

External Objects

This section describes the supplied programs, subprograms, and help routines that help simplify and standardize the model creation process. These utilities can be invoked by the supplied models or by user-written models.

Note:

The source code for external objects is not supplied.

This section covers the following topics:

- Introduction
- Natural-Related Subprograms (CNU*)
- Natural-Related Help routines (CNH*)
- Natural Construct Generation Utility Subprograms (CSU*)
- Predict-Related Subprograms (CPU*)
- Predict-Related Help routines (CPH*)
- General Purpose Generation Subprograms (CU--*)

Introduction

All model subprograms use external parameter data areas (PDAs) stored in the SYSCST library. The source for the PDAs is provided and contains details about each parameter. For example, some of the listings for the CPAEL PDA are:

Parameter	CPAEL	Library	SAG	DBID	18	FNR	4
Command							> +
I T L	Name		F	Leng	Index/Init/EM/Name/Comment		
Top	-----						
	1	CPAEL					
	2	INPUTS					
	3	FILE-NAME	A	32	/* File Name.		
	3	FIELD-NAME	A	32	/* Field name to be found in the		
	3	#SIMPLE-OUTPUTS-ONLY	L		/* True if interested in		
	*				/* #FIELD-FOUND only		
	*				/* given file		
	2	INPUT-OUTPUTS					
	3	FILE-CODE	P	8	/* If this code is known,		
	*				/* NSC checks are avoided.		
	3	DDM-PREFIX	A	16	/* Field prefix on DDM,		
	*				/* this will be set if correct		
	*				/* FILE-CODE is not provided.		
	2	SIMPLE-OUTPUTS					
	3	#FIELD-FOUND	L		/* True if field found on file		
	3	FIELD-IS-REDEFINED	L		/* The field is redefined.		

					S 70	L 1	

CPAEL contains a level 1 structure called CPAEL. Depending on the type of parameter, the remaining parameters are grouped into the following structures: INPUTS, INPUT-OUTPUTS, and OUTPUTS. This layout is the same for all PDAs used by the supplied subprograms.

Note:

Be careful when modifying fields in the INPUT-OUTPUTS structure; these fields may retain information across multiple calls.

You can define the PDAs as local data areas (LDAs) within the model subprograms that invoke the utilities. CPAEL is the PDA corresponding to the CPUEL subprogram utility, which returns information about a field in Predict.

The following example shows a model subprogram that requires field information from Predict:

```

DEFINE DATA PARAMETER
  PARAMETER
  .
  .
  .
LOCAL USING CPAEL
LOCAL USING CSASTD
  .
  .
  .
END-DEFINE
  .
  .
  .
ASSIGN CPAEL.FILE-NAME = #PDAX-FILE-NAME
ASSIGN CPAEL.FIELD-NAME = #PDAX-FIELD-NAME
CALLNAT 'CPUEL' CPAEL CSASTD
*
*Check outputs of CPUEL
  .
  .
  .
END

```

This section provides a brief description of the supplied program, subprogram, and help routine utilities. For examples of how to invoke the utilities, refer to the source code for the supplied model subprograms in the SYSCST library (prefixed by CU).

Note:

Driver programs for many of the supplied model programs and subprograms are included on the Natural Construct tape (prefixed by CTE). These driver programs are also available through the Drivers menu option on the Administration main menu. If a driver program is available, its location is listed in the *Drivers Menu Option* section for the program or subprogram. For information about invoking the driver programs, see Drivers Menu Function.

This section covers the following topics:

- Object Categories
- Error Processing

- Passing of Structure Names
- Restricted Data Areas
- Callback Functions
- Subprogram Chaining

Object Categories

The supplied objects are divided into three categories, based on the type of information they access. Each category is identified by its prefix as follows:

Prefix	Object Categories
CN*	Identifies objects that return or generate data based on information in the Natural system files.
CP*	Identifies objects that return or generate data based on information in Predict.
CS*	Identifies objects that are miscellaneous validation, calculation, or translation routines. Most of these routines do not access system file information, but some access Natural Construct system files.

Whenever possible, use the supplied programs, subprograms, and help routines instead of accessing the system file information directly. This helps protect your programs from unwanted changes to the internal structure. Natural Construct maintains the upward compatibility of the supplied programs, subprograms, and help routines.

Error Processing

Many of the supplied subprograms return information through the CSASTD parameter data area (PDA). The value in the RETURN-CODE field should be checked after each call. If it is not blank, it should be passed back to the generation nucleus so the user is aware of the problem.

The following example shows a model subprogram that invokes the CPUEL utility:

```

DEFINE DATA
  PARAMETER USING CUMYPDA
  PARAMETER USING CU--PDA
  PARAMETER USING CSASTD
  LOCAL USING CPAEL
  .
  .
  .
END-DEFINE
.
.
.
CALLNAT 'CPUEL' CPAEL CSASTD
IF CSASTD.RETURN-CODE NE ' ' THEN
  ESCAPE ROUTINE IMMEDIATE
END-IF

```

Passing of Structure Names

To invoke the supplied subprograms, pass only the level 1 structures in the PDA. This way, if new parameters are added to the utilities in future versions of Natural Construct, you need only recatalog your model subprograms to incorporate the changes.

Restricted Data Areas

Some subprograms have restricted data areas to retain information across multiple calls. The restricted data areas are identified by an R in the third position of the data area name (CPRELNX, for example).

You do not need to be concerned with the contents of these data areas. Define them as local data areas within the invoking subprograms and pass them to the subprogram that is invoked.

Tip:

As with all PDAs, the name of the structure passed to the subprogram always matches the name of the data area itself.

Callback Functions

Many of the Natural Construct utility subprograms iterate through system data and, for each record found, call a user-supplied routine. For example, CPURLRD is used to retrieve all relationships related to a particular file. Rather than returning these relationships to the caller of CPURLRD, the caller must supply the name of a subprogram CPURLRD should call for each relationship found.

These routines accept an A1 array to allow the caller of the utility to communicate information to and from the subprogram called by the utility. This data area is represented by CSAPASS. It is accepted by the utility as a 1:v array so that the actual size of the data area can be determined by the requirements of the caller.

Subprogram Chaining

When a subprogram performs read logical processing and returns a series of records, it is sometimes difficult or inefficient for the subprogram to “remember” where it left off in a previous call. Also, this type of processing can be awkward to code in the invoking object because it must define looping logic and issue iterative CALLNATs until a certain end condition is reached.

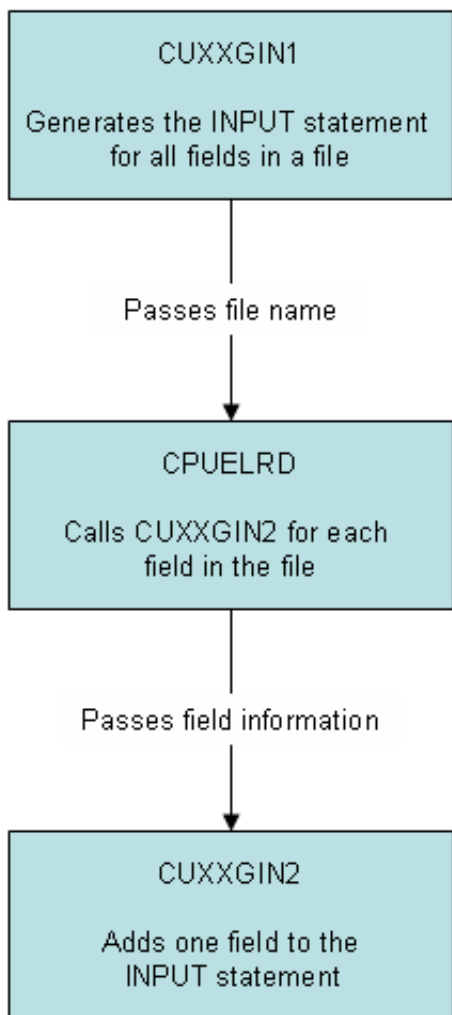
To avoid these problems, some subprograms do not return the information to the calling object. Instead, the calling object passes the name of a subprogram that is invoked for each record encountered. To generate an INPUT statement containing all fields in a file, for example, you can use the CPUELNX and CPUELRD subprograms. This section describes these subprograms.

Without Subprogram Chaining (CPUELNX)

The CPUELNX subprogram can be called iteratively to continually return the next field in the file until an end-of-file condition is reached. The model subprogram that generates the INPUT statement must define the looping logic and make iterative CALLNATs to include each field in the INPUT statement.

With Subprogram Chaining (CPUELRD)

The CPUELRD subprogram can be invoked once by the model subprogram (CUXXGIN1, for example). This subprogram receives the name of a file and a subprogram to CALLNAT (CUXXGIN2, for example). It traverses the file and CALLNATs the subprogram for each field. That subprogram adds the current field to the INPUT statement generated. For example:



To allow CPUELRD to remember information across iterative calls, a 1K area is passed to CUXXGIN2. This area can be redefined into individual fields, such as current status information, that are required by CUXXGIN2 across multiple calls. It can also pass additional information between CUXXGIN1 and CUXXGIN2.

Note:

For an example of how subprogram chaining is used, refer to the CUFMGIN1 and CUFMGIN2 programs in the SYSCST library.

Natural-Related Subprograms (CNU*)

The subprograms described in this section retrieve information from the Natural system files to assist in the generation process. For subprograms that return information about Natural objects (programs, data areas, etc.), the specified data area object must exist in the current library or one of its steplibs.

Tip:

Driver programs for many of the supplied model programs and subprograms are included on the Natural Construct tape (prefixed by CTE). These driver programs are also available through the Drivers menu option on the Administration main menu. If a driver program is available, its location is listed in the *Drivers Menu Option* section for the program or subprogram. For information about invoking the driver programs, see Drivers Menu Function.

