also the variation in absorption - which in uncorrected data sets is to some extent compensated by a variation of the B-factor - is a smooth function. *mar Scale* refuses, therefore, to improve R-factors at the expense of physically meaningless jumps in the temperature factor. Rather it offers a variety of analytical functions, both of time and of the rotation angle PHI, and fits their coefficients.
 - "Time" is represented, as well as possible, by the exposure number; these functions are aimed at the crystal decay.
 - Functions of PHI (using the centroid of the rocking curve) are aimed at absorption.
 - If detector coordinates and sample-to-detector distance are given (default in *automar* data), the fitted "absorption surface" takes into account not only the image's PHI setting but the angular coordinates of all individual reflexions.
- Scale factors can be chosen to be individually assigned to each source, and then possibly be smoothed by a digital filter, or be approximated by analytical function coefficients as well.
- Functions of the detector coordinates (correction of detector efficiency), or spherical harmonics to describe a 3-dimensional "absorption surface" (by combination of X,Y,PHI) might easily be implemented if necessary.

7.2 Method

- Linear least squares for the logarithm of intensities versus some reference intensity for each hkl, which may come from an external reference data set, or which may be the (scaled) mean value of current measurements. The logarithm renders scale [and temperature] factors additive

constants, at the expense that effectively not the arithmetic but the geometric mean determines the scale factors.

- *Note:* The logarithm implies that only positive intensities can participate in the scaling procedure. It is in any case advisable to include significantly measured reflexions only (default: $I > 3 \sigma$). Nevertheless are all measurements (i.e. also negative net I) included in the calculation of the final (arithmetic) mean intensity.
- More details can be found in the parameter / options description 7.1.2

7.3 Usage

```
> marScale @Parameter.file [options]
      ◀---6.1---▶◀-6.2-▶
or : > marScale '.Fsq-file' [options]
```

- The simplest way of calling **marScale** is

```
> marScale marPost.Fsq
```

All parameters will be inferred from the marPost.Fsq file header; in particular are the B-functions derived from crystal symmetry:

The 1- and 2-fold symmetry (360 & 180 deg. periodicity) are always assumed in addition to the following higher symmetries.

The 4-fold symmetry (90 deg. periodicity) is additionally used for tetragonal or cubic space groups.

The 6-fold symmetry (60 deg. periodicity) is used instead (for hexagonal) or in addition (for cubic space groups).

The 3-fold symmetry (120 deg. periodicity) is used in addition to the 6-fold for trigonal crystals which may have a triangular cross section.

- Many more options are available by way of parameter (control) file, both different restrictions and modes of scaling.

7.4 Parameter File

The file-name on the command-line is interpreted as control parameter file if either its file extension is '.ctrl' or if it is preceded by an @ sign.

7.4.1 Valid keywords and parameters

7.4.1.1 Crystal parameters

```
unit cell
space group
crys[tal] | xtal | run
```

7.4.1.2 File names

```
data | file | name
list | out
incl[ude] | @
back[ground] | bgr
```

7.4.1.3 Input/output format

```
format
service | select
..sources
```

7.4.1.4 Options

ano
 norm
 noref
 smooth
 resol[ution]
 dist[ance]
 wave[length]
 acc[ept] | incl[ude] | rej[ect]
 ice | ring
 thres[hold]
 refer[ence]
 mult[iply]
 mat[rix]
 deri[vative]
 Wilson
 log[-level]
 fin[ish] | end | eof

7.4.2 Description of keywords, parameters and options

unit cell
space group
data | file | name

} description → see [APPENDIX](#)

unit cell & space group override previously used parameters (as saved in *mar Post* header information). They are mandatory if other input formats are used.

The remainder is non-standard in the *automar* context since most of these keywords would make no sense elsewhere. They are therefore described here and not in the APPENDIX.

>>> output files:

- list | out file-name
 - ▶ This is in the first place the file name for the output listing.
 - ▶ Its default is the data file name (e.g. marPost.Fsq) with the extension ".marScale" (e.g. marPost.marScale)
 - ▶ All other output files, standard or optional, are constructed from this file name by replacement of the file extension:
 - standard:
 - xx.unmerged for output of individual measurements with original hkl;
 - both scaled and unscaled data;
 - coordinates (X,Y,PHI), S and B factors;
 - scaled sigma according to variance fit
 - xx.asu for individual measurements with hkl transformed into asymmetric unit
 - with symmetry operator number as listed in the .marScale listing;
 - both scaled and unscaled data (no omissions due to resol keyword)
 - xx.mean for scaled mean intensities & sigmas with hkl in the asymmetric unit;
 - Bijvoet +/- and /- for non-centric reflexions.
 - optional:
 - xx.log (if log-level > 0 for additional output)
 - xx.cmp (if log-level > 0 and reference data)
 - xx.df anomalous delta-F (SHELX format)
 - xx.rfl for output of scaled mean <F_hkl> (PROTEIN format)
 - xx.scl for output of scaled mean <I_hkl> (SCALEPACK format)
 - xx.shelx unmerged scaled measurements in SHELX format

>>> input data formats:

- format (FORTRAN format string) ... enclosing brackets are mandatory.
 - ▶ This keyword may be used after each input data file name and/or after the optional "refer[ence]" keyword:
 - (1) to describe input data format (for non-standard data files).
 - Special definition: "format **SHELX** [**I**]"
 - **mar Scale** has the SHELX format as (3I4,2F8.2,I4)
 - It reads h,k,l, F,sigma(F) [, num]
 - [or I,sigma(I), if "SHELX I "]
 - If **I** is not specified, intensities and sigmas are calculated from the structure factors (not recommended, not by G.Sheldrick either).
 - num = exposure number is optional. If not given, no scaling can be done; but outlier rejection, Wilson- and R-factor statistics are in effect.
 - (2) subsequent to "refer[ence]" keyword to describe the reference data input file format of non-standard <I>-files.
- mar Scale** recognizes its own 'mean' format. Otherwise the default is SCALEPACK mean format (3I4,4F8.1). (In both cases the format need not explicitly be specified.)
- mar Scale** reads h,k,l, I[+/-], sigI[+/-, I/-, sigI/-/)] for centric [non-centric] reflexions.
- Sour[ces] number_of_sources
 - ▶ This keyword has been implemented for non-automar input files only. It is needed in very few cases where a large number of images (=> sources) is unmatched by the number of measurements (such that the automatic allocation of internal workspace is insufficient). **mar Scale** will warn of such a situation, indicating the required number, and stop.

>>> scaling modes:

- ano (no subkeys or numbers)
- norm[a1]
 - ▶ Scale anomalous I/+ and I/- as independent values.
 - This will usually decrease the R-factors because the number of independent variables increases nearly by a factor of 2 - it will not necessarily improve the data quality, though, unless anomalous differences can be expected to be considerable and the remnant multiplicity of each I/+ and I/- is still good.
 - ▶ The default is "norm" :
 - ... assume that anomalous differences are negligible: I/+ = <I> = I/-
- refer[ence] reference_file_name \ none
 - ▶ Data may be scaled not for internal consistency but against another "reference" data set - e.g. native data for the scaling of a derivative.
 - ▶ See also the "format" keyword.
- noref[ine] exposure_number
 - ▶ This is an obsolete way to remove the singularity of internal scaling (= without external reference data) of both S and B which is due to the missing absolute scale and B factor.

- ▶ This input is still allowed but not maintained.
- ▶ **mar Scale** applies [eigenvalue-filtering](#), but it writes a WARNING if more than 1 (if S alone), or 2 (if S and B are determined) eigenvalues are filtered since it may mean that regions exist in reciprocal space that have no cross-correlation with any other region. Still this approach gives the best estimate and is probably a good answer, if these isolated regions have been measured under similar conditions as the rest (exposure time, primary beam intensity, etc.)
- ▶ Eigenvalues filtered may also indicate that the selected or default functions are highly correlated for the total rotation range available.
Example:
Phi = 0 - 25 deg. => the linear function of time (i.e. of numbers) is nearly identical with the sine of 1•PHI and 2•PHI).
Still the filtered result is the linear combination of those functions that fits the data best.

- **noref**[ine] B [**except** # aaaa:fff]

- **ref**[ine] B [sym[metry] a [a ..]] [fun[ction] f [f ..]]

