technical memorandum Daresbury Laboratory

DL/SCI/TM25T

DL/SCI/TM25T

# PROGRAM CASCADE: An Input Processor for use in Crystal Defect Calculations

by

W. Smith, Daresbury Laboratory

FEBRUARY 1981

Science Research Council

Daresbury Laboratory

**IMPORTANT**

The SRC does not accept any responsibility for loss or damage arising from the use of information contained in any of its reports or in any communication about its tests or investigations.

# PROGRAM CASCADE: An Input Processor for use in Crystal Defect Calculations

by

W. SMITH, Daresbury Laboratory

January 1981

## 1./ INTRODUCTION.

The purpose of this document is to describe the function and data input of the program known as CASCADE (Cray - 1 Automatic System for the CAlculation of Defect Energies). The program, written specifically for the Cray - 1 computer at Daresbury Laboratory, is intended to form an integral part of the proposed replacement to the HADES III program currently used to calculate defect energies (and related phenomena) in crystalline solids. Development of this new program has now reached a point where prospective users have expressed an interest in attempting trial executions of the program. For this reason, prior to the production of a more comprehensive text, this document has been prepared to allow such users some understanding of the program input requirements.

## 2./ THE INPUT SUBROUTINE.

One of the expressed requirements of the new defect energy program was that it should allow the user the convenience of free formatted input. This would mean that the user need not waste time producing regimented data sets, but could nevertheless rely on the program to interpret the data correctly. For this reason the program CASCADE includes a subroutine called INPUT, which enables this facility. The subroutine is in fact a collection of appropriate routines taken from the ATMOL III quantum chemistry package and gathered into one subroutine with several entry points. An understanding of the function of this subroutine is a necessary prerequisite to using the program CASCADE and therefore it is described here in some detail.

In order to read in a data card, the program CASCADE calls the subroutine INPUT, which reads the card in the FORTRAN A format as a field of 80 characters. The data is stored in the common block named /WORK/ in the array IREC(80). Within this 80 character field there are expected to be one or more functional fields separated by one or more spaces. A functional field can define an integer number, a real number or an alphanumeric character string. The subroutine scans the 80 character field and records the number of functional fields their length in characters and their starting location in the 80 character field. These quantities are stored in the common block /WORK/ as the variable JUMP and arrays INUM(40) and ISTRT(40) respectively. It does not at this stage attempt to interpret the functional fields since the nature of each functional field has to be specified by the main program.

To interpret a string of characters in the functional field as an integer the main program calls the entry point INPI(J). The routine then processes the character string into an integer number which is returned as the argument J to the main program. If the subroutine cannot convert the character string into an integer it sets an error indicator (the variable IERR in common block /WORK/) to a value of 1. How this error is dealt with is a matter for the main program, however the usual procedure in program CASCADE is to refer to the entry point OUTERR described below.

A real number present in the functional field is treated in a similar manner to an integer. In this case the main program calls the entry point INPF(R), from which the real number is returned as the real argument R. Failure to interpret the functional field as a real number causes the error indicator IERR to be set to 2. It should be noted that this routine will not recognise the E or D formats frequently encountered in FORTRAN, but in a realistically scaled problem the use of ordinary real numbers should not cause undue difficulties.

A functional field representing a character string may be extracted by calling the entry point INPA(A), from which the character string is returned in the argument A. For the purposes of the program CASCADE the character string will automatically be truncated to the first four characters, irrespective of the original length of the string. (Thus the character string DARESBURY would be truncated to DARE by the INPA routine.) The reason for this is to allow the use of shortened codewords in the input data. This aspect will have more relevance later in this document. Note that while it is not possible to misinterpret the character string in the way that the real and integer numbers may be misinterpreted it is possible that the given character string is not that expected. When this occurs in the program CASCADE the usual procedure is to refer to the OUTERR routine with the error indicator set to zero.

When the main program detects an error indicator (IERR=1 for integer error, IERR=2 for real number error or IERR=0 for incorrect character string), CASCADE usually calls the entry point OUTERR. This will result in the printing of an error message indicating the type of error (0,1 or 2) and a print of the data card on which the error was detected. Control will then be returned to the main program. In this way CASCADE is able to ignore incorrect data and continue to process subsequent data cards.

To conclude this section a number of important features of the INPUT subroutine should be mentioned.

Firstly it is important to realise that the functional fields are identified in the INPUT subroutine as continuous character strings separated by one or more spaces. It is obvious therefore that errors will result if spaces are carelessly inserted within an intended functional field. Functional fields cannot be written across data cards

Secondly it should be noted that each call to the subroutine INPUT will result in a new data card being read. Calls to the entry points INPI, INPF or INPA will cause a register of the current functional field to be incremented by 1. This register which is stored as the variable JREC in the /WORK/ common block, may be reset in the main program, thus allowing functional fields to be reprocessed or bypassed. This feature is often used by CASCADE in identifying the nature of the input data.

Lastly should the INPUT subroutine encounter the end of the input dataset when asked to read in a new data card, an appropriate error message is printed and the program is terminated. This is the only occasion when the INPUT subroutine may terminate the program.

## 3./ THE INPUT PROCESSOR.

The control of the data read into the program CASCADE is handled by an input processor, the function of which is to search the input dataset for the directives that identify the data and indicate how the data are to be processed. At the time of writing there are approximately 20 such directives but this number is very likely to increase as the program is extended to incorporate other procedures. However the basic structure of the input processor will not change and this document may be thought of as outlining a minimum set of directives to be found in the program CASCADE.