This section describes the following subprograms:

- CNUEL Subprogram
- CNUELNX Subprogram
- CNUERMSG Subprogram
- CNUEXIST Subprogram
- CNUGDABL Subprogram
- CNUGDAEL Subprogram
- CNUGENDA Subprogram
- CNUMPPRF Subprogram
- CNUMSG Subprogram
- CNUPEXST Subprogram
- CNUSEL Subprogram
- CNUSRCNX Subprogram
- CNUSRCRD Subprogram

CNUEL Subprogram

CNUEL	Description
What it does	Retrieves information about a field in a local data area (LDA) or parameter data area (PDA). This subprogram receives the name of a field and data area (CNAEL.INPUTS) and returns information about the field (CNAEL.OUTPUTS), such as the structure to which the field belongs, the field format and type, the level number, and the starting and ending index for arrays.
PDA's used	<ul style="list-style-type: none"> ● CNAEL ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSTEM-FUSER ● SYSTEM-FNAT

Drivers Menu Option



```

CTEELN          ***** Natural Related Subprograms *****          CTEELN1
May 07          - Driver for subprogram CNUEL -                          08:09 PM

*Data Area Name : _____
Field Name.....: _____
Structure Name  : _____

View Of Name...:

Field Found....:      Field Format:          Lvl Number....:
Constant Field :      Field Length:
Field Redefined:      Rank.....:
Lvl Type Trail :
From Index  Thru Index  1:V  Field Occurrences
-----

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                          mai
  
```

CNUELNX Subprogram

CNUELNX	Description
What it does	<p>Returns information about the next field in a data area. This subprogram receives the name of an external data area and returns information about the next field in that data area. On the first call to this subprogram, the specified field is returned. On subsequent calls, the next fields are returned.</p> <p>CNRELNX (PDA containing reserved variables) keeps track of the current position of the data area and must not be modified by the calling program.</p> <p>Note: For information about INPUT/OUTPUT parameters, refer to the CNAELNX data area in the SYSCST library.</p>
PDAs used	<ul style="list-style-type: none"> ● CNAELNX ● CNRELNX ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSTEM-FNAT

CNUELNX On Unix Platforms

On Unix platforms, it is necessary to explicitly close any open cursors. CNUELNX does this automatically whenever a data area is read in its entirety. However, if you want the calling program to only read a portion of the data area, you must insert additional code to close the open cursor. For example:

```

/* close the object
IF CNRELNX.NATA1500-END-OF-FILE
  IGNORE
ELSE
  CNAELNX.#CLOSE-OBJECT := TRUE
  CALLNAT 'CNUELNX' CNAELNX CNRELNX CSASTD
END-IF

```

Drivers Menu Option




```

CTENLNX          ***** Natural Related subprograms *****          CTENLNX1
Nov 30,06        - Driver for subprogram CNUELNX -                      09:20 PM

*Data Area Name...: ST5A_____      Field Count: 17      Constant Field :
First Time.....: _                  End Of File:      Dynamic Field...:
Structure Name...: MY-GROUP          Field Redefined:
Field Name.....: ALPHA2-R           Field Format...: A
Field Length....: 5.0                Units: 5         Decimals.....:
View Of Name.....:
Level Number....: 11      Basic Occurrences: _      Rank.....: 1
Level Type Trail.: S S S S S S S S R F
Occurrences Found: X
Starting At: 2_

                                From Index Thru index 1:V Field Occur
Object location  -----
Library: C52_____                      1
DBID...: 13000
FNR...: 1301_
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                mai
    
```

Tip:

As this subprogram can have up to 99 field levels, enter a level number in the Starting At field to display the specified level plus the next nine levels.

CNUERMSG Subprogram

CNUERMSG	Description
What it does	Receives a Natural error message number and returns the error message text. This subprogram receives a Natural error message number (CSASTD.MSG-NR) and returns the corresponding error message text (CSASTD.MSG). For example, the message text for Natural message number 0888 is Storage Overflow During Compilation or Execution.
PDA's used	<ul style="list-style-type: none"> ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSTEM-FNAT

Note:

This subprogram returns system error messages, rather than application error messages. For information about application error messages, see CNUMSG Subprogram.

Drivers Menu Option



```

CTEERMSG          N a t u r a l   C o n s t r u c t          CTERMSG1
Aug 14            D r i v e r   f o r   s u b p r o g r a m   C N U E R M S G          1   o f   1

Msg Nr...: ____  Error Fld:
Ret Code :

Msg:

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                     mai
    
```

CNUEXIST Subprogram

CNUEXIST	Description
What it does	<p>Checks for the existence of a Natural module. This subprogram receives the name of a Natural module and determines whether its source, compiled object, or both exist. If the source and/or compiled object exist, the subprogram returns the module type (P for program) and library name(s) in which the source and/or compiled object(s) were found.</p> <p>If the module is not found in the current library, you can request a search of all steplibs. In this case, the name of the first library in which the module was found is returned.</p>
PDAAs used	<ul style="list-style-type: none"> ● CNAEXIST ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSTEM-FUSER ● SYSTEM-FNAT

Drivers Menu Option



```

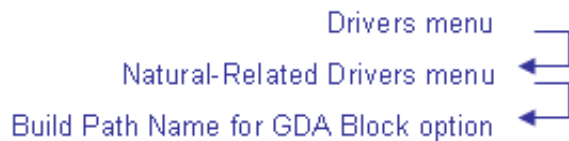
CTEEXIST          ***** Natural Related Subprograms *****          CTEEXIST
Feb 09            - Driver for subprogram CNUEXIST -                    05:31 P

*Object/Source Name.....: _____          Source      Object
Object/Source or Both...: _                -----      -----
Search type...                               Exists.:
Library + Steplib Search: _                   Type...:
or                                              Library:
Specific library search                       DBID...:
Library Name.....: _____                 FNR...:
DBID.....: _____                         User...:
FNR.....: _____                          Date...:
(Blank implies current library)              Time...:
    
```

CNUGDABL Subprogram

CNUGDABL	Description
What it does	Builds a full path name for a global data area (GDA) block. This subprogram receives a GDA name and the name of a GDA block. It returns the full path name from the master block to the specified block. For example, if BLOCK11 is a sub-block of BLOCK1, which is a sub-block of MASTER-BLOCK, the following full path name is returned: MASTER-BLOCK.BLOCK1.BLOCK11
PDA's used	<ul style="list-style-type: none"> ● CNAGDABL ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSTEM-FUSER ● SYSTEM-FNAT

Drivers Menu Option



```

CTEGDABL          N a t u r a l   C o n s t r u c t          CTEGDAB1
Aug 14            Driver for subprogram CNUGDABL          1 of 1

 *GDA Name.....: _____
Block Name.....: _____

Full Path Name:

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
help retrn quit                                          mai
    
```

CNUGDAEL Subprogram

CNUGDAEL	Description
What it does	Verifies that a field is contained in a global data area (GDA). This subprogram receives the name of a GDA and the name of a field. If the field exists in the GDA, this subprogram returns a confirmation flag.
PDA's used	<ul style="list-style-type: none"> ● CNAGDAEL
Files accessed	<ul style="list-style-type: none"> ● SYSTEM-FNAT ● SYSTEM-FUSER

Drivers Menu Option



```

CTEGDAEL          N a t u r a l   C o n s t r u c t          CTEGDAE1
Aug 14            D r i v e r   f o r   s u b p r o g r a m   C N U G D A E L          1 o f   1

*GDA Name...: _____
Field Name : _____
Field Found:

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                     mai
  
```

CNUGENDA Subprogram

CNUGENDA	Description
What it does	<p>Adds a field to a data area. This subprogram receives the definition of a field (field type, level number, field name, field format and length, and the number of occurrences, for example) to be added to a data area and generates the field definition at the end of the current edit buffer.</p> <p>For information about INPUT/OUTPUT parameters, refer to the CNAGENDA data area in the SYSCST library.</p> <p>Note: Before this subprogram is invoked, the calling program must set the Natural editor to a data area type of A, L, or G.</p>
PDAs used	<ul style="list-style-type: none"> ● CNAGENDA ● CNRGENDA ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

To use this utility internally, issue a CALLNAT to the following subprogram immediately after calling CNUGENDA:

```
CALLNAT 'CNUGENDU'
```

There are no parameters for this subprogram.

Drivers Menu Option



```

CTEGENDA          N a t u r a l   C o n s t r u c t          CTEGEND1
Aug 14            D r i v e r   f o r   s u b p r o g r a m   C N U G E N D A          1   o f   1

Field Name: _____

Field Type: _ Format: _ Occurrences: ____
Level.....: _ Length: ____ Comment....: _____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                     mai
    
```

CNUMPPRF Subprogram

CNUMPPRF	Description
What it does	<p>Reads a map profile from a Natural system file. This subprogram receives the name of the map profile in the CSAMPSET.#PROFILE field. It reads the profile from the Natural system file (FNAT) and returns the map settings.</p> <p>For information about the OUTPUT parameters, refer to the CSAMPSET data area in the SYSCST library.</p>
PDA's used	<ul style="list-style-type: none"> ● CSAMPSET ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSTEM-FNAT

Note:

This routine is not available on all platforms.

Drivers Menu Option



```

CTEMPPRF          N a t u r a l   C o n s t r u c t          CTEMPPRF1
Aug 14            Driver for subprogram CNUMPPRF           1 of 1

Map Profile....: _____ Layout.....:                Map Type.....:
Map Version....:                Map Name....:            Std Keys.....:

      +-----+-----+-----+-----+
      | 1__ | Delimiter Class  AD  CD  Delimiter Char |
DC:   |-----|-----|-----|-----|
PS:   |-----|-----|-----|-----|
LS:   |-----|-----|-----|-----|
ZP:   |-----|-----|-----|-----|
      +-----+-----+-----+-----+

Write Statement:                CV.....:                Justification:
Input Statement:                Error Code..:            Enforce Attr :
Auto Rule Rank :                Hlp Fld Dflt:
Fill Character :                Help.....:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help retrn quit                bkwrđ frwrđ                mai
  
```

CNUMSG Subprogram

CNUMSG	Description
What it does	<p>Returns application message text from the SYSERR message file. This subprogram receives the following input:</p> <ul style="list-style-type: none"> • message number • message library (CSTMSG by default) • message text • substitution data members • message libraries for data members (CSTLDA by default) • retrieval method • default languages (used if message number is not located using *Language) <p>It processes message text based on one of the following retrieval methods:</p> <ul style="list-style-type: none"> • R <p>Performs text retrieval based on message numbers. A message number can be entered in either the Message Number or Message Text (Input) field. If a message number is entered in the Message Number field, the corresponding text is retrieved from the message library (CSTMSG by default) and displayed at runtime. If the Message Number field is blank, the subprogram scans the Message Text (Input) field for a message number. If one is located, it is replaced with its corresponding text from the message library.</p> <p>For example, assume message number *2309 corresponds to the message text ":1::2::3:does not exist". If this message number is located in either the Message Number or Message Text (Input) fields, the subsystem will retrieve the message text ":1::2::3:does not exist".</p> • S <p>Performs text substitutions in the Message Text (Input) field. A substitution will occur if placeholders are found in the message text. Placeholders are replaced at runtime with a value entered in one of the Message Substitution Data fields (1, 2, and 3). Placeholders are entered in the following format: ":N:", where N identifies one of the three Message Substitution Data fields.</p> <p>For example, if you enter the following message text: ":1::2::3:does not exist", and the Message Substitution Data field 1 is "File", and the Message Substitution Data field 2 is "NCST-CUSTOMER", the message text "File NCST-CUSTOMER does not exist" is returned.</p> • B <p>Performs text retrieval using methods R and S. This method also supports inline retrieval and substitution; that is, typing the message number and substitution values directly in the Message Text (Input) field.</p> <p>For example, if you enter the following entry in the Message Text (Input) field: "*2309.*2075.1.NCST-CUSTOMER", the subprogram assigns 2309 as the message number and retrieves the message ":1::2::3:does not exist". The first substitution value is retrieved from message 2075.1, which is "File". The second substitution value is the text "NCST-CUSTOMER". At runtime, "File NCST-CUSTOMER does not exist" is displayed.</p> <p>If you are using message numbers, you can specify up to eight default languages. If the message text for the message number is not found using the currently selected language (*Language), the subprogram will search for the message in each of the specified default languages.</p> <p>The search begins with the *Language code specified in the first Default Language field through to the last Default Language field in which a code is specified. If the message is still not located, the subprogram will search the message text for the default system *Language code of 1 (English).</p> <p>Note: You can center text entered in the Message Text (Input) field using the ",+/NN" notation, where NN is the number of characters to be centered. For more information about message numbers and placeholders, see Use SYSERR References.</p>