- ▶ "noref B" will suppress determination of relative temperature factors.
- ▶ The optional digits after the "[except] #" sign tell the program to which of the possible functions the B-factor shall be restricted.
- ▶ Alternatively, "symmetry" and "function" sub-keywords define B-functions.
- ▶ For a description of functions, see "[smooth](#)" option
- ▶ Cf. previous paragraph (noref num) for a discussion of [correlation and eigenvalue-filtering](#).

- **smooth** [# [aaaa:]ffff | [sym[metry] a [a ..]] [fun[ction] f [f ..]] \ *digital filter*]

- ▶ The default, i.e. "smooth" without "#" etc. invokes a *digital filter* (5 point 2nd order Savitzky-Golay filter which is a very conservative measure).
- ▶ Otherwise (with "# [digits:]digits" or with "sym ... fun .."): Do not determine individual scale factors for each exposure. Instead approximate each run by selected analytical functions (determining the coefficients).
- ▶ [5 functions f of time](#) (i.e. of exposure number n) are probably overkill. Anyway. They are expected to describe the crystal decay, but may for example also reflect the decreasing synchrotron beam, in which case the appropriate assignment of run numbers is crucial:

1. lin[ear]	n	p[roportional]
2. square-root	\sqrt{n}	r[oot] sqrt
3. s[quare]	n^2	
4. l[ogarithm]	log(n)	
5. rec[iprocal]	1/n	i[nverse]

(The first 3 only are implemented for B.)

- ▶ These functions are user-selectable by the binary representation :ffff where f = 0 or 1, and the bit position is the above running function number.
- ▶ Empty space to the right of the colon is equivalent with ":0" = no time-dependent functions.
- ▶ More conveniently now, "*function prsl*" or "*function lin|pro sqrt|root squ log rec|inv*"

Examples:

"smooth # :1" = "smooth func lin" selects the linear function only;

"smooth # :1101" = sum of linear, square and logarithmic dependence on time (from right to left); this construction may describe a sigmoidal curve.

- ▶ [5 x 2 angular functions a](#) are implemented which are crystallographically justifiable: 1-, 2-, 3-, 4- and 6-fold with sine and cosine each. These are expected to be a description of long-range intensity variations due to absorption. (Mind that both the

primary beam and the reflected rays are attenuated on their paths through the crystal.)

- These functions are user-selectable in addition to, or alone without, the above time-dependent functions, by the analogous binary representation "# aaaaa:", left of the colon which is mandatory here. Each bit position determines the symmetry number (except for the 6-fold at bit 5, e.g. "# 10000:" = 6-fold alone.
 - ▶ Empty space to the left of the colon is equivalent with "0:" = no angular functions. In this case, the colon is optional.
 - ▶ For these angular functions it is possible to make the input easier, more mnemonic, giving the symmetry number(s) as digits (in any order).
- Alternatively, use sub-keywords "sym[metry] one[fold] two... three... four... six[fold]" ... or simply "sym 12346".

Examples:

"smooth # 1:" = 1-fold only, to describe the absorption the mother liquor.

"smooth # 1011:" = 1-, 2- and 4-fold, which may describe a square cross section of a tetragonal crystal with its 4-fold along the rotation axis, with a mother liquor droplet.

- ▶ same as "smooth # 4021:"
- or even "smooth # 124:"

- ▶ A constant S [and B] each are fitted if there are more than one run or crystal. (One at least each is filtered.)
- ▶ Added partials are treated as a different source, as much as scale factors are concerned. Their B-factor is taken the same as for fully recorded reflexions except for an additive constant delta_B, "dcpart".
- ▶ Cf. previous paragraph ('noref num') for a discussion of [correlation and eigenvalue-filtering](#).

>>> modify input data:

- crys[tal] | xtal | run n
 - ▶ optionally re-name the intensities from the preceding input file to make them distinct from other input data in the scaling procedure.
- dist[ance] dist
 - ▶ optionally correct, or supply missing, sample-to-detector distance to enable absorption surface fit. Only useful if input data file holds detector coordinates of each reflexion as well.
- mult[iply] factor \ 1.0
 - ▶ optionally pre-scale the intensities from the preceding input file or the reference data before their use in the scaling procedure.
- mat[rix] m₁₁ m₁₂ ... m₃₃
 - ▶ optionally re-index = transform hkl indices (immediately upon input)

>>> other options:

- acc[ept] limit_I_over_sigma \ 2.3 | sat[urated]

- ▶ `limit_I_over_sigma` : measurements are accepted (i.e. not further checked for outliers) if their intensity deviates from its hkl mean value by less than this times the sigma (r.m.s.) of the mean value. - Default 2.3
- ▶ `sat[urated]` : reflexions flagged by $\sigma(I) < 0$ are excluded from the scaling process, but then they are treated like any other measurement and possibly checked if they are outliers.
- ▶ *REJECTION ALGORITHM STILL INCOMPLETE !!!*
- `inc[lude]` `limit_I_over_sigma \ 2.3 | sat[urated]`
 - ▶ synonymous with `accept`
- `ice`
 - ▶ hkl reflexions which coincide in their d-spacings with ice rings are excluded from the scaling process. Later in the outlier rejection step, in case of doubt, the strongest intensities are rejected since they are assumed to be superimposed with ice spots.
- `rej[ect]` `limit_I_over_sigma \ 2.3 | sat[urated]`
 - ▶ Reject **saturated** reflexion intensities (i.e. flagged by $\sigma(I) < 0$), both in the scaling and merging, no matter how well they may fit
 - ▶ otherwise synonymous with `accept` which is a more appropriate word for the behaviour
 - ▶ *REJECTION ALGORITHM STILL INCOMPLETE !!!*
- `resol[ution]` `[d_min d_max (in [A])] [shells n_shells \ 10] [keep \ omit]`
 - ▶ Omit reflexions from the scaling algorithm unless they are between these resolution limits.
 - ▶ Omit reflexions outside these resolution limits from the output files (except .asu file) unless the `keep` keyword is specified.
 - ▶ Alternatively, or in addition to `d_min` & `d_max`, the number of resolution shells for R-factor statistics may be modified; default 10, maximum 50
- `ring` `d1 [d2]`
 - ▶ same strategy as `ice`, but define d-spacings for other powder rings. For the definition, see the APPENDIX ("`exclude rings`").
- `thres[hold]` `[I or Fsq I_minimum \ 1.] [sig greater_than \ 3.]`
 - ▶ Both modes may be used together in the same **mar Scale** run by repetition of the keyword on different lines.
 - ▶ (1) `I` (or `Fsq`) as a subkey defines a fixed lower threshold for intensities used in the scaling:

$$\text{reflexion_intensity} > I_{\text{minimum}}$$
 Note that this threshold is applied to input intensities before scaling.
 Note also that only positive intensities ($> 1.$) are usable in the scaling algorithm (cf. scaling method, section 7.1) such that $I_{\text{minimum}} < 1$ is rejected.
 - ▶ (2) `sig` as a subkey defines a multiplier of each intensity's sigma to define a variable threshold

$$\text{reflexion_intensity} > \text{greater_than} \cdot \text{reflexion_sigma}$$

- deri[vative] DERI (4 characters) \ 'NATI'
 - ▶ special to "*service PROTEIN*" which writes a file (xx.rfl) that is suited for the LOAD step. The PROTEIN system keeps all reflexion data in one big "reflexion-file" where all native or derivative data sets are marked by their 4-character name.
- Wilson I_000 [W_shells \ 20]
 - ▶ special condition for scaling ▶▶ not functional yet!
!!!!!!!!!!!!!!!!!!!!
 - ▶ The idea is to include Wilson-statistics' absolute scale [and B] factor to remove the singularity in internal relative scaling. The output intensities should then be on a (more or less) absolute scale.
- log[-level] level \ 0
 - ▶ write extra information to *xxx.log*
 - ▶ = 1 : Assignment of source indices;
Listing of equivalent hkl with $|I_j - \langle I \rangle| > 10\%$ (of unscaled I_j and their $\langle I \rangle$);
Report on eigenvalue-filtering;
Coefficients of B- [and S-] functions.
 - ▶ =2,3 : **•very•** extensive and unreadable debug stuff.

7.5 Command Line Options

7.5.1 -vp ("verbose")

prints more details to the screen and the listing.