The input dataset consists of the usual items of data necessary for a crystal defect calculation, such as a description of the lattice vectors, the contents and spatial distribution of the unit cell etc.,interspersed with the directives. The directives appear in the text of the data as ordinary English words which act as signals to the input processor to identify and process the subsequent data. For example the appearance of the word LATTICE in the data indicates that the lattice vectors are to follow. A detailed description of each directive is given in the next section.

It should be noted that in its usual mode of operation each directive acts as a switch which lifts the program out of the input processor mode into a data processing mode in which the program performs predetermined operations on the data. It is not until this processing has been completed that the program returns to the input processor mode. As a consequence of this the user needs to be wary of inserting directives in the input data without due care in case a directive is misinterpreted as genuine data. Although the program does have a well developed capacity to deal with such errors it is not always possible to deal with them in ways the user would like.

An important feature of the input processor is that it will allow the user to make mistakes in the data without bringing the program to an immediate halt. The program adopts the general strategy of ignoring nonsensical data unless the data it is expecting is either essential to its subsequent operations or is needed to allow efficient processing. Thus the program will often proceed quite some way with imperfect data before finally rejecting the problem as hopeless. This feature means that the unfamiliar user will learn of the errors in his input data without the ordeal of performing many trial runs.

There now follows a description of the various directives available in the program CASCADE. The order in which they are presented has no particular significance, but only reflects the order in which they are programmed. This will be followed by the recommended sequence of input to guide the user in constructing input datasets.

## 4./ THE DIRECTIVES.

### (i) Directive 1. TITLE.

The TITLE directive is an instruction to print a title or heading in the line printer listing of the output. The position of the printed title in the output is determined by the position of the directive in the input dataset. Thus the user can arrange to have comments inserted at different points in the output. The titles are printed immediately after the directive is read in. The title may be any number of lines of 80 characters each. Each line is written out in the output in exactly the same form as it was input, as a field of 80 characters placed centrally in the 132 charcters of the line printer hollerith field.

Format:

TITLE

Operands: None

Subsequent Data Cards: Any number with any contents. The collection of TITLE cards is terminated by a card with the single entry ENDS.

Example.

```
TITLE
CALCULATION OF SODIUM CHLORIDE LATTICE WITH SODIUM ION VACANCY.
ENDS
```

### (ii) Directive 2. PRINT.

The PRINT directive is an option used to define some of the data that the user wants to be printed out. The option allows control over the printing of the region I coordinates region II coordinates the reciprocal lattice the symmetry eigenvectors and the structure factors. Ideally this option is presented to the input processor early on in the input dataset so that the user may control the output of the program from the start of the job. The quantity of output produced under these options is determined by the DUMP directive (See below). The default option specifies the printing of none of the data types mentioned above.

Format:

PRINT

Operands: There are six operands (currently), each specifying a particular type of data the user wishes to be printed out.

PRINT aaaa bbbb cccc
Where aaaa, bbbb and cccc etc. are the operands. These are as follows:

REG1 - Print region I coordinates.
REG2 - Print region II coordinates.
RECI - Print reciprocal lattice coordinates.
SYMM - Print symmetry eigenvectors.
STRU - Print structure factors.
NONE - Print none of the above types. (Default option ).

Note that any of these options may be specified in any order. The NONE option should not be used if any other option is specified. If the user includes more than one PRINT directive in the input dataset the parameters of the previous PRINT directive are overwritten. This allows the user to respecify the output options at other points in the program execution.

Subsequent Data Cards: None.

Example:

```
PRINT REG1 STRU REG2
```

The above card result in the printing of region I and region II coordinates and the structure factors.

### (iii) Directive 3. DIMENSION.

The DIMENSION directive is used to specify the amount of dynamic core that the program requires. Its single operand specifies the number of words of storage the program is to use. A typical value for this would be 250000 words which would be large enough for virtually any crystal system. The dynamic core allows the program dimensions to be specified at the time of execution.

Format:

DIMENSION (Acceptable abbreviation DIME)

Operands: One, indicating the required number of words of core.

```
DIMENSION nnnnnn
```

Where nnnnnn=number of required words of storage.

Subsequent Data Cards: None.

<u>Note that in the present version of CASCADE it is essential that the DIMENSION directive be first directive that the program encounters in the input dataset. (See Section 5 in this document.)</u>

### (iv) Directive 4. LATTICE.

The LATTICE directive indicates to the input processor that the data cards to follow specify the lattice vectors for the crystal system under study. On encountering the LATTICE directive the program CASCADE enters the subroutine named LATICE where the lattice vectors are read in and processed. This processing includes a check for coplanarity of the given lattice vectors and the calculation of the first three reciprocal lattice vectors.

Format

LATTICE (Acceptable abbreviation: LATT)

Operands: None

Subsequent Data Cards: Three, each with three real numbers specifying the components of one of the lattice vectors with respect to a cartesian set of axes. The units are the usual lattice units i.e. Distances are expressed as multiples of the lattice constant.

Example:

```
LATTICE
1.0 0.0 0.0
0.0 1.0 0.0
0.0 0.0 1.0
```

Note that the accuracy of the numbers used to define the lattice vectors must be carefully considered. If the numbers are not specified accurately enough the program CASCADE will be unable to produce a sufficiently accurate representation of the lattice to define the symmetry properties correctly. This note applies to the basis and defect vectors also. (See however, the ACCURACY directive below.)