CNUMSG	Description
PDA's used	<ul style="list-style-type: none"> • CNAMSG • CSASTD
Files accessed	<ul style="list-style-type: none"> • SYSTEM-FUSER

Drivers Menu Option



```

CTEMSG          ***** Natural Related subprograms *****          CTEMSG1
Oct 16          - Driver for subprogram CNUMSG -                      08:53 AM

Message Number.: 0008  *Message Library: CSTMSG__
Message Text (Input)
_____

Retrieval Method: R ('R' for Retrieve, 'S' for Substitute, 'B' for Both)

Message Substitution
Data(1): _____ *Message Library: CSTLDA__
Data(2): _____ *Message Library: CSTLDA__
Data(3): _____ *Message Library: CSTLDA__

Default Languages
*LANGUAGE: 1 1) 1_ 2) 1_ 3) 1_ 4) 1_ 5) 1_ 6) 1_ 7) 1_ 8) 1_

Response Code: 0      ( 9 - unsuccessful )

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                          mai
    
```

CNUPEXST Subprogram

CNUPEXST	Description
What it does	Checks for the existence of a map profile. This subprogram receives the name of a map profile and verifies that it exists in the Natural FNAT system file.
PDA's used	<ul style="list-style-type: none"> • CNAPEXST
Files accessed	<ul style="list-style-type: none"> • SYSTEM-FNAT

Note:

This subprogram is not available on all platforms.

Drivers Menu Option



```

CTEPEXST          N a t u r a l   C o n s t r u c t          CTEPXST1
Aug 14           Driver for subprogram CNUPEXST           1 of 1

Map Profile Name.: _____
Map Profile Exists:

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                     mai
  
```

CNUSEL Subprogram

CNUSEL	Description
What it does	Selects fields from data areas (local or parameter). This subprogram receives the name of a local (LDA) or parameter data area (PDA) and browses fields in the data area. To select a field, mark it. If more than one field is marked, only the first field is selected. You can enter "X" to terminate the display or "T" to position the list to the top.
PDA's used	<ul style="list-style-type: none"> ● CNASEL ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

Drivers Menu Option

```

CTESEL          ***** Construct Related Subprograms *****      CTESEL1
Oct 09,96          - Driver for subprogram CNUSEL -                  01:52 PM

>Data Area Name..: _____ Fld Name:

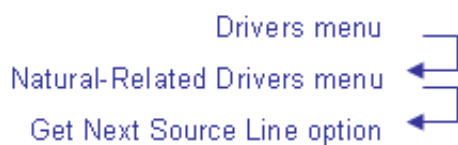
Structure Number:      Field Format:      Field Occurrences
Type Of Field...:     Field Length:
Level Number...:      Units.....:
Total Fields Cnt: 0   Decimals....:

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help retrn quit                                          mai
    
```

CNUSRCNX Subprogram

CNUSRCNX	Description
What it does	Receives the name of the Natural object and returns the next source line. The first call to the subprogram returns the first source line. Subsequent calls return the next lines.
PDA's used	<ul style="list-style-type: none"> ● CNASRCNX ● CNRSRCNX ● CSASTD <p>Note: The CNRSRCNX data area (containing reserved variables) keeps track of the current position of the object source and must not be modified by the calling program.</p>
Files accessed	<ul style="list-style-type: none"> ● SYSTEM-FUSER ● SYSTEM-FNAT

Drivers Menu Option



```

CTESRCNX          N a t u r a l   C o n s t r u c t          CTESRCN1
Aug 14            Driver for subprogram CNUSRCNX           1 of 1

*Object Name: CTELRDSM          Version:
First Time : X                  Include Comments: _

Src Line...:   Userid:          Date...:   - -          Type:
End Of Src :   Level :          Time...:   . . .          SM...

Src Code...:

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                     mai
    
```

CNUSRCRD Subprogram

CNUSRCRD	Description
What it does	<p>Reads source text and performs specified processing. This subprogram receives the name of a Natural object (in the CNASRCRD.#OBJECT-NAME field) and the name of the subprogram invoked to process each source line (in the CNASRCRD.#CALLNAT field). It passes the fields it receives to the subprogram it invokes.</p> <p>CU--PDA, which contains the model parameters, is also passed to CNUSRCRD, as well as CSAPASS (redefined as required). It “remembers” information between calls to the subprogram that processes each source line.</p>
PDAs used	<ul style="list-style-type: none"> ● CNASRCRD ● CU--PDA (model PDA) ● CSAPASS (redefined as required) ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSTEM-FUSER ● SYSTEM-FNAT

Drivers Menu Option



```

CTESRCRD          N a t u r a l   C o n s t r u c t          CTESRCR1
Aug 14            D r i v e r   f o r   s u b p r o g r a m   C N U S R C R D          1   o f   1

*Object Name: _____      Finished:
CALLNAT.....: CTESRCSM        Include Comments: _

Object Information
-----
Type.....:      Version:      Userid:      Time:      . . .
SM.....:      Level...:      Date:      - -

Src Line...:
Source Code:
      :

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                          mai

```

Note:

If you change the name of the subprogram in the CALLNAT field, the specified subprogram must have the same parameters as those in the PDAs used by CNUSRCRD.

Natural-Related Helprouines (CNH*)

You can attach the helprouines in this section to fields that require the input of Natural information (such as object names, message numbers, etc.). They are active helprouines that populate the field to which they are attached.

CNHMDL Helprouine

CNHMDL	Description
What it does	Browses all the Natural Construct models for selection. Valid restriction parameters are: <ul style="list-style-type: none"> ● S (display statement models only) ● M (display program models only) ● B (display all models)
Attached to	Input of a Natural Construct model name.
Parameters used	<ul style="list-style-type: none"> ● #PDA-RESTRICTION(A1) ● #PDA-KEY(A32) (model name)
Files accessed	<ul style="list-style-type: none"> ● NCST-MODEL

CNHMSG Helprouine

CNHMSG	Description
What it does	Browses for and displays the application error message text. You can add new messages to the application by pressing the Add PF-key (the new message number is always adjusted to the next available number).
Attached to	Input of a message number field.
Parameters used	<ul style="list-style-type: none">● #PDA-MESSAGE(A65)● #PDA-MESSAGE-LIBRARY(A8)● #PDA-KEY(N4)
Files accessed	<ul style="list-style-type: none">● SYSTEM-FUSER

CNHOBJ Helproutine

CNHOBJ	Description
What it does	<p>Browses all objects of a specified type in the current library. This helproutine receives an object type and browses all the objects with that type that exist in the current library. Valid object types are:</p> <ul style="list-style-type: none"> ● P (program) ● N (subprogram) ● S (subroutine) ● M (map) ● H (helproutine) ● C (copycode) ● A (parameter) ● G (global) ● L (local) ● T (text) ● * (all) ● 2 (subprogram/helproutine) ● 3 (subprogram/helproutine/subroutine) ● 4 (program/subprogram/helproutine/subroutine) ● 5 (command processor) ● D (data area)
Attached to	Input of a Natural object name field.
Parameters used	<ul style="list-style-type: none"> ● #PDA-TYPE(A1) ● #PDA-KEY(A8) /* Start/Return key
Files accessed	<ul style="list-style-type: none"> ● SYSTEM-FUSER

Natural Construct Generation Utility Subprograms (CSU*)

The subprograms in this section perform specialized functions to assist in the generation process.

Note:

Driver programs for many of the supplied programs/subprograms are available through the Drivers menu option on the Administration main menu. If a driver program is available, its location is listed in the *Drivers Menu Option* section in the program/subprogram description. For more information about the supplied driver programs, see Drivers Menu Function.

These subprograms are:

- CSU-VAR Subprogram
- CSUBANN Subprogram
- CSUBLDRP Subprogram
- CSUBMIT Subprogram (Mainframe)
- CSUBYTES Subprogram
- CSUCASE Subprogram
- CSUCCMD Subprogram
- CSUCENTR Subprogram
- CSUCOMPR Subprogram
- CSUCTRL Subprogram
- CSUCURS Subprogram
- CSUCURS1 Subprogram
- CSUDB2SP Subprogram
- CSUDELEFF Subprogram
- CSUDEFLT Subprogram
- CSUDYNAT Subprogram
- CSUEMLEN Subprogram
- CSUENDX Subprogram
- CSUFDEF Subprogram
- CSUFRVAR Subprogram
- CSUGEN Subprogram
- CSUHEADS Subprogram
- CSUINCL Subprogram
- CSUIS Subprogram
- CSULABEL Subprogram
- CSULENGT Subprogram

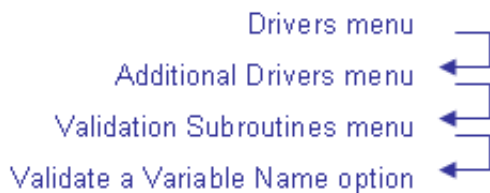
- CSULPS Subprogram
- CSUMAX Subprogram
- CSUMIMAX Subprogram
- CSUMODEL Subprogram
- CSUMORE Subprogram
- CSUMPBOX Subprogram
- CSUMPCPR Subprogram
- CSUMPDUP Subprogram
- CSUMPLAY Subprogram
- CSUMPMMS Subprogram
- CSUMPOVL Subprogram
- CSUMPREG Subprogram
- CSUMPTAB Subprogram
- CSUMPTST Subprogram
- CSUNATFM Subprogram
- CSUNEWX Subprogram
- CSUOG Subprogram
- CSUPARMS Subprogram
- CSUPARTY Subprogram
- CSUPPER Program
- CSUREADS Subprogram
- CSUREF Subprogram
- CSUSCAN Subprogram
- CSUSELFV Subprogram
- CSUSETKY Subprogram
- CSUSETW Subprogram
- CSUSORT Program

- CSUSPLIT Program
- CSUSUB Program (Mainframe)
- CSUSUBP Subprogram
- CSUTEST Program
- CSUTLATE Subprogram
- CSUTRANS Subprogram
- CSUXCHK Subprogram
- CSU2LONG Subprogram