7.5.2 -s ("service")

1. input data formats: *marScale* (i.e. **mar Scale** output recycled)
(*SHELX* is available by way of "**format**" keyword)
Other formats will be implemented when **mar Scale** learns to scale and add partials.
Default is **mar Post** output
2. output data formats: PROTEIN
SCALEPACK
SHELX
Default is *marScale* (merged intensities)

7.6 Input and Output files

7.6.1 The reflexion input

automar parameters are read from the header of reflexion intensity files; concatenated files (thus with repeating headers) are allowed. ▶▶ The present version does not accept .i-files but only [marPost.Fsq], or also [marScale.asu or .unmerged].

Up to 20 input data files are accepted. A single file name may be given on the command line, or alternatively be specified by "data" keyword (or alternatives) in a control parameter file.

7.6.2 The *mar* Scale listing file

Its name is specified by the "list" keyword; it defaults to the input data file name with extension '.marScale'; e.g. "data marPost.Fsq" implies default 'list marPost.marScale'

The listing reflects all input, both the file names with accepted input header parameters and optional input or active defaults.

► The only section that is not obvious from previous activities is the table of symmetry operators, sorted in Bijvoet positive and negative columns, as they are encoded in the program for all space groups.

The numbers of these symmetry operators are used to flag hkl after they have been put into the asymmetric unit (see for example [marS].asu file).

► Also, input or default function coefficients may be checked here (see above description of "[noref B \[except\] #...](#)" and "[smooth #...](#)")

A table summarizes all input reflexions and rejections.

The (unweighted) R-value of unscaled intensities is printed. This will only be reasonable if all intensities are measured at, or pre-scaled to, constant primary beam intensity (pre-scaled e.g. by means of background).

Results of scaling procedure:

- Possibly warning on [filtered eigenvalues](#) (cf. "[noref num](#)" above).
- Coefficients of [analytical functions](#) (B, possibly "[smooth](#)" S).
- B (temperature) and S(scale) factors each for full and summed partial intensities, for all sorted exposure numbers, wherever determined.
- Wilson-plot of scaled data
- Fit of experimental variances
 - Explanation:

The input $\sigma(I)$ is assumed to be purely Poissonian on the basis of 1 ADC unit (pixel count) per X-ray quantum. This is true for **mar Process** integration; for a discussion of a possible gain factor, see [section 5.6.2 above: mar Process log-file, "histogram of gain"](#). Experience shows that surplus systematic errors are more or less proportional to the measured intensity. It is, therefore, reasonable to assume that the actual variance (i.e. squared sigma) of multiple measurements can be described by

$$\sigma(I) = g \cdot \sigma(I|\text{Poisson}) + f \cdot I$$

where g = gain, and f = fraction of I to describe systematic errors. (If the primary assumption be wrong and input sigmas be adequate to describe the variances, then this fit ought to yield $g=1$ & $f=0$, which is also ok.)
 - A table summarizes, in intensity shells as derived from the Wilson distribution:
 - the number of contributing hkl;
 - their average intensity and r.m.s. sigma as derived from input sigmas;
 - the r.m.s.-sigma found from the variance, and
 - their mean absolute deviation.
 - It shows by how much the compound input sigma would have to be increased (quadratically) to make up for the variance, and how much it would be in terms of systematic error fraction f alone (assuming $g=1$).
 - Resulting g and f are applied to all $\sigma(I)$ to yield the appropriate output $\sigma(\langle I \rangle)$
- 2-dim. table (intensity shell vs. resolution shell) of R-factors and number of contributing hkl.
 - This table is divided in four to fit 20 resolution shells on the print output. Each column is summed to give the R-factor of each resolution shell for all intensities (..presently printed for -vp option only).

► Finally the "horizontal" sums are printed to give R-factors of each intensity shell independent of resolution. This last column is accompanied by a graph of the reflexion numbers in each shell, both differential and incremental. As mentioned above, the shells are derived from Wilson statistics. If there were no $\sin\theta/\lambda$ dependence, one would expect two straight lines, according to constant numbers of reflexions per shell. As it stands, the shells are a compromise between the different resolution ranges.

```

@ s-shell :    0.000      0.050      0.071      0.087      0.100      0.112
  resol.[A]:    |          9.98      7.06      5.76      4.99      4.46
Fsq-shell :    --|--      --|--      --|--      --|--      --|--      --|--
< 134816.    |  4.%  547|  3.%  875|  3.%  608|  3.%  816|  3.% 1396|
<  8663.     |  8.%  146|  4.%  286|  4.%  426|  3.%  511|  3.%  593|
<  5431.     |  8.%   97|  4.%  215|  4.%  256|  4.%  361|  4.%  343|
<  3928.     |  9.%   45|  5.%  160|  5.%  189|  4.%  317|  5.%  309|
<  2938.     | 11.%   46|  6.%  152|  5.%  263|  5.%  236|  5.%  235|
<  2198.     | 20.%   34|  7.%  111|  6.%  204|  5.%  271|  7.%  222|
<  1608.     |  7.%   41|  8.%   93|  7.%  253|  8.%  168|  8.%  166|
<  1117.     | 18.%   51|  9.%   95| 10.%  181| 11.%  157| 10.%  146|
<   695.     | 25.%   39| 15.%   78| 15.%  156| 14.%  175| 16.%  117|
<   327.     | 48.%   60| 24.%   98| 31.%  208| 34.%  202| 36.%  135|
sum(I_shells) --|--      --|--      --|--      --|--      --|--      --|--
< 134816.     |  5.% 1106|  3.% 2163|  4.% 2744|  3.% 3214|  3.% 3662|
                | 0.11 0.01| 0.06 0.00| 0.10 0.02| 0.08 0.02| 0.08 0.03|
    
```

```

@ s-shell :    0.112      0.123      0.133      0.142      0.150      0.158
  resol.[A]:    |          4.08      3.77      3.53      3.33      3.16
Fsq-shell :    --|--      --|--      --|--      --|--      --|--      --|--
...
...
    
```

```

@ s-shell :    0.000      0.224
  resol.[A]:    |          2.23
Fsq-shell :    --|--      --|--
< 134816.    |  3.% 8792| : %
<  8663.     |  4.% 6270| : + %
<  5431.     |  5.% 5015| : + %
<  3928.     |  5.% 5012| : + %
<  2938.     |  7.% 5342| : + %
<  2198.     |  9.% 6310| : + %
<  1608.     | 12.% 6932| : + %
<  1117.     | 16.% 8893| : + %
<   695.     | 23.% 15130| : + %
<   327.     | 40.% 22962| : + %
sum(I_shells) --|--      --|--
< 134816.     |  6.% 90658|
                | 0.37 0.20|
    
```

► It is the weakness of the R-factor that it is dominated by the strong reflexions. Whether or not weak spots agree well, does not matter much. This is different for the average relative error.

While the

$$R\text{-factor} = \frac{\sum \{ dI \}}{\sum \{ I \}}$$

for all multiple measurements, the

$$\begin{aligned} \text{average relative error} &= \frac{\sum \{ dI / I \}}{N} && \text{(signed)} \\ \text{or} \quad \sum \{ |dI| / I \} &= N && \text{(absolute)} \end{aligned}$$

The bottom line of each column of the table lists also the average absolute and signed relative error.