### (v) Directive 5. BASIS.

The BASIS directive indicates to the input processor that the crystal basis vectors are about to be input (i.e. the components of the unit cell.) The directive causes the program CASCADE to enter the subroutine CRYBAS which processes the basis vectors. The BASIS directive is immediately followed by several cards, each with the the details of one of the species in the unit cell. The species may be input in any order. Subsequent processing will orgnnise them more suitably for calculation. This processing also checks the basis species for errors such as duplication. The data cards are terminated by a card with the single entry ENDS.

Format:

BASIS

Operands: None.

Subsequent Data Cards: Any number. one for each species or component of the unit cell plus one card specifying a new position of the origin if desired. (This origin is known as the centre of symmetry about which the symmetry operations of the crystal point group are applied. This is also frequently taken to be the defect origin.) Each species card carries the species label, its type (core or shell) and the position vector of the species specified by three real numbers. The position of the species in the unit cell is defined initially with respect to some arbitrary cartesian axes and the centre of symmetry is defined in terms of the same axes. During the processing of the basis coordinates, the basis is relocated with respect to the centre of symmetry. Note that if no centre of symmetry is given, the centre is taken to be the origin of the cartesion axes used to define the basis initially. As with the lattice vectors, the coordinates are expressed in lattice units.

Example:

```
BASIS
NA+ CORE 0.0 0.0 0.0
CENTRE 1.0 0.0 0.0
CL- SHELL 1.0 0.0 0.0
CL- CORE 1.0 0.0 0.0
NA+ SHELL 0.0 0.0 0.0
ENDS
```

Note that the example uses standard chemical nomenclature for labelling the different species. This is strongly recommended since the input processor does not recognise the characters of any label beyond the fourth and using full names can cause ambiguity. (e.g. OXYGEN2- and OXYGEN- both become OXYG.) The use of the codewords CORE, SHELL and CENTRE in the appropriate places is mandatory. The earlier comments about the need for sufficient accuracy in presenting the coordinates apply here also.

### (vi) Directive 6. DEFECT.

The DEFECT directive signals to the input processor that the details of the crystal defects are to follow. The directive causes the program CASCADE to enter the subroutine DEFECT, where the defect vectors are processed. The directive is followed by any number of cards one for each defect species, up to a (current) maximum of 40 and is terminated by a card with the single entry ENDS. Each card carries the details of the defects to be introduced into the perfect lattice.

Format:

DEFECTS (Acceptable abbreviation: DEFE)

Operands: None.

Subsequent Data Cards: Any number, in any order, each with the defect species label, type (core or shell), defect type (interstitial or vacancy) and the defect coordinates defined with respect to the unit cell cartesian axes (i.e. the same coordinate system as the basis vectors. )

Example:

```
DEFECTS
NA+ CORE VACANCY 0.0 0.0 0.0
CL- CORE INTERSTITIAL 0.0 0.0 0.0
CL- SHELL INTERSTITIAL 0.0 0.0 0.0
NA+ SHELL VACANCY 0.0 0.0 0.0
ENDS
```

In the case of vacancy defects it is essential to specify the vacancy as occuring at a valid lattice site. A consistent use of labels for the species concerned is also necessary. The codewords VACANCY and INTERSTITIAL may be abbreviated to VACA and INTE respectively. (Note if it is required that the interstitials remain fixed i.e. unrelaxed during a defect calculation the codeword INTF must be used in place of INTE). The earlier comments about the accuracy of the coordinates apply here also.

## (vii) Directive 7. ACCURACY.

The ACCURACY directive is used to respecify the internal tolerances of the program CASCADE for the purposes of defining the crystal lattice coordinates and its associated symmetry. This directive is necessary when the user attempts to input lattice, basis and defect vectors that are not accurate to at least 0.0000001 of a lattice unit. If this is the case then user may either respecify the vectors to achieve this accuracy or simply reset the tolerances of the internal testing procedures of the program. The ACCURACY directive enables the latter option. If the user fails to pay attention to this matter there is a good chance that CASCADE will be unable to define the crystal lattice or its symmetry correctly. (See the SYMCHECK directive below ).

Format:

ACCURACY (Acceptable abbreviation: ACCU)

Operands: One, indicating the required tolerance of the test procedures.

ACCURACY rrrrrrr
Where rrrrrrr=required tolerance.

Subsequent Data Cards None.

Example:

```
ACCURACY 0.00001
```

This example indicates that the basis vectors etc. are accurate to the sixth decimal place only. Note that the ACCURACY operand is set to be ten times bigger than the accuracy of the coordinates. This is necessary because the ACCURACY operand specifies the tolerance on the internal testing procedures of the program hence to ensure that the coordinates etc. will pass the required tests a safety margin is used that allows for rounding errors.

## (viii) Directive 8. REGION.

The REGION directive is used to input details about the crystal region sizes and cutoff radii for various parameters used in the program.

Format:

REGION (Acceptable abbreviation: REGI)

Operands Seven, (currently), specifying the magnitudes of the various parameters.

REGION aaaa bbbb cccc ...
Where aaaa, bbbb and cccc etc. are numbers specifying the following parameters:

Parameter 1: NREG1. Number of classes required in region I. (Integer)
Parameter 2: REG2. Radius of crystal region in lattice units. (Real number)
Parameter 3: RLAT. Lattice constant in Angstroms (Real number).
Parameter 4: CUTPOT. Short range potential energy function cutoff radius in lattice units. (Real number).
Parameter 5: CUTSHL. Maximum permissible separation of cores and corresponding shells in the defect crystal relaxation in lattice units. (Real number).