CSU-VAR Subprogram

CSU-VAR	Description
What it does	Validates a specified variable name. This subprogram receives a string and checks for a valid Natural naming convention. Use it whenever a name used as a Natural variable is entered. If the name is invalid, the subprogram returns a message containing the reason. Note: The variable name can be fully qualified (contain a prefix).
Parameters used	<ul style="list-style-type: none"> ● #PDA-STRING(A65) /*INPUT ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

Drivers Menu Option



```

CTE-VAR          ***** Construct Related Subprograms *****      CTE-VAR1
Oct 09           - Driver for subprogram CSU-VAR -                    02:58 PM

String: _____

Msg...:

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                          mai

```

CSUBANN Subprogram

CSUBANN	Description
What it does	Generates the standard banner into the source buffer. Use this subprogram to generate Natural or Visual Basic comments.
PDAs used	<ul style="list-style-type: none"> ● CSABANN ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

CSUBLDRP Subprogram

CSUBLDRP	Description
What it does	<p>Builds a report layout. This subprogram builds a report layout for the Batch, Browse, and Browse-Select models. It can be invoked from a sample subprogram within a user exit. The invoking subprogram must issue an initial RESET statement to clear the structures in CSASELFV. For example:</p> <pre> RESET CSASELFV CSASELFV.GENERAL-INFORMATION CSASELFV.FIELD-SPECIFICATION(*) </pre> <p>The sample subprogram must also contain a SET KEY ALL statement.</p> <p>For an example of how to invoke the CSUBLDRP utility, refer to the CUSCSR subprogram in the SYSCST library.</p>
PDAs used	<ul style="list-style-type: none"> ● CSABLDRP ● CSASELFV ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

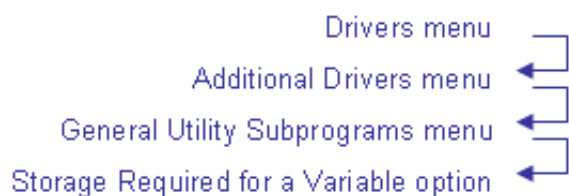
CSUBMIT Subprogram (Mainframe)

CSUBMIT	Description
What it does	Submits a job for execution. The JCL for the job must be in the source buffer. Note: This subprogram is used in conjunction with the CSUSUB command. For more information, see <i>JCL Submit Utility (Mainframe)</i> , <i>Natural Construct Generation</i> .
PDA's used	<ul style="list-style-type: none"> ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

CSUBBYTES Subprogram

CSUBBYTES	Description
What it does	Calculates the required bytes for a field, based on the field's Natural format and length. This subprogram receives the length and format of a field and returns the number of bytes occupied by the field.
PDA's used	<ul style="list-style-type: none"> ● CSABYTES ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

Drivers Menu Option



```

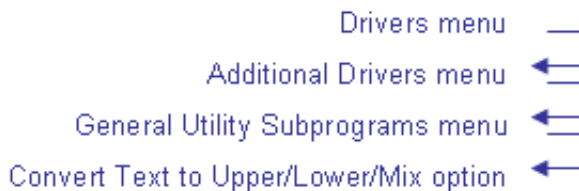
CTEBYTES          N a t u r a l   C o n s t r u c t          CTEBYTE1
Aug 14            D r i v e r   f o r   s u b p r o g r a m   C S U B Y T E S          1   o f   1

Field Format: _      Bytes.....:
Field Length: _____
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                     mai
    
```

CSUCASE Subprogram

CSUCASE	Description
What it does	<p>Converts a string to upper/lower/mixed case. This subprogram receives a string and a desired function. It converts and returns the string as follows:</p> <ul style="list-style-type: none"> ● If the function is "U", this subprogram converts all alpha characters in the string to upper case. ● If the function is "L", it converts all alpha characters to lower case. ● If the function is "M", it converts the alpha characters as follows: <ul style="list-style-type: none"> ○ Removes leading hash (#) or plus (+) characters ○ Replaces all dashes (-) and underscores (_) with blanks ○ Converts the first character, as well as all characters following a dash or underscore, to upper case
PDA's used	<ul style="list-style-type: none"> ● CSACASE ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

Drivers Menu Option



```

CTECASE          N a t u r a l   C o n s t r u c t          CTECASE1
Aug 14           D r i v e r   f o r   s u b p r o g r a m   C S U C A S E          1   o f   1

Function: _      U=Upper, L=Lower, M=Mixed Case
String...: _____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                     mai

```

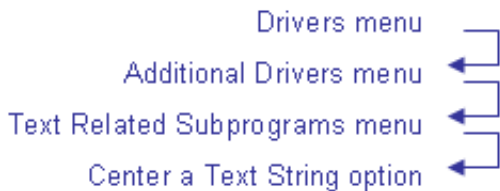
CSUCCMD Subprogram

CSUCCMD	Description
What it does	<p>Generates command block delimiters into the Natural source buffer for super models (generate multiple modules). This subprogram receives a command type, an eight-character module name, a module type, and, optionally, a model name.</p> <p>Natural Construct evaluates the contents of these command blocks after it processes the pre-generation subprogram for the super model. Before continuing the generation, Natural Construct either creates the child model specification or saves, stows, and catalogs the contents of the command block.</p> <p>CSUCCMD must always be called twice — first to initialize the command block and then to close it after generating the contents of the command block.</p> <p>Notes:</p> <ol style="list-style-type: none"> 1. See the CSLCCMD local data area for valid command values. 2. You cannot use nested command blocks.
PDAAs used	<ul style="list-style-type: none"> ● CSACCMD ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

CSUCENTR Subprogram

CSUCENTR	Description
What it does	<p>Centers a text string. This subprogram centers text, such as headings, within a variable. The length passed to this subprogram should be one of the following:</p> <ul style="list-style-type: none"> ● the length of the variable that stores the heading ● the length of the AL parameter that displays the variable that stores the heading
PDAAs used	<ul style="list-style-type: none"> ● CSACENTR ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

Drivers Menu Option



```

CTECENTR          N a t u r a l   C o n s t r u c t          CTECNTR1
Aug 14            Driver for subprogram CSUCENTR           1 of 1

Length:  _____

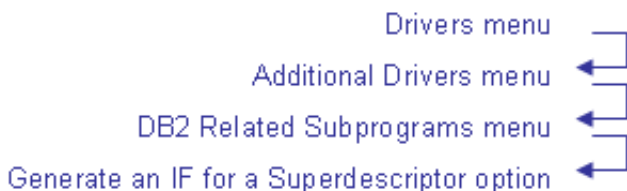
String:  _____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                     mai
  
```

CSUCOMPR Subprogram

CSUCOMPR	Description
What it does	<p>Generates an IF clause for two structures. The subprogram receives two structure names and a list of underlying components to compare. It generates the IF clause according to the criteria requested (LT, LE, GT, GE).</p> <p>Note: DB2 users should use the CSUDB2SP subprogram to compare key values (see CSUDB2SP Subprogram for a description).</p>
PDA's used	<ul style="list-style-type: none"> ● CSACOMPR CSASTD ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

Drivers Menu Option



```

CTECOMPR          N a t u r a l   C o n s t r u c t          CTECOMP1
Aug 14            Driver for subprogram CSUCOMPR           1 of 1

Comparison Operator.: ___   Lhs Structure: _____
Tab.....: ___   Rhs Structure: _____
No. Of Components...: ___

                        Component Fld Name
                        +-----+
1___ | _____ |
    | _____ |
    | _____ |
    | _____ |
    | _____ |
    +-----+

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                bkwrđ frwrđ                mai
    
```

CSUCTRL Subprogram

CSUCTRL	Description
What it does	Retrieves information from the Natural Construct control record and sets the PF-keys, help indicator, underscore characters, position indicators, disable indicator, scroll indicator, “of” right prompt, and dynamic attributes for Natural Construct.
PDAs used	<ul style="list-style-type: none"> ● CU--PDA ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● NCST-CONTROL

CSUCURS Subprogram

CSUCURS	Description
What it does	Determines the position of the field in which the cursor is placed. This subprogram is invoked when runtime translation is requested. It determines the message numbers and positions associated with fields in a translation LDA and invokes the CSUTLATE subprogram to perform runtime translation. For more information, see CSUTLATE Subprogram.
Parameters/PDAs used	<ul style="list-style-type: none"> ● #TRANSLATION-DATA(A1/1:V) ● #SYSERR-APPL(A8) ● #DATA-AREA-NAME(A8) ● #TEXT-REQUIRED(L) ● #LENGTH-OVERRIDE(I4) ● CSACURS ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

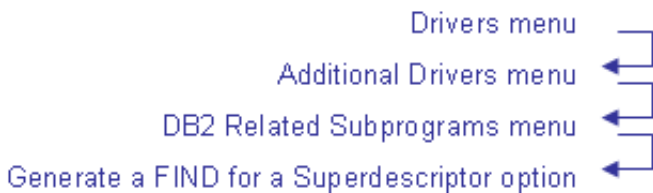
CSUCURS1 Subprogram

CSUCURS1	Description
What it does	Determines the position of a single field in which the cursor is placed. This subprogram is invoked whenever runtime translation of a single field is requested. It determines the message number and position associated with the field and invokes the CSUTLATE subprogram to perform runtime translation. For more information, see CSUTLATE Subprogram.
Parameters/PDAs used	<ul style="list-style-type: none"> ● #TRANSLATION-DATA(A1/1:V) ● #SYSERR-APPL(A8) ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

CSUDB2SP Subprogram

CSUDB2SP	Description
What it does	<p>Generates a FIND statement for a superdescriptor. This statement retrieves DB2 records based on a complex key definition. If a complex key is composed of 5 fields (Field1, Field2, Field3, Field4, and Field5), for example, the generated FIND/WHERE clause is:</p> <pre>Field1 GE #INPUT.Field1 SORTED BY Field1 Field2 Field3 Field4 Field5 WHERE Field2 GE #INPUT.Field2 AND Field3 GE #INPUT.Field3 AND Field4 GE #INPUT.Field4 AND Field5 GE #INPUT.Field5 OR Field1 GE #INPUT.Field1 AND Field2 GE #INPUT.Field2 AND Field3 GE #INPUT.Field3 AND Field4 GT #INPUT.Field4 OR Field1 GE #INPUT.Field1 AND Field2 GE #INPUT.Field2 AND Field3 GT #INPUT.Field3 OR Field1 GE #INPUT.Field1 AND Field2 GT #INPUT.Field2 OR Field1 GT #INPUT.Field1</pre> <p>Note: #INPUT is the qualifier for the RHS fields of the in equations.</p>
PDAs used	<ul style="list-style-type: none"> ● CSADB2SP ● CU--PDA ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