These numbers are additionally printed *for each individual I-shell* if command-line option `-vp` (verbose protocol) is specified. Then the same table looks like this, less well readable but more informative:

```

@ s-shell : 0.000      0.050      0.071      0.087      0.100      0.112
  resol.[A]: |          9.98      7.06      5.76      4.99      4.46
Fsq-shell : --|--      --|--      --|--      --|--      --|--      --|--
< 134816.  |  4.%  547|  3.%  875|  3.%  608|  3.%  816|  3.% 1396|
              | 0.04 0.01| 0.03 0.00| 0.03 0.00| 0.03 0.00| 0.03 0.00|
<  8663.   |  8.%  146|  4.%  286|  4.%  426|  3.%  511|  3.%  593|
              | 0.08 0.02| 0.04 0.00| 0.04 0.00| 0.03 0.00| 0.03 0.00|
<  5431.   |  8.%   97|  4.%  215|  4.%  256|  4.%  361|  4.%  343|
              | 0.08 -0.01| 0.04 0.00| 0.04 0.00| 0.04 0.00| 0.04 0.00|
<  3928.   |  9.%   45|  5.%  160|  5.%  189|  4.%  317|  5.%  309|
              | 0.09 -0.01| 0.05 0.00| 0.05 0.00| 0.04 0.00| 0.05 0.00|
<  2938.   | 11.%   46|  6.%  152|  5.%  263|  5.%  236|  5.%  235|
              | 0.11 -0.03| 0.06 0.01| 0.05 0.00| 0.05 0.00| 0.05 0.01|
<  2198.   | 20.%   34|  7.%  111|  6.%  204|  5.%  271|  7.%  222|
              | 0.22 0.06| 0.07 -0.01| 0.06 0.01| 0.06 0.00| 0.07 0.00|
<  1608.   |  7.%   41|  8.%   93|  7.%  253|  8.%  168|  8.%  166|
              | 0.07 0.00| 0.08 0.01| 0.07 0.00| 0.08 0.01| 0.09 0.01|
<  1117.   | 18.%   51|  9.%   95| 10.%  181| 11.%  157| 10.%  146|
              | 0.20 0.07| 0.09 0.00| 0.11 0.00| 0.11 0.01| 0.11 0.00|
<   695.   | 25.%   39| 15.%   78| 15.%  156| 14.%  175| 16.%  117|
              | 0.28 0.08| 0.16 0.02| 0.16 0.02| 0.15 0.02| 0.18 0.06|
<   327.   | 48.%   60| 24.%   98| 31.%  208| 34.%  202| 36.%  135|
              | 0.55 -0.09| 0.31 0.02| 0.62 0.26| 0.56 0.27| 1.02 0.72|
sum(I_shells) --|--      --|--      --|--      --|--      --|--      --|--
< 134816.  |  5.% 1106|  3.% 2163|  4.% 2744|  3.% 3214|  3.% 3662|
              | 0.11 0.01| 0.06 0.00| 0.10 0.02| 0.08 0.02| 0.08 0.03|

...
...

@ s-shell : 0.000      0.224
  resol.[A]: |          2.23
Fsq-shell : --|--      --|--
< 134816.  |  3.% 8792| : %
              | 0.03 0.00|
<  8663.   |  4.% 6270| : + %
              | 0.04 0.00|
<  5431.   |  5.% 5015| : + %
              | 0.05 0.00|
<  3928.   |  5.% 5012| : + %
              | 0.06 0.01|
<  2938.   |  7.% 5342| : + %
              | 0.07 0.02|
<  2198.   |  9.% 6310| : + %
              | 0.09 0.03|
<  1608.   | 12.% 6932| : + %
              | 0.13 0.04|
<  1117.   | 16.% 8893| : + %
              | 0.18 0.06|
<   695.   | 23.% 15130| : + %
              | 0.30 0.15|
<   327.   | 40.% 22962| : + %
              | 1.08 0.64|
sum(I_shells) --|--      --|--
< 134816.  |  6.% 90658|
  
```

- The final R-values are introduced, if appropriate, by relative consistency values w.r.t. reference data.

Thereafter, or otherwise :

R-values of internal consistency, including their formulae; both the "normal", and the anomalous R-values of <I+> vs. <I->, are printed, both without and with sigma-weights.

- The bottom tables give resolution-dependent statistical information: **multiplicities** and **completeness** of measurements, and various **R-values**, both anomalous and normal and for centric reflexions alone, along with their definitions. In addition to columns with the conventional linear (R_lin.) and quadratic (R_squ.) R-values, *mar* Scale computes multiplicity-weighted PCV, R_rim & R_pim.

For reference, see

[1] M.S.Weiss, J.Appl.Crys.(2001) **34**, 130-135

[2]K.Diederichs & P.A.Karplus, nature struct.biol. (april 1997), vol.4 nr.4, p.269-275

If option `-vp` is specified, a non-standard square-root R-value, R_sqrt., is given, too. Mind that the quadratic R is dominated by the strong intensities. The square-root R gives implicitly more weight to weak reflexions:

Table of s-dependent R-factors in equal-volume shells :

```
R_linear = sum { |I - <I>| } / sum {I}
R_square = SQRT( sum{|I - <I>|**2} / sum{I**2} )
R_s.root = ( sum{ SQRT(|I - <I>|) } / sum{F} )**2, for I>0
(Single measurements or negative <I> omitted from all sums.)
```

Shell limits [A]	s	contributions	<I>	R_lin.	R_squ.	R_srt.	(I>0)
inf.-4.81	0.000-0.104	10405	7487.	3.63 %	4.26 %	3.47 %	10356
4.81-3.82	0.104-0.131	10300	8267.	3.46 %	3.62 %	3.39 %	10237
3.82-3.33	0.131-0.150	10152	4602.	4.60 %	4.58 %	4.71 %	10015
3.33-3.03	0.150-0.165	9953	2539.	6.63 %	6.16 %	6.98 %	9748
3.03-2.81	0.165-0.178	9736	1637.	9.03 %	8.00 %	9.54 %	9483
2.81-2.65	0.178-0.189	9530	1179.	11.63 %	10.31 %	12.02 %	9209
2.65-2.51	0.189-0.199	9271	800.	15.89 %	13.74 %	15.99 %	8824
2.51-2.40	0.199-0.208	9024	650.	19.10 %	16.87 %	18.67 %	8574
2.40-2.31	0.208-0.216	8543	557.	22.04 %	19.39 %	21.19 %	8063
2.31-2.23	0.216-0.224	6574	452.	26.14 %	23.04 %	24.21 %	6149
total :		93488	3030.	5.81 %	4.61 %	7.09 %	90658
579 non-pos.<I> with		2079	measurements	omitted.			

7.6.3 The output intensity files

7.6.3.1 The mean intensity file (.mean)

(1) header with identifier, cell, and space group;

(2) h k l * <I> sig<I> [* <I+> sig<I+> * <I-> sig<I->]
 FORMAT (3I4, 3(1X,A3,F11.1,F9.1))

where * represents an identifier:

/0/ for centric reflexions;

<I> for mean of acentric reflexions; with

/+ / for Friedel /+/,

/- / for Friedel /- /

If either /+ / or /- / are not measured, this field has I=0., sigma=-1.

7.6.3.2 The asymmetric unit file (.asu)

(1) header with identifiers and relevant automar parameters.

(2) h k l sym source scaled(I, sigI) PHI,X,Y S,B unscaled(I, sigI)

FORMAT(4I4, 1X,A12, F11.1,F9.1, F9.3,1X,2F7.1,1X, 2F8.3, F11.1,F9.1)

All hkl are sorted into the asymmetric unit and printed with the applied symmetry operator (even numbers are *l-l*, odd are *l+l*).

Scaled intensities and sigmas (not yet modified by variances) can thus be compared by inspection of successive lines. Source-id., coordinates (X,Y,PHI) and original I,sigI are copied from the input file.

7.6.3.3 The scaled measurements file (.unmerged)

(1) Header with identifiers and relevant automar parameters.

(2) Similar to the .asu-file; but with the following differences:

- hkl are the original ones, and the symmetry number is flagged negative, telling that this would be the operator to put hkl into the asymmetric unit.
- sigma(scaled I) is not only scaled sigma(unscaled I) but is already modified by gain g and systematic error fraction f (see above 7.3.2, "The *mar Scale* listing file": Fit of experimental variances). Also, sigma is flagged negative for measurements rejected by *mar Scale* (in addition to those already flagged in the input file, e.g. saturated spots).

7.6.4 The optional output files

7.6.4.1 service PROTEIN

Write a structure factor file [marPost].rfl suited for PROTEIN (LOAD):

h k l F_{mean} DERI SRCE sigma(F_{mean})

FORMAT (3I4, F8.2, 2(1X, A4), F8.2)

If both Friedel *l+l* and *l-l* are measured, subsequent records are written with (h,k,l) and (-h,-k,-l)

7.6.4.2 service SCALEPACK

Write mean intensities in SCALEPACK (merged) format:

h k l I_{mean} sigma(I_{mean}) (centric), or:

h k l I/*+* sigma(I/*+*) I/*-* sigma(I/*-*)

FORMAT (3I4, 4F8.1)

- ▶ Note that mean intensities can be written either in SCALEPACK (.scl) or in automar (.mean) format *only*.

7.6.4.3 service SHELX

Write unmerged intensities in SHELX (unmerged) format

h k l I_{scaled} sigma(I_{scaled}) image_number

FORMAT (3I4, 2F8.1, I4)

- ▶ Note that only the image number is written (no more space in I4 !); any run or crystal number is lost, as well as the "added partials" flag or any detector number. This may hamper re-scaling attempts in SHELX if any image number exists in more than one input data set, or if more than one lattice was evaluated (twin or split patterns)!
- ▶ Note also that all sigmas are written positive, i.e. the rejection flag is lost.