Parameter 6: DISMAX. Maximum permissible relaxation of the defect crystal cores in lattice units (Real number).
Parameter 7: CUTMAX. Maximum range of any of the short range potential functions used in the crystal model in Angstroms. (Real number).

Subsequent Data Cards: None

Example:

```
REGION 80 4.94 2.789 1.5 0 1 0 2 0.0
```

Where the number of symmetry classes is 80, the radius of the crystal region is 4 94 lattice units, the lattice constant is 2.789 Angstroms. the potential cutoff is 1.5 lattice units, the maximum core - shell separation is 0.1 lattice units, the maximum relaxation of the cores is 0.2 lattice units and the maximum potential range possible in the system is 0.0 Angstroms, (which means that the program will default to the largest value it finds in the potential data input. See below.)

### (ix) Directive 9. OPTICAL.

The OPTICAL directive indicates to the program CASCADE that the crystal defect calculation is to be performed in an optical mode. This means that the relaxation of the crystal is to permit the shells only to relax.

Format:

OPTICAL (acceptable abbreviation: OPTI)

Operands: None.

Subsequent Data Cards: None.

Note that the OPTICAL and THERMAL directives (below) are mutually exclusive directives. One of them must be selected for a complete defect calculation.

### (x) Directive 10. THERMAL.

The THERMAL directive indicates to the program CASCADE that during the crystal defect calculation the relaxation of the crystal is to allow both cores and shells to relax.

Format:

THERMAL (Acceptable abbreviation: THER)

Operands: None.

Subsequent Data Cards: None.

See comments on the OPTICAL directive above.

### (xi) Directive 11. CUBIC.

The CUBIC directive indicates to the program CASCADE that the crystal system under study has (at the outset) cubic or Oh symmetry. This allows the program to call the subroutine CUBMAT to construct the appropriate transformation matrices. These matrices are applied to the defect crystal to establish the true symmetry of the defect crystal as a subgroup of the Oh point group.

Format:

CUBIC

Operands: None.

Subsequent Data Cards: None.

The CUBIC and HEXAGONAL directives (see below) are mutually exclusive. One of them must be specified in the input dataset.

### (xii) Directive 12. HEXAGONAL.

The HEXAGONAL directive indicates that the crystal system under study has hexagonal or D6h symmetry at the outset. This enables CASCADE to call the subroutine HEXMAT to construct the required transformation matrices. These matrices are applied to the defect crystal to establish the correct subgroup of the D6h point group that applies to the defect crystal.

Format:

HEXAGONAL (Acceptable abbreviation: HEXA)

Operands: None.

Subsequent Data Cards: None.

See the comments on the CUBIC directive.

### (xiii) Directive 13. DUMP.

The DUMP directive indicates to the program CASCADE the level of output the program is required to produce. Three output levels are available and the user may code for any one of these.

Format:

```
DUMP
```

Operands: One, which indicates the desired output level.

```
DUMP n
```

Where n= 0 1 or 2. The significance of each being as follows:

n = 0. This is the default option and indicates the lowest level of output. This produces output of the type most suited to the beginner or casual user where output is kept to an essential minimum.
n = 1. This is the level of output that an experienced user may require or someone who is exploring a given crystal system for the first time. It provides the user with more data than the previous option.
n = 2. This is the diagnostic level of output and is primarily of interest to programmers tracing errors or who wish to modify or examine the workings of the code.

Subsequent Data Cards: None.

This directive may be used more than once in the input dataset if the user wishes to alter the output level in run time. Note that the dump level alters the quantity of output produced by the PRINT options listed earlier, but does not prevent them being printed altogether.

## (xiv) Directive 14. POTENTIAL.

The POTENTIAL directive is a signal to the input processor that the data describing the short range potential energy functions are to follow. On encountering this directive in the input dataset the program CASCADE enters the subroutine POTENT, which processes the potential data. The subroutine POTENT has for this purpose its own input processor to handle the potential data. This too uses codewords in the form of subdirectives to qualify the data being input. These will be deacibed below.

Format:

POTENTIAL (Acceptable abbreviation: POTE)

Operands: None.

Subsequent Data Cards: Any number of data cards may follow including the subdirectives. These subdirectives and their assosciated data are as follows:

(a) Subdirective 1. SPECIES.

This subdirective indicates that a description of the different species present in the crystal follows. This is followed by any number of data cards, one for each species, giving the species label, species type (core or shell), the species charge and the mass. (The latter quantities relate to the shell model for a given ion described as core - shell pair.) The sequence of cards is terminated by a card with the single entry ENDS.
Example:

```
SPECIES
NA+ CORE -1.140 1.0
CL- CORE 1.497 1.0
NA+ SHELL 2.128 0.0
CL- SHELL -2.485 0.0
ENDS
```

Where the first real number on each card refers to the charge and the second real number refers to the mass.

(b) Subdirective 2. BUCK.

This subdirective indicates that the data relevant to one of the short range Buckingham potential functions follows.i e.