Drivers Menu Option



```

CTEDB2SP          N a t u r a l   C o n s t r u c t          CTEDB2S1
Aug 14            Driver for subprogram CSUDB2SP           1 of 1

*File Name.....: _____ Find Next Record: _
*Field Name.....: _____
Function.....: _____

LHS Structure.....: _____
LHS Index.....: _____
RHS Structure.....: _____
RHS Index.....: _____

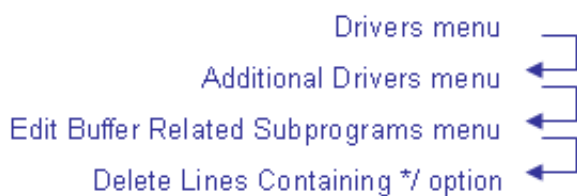
Prefix Length.....: ____
Low Key Structure : _____
High Key Structure: _____

Tab.....:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                     mai
    
```

CSUDELFF Subprogram

CSUDELFF	Description
What it does	Deletes the lines containing */ in the edit buffer. This subprogram searches for and deletes the lines containing */ in the edit buffer. These lines are written by WRITE/PRINT statements when the DEFINE PRINTER OUTPUT 'SOURCE' statement is used.
PDA's used	● None
Files accessed	● None

Drivers Menu Option



```

CTEDELFF          N a t u r a l   C o n s t r u c t          C T E M A P 1
Aug 14           D r i v e r   f o r   s u b p r o g r a m   C S U D E F L F          1   o f   1

+-----+
|               |
|   P R E S S   |
|   E N T E R   |
|   T O   E X E |
|   C U T E .   |
|               |
+-----+

      R e a d   i n   N e w   S o u r c e :   _
      * N e w   S o u r c e   N a m e . . . :   _____
      N e w   S o u r c e   L i b r a r y :   D E V P R ____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                     mai
    
```

CSUDEFLT Subprogram

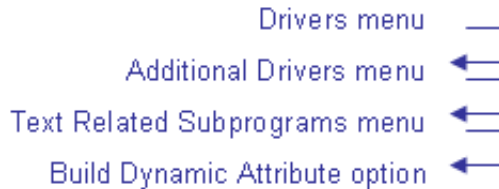
CSUDEFLT	Description
What it does	Provides default specification values for Natural Construct models. This subprogram provides an interface between generated applications and the user-maintained CSXDEFLT sample exit subprogram. To override the default settings, modify CSXDEFLT. The CCDEFLTA, CCDEFLTL, and CCDEFLTN copycode members return defaults for alphanumeric, logical, and numeric values, respectively.
PDA's used	<ul style="list-style-type: none"> ● CSADEFLT ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

CSUDYNAT Subprogram

CSUDYNAT	Description
What it does	<p>Builds parameters containing dynamic attributes. This subprogram receives a set of dynamic attribute characters in the CSADYNA.#ATTRIBUTE-CHARS(A11/1:13) field and builds the definition for the DY= parameter. The positioning within this array indicates the type of dynamic attribute assigned. The positions and attributes are:</p> <ul style="list-style-type: none"> ● 1 (normal intensity) ● 2 (intensified) ● 3 (blinking) ● 4 (cursive/italic) ● 5 (underlined) ● 6 (reversed video) ● 7 (blue) ● 8 (green) ● 9 (neutral/white) ● 10 (pink) ● 11 (red) ● 12 (turquoise) ● 13 (yellow) <p>For example, if you enter:</p> <pre>#ATTRIBUTE-CHARS(1) = '}' #ATTRIBUTE-CHARS(2) = '{'</pre> <p>This subprogram returns:</p> <pre>#DY-PARAMETER = DY={I</pre> <p>If the caller's attributes are printable special characters, they are represented literally. Otherwise, they are represented using the HH syntax.</p> <p>Notes:</p> <ol style="list-style-type: none"> 1. The dynamic attribute character specified in position 1, which corresponds to normal intensity, is always coded at the end of the DY= parameter. 2. Programs containing those represented in hex may not be portable.
PDAs used	<ul style="list-style-type: none"> ● CSADYNAT ● CSASTD

CSUDYNAT	Description
Files accessed	<ul style="list-style-type: none"> • None

Drivers Menu Option



```

CTEDYNAT          N a t u r a l   C o n s t r u c t          CTEDYNT1
Aug 14            Driver for subprogram CSUDYNAT           1 of 1

                Attribute Characters
                -----

(1) Normal Intensity...: _          (8) Green.....: _
(2) Intensified.....: _          (9) Neutral (white)...: _
(3) Blinking.....: _            (10) Pink.....: _
(4) Cursive/Italic....: _        (11) Red.....: _
(5) Underlined.....: _          (12) Turquoise.....: _
(6) Reversed Video....: _        (13) Yellow.....: _
(7) Blue.....: _

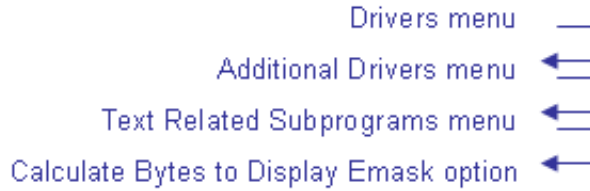
Dynamic Attribute Parameter:

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                     mai
    
```

CSUEMLEN Subprogram

CSUEMLEN	Description
What it does	Determines the number of characters (bytes) required to display an edit mask. This subprogram receives the name of an edit mask and the format of the field to which the edit mask is applied. It returns the number of characters (bytes) required to display the edit mask.
PDA's used	<ul style="list-style-type: none"> • CSAEMLEN • CSASTD
Files accessed	<ul style="list-style-type: none"> • None

Drivers Menu Option



```

CTEEMLEN          N a t u r a l   C o n s t r u c t          CTEMLEN1
Aug 14            D r i v e r   f o r   s u b p r o g r a m   CSUEMLEN          1   o f   1

Edit Mask.....: _____
Field Format...: __

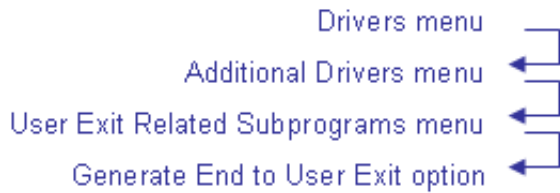
Display Length:

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                     mai
  
```

CSUENDX Subprogram

CSUENDX	Description
What it does	Generates the end of a user exit prompt. This subprogram is used by sample subprograms that generate multiple user exits. Call this subprogram after each user exit is generated. Note: You do not need to call this subprogram after the last user exit.
PDA's used	● None
Files accessed	● None

Drivers Menu Option



```

CTEENDX          N a t u r a l   C o n s t r u c t          CTEMAP1
Aug 14          D r i v e r   f o r   s u b p r o g r a m   C S U E N D X          1   o f   1

+-----+
|               |
|               |
|               |
|               |
|               |
|               |
|               |
|               |
|               |
|               |
+-----+

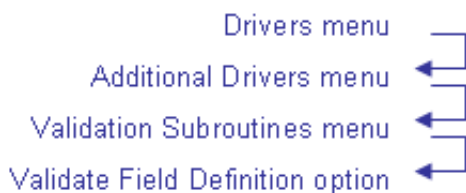
      Read in New Source:  _
      *New Source Name...:  _____
      New Source Library:  DEVPR____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                          mai
    
```

CSUFDEF Subprogram

CSUFDEF	Description
What it does	Validates a field definition. This subprogram receives the Natural format and length of a field and a list of invalid field formats to disallow. To disallow control variables as input variables, for example, list the invalid formats in the CSAFDEF.#INVALID FORMATS field. If the field definition is valid, nothing is returned in CSUFDEF. If the field definition is invalid, CSASTD.MSG and CSASTD.ERROR-FIELD contain an error message and the invalid component of the field (FIELD-FORMAT, DECIMALS, or UNIT).
PDA's used	<ul style="list-style-type: none"> ● CSAFDEF ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

Drivers Menu Option



```

CTEFDEF          N a t u r a l   C o n s t r u c t          CTEFDEF1
Aug 14          D r i v e r   f o r   s u b p r o g r a m   C S U F D E F          1   o f   1

Field Format...:  _          Invalid Formats:  _____
Field Length...:  _____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                          mai
    
```

CSUFRVAR Subprogram

CSUFRVAR	Description
What it does	Returns the parameters and conditions from the model code frames. This subprogram receives a model name and traverses its code frames. It returns the code frame parameters and conditions.
PDA's used	<ul style="list-style-type: none"> ● CSAFRVAR ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● NCST-FRAME-LINES ● NCST-MODEL

Drivers Menu Option



```

CTEFRVAR          N a t u r a l   C o n s t r u c t          CTEFRVR1
Aug 14            D r i v e r   f o r   s u b p r o g r a m   C S U F R V A R          1   o f   1

*Model Name: _____

No. Of Conditions : 0
No. Of Frame Parm: 0

+-----+-----+
| 1__   Conditions                | 1__   Frame Parameters          |
|-----|-----|
|                                     |                                     |
|                                     |                                     |
|                                     |                                     |
+-----+-----+

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help retn quit                    bkwrd frwrd                    mai
  
```

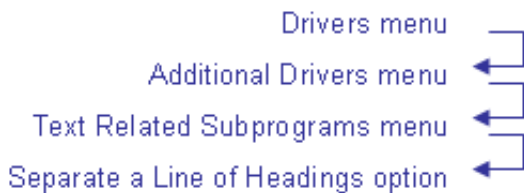
CSUGEN Subprogram

CSUGEN	Description
What it does	<p>Issues a CALLNAT to the Natural Construct Generate function for a specified module. This subprogram receives the names of a model PDA and a model information PDA (CSAMODEL, which must contain the name of the model) and uses the inputs to generate the module code into the Natural source buffer. When the CALLNAT is made to the module, the code is appended to the contents of the source buffer. The source buffer name or type does not change.</p> <p>Notes:</p> <ol style="list-style-type: none"> 1. The specified model PDA must contain the model parameters required for generation. 2. This subprogram requires a NATPARM SSIZE of 55 or greater.
PDA's used	<ul style="list-style-type: none"> ● CSAGEN ● CSAMODEL ● CU--PDA ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● NCST-ADA

CSUHEADS Subprogram

CSUHEADS	Description
What it does	<p>Separates a line of headings into separate headings. This subprogram receives a line of headings and returns three separate headings (each with the length of longest heading).</p>
PDA's used	<ul style="list-style-type: none"> ● CSAHEADS ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

Drivers Menu Option



```

CTEHEADS          N a t u r a l   C o n s t r u c t          CTEHEAD1
Aug 14            Driver for subprogram CSUHEADS           1 of 1

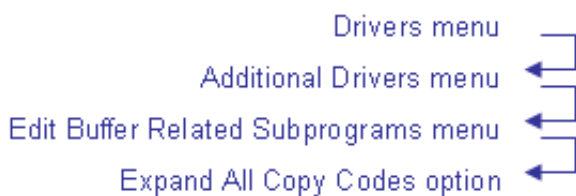
Headings: _____ Field Headings Stacked
                -----
Field Heading Width: 0

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                     mai
    
```