8. Data Conversion

8.1 mar2mtz

8.1.1 Usage

Usage of mar2mtz is very much alike **mar Post**, however partials are not added. Instead the whole lot of reflexion data with concomitant batch information is copied to a CCP4 mtz-file ready for subsequent

sortmtz → **scala** → etc.

Again up to 20 file names can be specified with

- optional 'add' and 'crystal' | 'run' numbers each, and
- up to 5 image ranges with PHI start and rotation range each.

All other parameters are allowed once only. Header values are copied to the batch records unless redefined by one of the keyworded parameter definitions below.

8.1.2 Valid keywords and parameters

If not commented, the keyword is described in the APPENDIX.

Input is digested by way of CCP4 routines, therefore the following syntax differs from **automar** : The essential 4 characters are given in upper case, the optional part of the keyword in lower case [not in square brackets].

Repeated input allowed:

TEMPlate	file_template	→ cf. data keyword
FILE		
IMAGe		
ADD	num_add	(difference number)
CRYStal XTAL RUN	n	(identifying number)

Input only once to override header value(s):

TITLe		
TYPE	SUMP AUTO MARS .. FSQ I F	
	.. obsolete by way of automar header code numbers	
AXIS AXES		
ORIE		
RESOLution		
CELL		
SYMMetries	space-group number or name	→ cf. space-group keyword
WAVElength		
DISTance		
BEAM CENTre		
PIXElsize		
DETEctor-limits	$x_{min}, x_{max}, y_{min}, y_{max}$	→ a warning is printed if reflexions are found outside these limits.
MONItor	n	→ list every n-th reflexion
REINdex	$x_{11} x_{12} x_{13} x_{21} x_{22} x_{23} x_{31} x_{32} x_{33}$	= 3x3-matrix to yield: (hkl) → [matrix] • (hkl)
SCALE-factor	sf	→ pre-scale all intensities: I → sf • I
UMAT	$x_{11} x_{12} x_{13}$	
UMAT	$x_{21} x_{22} x_{23}$	
UMAT	$x_{31} x_{32} x_{33}$	

END | GO —▶ stop reading input, begin conversion

8.1.3 Command line options

[hklin] input-filename (usually first .i-file)
hklout output-filename (mtz-file)

APPENDIX: Keywords in Parameter Files

A1.1 Syntax

Keywords are case-insensitive.

All keywords can be abbreviated to 4 characters, some to 3 (where indicated) but they may be written at any length as long as the entity of one keyword is recognizable. Valid separators are blanks and/or tabs.

To avoid frequent (though unnecessary) errors, some composite keywords are accepted with or without embedded dash and/or blank such as :

wavelength / wave length / wave - length / wave-length;

or similarly :

beam / beam centre / beam-centre /...

!! Here is the only exception from the abbreviation rule:

The [beamstop](#) keyword would be abbreviated to "beam" just like 'beamcenter'; it must, therefore, at least have the "stop" as keyword, i.e.:

beamstop / beam stop / beam-stop / stop
(!!!!)

A1.2 Comments in the Parameter File

... ! ./ . ! everything behind an exclamation mark, on any line, is ignored (i.e. treated as comment);
 # ./ . lines with # or * as 1st non-blank character are ignored,
 * ./ . such that comment[ed-out] lines may start with ! or # or *
 remark ... or with key-word "remark" which is assumed to terminate the key-word lines of interest for parameter reading in the **mar Peaks** file-header.

A1.3 Basic Parameters

Here is a quick reference of the essential parameter set, to make the programs work. (**mar Index** ignores crystal and orientation parameters; rather it the purpose of this program to find these values. Except for the permute keyword, there is no way, as yet, to impose known cell constants to **mar Index**.)

<u>keyword</u>	<u>value(s)</u>	<u>description</u>
<u>[unit-]cell</u>	a,b,c, α,β,γ	lattice constants
<u>axes</u> <u>ori</u> [entation] or: <u>set</u> [ting]	l_{axis} , l_{up} , l_{in} phi1, phi2, phi3) \rightarrow cf. section 1.2 in > mar Index manual) "Coordinate systems"
<u>wave</u> [-length]	λ , $\delta\lambda/\lambda$ [, λ_2 , $\delta\lambda_2/\lambda_2$]	\rightarrow cf. description of the synonymous keyword lam[bda]

<u>dist</u> [ance]	D [+/- tolerance(D)] in [mm]	Sample-to-Detector-Distance
<u>rad</u> [ius]	R_{\min} [, R_{\max} [, resolution_limit]]	inner [& outer] radius in [mm] (, resolution limit in [Å])
<u>imag</u> [e-range]	first_image & last_image, START startPHI OSC deltaPHI	image range, PHI rotation range ("START" & "OSC" are mandatory sub-keywords)
<u>file</u> [-name] <u>name</u>	image_file_name (→ give the correct name, not a template)	→ see also keywords for templates " <u>data image</u> "
<u>scan</u>	image[_x]_size [, image_y_size]	scanner / image width(s)
<u>beam</u> [-centre]	c[x]_image [, cy_image]	beam centre [pixel]
<u>pix</u> [elsize]	p[x]_image [,py_image]	pixel size [mm]

HINT: For historical reasons, these parameters are acceptable without keywords as long as they appear in exactly this order. The remainder is only available by way of keywords.

A1.4 Note on Eulerian angles and cradles & non-*mar* goniostats:

Unfortunately there is a mismatch in the definition of χ (chi) on Eulerian cradles:

- Standard textbooks have $\chi=0$ if $\varphi \parallel \omega$ (e.g. Stout & Jensen; Blundell & Johnson; probably all originating from Arndt & Willis).
- By contrast, the *Huber* goniostat (4-circle diffractometer) has the φ -axis horizontal if $\chi=90^\circ$ (in the "upright" mounting position, i.e. ω -axis vertical, as is usual in the home laboratory). This assignment is certainly more intuitive – mathematically seen – because it is at $0 \mid 0 \mid 0$ that all three rotations are independent (orthogonal).

This conflict has been exploited to adapt *automar* to two different (non – *mar*) Eulerian cradles equipped with *mar*-detectors, while still no ω -rotation proper is implemented (just small missetting angles δ_2 of the φ -axis which would correspond to ω_{offset} if φ horizontal and ω vertical):

- Synchrotron beamlines are bound to have Eulerian cradles mounted horizontally, due to the horizontal beam polarization. Kappa goniostats tend to scan ω (like SER-CAT at APS) instead of positioning the φ -axis horizontally. The standard textbook definition of $\chi=0$ (or 180°) causes *automar* to emulate the ω -scan as a φ -scan.
- Otherwise (i.e. consistently, as seen in the laboratory coordinates), *automar* takes $\chi=0$ horizontal to match the default position of the φ -axis of *mar*-bases.
- automar* assumes 2θ -swing horizontal if $\chi=90^\circ \mid 270^\circ$ (*Huber* diffractometer).

A2.1 Input Parameters in Alphabetical Order

align[ment] delta1,delta2 in [deg.]
or: **mis**[setting]

- Deviation of the PHI axis from the x-axis: if the PHI axis points into direction $[e1,e2,e3]$ (where $[1,0,0]$ is the x-axis), then

$$\text{delta1} = \arctan(e2/e1)$$

$$\text{delta2} = \arctan(e3/e1)$$

→ if $\text{CHI}=0$, then $\text{delta1} = \text{CHI}_{\text{off}}$, $\text{delta2} = -\text{OMG}_{\text{off}}$

- See the introductory [note on Eulerian angles and cradles](#) above.

a.s. | **ammo**[nium - sulfate]

—► see the '[exclude](#)' keyword

axes $l_{axis}, l_{up}, l_{in} = 3$ permutation indices

- permutation indices l for the standard orientation of the crystal lattice (**A**-matrix) chosen such as to allow the orientation angles to be always < 45 deg.: $|l|=1,2,3$ denote **a***,**b***,**c***; the sign of l means "+" = parallel, "-" = antiparallel to the closest coordinate axis. The standard orientation is:
- (1) the 1st axis is put parallel to the x-axis (\sim PHI axis), or antiparallel if $l < 0$;
 - (2) the 2nd in the x-y-plane (with $l > 0$ pointing up),
 - (3) the 3rd axis ought to make a right-handed system pointing "inward" if $l > 0$ = into the Ewald sphere. (This idea was due to P.Schwager in FILME program.)