V(r) = A*exp(-r/RHO) - C/r**6

The subdirective card has the label and type of the two species relevant to the potential function, written after the BUCK subdirective. It is assumed that the potential function is divided into different ranges and the cards that follow this subdirective each specify the parameters appropriate to one range. It is further assumed that the ranges will be presented in order of increasing inter - species separation. The series of cards is terminated by a card with the single entry ENDS.
Example:

```
BUCK NA+ SHELL NA+ SHELL
7895.4 0.1709 29.06 4.1835
ENDS
```

Where the first real number is the pre - exponential factor A in the Buckingham formula, the second is the constant RHO within the exponent, the third is the Van der Waals constant C and the fourth is the outer limit RMAX of the range over which the potential applies. The appropriate units for these parameters are electron volts and Angstroms. When more than one range is used for a potential function, then the outer limit for a given range is taken to be the inner limit of the next function range.

(c) Subdirective 3. MOD1.

The subdirective MOD1 indicates that the data cards following it describe the potential of a modified Buckingham function. i.e.

V(r) = A*exp(-r/RHO) - C/r**N

The input of the MODI subdirective is identical to that of the Buckingham function described above except for the use of the MODI codeword in place of BUCK and the presence of an integer constant N on the subdirective card indicating the order of the power law describing the Van der Waals factor.
Example:

```
MODI NA+ SHELL NA+ SHELL 8
7895.4 0.1709 29.06 4.1835
ENDS
```

Where the numerical constants have the same significance as in the previous example. The presence of the number 8 on the MODI subdirective card indicates that the Van der Waals factor has a 1/r**8 power law Here again the units are Angstroms and electron volts.

(d) Subdirective 4. LENN.

This directive indicates that the data describing a Lennard - Jones potential function is to follow i.e. a function of the form:

V(r) = A/r**N - B/r**M

In this instance, as with the previous potential functions, the subdirective card specifies the labels and type (core or shell) of the two interacting species and it also specifies the indices N and M in the above expression. Here again it is assumed that the potential function is divided into ranges that are presented in order of increasing inter species separation, one data card representing one such range. The sequence of cards is terminated by a single card with the entry ENDS.
Example:

```
LENN CL- SHELL CL- SHELL 12 6
8560.7 30.67 5.78
ENDS
```

Where the subdirective card specifies the species concerned and the required indices (N = 12 M = 6) of the expression. The real numbers 8560.7, 30.67 and 5.78 represent the constants A and B respectively and the maximum range for the function (RMAX). The appropriate units are electron volts and Angstroms.

(e) Subdirective 5. MORSE.

This is the subdirective used to input a Morse potential function i.e.

V(r) = A*(1-exp(-B(r-C)))**2

Where A,B and C are constants to be supplied. The method of reading in this function is the same as for the previous cases.
Example:

```
MORSE CL- SHELL CL- SHELL
23.57 0.5 2.46
ENDS
```

Where the three real numbers correspond to the constants A,B and C of the above expression. The units are the same as the previous cases.

(f) Subdirective 6. HARM.

This subdirective indicates the input of the harmonic spring constants of the core - shell model. i.e.

V(r) = C*r**2

Where C is the so - called force constant. Unlike the previous potentials the relevant data are placed all on one card with the HARM subdirective.
Example:

```
HARM NA+ CORE NA+ SHELL 96.44
```

Where 96.44 is the force constant between the core and shell concerned. The units are again electron volts and Angstroms.

(g) Subdirective 7. ENDS.

The potential data input is terminated by a single card with the entry ENDS.

The following points are of importance when preparing the potential data.

The POTENT subroutine in its present form will not allow the use of more than one type of short range potential except when the second potential type is the harmonic potential. Any attempt to run CASCADE for a mixed potential function system will result in an error message and program termination. This is a temporary restriction only and later models will not be so hindered.

The POTENT subroutine is designed so that it is necessary to read in the data associated with the SPECIES subdirective before the potential data. An error message will result if this sequence is not adhered to. This is necessary because the species data are needed to classify the potentials correctly. The potentials themselves however may be presented in any order.

The species labels and types must be consistent with the information given to the BASIS and DEFECT directives mentioned earlier. The program will terminate prematurely if such inconsistencies are found.

The POTENT subroutine input processor has been written with the same philosophy as the main input processor and will process as much of the potential data as is possible, ignoring user errors where it can. If such errors are found however, the program will terminate before it can proceed to the next directive in the main program.

### (xv) Directive 15. MAXIT.

The MAXIT directive is used to specify the maximum number of iterations the program is allowed to complete in the relaxation of the crystal coordinates about the defect. The program default is currently 25 such iterations. This directive allows the user to change the specification.

Format: MAXIT

Operands: One, specifying the new maximum iteration number (integer).

MAXIT nnn
Where nnn = new iteration number (integer).

Subsequent Data Cards: None.

### (xvi) Directive 16. START.

The START directive is responsible for controlling the major operations of the crystal defect calculation. The program CASCADE may be thought of as performing four fundamental operations. The first of these is the preliminary processing of data such as reading in the fundamental lattice vectors, the basis of the unit cell, the defect vectors and determining the symmetry of the system. The second operation is the calculation of the lattice vectors for the entire crystal region and determining the symmetry variables etc. (i.e. CRYSTAL definition). The third operation is the calculation of the perfect lattice properties such as lattice energy dielectric properties etc (i.e. a PLUTO calculation). The last operation is the relaxation of the crystal about the defect and the determination of the defect energy etc. (i.e. a RELAXATION calculation). The START directive is used to control the program when proceeding with each of the three operations crystal definition, perfect lattice calculation and crystal relaxation.