CSUINCL Subprogram

CSUINCL	Description
What it does	Inserts the source for all copycode (currently in the edit buffer) into the edit buffer.
PDA's used	<ul style="list-style-type: none"> • None
Files accessed	<ul style="list-style-type: none"> • None

Drivers Menu Option



```

CTEINCL          N a t u r a l   C o n s t r u c t          CTEMAP1
Aug 14            Driver for program CSUINCL             1 of 1

                +-----+
                |               |
                | PRESS ENTER TO EXECUTE. |
                |               |
                +-----+

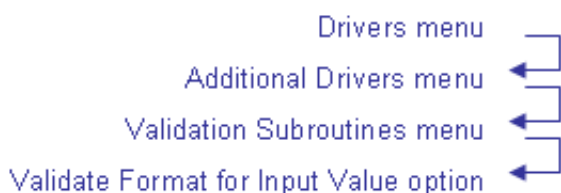
                Read in New Source: _
                *New Source Name...: _____
                New Source Library: DEVPR____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                     mai
    
```

CSUIS Subprogram

CSUIS	Description
What it does	<p>Verifies whether the contents of an alphanumeric field can be converted to a specified format and length. If the format and length are invalid Natural formats, CSASTD.MSG contains an error message when this subprogram is invoked. If the format and length are valid, CSASTD.MSG is blank.</p> <p>In some cases, a user must specify a value using a certain (variable) format and length. For example, the minimum/maximum key values should be valid values corresponding to the format and length of the key. You cannot use the Natural IS function because the format is not known until runtime.</p>
PDAs used	<ul style="list-style-type: none"> ● CSAIS ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

Drivers Menu Option



```

CTEIS              N a t u r a l   C o n s t r u c t           CTEIS1
Aug 14              D r i v e r   f o r   s u b p r o g r a m   C S U I S           1   o f   1

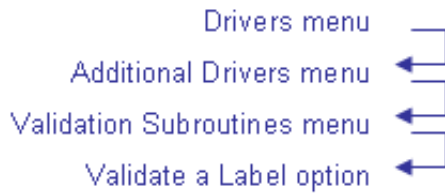
Field Value.: _____
Field Format:  _
Field Length:  ___

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                           mai
    
```

CSULABEL Subprogram

CSULABEL	Description
What it does	<p>Verifies a Natural looping label. This subprogram receives a string of characters and validates it against the Natural label naming convention. ; if the label is not valid, CSASTD.MSG contains an error message.</p> <ul style="list-style-type: none"> ● If the label is valid, CSASTD.MSG is blank ● If the label is not valid, CSASTD.MSG contains an error message
Parameters/PDAs used	<ul style="list-style-type: none"> ● #PDA-LABEL(A32) ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

Drivers Menu Option

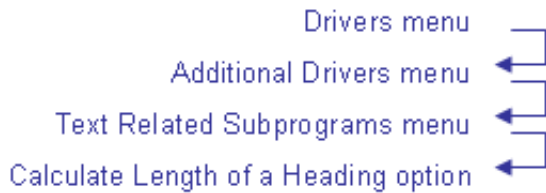


CTELABEL	N a t u r a l C o n s t r u c t	CTELABL1
Aug 14	Driver for subprogram CSULABEL	1 of 1
Label: _____		
Msg..: _____		
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1		
help	retrn quit	mai

CSULENGT Subprogram

CSULENGT	Description
What it does	Builds an input prompt and calculates the length of the heading. This subprogram receives a field name, format, and length. It builds the input prompt from the field headings (if no heading was given, the field name is converted to mixed case) and calculates the length from the format, length, and edit mask. It also returns the heading length and sign option (based on the field format and edit mask).
PDA's used	<ul style="list-style-type: none"> ● CSALENGT ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

Drivers Menu Option



```

CTELENGT          N a t u r a l   C o n s t r u c t          CTELNGT1
Aug 14            D r i v e r   f o r   s u b p r o g r a m   C S U L E N G T          1   o f   1

Field Name....: _____      Field Length....: _____
Field Headings: _____      Field Format....: _
                : _____      Sign.....: _
                : _____

Edit Mask.....: _____

Input Prompt...: _____      Heading Length..:
Sg Option.....: _____      Fld Displ Length:

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                     mai
    
```

CSULPS Subprogram

CSULPS	Description
What it does	Changes the display language (*Language value) and sets the translation required flag to True. This subprogram displays a list of all available languages supported by Natural. When a new language is selected, it switches the user's session to that language and sets the translation required flag to True.
Parameter/PDAs used	<ul style="list-style-type: none"> ● #PDA-TRANSLATION-REQUIRED (L) ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-FI

CSUMAX Subprogram

CSUMAX	Description
What it does	Generates the assignment of a maximum value for a field. This subprogram receives the name, format, and length of a variable and generates the assignment of the maximum value for the field into the edit buffer. It is used when reading a file for all values with a specified prefix, where the suffix extends from the lowest to the highest value.
PDAs used	<ul style="list-style-type: none"> ● CSAMAX ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

Drivers Menu Option

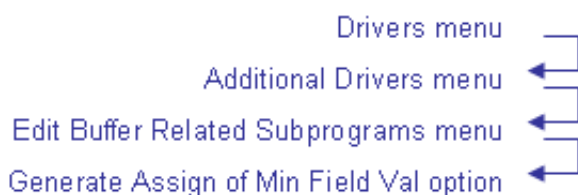
Drivers menu
 Additional Drivers menu
 Edit Buffer Related Subprograms menu
 Generate Assign of Max Field Val option

CTEMAX	N a t u r a l C o n s t r u c t	CTEMAX1
Aug 14	Driver for subprogram CSUMAX	1 of 1
Field : _____		
Format: _		
Length: _____		
Tab...: _		
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1		
help retrn quit		mai

CSUMIMAX Subprogram

CSUMAX	Description
What it does	Generates the assignment of a minimum value for a field. This subprogram receives the name of a variable and its format and length. It generates the assignment of the minimum/maximum values for the field into the edit buffer.
PDA's used	<ul style="list-style-type: none"> ● CSAMIMAX ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

Drivers Menu Option



```

      CTEMIMAX          N a t u r a l   C o n s t r u c t          CTEMIMX1
Aug 14                Driver for subprogram CSUMIMAX          1 of 1

Field : _____

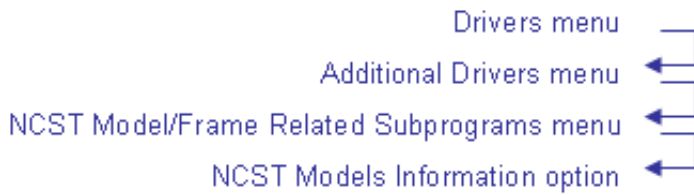
Format: __           Minimum Value: _           Non Negative Min/Max: _           Tab: __
Length: _____   Descending...: _           DB2 Date/Time Stamp : _

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                     mai
  
```

CSUMODEL Subprogram

CSUMORE	Description
What it does	Returns information about a Natural Construct model. This subprogram receives the name of a model and returns the model description, generator mode and type, and the names of the model PDA, subprograms, and code frames.
PDA's used	<ul style="list-style-type: none"> ● CSAMODEL ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

Drivers Menu Option



```

CTEMODEL          N a t u r a l   C o n s t r u c t          CTEMODL1
Aug 14            D r i v e r   f o r   s u b p r o g r a m   C S U M O D E L          1   o f   1

*Model Name.....: _____
Model Description:

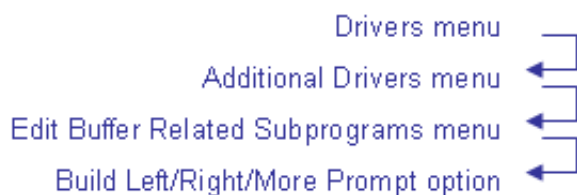
No. Modify Subps:      Modify Subps   Code Frames   Clear Subp...:
No. Code Frames :      -----      -----      Read Subp....:
Generator Mode..:                                Save Subp....:
Generator Type..:                                Pre-Gen Subp.:
Display Window..:                                Post-Gen Subp:
Start Comment...:                                Doc Subp.....:
End Comment.....:                                Pda Name.....:

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                  mai
  
```

CSUMORE Subprogram

CSUMORE	Description
<p>What it does</p>	<p>Builds the initial assignment for the LEFT-MORE/RIGHT-MORE array. This subprogram receives a function (L for the LEFT-MORE array, R for the RIGHT-MORE) and the number of panels used by a program. These arrays contain the prompts displayed at the top left or right corner of the panels. The prompts indicate the number of panels located to the left or right of the current panel.</p> <p>For example, to generate the initial value for the LEFT-MORE-PROMPT array for a program with two panels, enter:</p> <pre>CSAMORE.#LEFT-RIGHT = 'L' CSAMORE.#MAX-WINDOW = 2</pre> <p>The subprogram writes the following to the source buffer:</p> <pre>INIT < ' ', '<1 more' ></pre> <p>To generate the initial value for the RIGHT-MORE-PROMPT array for a program with two panels, enter:</p> <pre>CSAMORE.#LEFT-RIGHT = 'R'</pre> <p>The subprogram writes the following to the source buffer:</p> <pre>INIT < '1 more >', '' ></pre> <p>Note: If the value of *Language is not 1 during generation, the word “more” is not included in the initial values.</p> <p>Tip: Use a scalar field rather than an occurrence of this array. Before the map is displayed, assign the array occurrence to the scalar field. Using arrays on maps makes them difficult to maintain and less suitable to use as standard layouts.</p>
<p>PDA's used</p>	<ul style="list-style-type: none"> ● CSAMORE ● CSASTD
<p>Files accessed</p>	<ul style="list-style-type: none"> ● None

Drivers Menu Option



```

CTEMORE          N a t u r a l   C o n s t r u c t          CTEMORE1
Aug 14           D r i v e r   f o r   s u b p r o g r a m   C S U M O R E          1   o f   1

Left/Right:    _ (L or R)
Max Windows:   __

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
             help retn quit                                     mai

```

Note:

For more information on changing the size of the left or right prompt, see Use CSXDEFLT Overrides.