Example:

Monoclinic cell, b-setting, $\beta = 119$ deg.:
 "axes 2 3 1" rotates the crystal around its twofold axis **b***, with **c*** initially pointing up; $\phi_1 = 29$. ($= 90 - \beta^*$) makes the **a*** axis point into the direction of the collimator.

beam[. [fix]] $c[x_image]$ [, cy_image] in [pixel]

- beam centre, like distance, may be fixed, i.e. not included in refinement.

[beam -]**stop** X_1, Y_1, X_2, Y_2 [, W] = beam stop boundaries (4 or 5 values)
 !!! "[stop](#)" is mandatory !!!

- see the '[exclude](#)' keyword

[unit-]**cell** a, b, c [, $\alpha, \beta, \gamma \setminus 90\ 90\ 90$]

- crystal lattice constants

ccd

- short for "det[ector] ccd" —► see [det\[ector\]](#) keyword

chi χ [, χ_{off}] = Eulerian angle chi [deg.]

- since Eulerian omega $\omega = 0$ in X-ray cameras, χ may be used to describe the Φ (PHI) axis direction w.r.t. horizontal.
 —► Hint: χ (chi) is used to rotate the complete predicted pattern around the beam, assuming $\omega = 0$
 —► See the introductory [note on Eulerian angles and cradles](#) above.

data | **imag**[es] $file_name_template$ (character string)

- (data and image keywords are synonymous)

- image file name with an indication which characters to substitute with the current image number.
- 2 conventions are accepted:
 - root###ext
 - the number of hashes indicates the number of significant digits;
 - prefix<digits>suffix
 - the pointed brackets are removed,
 - the number of enclosed characters indicates significant digits to be replaced by current image numbers,
 - the enclosed digits may pass an image number.
- this "**imag**[e-file]" keyword is discerned from the "imag[e-range]" by the occurrence of either **##** or **<, >**

det[ector] [mar]CCD | [mar]TFT

- The detector type is usually inferred from the image type (header). Zinger rejection or special corrections are applied where appropriate. However *marcv*t may have been used to convert the image format; or the header information may else be corrupted. Use this definition to override the default detector type.

dist[ance [fix]] Sample-to-Detector-Distance (in mm) [, D-tolerance (in %)]

- D-tolerance=0% is equivalent with optional keyword fix[ed] on the same line => no refinement

div[ergence] divergence[(x), (y)] [in deg.]

- [vertical and horizontal] divergence of primary beam

Comment:

Divergence and mosaicity in combination with the particular camera goniometer geometry define how long any part of the crystal volume is in reflecting condition for any hkl.

The precession geometry had the advantage of being rotational symmetric around the beam (at the expense of unusable partials). Divergence and mosaicity together would be described by the "angular spread", hence the picture of a ball-like reciprocal spot.

The rotation method is quite complicated in terms of spot movement and concomitant projection centre, which shows up as an elongated spot shape and/or lateral movement of long-running partials.

In the vertical mid range (yz-plane) the reciprocal spot moves rather perpendicularly through the sphere of reflection. This is why the vertical width cannot normally be separated into mosaicity and divergence, at least not on the basis of image spot positions.

Quite contrary in the horizontal area (neighbourhood of xz-plane) where the reciprocal spot trajectory passes through the sphere at a very oblique angle:

horizontal divergence and mosaicity contribute to the reflexion width and the spot coordinates with different coefficients and can thus be separated, and are therefore refined both from spot centres in *mar Process* integration and from [rocking curves in mar Post](#).

[excl[ude]]

- ▶ optional introduction to the following keywords that define one or more areas where useful data seem to be non-existent (shaded) or corrupted (e.g. by ice or other powder diffraction rings)
- ▶ all parameters are in [pixel] as read from the **automar** display window (or **mar View**)
- ▶ **mar Index** rejects **mar Peaks** coordinates within such area.
- ▶ **mar Predict** | **mar Process** reject spot coordinates if predicted within any excluded area.

— [beam -]stop X_1, Y_1, X_2, Y_2 [,W] = beam-stop boundaries (4 or 5 values)
 !!! "stop" is mandatory !!!

- ▶ shadowed rectangle (meant for the beam stop holder)
- ▶ either given by lower left and upper right corners, or by 2 end points and width of central line (in [pixel] as read from **mar View**)

— shad[ow=shade] x, y, r | x_1, y_1, x_2, y_2 , [w] (in [pixel])

- ▶ up to 12 shaded areas may be defined in addition to beam**stop** by repeated definition of this keyword with one each of three possible boundary descriptions:
 - 3 values = x, y, r :
circular region with centre (x,y) and radius r
 - 4 values =: x_1, y_1, x_2, y_2
rectangular region given by lower left (x_1, y_1) and upper right (x_2, y_2) corner
 - 5 values = x_1, y_1, x_2, y_2, w :
rectangular bar (any direction) specified by central line from (x_1, y_1) to (x_2, y_2) with the width w

— ice[-rings] [$n_1 n_2 ..$ | $d_1 d_2 ..$]

- ▶ exclude annuli (ring-shaped areas) according to the first tabulated d-spacings of ice:
3.9, 3.67, 3.45, 2.67, 2.25, 2.07, 1.95, 1.92, 1.88, 1.72, 1.53 [Å]
- ▶ you may decide to take the risk and exclude some of the ice rings only by specification of their running numbers (in ascending order, because the internal data are overwritten - like also for the keyword "ring"). (If at all, the advice is: "exclude ice-ring 2 3 4 6")
- ▶ alternatively, or in addition to the previous ice ring definition - on the same or another line, up to 12 d-spacings may be defined (e.g. for other powder rings. No more than 12 rings in total.).

Note: The "ice-ring" keyword defines the centre only of each ring by its d-spacing, its width is constructed from the spot size such as to avoid overlap.

— a.s. | ammo[nium - sulfate]

- ▶ exclude annuli (ring-shaped areas) according to the d-spacings of ammonium sulfate (AS). The lattice constants are taken from the low-temperature structure (Pna2₁) by E.O.Schlemper & W.C. Hamilton, J.Chem.Phys.**44**(1966) 4498-4509. Few of the AS reflexions yield clearly separated Debye-Scherrer rings; most of these annuli have been defined to cover 2 to 4 close rings:
5-20-5.30, 4.33-4.39, 3.92, 3.15, 2.98-3.04, 2.60-2.70,

- 2.47-2.54, 2.32-2.34, 2.17-2.20, 2.05-2.10, 1.93-1.99 [A]
 —► Contrary to the ice ring definition, no sub-selection is provided. However, a definition of, e.g. "exclude ice-ring 2 3" may be followed by an "exclude a.s." command which will then fill up the remaining space (10 more rings in this example. Note that 4-th and 6-th ice ring are then implicitly excluded as well.)

— **ring[s]** d_1 [, d_2]

- repeated definition of the "ring" keyword defines one annulus each.
- specification of d_1 only is equivalent with an "ice" keyword definition: d_1 is the central d-spacing and the width is made up from spotsize.
- d_1 and d_2 define the inner and outer boundary of the annulus to be excluded (e.g. powder ring that is wider than the spot size parameter suggests; or any other ring-like shadow, stray radiation or other reasons to exclude such an area).

Note: In total 12 annulus definitions are acceptable, no matter by which combination of "**ice**", "**ring**" and "**a.s.**" keywords they are defined. (They do not interfere with beam-**stop** and/or **shadow** definitions.)

ext[ension] suffix (character string)

- root and ext keywords may help marIndex to construct the image file template that it is to write in the .amp-file

file[-name] image_file_name (character string)
 or: **name**

- specify the correct name, not a template
- **file** and/or **name** keyword are accepted
- cf. "[data | image](#)" for templates

fix [beam[centre] | cent[re]] [dist[ance]] [ori[entation]]
 [cell] [mos[aic]] [div[ergence]] [spot[size] | [size]]
 [tilt] [chi[_offset]] [omega|omg[_offset]]
 (or: alig[nement]=mis[setting] == both chi_ and omg_offset)

- this keyword specifies which parameters *not* to include in refinement; the values are not numbers but rather keywords.

Explanation:

Since **mar Predict** is an integral part of **mar Process**, the same parameters are used by both programs. But contrary to stand-alone **mar Predict**, all sorts of refinement may there alter a number of variables in due course, based on the significantly observed spots. Here is a means to suppress any one of those updates, if the user has good reasons why he or she does not like it.

Special keyword "fix" (immaterial for stand-alone **mar Predict**) may on the one hand be used in its own right with the above list of subkeys.