Format START

Operands: One operand is needed and it must be one of the following codewords:

CRYSTAL – When this codeword is used the program CASCADE will proceed with the crystal definition. The codeword indicates to the input processor to enter the CRYDFN subroutine.
PLUTO – When this codeword is used the program proceeds with the calculation of the perfect lattice properties. This codeword indicates to the input processor to enter the PLUTO subroutine.
RELAX – When this codeword is used the program proceeds with the calculation of the relaxed defect crystal. This option indicates to the input processor to enter the subroutine RELAX.

Subsequent Data Cards: None.

Example:

START CRYSTAL
Or:
START PLUTO
Or:
START RELAX

It can be seen that for a complete defect calculation three START directive cards will be required. The advantage of this method of controlling the calculation is that when used in conjunction with the STOP directive (see below), it is possible to stop the calculation at the end of any one of these operations. This can be very convenient when exploring a new crystal system. It is worth noting that the START directive is not obeyed if the program has detected errors in the previous operations. (The program in fact terminates.)

### (xvii) Directive 17. STOP.

The STOP directive does exactly what it says, it stops the program execution immediately it is encountered in the input dataset. Thus it allows the user to end an execution at whatever point is convenient.

Format:

STOP

Operands: None.

Subsequent Data Cards: None.

### (xviii) Directive 19, FREEZE,

This directive has not yet been implemented. Its function is to instruct the program to fix or 'freeze' the coordinates of the species of the region IIA during a relaxation calculation. It will be fully specified at a later date.

### (xix) Directive 20. RESTART.

This directive has not yet been implemented. The proposed directive will enable the defect calculation to be resarted at some semi relaxed stage after a given calculation has failed through time limitations.

### (xx) Directive 21. RESET.

The RESET directive enables the user to input several crystal systems if it is desired. If this is what the user wishes it will be appreciated that the program control variables will have to be reset before a second calculation can begin. The RESET directive removes all data relevant to the previous calculation from the body of the program.

Format:

RESET

Operands: None.

Subsequent Data Cards: None.

### (xxi) Directive 22. PLOT.

The PLOT directive is an instruction to the program CASCADE to prepare an output dataset complete with the coordinates of the perfect and defective crystal lattices. This data is then used as input data to a general plotting package. Users should note that this option will require different CRAY JCL to the usual jobs, since it needs to create and store a new dataset containing the plotting data.

Format:

PLOT

Operands: None.

Subsequent Data Cards: None.

### (xxii) Directive 23. SYMCHECK.

The SYMCHECK directive is used when the user wants the program to check that it has defined the defect crystal symmetry correctly. This need may arise for example if he is unsure that he has specified the tolerance on the ACCURACY directive correctly or he may simply wish to perform the checking routinely. The directive results in a call to the subroutine SYMCHK, which checks that the set of symmetry operations the program applies to the defect crystal do in fact form a group.

Format:

SYMCHECK (Acceptable abbreviation: SYMC)

Operands: None.

Subsequent Data Cards: None.

## 5./ THE INPUT DATASET STRUCTURE

The input processor described above allows some degree of flexibility in the order in which the data are input. This flexibility is not as great as might be imagined however because there is a need for most of the data to be presented in a logical sequence. This is because most of the calculated quantities are dependent on the data of previous steps. Thus as a general rule it is necessary to be aware of the requirements of the defect calculation at each stage to be sure that the data is presented in the correct logical order. As it is unlikely that a novice will be fully aware of these needs, a standard sequence for the directives is presented here which refers to the requirements of a full defect calculation. The experienced user may then alter the standard sequence to suit his own requirements.

### Standard Input Sequence for Defect Calculation.

Item 1. Directive 3, DIMENSION. (Mandatory).

Item 2. Directive 1, TITLE. (Optional).

Item 3. Directive 13, DUMP. (Optional).

Item 4. Directive 2, PRINT. (Optional).

Item 5. Directive 11, CUBIC or
Directive 12, HEXAGONAL. (One of which is mandatory).

Item 6. Directive 8, REGION. (Mandatory).

Item 7. Directive 9, OPTICAL, or
Directive 10, THERMAL. (One of which is mandatory).

Item 8. Directive 15, MAXIT. (Optional).

Item 9. Directive 22, PLOT. (Optional).

Item 10. Directive 7, ACCURACY. (Optional).

Item 11. Directive 4, LATTICE. .(Mandatory).

Item 12. Directive 5, BASIS. (Mandatory).

Item 13. Directive 6, DEFECTS. (Mandatory).

Item 14. Directive 23, SYMCHECK. (Optional).

Item 15. Directive 16, START CRYSTAL. (Mandatory).

Item 16. Directive 14, POTENTIALS. Followed by subdirectives:
SPECIES
BUCK/MODI/LENN/MORS (one type only)
HARM
(All items under POTENTIALS directive are mandatory).

Item 17. Directive 16, START PLUTO. (Mandatory).

Item 18. Directive 16, START RELAX. (Mandatory).

Item 19. Directive 17, STOP. (Mandatory).

Item 20. Directive 21, RESET. (Optional).