CSUMPBOX Subprogram

CSUMPBOX	Description
What it does	Handles the map edit buffer. This subprogram receives a function and parameters (in CSAMPBOX). It initializes the map edit buffer or generates variable, array, and text control blocks into the edit buffer.
PDA's used	<ul style="list-style-type: none"> ● CSAMPBOX ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

CSUMPCPR Subprogram

CSUMPCPR	Description
What it does	Replaces the map settings in the edit buffer with values from the CSAMPSET parameter data area.
PDA's used	<ul style="list-style-type: none"> ● CSAMPSET ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

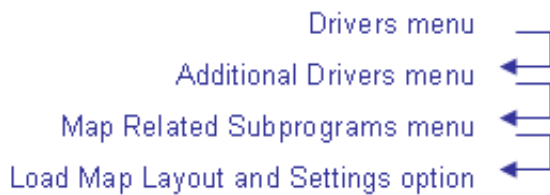
CSUMPDUP Subprogram

CSUMPDUP	Description
What it does	Checks for the duplication of fields on a map. This subprogram determines whether there are any fields duplicated in the CSAMPFLD.FIELD-INFO(*) structure. If there are duplicate fields, CSASTD.MSG contains an error message when this subprogram is invoked.
PDA's used	<ul style="list-style-type: none"> ● CSAMPFLD ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

CSUMPLAY Subprogram

CSUMPLAY	Description
What it does	Loads the map layout into the edit buffer and returns the map settings. This subprogram receives the name, layout, and type of map and loads the specified map into the edit buffer. It returns the map settings.
PDA's used	<ul style="list-style-type: none"> ● CSAMPSET ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

Drivers Menu Option



```

CTEMPLAY          N a t u r a l   C o n s t r u c t          CTEMPLY1
Aug 14            D r i v e r   f o r   s u b p r o g r a m   C S U M P L A Y          1   o f   1

*Layout..: _____      Error Code :           Dc:      Zp.....:
                               Map Version:         Ps:      Pm.....:
                               Profile.....:         Ls:      Cursor Skip...:

Delimiter Class..:                               Std Keys.....:
Ad.....:                               Justification :
Delimiter Char...:                               Col Shift.....:
Cd.....:                               Case Deflt....:

Write Statement..:   CV.....:                   Auto Rule Rank:
Input Statement..:   Filler Char:                 Enforce Attr...:

Help.....:
Help-As-Fld-Deflt:

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                     mai
    
```

CSUMPMMS Subprogram

CSUMPMMS	Description
What it does	Merges the settings for two maps. This subprogram merges the map settings from CSAMPSET and CSAMPOUT. The settings in CSAMPSET override the settings in CSAMPOUT and the result is stored in CSAMPOUT.
PDA's used	<ul style="list-style-type: none"> ● CSAMPSET ● CSAMPOUT
Files accessed	<ul style="list-style-type: none"> ● None

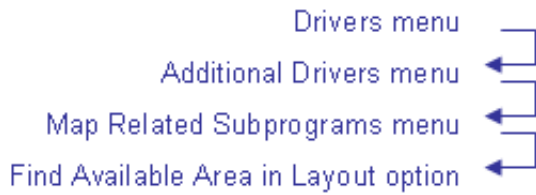
CSUMPOVL Subprogram

CSUMPOVL	Description
What it does	<p>Checks the boundary on a map and determines if there are overlapping fields. This subprogram checks whether the fields specified in CSAMPFLD exceed the line size or page size of the available map panel.</p> <p>The available map panel is a block of consecutive lines on the panel. This block is determined by the specified page and line size, excluding the map layout and any PF-keys. The fields on the map cannot overlay the layout or PF-keys.</p>
PDA's used	<ul style="list-style-type: none"> ● CSAMPFLD ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

CSUMPREG Subprogram

CSUMPREG	Description
What it does	Determines the available map area in a map layout. This subprogram determines the first and last line on a map that is available for editing in a specified map layout.
PDA's used	<ul style="list-style-type: none"> ● CSAMPREG ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

Drivers Menu Option



```

CTEMPREG          N a t u r a l   C o n s t r u c t          CTEMPRG1
Aug 14            D r i v e r   f o r   s u b p r o g r a m   C S U M P R E G          1   o f   1

*Layout: _____  F i r s t   A v a i l a b l e   L i n e :          L a y o u t   P a g e   S i z e :
                   L a s t   A v a i l a b l e   L i n e :          L a y o u t   L i n e   S i z e :

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                          mai
    
```

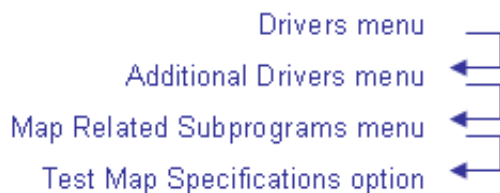
CSUMPTAB Subprogram

CSUMPTAB	Description
What it does	Calculates the absolute field coordinates on a map and creates the field prompts. This subprogram receives field information from CSAMPFLD and returns the absolute field positions and prompts in CSAMPX-Y. Dots are added to each field prompt in a region to extend its length to that of the longest prompt in that region (... for ISA format and . . . for SAA format). Note: For more information about the data returned, refer to the CSAMPX-Y data area in the SYSCST library.
PDA's used	<ul style="list-style-type: none"> ● CSAMPFLD ● CSAMPX-Y ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

CSUMPTST Subprogram

CSUMPTST	Description
What it does	Tests the specifications for the map currently in the edit buffer.
PDA's used	<ul style="list-style-type: none"> ● CSAMPTST ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

Drivers Menu Option



```

CTEMPTST          N a t u r a l   C o n s t r u c t          CTEMPTST1
Aug 14            D r i v e r   f o r   s u b p r o g r a m   C S U M P T S T          1   o f   1

  R e a d   i n   N e w   M a p :   _           P a g e   S i z e :   2 3 _
 * M a p   N a m e . . . . . :   _____   L i n e   S i z e :   8 0 _
  M a p   L i b r a r y . . . . :   D E V P R _____

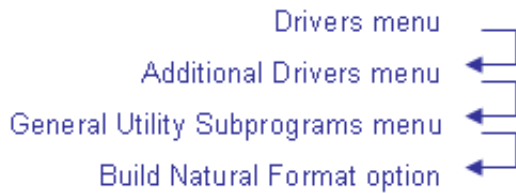
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                     mai

```

CSUNATFM Subprogram

CSUNATFM	Description
What it does	<p>Builds a valid Natural format definition from the formats and lengths specified. This subprogram receives the format and length values and combines these to build a valid Natural format string. For example, if you enter:</p> <pre>CSANATFM.FIELD-LENGTH = 9.0 CSANATFM.FIELD-FORMAT = 'P'</pre> <p>CSUNATFM produces the following output:</p> <pre>CSANATFM.#Natural-FORMAT = P9</pre>
PDA's used	<ul style="list-style-type: none"> ● CSANATFM ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

Drivers Menu Option



```

CTENATFM          N a t u r a l   C o n s t r u c t          CTENTFM1
Aug 14            D r i v e r   f o r   s u b p r o g r a m   C S U N A T F M          1   o f   1

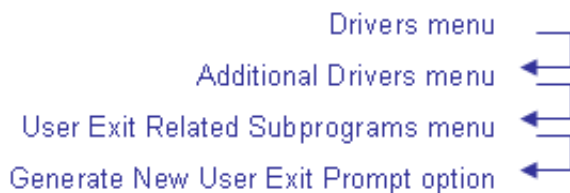
Field Format: _      Natural Format:
Field Length: _____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                     mai
    
```

CSUNEWX Subprogram

CSUNEWX	Description
What it does	Generates a new user exit prompt. This subprogram receives the name of a user exit and generates a starting point (DEFINE EXIT <i>exit-name</i> , for example) for the user exit. It initiates a new user exit for sample subprograms that are capable of generating more than one exit.
PDA's used	<ul style="list-style-type: none"> ● CSANEWX
Files accessed	<ul style="list-style-type: none"> ● None

Drivers Menu Option



CTENEWX Aug 14	N a t u r a l C o n s t r u c t Driver for subprogram CSUNEWX	CTENEWX1 1 of 1
User Exit Name: _____		
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--- help retrn quit		

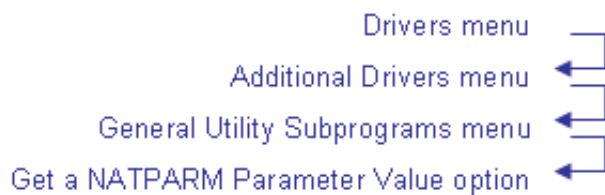
CSUOG Subprogram

CSUOG	Description
What it does	<p>Comments out all code within a specified user exit. This subprogram receives the name of a user exit and inserts comment indicators at the beginning of each line of code within the specified exit.</p> <p>Specify the name of the user exit in the #USER-EXIT (A65) variable. For example, to comment out all code within the MOVE-TO user exit, specify the following:</p> <pre>0040 01 #USER-EXIT (A65) . . . 3800 #USER-EXIT := 'MOVE-TO' 3810 CALLNAT 'CSUOG' #USER-EXIT</pre>
PDA's used	<ul style="list-style-type: none"> ● CSAOG
Files accessed	<ul style="list-style-type: none"> ● None

CSUPARMS Subprogram

CSUPARMS	Description
What it does	<p>Returns the value of a NATPARM parameter. This subprogram receives a NATPARM parameter and returns its corresponding value. Valid NATPARM parameters are:</p> <ul style="list-style-type: none"> ● CF ● DC ● IA ● ID ● KD ● ML ● TB ● UL <p>Note: For information about INPUT/OUTPUT parameters, refer to the CSAPARMS data area in the SYSCST library.</p>
PDAs used	<ul style="list-style-type: none"> ● CDUPARMA ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

Drivers Menu Option



```

CTEPARMS          N a t u r a l   C o n s t r u c t          CTEPARM1
Aug 14            D r i v e r   f o r   s u b p r o g r a m   C S U P A R M S          1   o f   1

Parameter....:  _ _   ( I D , C F , U L , T B , I A , D C , K D , M L )
Alpha Value..:
Numeric Value:

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                mai

```

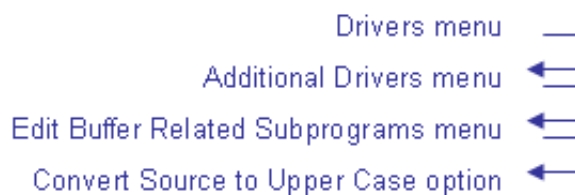
CSUPARTY Subprogram

CSUPARTY	Description
What it does	Determines Natural data types and returns the byte length. This subprogram receives the format and length for a data type and indicates whether it is a valid Natural data type. If it is, this subprogram returns the byte length.
PDA's used	<ul style="list-style-type: none"> ● CSAPARTY ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

CSUPPER Program

CSUPPER	Description
What it does	Converts the contents of the source buffer into upper case. This program reads through the source buffer and converts specified lower case characters into upper case.
PDA's used	<ul style="list-style-type: none"> ● None
Files accessed	<ul style="list-style-type: none"> ● None

Drivers Menu Option



```

CTEPPER                      N a t u r a l   C o n s t r u c t          C T E M A P 1
Aug 14                        D r i v e r   f o r   p r o g r a m   C S U P P E R          1   o f   1

+-----+
|                                     |
|           P R E S S   E N T E R   T O   E X E C U T E .           |
|                                     |
+-----+

      R e a d   i n   N e w   S o u r c e :   _
      * N e w   S o u r c e   N a m e . . . :   _____
      N e w   S o u r c e   L i b r a r y :   D E V P R ____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                          mai
    
```

CSUREADS Subprogram

CSUREADS	Description
What it does	<p>Reads the specification parameters for a module. This subprogram receives the name of a source module. If the module was generated using Natural Construct, the subprogram reads the source code and returns the model parameter data area (PDA) containing the parameters used to generate the module.</p> <p>You can use the passed model PDA to call the model subprograms for the model used to generate the module.</p> <p>This subprogram also returns a data area describing the model and listing the names of the model subprograms.</p> <p>Note: This subprogram requires a NATPARM SSIZE of 55 or greater.</p>
Parameters/PDAs used	<ul style="list-style-type: none"> ● #READ-THIS-MODULE(A8) ● CSAMODEL ● CU--PDA ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● NCST-ADA ● SYSTEM-FUSER

Tip:

If you know the name of the model used to generate the specified module, you can pass its model PDA to CSUREADS rather than CU--PDA. After the call to CSUREADS, the model PDA is populated with the parameters used to generate that module.