On the other hand, this way of definition is synonymous with typing "fix[ed]" on each of the keyword lines concerned.

ice[-rings] [n_1 n_2 .. | d_1 d_2 ..]

- see the '[exclude](#)' keyword
- imag[e_files]** file_name_template (character string)
- this "**imag[e_files]**" keyword is discerned from the "imag[e-range]" by the occurrence of either ## or < and >
- for description see the synonymous —► '[data](#)' keyword
- imag[e-range]** first_image last_image START start_osc OSC osc_width
- image range, and PHI range ("START" & "OSC" are mandatory sub-keywords)
- note that peaks file header and marIndex .amp-file hold the image range from which peaks were selected, i.e. from which the auto-indexing has worked out the present lattice and orientation. Therefore, if marProcess is run from the command-line using the .amp-file, the log-file can correctly monitor where the lattice originated.
- repeated "image ... START ... OSC ..." definitions (up to 5) are acceptable to specify multiple image ranges (cf. [Section 5.5 command line option "Image Sequences"](#))
- lam[[b]da]** wavelength₁₁ [, wavelength₂₁ [, ... wavelength₂₂]]
- synonymous with '[wave](#)[[*-*]length]'
- may either be specified as 'Cu|Mo|Ag' for stored values of $K\alpha_1$ and $K\alpha_2$; then $K\alpha_1/\alpha_2$ split will be considered in *mar* Index refinement and in *mar* Predict | *mar* Process if the maximum split is big enough not to be disguised by the given pixelsize.
- Up to 4 numbers may otherwise have one of two different meanings :
- (1) possibly composite wave length:
- | | |
|---|--------------------------------|
| <nominal λ > | (wavelength ₁₁) |
| [, $\delta\lambda/\lambda$ \ default=0.003 | (wavelength ₂₁) |
| [, $\lambda(\alpha_1), \lambda(\alpha_2)$]] | (wavelength _{12;22}) |
- This is also the internal data format for keyworded Cu|Mo|Ag specification:
- ```
Cu 1.54178 0.005 1.540562 1.544390
Mo 0.71069 0.01 0.70926 0.71354
Ag 0.5608 0.01 0.559401 0.563802
```
- Full data are used if the resol limit is big enough to make the split visible, i.e. more than a pixel.
- Comment: For the prediction of  $K\alpha_1$  &  $\alpha_2$  split parts, the internal wavelength is  $(K\alpha_1 + K\alpha_2) / 2$  such that the split is symmetric (X,Y) +/- (dX,dY)
- Alternatively:
- (2) two superimposed wave-lengths:
- |                                         |                                |
|-----------------------------------------|--------------------------------|
| $\lambda_1, \delta\lambda_1/\lambda_1,$ | (wavelength <sub>11;21</sub> ) |
| $\lambda_2, \delta\lambda_2/\lambda_2$  | (wavelength <sub>12;22</sub> ) |
- (e.g.  $K\alpha - K\beta$ ; or synchrotron 1st and 3rd order Ge111 - Ge333)
- Example: Cu  $K\alpha$  &  $K\beta$ : "*lam 1.54178 0.005 1.39222 0.01*"
- latt[ice]** Bravais-lattice (or extinction-group)
- Bravais-lattice = 2 characters as defined in new Int.Tables (Vol.I, table 9.2.2), e.g. oP, cI, etc.

- Unless special extinction rules (for screw axes or glide planes) are of interest, the Bravais-lattice definition is sufficient in *marPredict*:
- *extinction-group* is an internal pointer to the extinction rules, obsolete by way of "space-group" but still tolerated as an input parameter. It is printed in the *marPredict* log-file.

*Example:* space group #17 (P 2 2 2<sub>1</sub>), #93 (P 4<sub>2</sub> 2 2), #177 (P 6<sub>3</sub>), etc. have the same systematic absent reflexions ( $l=2n+1$ ), and identical *extinction-group* (5).

- list** [ list\_code \ all extra listings ]
- extra listings, or more extensive listings, as described for individual programs, where appropriate
  - cf. command line option *-vp* which causes partly the same, partly complementary extra information to be printed
- mar** peaks\_file (character string)
- synonymous with → [peak\[s\]](#) → see description there
- mis[setting]** delta<sub>1</sub>,delta<sub>2</sub> in [deg.]
- synonymous with → [alignment](#)
- mosa[ic..]** mosaicity[(x), (y)] in [deg.]
- in *mar Index* ignored as an input value, but passed on to the ".amp"-file overriding any *mar Index* results.
  - otherwise *mar Index* finds an approximate value for the angular spread, cf. *mar Index manual*, and outputs tentative mosaicity as angular spread minus default divergence of 0.1 deg.
  - For description of angular spread, vertical (x) and horizontal (y) mosaicity and divergence, cf. comment for [divergence](#).
- name** image\_file\_name (character string)
- synonymous with the → ['file'](#) keyword
- omg\_|omeg[a]** OMG [, OMG\_off]
- Eulerian omega is not yet consistently implemented within *automar*; thus the (optional) *OMG\_off* as one of the PHI axis misalignment angles is, at present, the essential value. But see the ['align'](#) keyword which is more appropriate.
  - See the introductory [note on Eulerian angles and cradles](#) above.
- ori[entation]** phi<sub>x</sub>,phi<sub>y</sub>,phi<sub>z</sub>  
or: **set[ting]**
- phi<sub>x</sub>=φ<sub>1</sub>, phi<sub>y</sub>=φ<sub>2</sub>, phi<sub>z</sub>=φ<sub>3</sub> = Eulerian orientation angles that rotate the lattice around x,y,z from the standard orientation (see [axes](#)) to its experimental setting for PHI=0.  
The phi<sub>1,2,3</sub> rotation matrices are applied from right to left :  
$$[\mathbf{A}(\Phi)] = [\Phi] \cdot [\varphi_1] \cdot [\varphi_2] \cdot [\varphi_3] \cdot [\mathbf{A}_0]$$
such that the diagonal setting of a face-centred lattice can be incorporated into phi<sub>3</sub> without being affected by the other setting angles; and [phi<sub>1</sub>] is a mere correction of [PHI] which may be chosen to re-define the PHI=0 setting.



- peak[s] | mar**      peaks\_file (character string)
- ▶ specific definition of marPeaks output file with extension .pks (or .mar .. obsolete) as an input to marIndex by way of control input file.
  - ▶ marIndex may, however, work out the .pks-file name from the primary image file name, or take the peaks file-name immediately from the command line.
- perm[ute]**       $l_{axis}, l_{up}, l_{in} = 3$  permutation indices
- ▶ special input to *marIndex*:  
By default, output axes are ordered according to the standard of the Int.Tables, i.e. increasing lengths (unless forced otherwise by symmetry conventions, e.g. 6-fold = **c** axis).  
Certain cases require re-ordering: P 2<sub>1</sub> 2<sub>1</sub> 2, or P 2 2 2<sub>1</sub>, have the exceptional (non-screw or screw) axis as **c**. *marIndex* cannot & does not examine screw axes and consequently ignores the difference.
  - Example:  
"perm 2 3 1" will make the c-axis the shortest.
  - ▶ GUI Index-window has a menu button for permutation of cell axes. Subsequent to initial auto-indexing, re-ordering of the axes can be forced (in the above example) by selection of "b c a" replacing standard.  
A message window warns the user to re-run **mar Index** such as to make axes and orientation consistent.
- pix[elsize]**      p[x]\_image [, py\_image]
- ▶ pixel-size [mm]
- plot**      [ plot\_code \ all plot options ]
- ▶ **obsolete** .. would write a *marPeaks* style coordinate file (before *marView* could understand .prd-files)
- pol[arization]**      \* (sub-keyword) [p]
- 3 alternatives:
- \* = "**mir**[rors]" implies no polarization
- ▶ *marIndex* will also try to index badly fitting spots by K\_beta radiation, if wave-length Cu is specified.
- \* = "**mon**[ochromator]" may be accompanied by :  
Ge[111] | Si[220] | graph[ite]      (default: graphite 002)
- ▶ **automar** assumes the monochromator to deflect the beam horizontally and knows the monochromator's lattice constants; so for each wave-length it can work out the degree of polarization (immaterial for auto-indexing).
- \* = "**S**[ynchrotron]" defines fractions of primary beam intensity polarized horizontally or vertically in one of the 3 following ways:

**S h**  $f_h$  horizontal fraction; if  $f_h > 1$ , taken as %:  $f_h \rightarrow f_h/100$ ;

$$f_v = 1 - f_h$$

**S v**  $f_v$  vertical fraction; if  $f_v > 1$ , taken as %:  $f_v \rightarrow f_v/100$ ;

$$f_h = 1 - f_v$$

**S fract** polarization factor  $\text{fract} = (h-v)/(h+v)$ ;  
if  $\text{fract} > 1$ , taken as %:  $\text{fract} \rightarrow \text{fract}/100$ ;

$$f_h = (1+\text{fract})/2$$

$$f_v = 1 - f_h$$

**S** with no values defaults to "S h 1" (100% horizontally pol.)

Comment: The latter input mode ( **S** ) is not restricted to S.R. but is the most general way to input any situation that contradicts the previous assumptions, e.g. monochromator deflecting out of the horizontal plane. However it is up to the user to work out what the polarization is.