The above sequence may be modified a number of ways. A trivial example would be to insert a TITLE directive between items 15 and 16. This is frequently done to provide a title for the potential data in the output. A more complicated change would be to place a DUMP directive between each pair of items from 10 to 18. This would allow finer control over the output level at each stage of the calculation. Similarly, the PRINT directive could be used between items 14 to 18 to control what is output at each stage. Also the STOP directive may be inserted between any two directives if the user wants the program to terminate at any point. Note that the STOP directive renders the subsequent data in the input dataset redundant.

The user is warned against using directives to repeat or respecify parts of a calculation. Such an attempt will almost certainly cause termination of the program. The user may repeat a calculation only after using the RESET directive and then including all the required data again after it. The user is also warned against using the mandatory directives presented above in any other sequence order than that given. If he does so he runs the risk of having the program fail through illogical sequencing. Note however, that it is permissible to omit some of the directives if a full defect calculation is not required, but even then the user must take care to ensure that all the directives appropriate to the required task are present in the input dataset and that their sequence is logically correct (i.e. following the sequence given above).

The next section provides examples of input datsets.

## 6./ EXAMPLE DATASETS.

### (1) SODIUM CHLORIDE LATTICE.

```
DIMENSION 180000
TITLE


                SODIUM CHLORIDE ANION SUBSTITUTED BY CATION



ENDS
DUMP 1
PRINT REG1 REG2 STRU RECIP SYMMETRY
CUBIC MATRICES
REGION 20 4.94 2.789 1.5 0.1 0.2 0.0
THERMAL
MAXIT 15
PLOT
LATTICE VECTORS
1.0 1.0 0.0
1.0 0.0 1.0
0.0 1.0 1.0
BASIS VECTORS AND CENTRE OF SYMMETRY
NA+ CORE 0.0 0.0 0.0
NA+ SHELL 0.0 0.0 0.0
CENTRE 1.0 0.0 0.0
CL- CORE 1.0 0.0 0.0
CL- SHELL 1.0 0.0 0.0
ENDS
DEFECTS
CL- CORE VACANCY 1.0 0.0 0.0
CL- SHELL VACANCY 1.0 0.0 0.0
NA+ CORE INTERSTITIAL 1.0 0.0 0.0
NA+ SHELL INTERSTITIAL 1.0 0.0 0.0
ENDS
START CRYSTAL
TITLE OF POTENTIAL DATA

                SODIUM CHLORIDE POTENTIALS

ENDS
POTENTIALS
SPECIES
NA+ CORE -1.140 1 0
CL- CORE 1.497 1 0
NA+ SHELL 2.120 0.0
```

```
CL- SHELL -2.485 0.0
ENDS
BUCK NA+ SHELL NA+ SHELL
7895.4 0.1709 29.06 4.1835
ENDS
BUCK CL- SHELL CL- SHELL
1227.2 0.3214 29.06 4.1835
ENDS
BUCK NA+ SHELL CL- SHELL
2314.7 0.2903 0.0 4.1835
ENDS
HARM NA+ CORE NA+ SHELL 96.44
HARM CL- CORE CL- SHELL 29.38
ENDS
START PLUTO
START RELAX
STOP
```

It is worth noting here that on some of the directive cards the directive has been expanded into a fuller description of its function e.g. LATTICE to LATTICE VECTORS. This expansion is permissible if the directive card has no operands. The additional wording is ignored.

## (ii) ALUMINIUM OXIDE LATTICE.

```
DIMENSION 180000
TITLE OF DATASET


          AL3+ VACANCY (V''') IN ALUMINIUM OXIDE



ENDS
DUMP 1
PRINT REG1 REG2 STRU RECIP CRYSTAL SYMMETRY
HEXAGONAL MATRICES EMPLOYED
REGION  80  2.487  4.3963  0.85  0.10  0.2  0,0
THERMAL
PLOT
LATTICE VECTORS
0.550810776   0.318010749   2.0
0.0   0.636021501   1.0
0.0   0.0   3.0
BASIS VECTORS AND CENTRE OF SYMMETRY
AL3+   CORE   0.0   0.0   0.0
AL3+   SHELL   0.0   0.0   0.0
AL3+   CORE   0.0   0.0   1.5
AL3+   SHELL   0.0   0.0   1.5
O2-   SHELL   -0.3396597525   0.0   -0.32205
O2-   CORE   0.1698298761   -0.2941539742   -0.32205
O2-   SHELL   0.1698298761   -0.2941539742   -0.32205
O2-   CORE   0.1698298761   0.2941539744   -0.32205
O2-   SHELL   0.1698298761   0.2941539744   -0.32205
CENTRE   0.0   0.0   0.0
AL3+   CORE   0.0   0.0   0.8559
AL3+   SHELL   0.0   0.0   0.8559
AL3+   CORE   0.0   0.0   -0.6441
AL3+   SHELL   0.0   0.0   -0.6441
O2-   CORE   0.3396597525   0.0   1.17795
O2-   SHELL   0.3396597525   0.0   1.17795
O2-   CORE   -0.1698298761   0.2941539744   1.17795
O2-   SHELL   -0.1698298761   0.2941539744   1.17795
O2-   CORE   -0.1698298761   -0.2941539742   1.17795
O2-   SHELL   -0.1698298761   -0.2941539742   1.17795
O2-   CORE   -0.3396597525   0.0   -0.32205
ENDS
DEFECTS
AL3+   CORE   VACANCY   0.0   0.0   0.0
AL3+   SHELL   VACANCY   0.0   0.0   0.0
ENDS
START CRYSTAL
TITLE OF POTENTIAL DATA

          AL2O3 POTENTIALS

ENDS
POTENTIALS
SPECIES LIST
AL3+ CORE 3.0 1.0
O2- CORE -0.026 1.0
AL3+ SHELL 0.0 0.0
O2- SHELL -1.974 0.0
ENDS
BUCK AL3+ SHELL AL3+ SHELL
28335.8 0.124670 0.366011 2.64589
0.0 1.0 0.0 10000000.0
ENDS
BUCK AL3+ SHELL O2- SHELL
2826.65 0.258549 2.30525 1.69337
2905.92 0.256896 1.16670 2.43421
2800.47 0.257803 0.895286 4.23342
0.0 1.0 0.0 10000000.0
ENDS
BUCK O2- SHELL O2- SHELL
267.652 0.439343 0.383073 1.79920
872.655 0.367087 70.3754 2.64589
1014.46 0.356088 57.2077 4.23342
0.0 1.0 0.0 10000000.0
ENDS
HARM AL3+ CORE AL3+ SHELL 10000000.0
HARM O2- CORE O2- SHELL 16.0
ENDS
START PLUTO
START RELAX
STOP
```