**rad**[ius]  $R_{\min}$  [,  $R_{\max}$  [, resolution\_limit] ]

- inner [& outer] radius limit(s) [mm], *optionally* resolution limit [A]
- usually  $R_{\min}$  excludes the inner beam stop shadow; if this is very excentric, and if low resolution data matters, then use the [\[exclude\] shadow](#) keyword instead; on the other hand exclude the beam-stop *holder* with the [\[exclude beam-\]stop](#) keyword.
- for  $R_{\max}$  and resolution\_limit, cf. comments for the "[reso](#)[lution]" keyword.

**reso**[lution] resolution\_limit =  $\lambda/(2 \cdot \sin\theta_{\max})$  [A]

- alternatively input as 3rd value with 'rad'
- The minimum of  $R_{\max}$  and resol limits the pattern. The difference may be important for double-wavelength patterns where the resolution limit corresponds to different radii. Also in  $2\Theta$ (detector)-settings the radius limit differs from the resolution limit, such that **resolution** is preferable.

**rest**[art]  $n \setminus 3$  | no [ne]

- (only for **mar Process**): suppress or limit the number of restarts after significant parameter refinements (restricted to 4 at most)

**ring**  $d_1$  [,  $d_2$ ]

- see the '[exclude](#)' keyword, sub-keyword **ring**

**root** prefix (character string)

- **root** and **ext** keywords may help **mar Index** to construct the image file template that it is to write in the .amp-file

- rot**[ate] AD\_rot, detector-rotation wrt. projection of TTH-axis  
—► for more description, cf. [tth. keyword](#)
- sat**[uration] limit  
—► threshold above which pixel values are considered over-exposed (only for marIntegrate / **mar Process**).  
Integration by profile-fitting may substitute few saturated pixels. All other methods must reject over-exposed spots.  
—► Default values:  
65534 for marCCD (16 bits)  
130000 for mar300 (16 bits with expansion)  
250000 for mar345 (binned sub-pixel)
- scan** image[\_x]\_size [, image\_y\_size]  
—► scanner / image width(s) in [pixel]
- serv**[ice] foreign program name[s]  
—► input and/or output files may be formatted to serve the transition to foreign programs.  
—► see description of the individual programs.
- set**[ting-angles]  $\phi_x, \phi_y, \phi_z$   
—► synonymous with the —► '[orientation](#)]' keyword
- shad**[ow=shade] x,y,r | x1,y1, x2,y2, [w] (in [pixel])  
—► see the '[exclude](#)' keyword
- skew** tilt\_X [, tilt\_Y [, AD\_rot]] (all 3 detector-rotations)  
—► for all 3 detector rotations, see '[tth keyword](#)', and/or section 1.3.1 on "[Coordinate systems](#)" above.
- spac**[e-group] *name or number in Int.Tables*  
—► Unlike with the Bravais-lattice definition, extinction rules due to screw axes and glide planes are observed in the prediction. All 230 space groups with their particular extinctions are implemented (keep your fingers crossed).  
—► The Bravais-lattice follows from this definition.

- spot**[-size]      spot\_size[\_hor.=X, spot\_size\_vert.=Y **asym**[metric]] [mm]
- size of the classical integration box (which may be adapted by spot profile analysis unless fixed).
  - The experimental spot profile is not restricted by this box; the analysed area, however, is limited by the box size (increased for oblique incidence) plus standard rim for box background (2 pixel wide) and search range for spot location.
  - in **mar Index** ignored as an input value, but passed on to the ".amp"-file overriding any **mar Index** results.
  - otherwise **mar Index** averages the number of pixels in the **mar Peaks** file, and computes an approximate spot size as diameter of an idealized circular spot. This value may rather be too small being dependent on the significance level set in **mar Peaks**.
- tft**
- short for "det[ector] tft" → see [det\[ector\]](#) keyword
- tilt**      tilt\_X [, tilt\_Y] [deg.]
- tilt of the detector about x-|y-axis in math.+ve direction. Tilt>0 brings the upper or left edge of the detector closer to the crystal.
  - Note: x is always the direction of the phi axis. if chi = 90 or 270 deg. then the x axis is upright.
- titl**[e]      ... (character string[s])
- (1st and 2nd title allowed, defined by their sequence)
- tth** | **twot**[h..]      TTH [, TTH<sub>off</sub>] in [deg.]
- detector position  $2\Theta$  [deg.]
  - 2Theta and Phi axes are assumed (nearly) coincident. More precisely, if  $2\Theta$  is different from zero, the x-axis is the projection of the 2Theta axis onto the plane perpendicular to the beam, while the Phi axis is described by the alignment parameters (see [align\[ment\] keyword](#)). If the detector pixels do not run parallel to the 2Theta axis, use the **rot** (or **skew**) keyword.
  - If  $2\Theta=0$ , rot is dispensable, and the x-axis is the projection of the X-pixel direction onto the plane perpendicular to the beam, while the phi axis is again described by the alignment parameters.
- unit**[-]cell      a, b, c [,  $\alpha$ ,  $\beta$ ,  $\gamma$  \ 90 90 90]
- synonymous to → **'cell'** keyword
- wave**[-length]      wavelength<sub>11</sub> [, wavelength<sub>21</sub> [, ... wavelength<sub>22</sub>]]
- synonymous with → [lambda](#) → see description there

**wind[ow]**       $x,y,r$  |  $x_1,y_1, x_2,y_2$  (in [pixel])

- ▶ restrict all activity to within the circle with radius  $r$  around  $(x,y)$ , or the rectangle given by lower left and upper right corner.
- ▶ may also be used to enforce full-size area for detector types that are active right into the corners, e.g. *mar-mosaic ccd* (disabled for image plate or circular ccd); must be accompanied by corresponding [R<sub>max</sub>](#) and [resol](#) values. Default is inscribed circle.

## A2.2 Keywords in Groups and Examples

```
!
! crystal parameters :
!
[unit-]cell 79.1 79.1 37.9 90 90 90
axes 3 1 2
permute 2 1 3
orientation | setting 15.327 -6.182 0.313
lattice tP
spacegroup 96
resolution 1.85
mosaicity 0.2
spotsize 0.7 0.5
!
! goniometer parameters :
!
image[-range] 1 90 START 0. OSC 0.5
lambda Mo | wave-length 1.54178
divergence 0.3 0.5
polarization mirrors | polarization monochromator graphite
| polarization Synchrotron h 85
| polarization S v 15
distance 119.8 [fix]
chi 90.
omg | omega 0. -0.14
tth | twotheta 30. 0.31
alignment | missetting 0.31 0.14
!
! detector parameters :
!
scan 2000 x 2000
pixelsize 0.069
saturation 32000
beamcentre 998.1 1002.3 [fix]
tilt 0.21 -0.08
rotate 0.15
skew 0.21 -0.08 0.15
radius 7.5 125.
window 100 1 1900 2000
[beam -]stop 1 965 1010 1020
[beam -]stop 1 982 1010 1003 35
```

```
[exclude] shadow 980 1005 150
[exclude] shade 1700 1800 2000 2000
[exclude] shadow 1 1400 800 2000 300
[exclude] ice 1 2 3
[exclude] ring 4.714
[exclude] ring 3.847 3.986
!
!file names:
!
data | image lyso_2_<001>.mar2000 or lyso_2_###.mar2000
file[-name] | name lyso_2_001.mar2000
root lyso_2
ext mar2000
peaks | mar lyso_2_001.pks
!
!options:
!
fix cell align
list 1
title Second run of the new crystal form
service SHELX
restart 2
```