## (iii) LITHIUM ALUMINATE LATTICE.

```
DIMENSION   25000
TITLE


          LITHIUM   ALUMINATE



ENDS
DUMP    1
PRIN   REGI   RECI   STRU
CUBIC  ROTATION  MATRICES
REGI  148  2.50  6.42  1.0774  0.10  5.  1.0774
THERMAL
MAXI    35
LATT
1.0  0.0  0.0
0.0  0.981619937694703  0.0
0.0  0.0  0.719626168224298
BASI
AL3+    CORE   0.250   0.245404984423676   0.162635514018691
AL3+    SHEL   0.250   0.245404984423676   0.162635514018691
AL3+    CORE   0.750   0.736214953271027   0.556990654205606
AL3+    SHEL   0.750   0.736214953271027   0.556990654205606
O2-    CORE   0.250   0.0157059193115   0.309439252336448
O2-    CORE   0.750   0.965914018691588   0.410186915887849
O2-    CORE   0.250   0.475104049844235   0.309439252336448
O2-    CORE   0.750   0.506515887850465   0.410186915887849
O2-    SHEL   0.250   0.0157059193115   0.309439252336448
O2-    SHEL   0.750   0.965914018691588   0.410186915887849
O2-    SHEL   0.250   0.475104049844235   0.309439252336448
O2-    SHEL   0.750   0.506515887850465   0.410186915887849
O2-    CORE   0.018   0.245404984423676   0.010794392523364
O2-    CORE   0.982   0.736214953271027   0.7088317757933
O2-    CORE   0.482   0.245404984423676   0.010794392523364
O2-    CORE   0.518   0.736214953271027   0.7088317757933
O2-    SHEL   0.018   0.245404984423676   0.010794392523364
O2-    SHEL   0.982   0.736214953271027   0.7088317757933
O2-    SHEL   0.482   0.245404984423676   0.010794392523364
O2-    SHEL   0.518   0.736214953271027   0.7088317757933
LI+    CORE   0.55274    0.53958    0.17859
LI+    CORE   0.94626    0.53958    0.17859
LI+    CORE   0.55274    0.93285    0.17859
LI+    CORE   0.94626    0.93285    0.17859
LI+    CORE   0.44726    0.44204    0.54104
LI+    CORE   0.05274    0.44204    0.54104
LI+    CORE   0.44726    0.04877    0.54104
LI+    CORE   0.05274    0.04877    0.54104
LI+    CORE   0.250   0.73621    0.17130
LI+    CORE   0.750   0.24540    0.54833
CENTRE    0.750    0.24540    0.54833
ENDS
DEFECT
LI+    CORE    VACA    0.750    0.24540    0.54833
LI+    CORE    INTE    0.750    0.24540    0.54833
ENDS
```

```
START    CRYS
TITLE

        LI5ALO4    LIALMOD1    WITH    NEW    COORDINATE    SET    POTENTIAL..

ENDS
POTE
SPEC
LI+    CORE    1.    6.9390
O2-    CORE    0.8106    15.9994
O2-    SHEL    -2.8106    0.0
AL3+    CORE    1.6170    26.9820
AL3+    SHEL    1.3830    0.0
ENDS
BUCK    LI+    CORE    O2-    SHEL
262.53703    0.3476    0.0    6.91703640
ENDS
BUCK    LI+    CORE    LI+    CORE
916.53970    0.1427    0.0    6.91703640
ENDS
BUCK    O2-    SHEL    O2-    SHEL
22764.3    0.1490    27.88    6.91703640
ENDS
BUCK    LI+    CORE    AL3+    SHEL
5187.642    0.13    0.0    6.91703640
ENDS
BUCK    O2-    SHEL    AL3+    SHEL
1460.3    0.299120    0.0    6.91703640
ENDS
HARM    O2-    CORE    O2-    SHEL    103.070
HARM    AL3+    CORE    AL3+    SHEL    92.488
ENDS
 START    PLUT
 START    RELA
 STOP
```

## 7./ ACKNOWLEDGEMENTS.

I am indebted to many people for advice in the development of the program CASCADE. In particular, I am indebted to J. Kendrick of I.C.I. Corporate Laboratory, Runcorn and M. Guest, M. Leslie and V.R. Saunders of Daresbury Laboratory.

W. Smith.  
Daresbury Laboratory  
January 1981