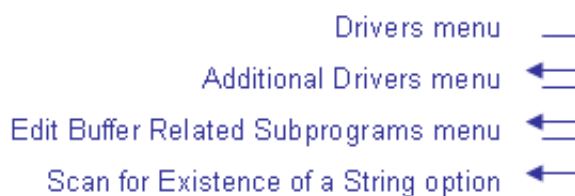
CSUREF Subprogram

CSUREF	Description
What it does	Generates referential integrity checks against foreign files. This subprogram is typically called three times: once to generate the data structures (DATA) required by the generated code, once to generate the update edits (UPDATE), and once to generate the delete edits (DELETE). Set the value of CSAREF.FUNCTION-CODE to either DATA, UPDATE, or DELETE. After the first call, this subprogram returns the number of update and delete edits found. This avoids unnecessary subsequent calls.
PDAAs used	<ul style="list-style-type: none"> ● CSAREF ● CU--PDA ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-RL ● SYSDIC-FI

CSUSCAN Subprogram

CSUSCAN	Description
What it does	Scans for the existence of a string in the edit buffer. This subprogram receives a string and scans for (not absolute) the existence of the string in the edit buffer.
PDAAs used	<ul style="list-style-type: none"> ● CSASCAN
Files accessed	<ul style="list-style-type: none"> ● None

Drivers Menu Option



```

CTESCAN          N a t u r a l   C o n s t r u c t          CTESCAN1
Aug 14           D r i v e r   f o r   s u b p r o g r a m   C S U S C A N          1   o f   1

String..: _____
Absolute: _ (Mark if scan string need not be delimited by special chars)
Found...: _

Read in New Source: _
*New Source Name...: _____
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                     mai
    
```

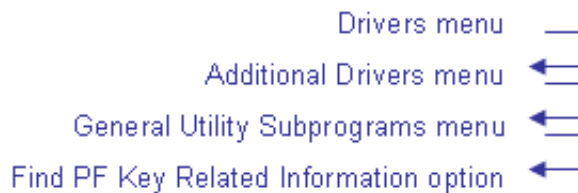
CSUSELFV Subprogram

CSUSELFV	Description
What it does	<p>Selects fields/variables from views, LDAs, or PDAs. This subprogram selects up to 40 fields/variables from up to 6 different views, LDAs, or PDAs and appends the selected fields/variables to CSASELFV. Existing fields/variables in CSASELFV cannot be re-selected.</p> <p>When selecting from data areas, you cannot select the following:</p> <ul style="list-style-type: none"> ● constants ● more than one structure <p>If you specify the select all option, then the first structure in the data area is selected.</p> <p>The invoking subprogram should issue an initial RESET statement to clear the structures in CSASELFV, such as:</p> <pre> RESET CSASELFV CSASELFV.GENERAL-INFORMATION CSASELFV.FIELD-SPECIFICATION(*) </pre>
PDAs used	<ul style="list-style-type: none"> ● CSASELFV ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

CSUSETKY Subprogram

CSUSETKY	Description
What it does	Returns PF-key definitions from the control record to support variable PF-keys in Natural Construct. The PF-key names are returned in the CSUSETKY.#PF-NAME(*) array. The index for each array element corresponds to the PF-key number. The following example indicates that PF1 is named "help": #PF-NAME(1) = 'help'
PDA's used	<ul style="list-style-type: none"> ● CSUSETKY ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● NCST-CONTROL

Drivers Menu Option



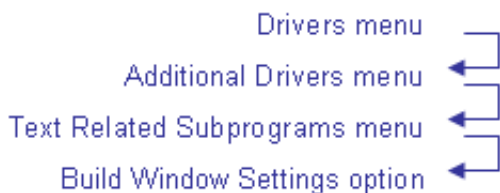
Pf Name	Pf Number	Pf Key
main	Main.....: 12	Pf Main.....: PF12
retrn	Return....: 2	Pf Return....: PF2
quit	Quit.....: 3	Pf Quit.....: PF3
test	Test.....: 4	Pf Test.....: PF4
bkwrđ	Backward..: 7	Pf Backward..: PF7
frwrđ	Forward...: 8	Pf Forward...: PF8
left	Left.....: 10	Pf Left.....: PF10
right	Right.....: 11	Pf Right.....: PF11
help	Help.....: 1	Pf Help.....: PF1
	Available1: 5	Pf Available1: PF5
	Available2: 6	Pf Available2: PF6
	Available3: 9	Pf Available3: PF9

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
 help retrn quit ma

CSUSETW Subprogram

CSUSETW	Description
<p>What it does</p>	<p>Returns the SET CONTROL parameters to define a window. This subprogram receives the parameters for a window (such as frame, line size, column size, base line, and base column). It returns the SET CONTROL parameters to define the window. For example, if the parameters are:</p> <pre>CSUSETW.FRAME=TRUE CSUSETW.LINE-SIZE=70 CSUSETW.COLUMN-SIZE=5</pre> <p>This subprogram returns:</p> <pre>CSUSETW.SET-CONTROL.PARM='WBFL70C5'</pre>
<p>PDA's used</p>	<ul style="list-style-type: none"> ● CSUSETW ● CSASTD
<p>Files accessed</p>	<ul style="list-style-type: none"> ● None

Drivers Menu Option



```

CTESETW          N a t u r a l   C o n s t r u c t          CTESETW1
Aug 14           D r i v e r   f o r   s u b p r o g r a m   CSUSETW          1 of 1

Frame.....: _ Line Size..: ___ Base Line..: ___ Required Width : ___
              Column Size: ___ Base Column: ___ Required Height: ___

Set Control Parm:

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help retn quit                                     mai
  
```

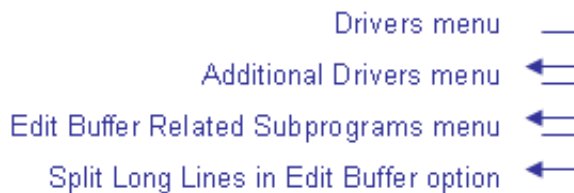
CSUSORT Program

CSUSORT	Description
What it does	Sorts a 2-dimensional array based on specified column positions. This subprogram receives a 2-dimensional array and sorts the array based on the desired column positions. A Natural SORTSIZE is not required because the sort uses an internal bubble sort algorithm. Note: For an example of how to call this subprogram, refer to the CSASORT data area.
Parameters/PDAs used	<ul style="list-style-type: none"> ● CSASORT ● #SORT-DATA(A1/1:V,1:V) ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● None

CSUSPLIT Program

CSUSPLIT	Description
What it does	Splits lines in the source buffer that are longer than 72 characters. Only lines with code extending beyond column 72 are split; lines with comments extending beyond column 72, but not code, are ignored. If a text string (enclosed within quotes) extends beyond column 72, the entire string is moved to the next line.
PDAs used	<ul style="list-style-type: none"> ● None
Files accessed	<ul style="list-style-type: none"> ● None

Drivers Menu Option




```

CTESPLIT          N a t u r a l   C o n s t r u c t          CTEMAP1
Aug 14             D r i v e r   f o r   p r o g r a m   C S U S P L I T          1   o f   1

+-----+
|               |
|               |
|               |
|               |
|               |
|               |
|               |
|               |
|               |
|               |
+-----+

          Read in New Source: _
          *New Source Name...: _____
          New Source Library: DEVPR____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                     mai
    
```

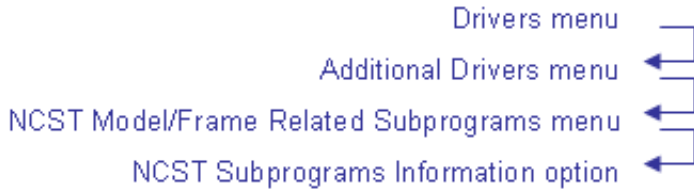
CSUSUB Program (Mainframe)

CSUSUB	Description
What it does	Submits a job for execution. The JCL for the job must be in the source buffer. This subprogram is used in conjunction with the CSUSUB command. For information, see <i>JCL Submit Utility (Mainframe), Natural Construct Generation</i> .
PDA's used	<ul style="list-style-type: none"> ● None
Files accessed	<ul style="list-style-type: none"> ● None

CSUSUBP Subprogram

CSUSUBP	Description
What it does	Returns information about a Natural Construct model subprogram, such as the PF-key settings and the window sizes. This subprogram receives the name of a model subprogram and returns information about that subprogram. The information corresponds to the data accessed through the Maintain Subprograms function. Note: For more information, see Maintain Subprograms Function.
PDA's used	<ul style="list-style-type: none"> ● CSASUBP ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● NCST-SUBPROGRAM

Drivers Menu Option



```

CTESUBP          N a t u r a l   C o n s t r u c t          CTESUBP1
Aug 15           D r i v e r   f o r   s u b p r o g r a m   C S U S U B P          1   o f   1

Subprogram Name: _____
Description....:

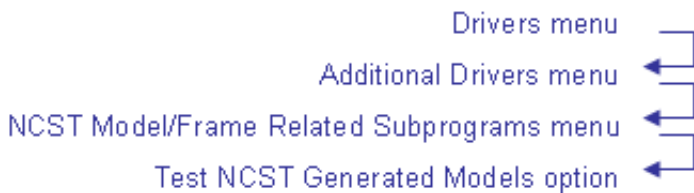
Backward Forward Flag:      Window Length :      Key Name  No.  Other Keys:  _
Left Right Flag.....:      Window Columns:  -----
Test Key Flag.....:

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                          mai
  
```

CSUTEST Program

CSUTEST	Description
What it does	Tests the subprograms for Natural Construct-generated models. This program tests the individual subprograms for Natural Construct-generated models. For information, see Test the Model Subprograms.
PDA's used	<ul style="list-style-type: none"> • None
Files accessed	<ul style="list-style-type: none"> • NCST-SUBPROGRAM • NCST-CONTROL

Drivers Menu Option



```

CSUTEST                      N a t u r a l   C o n s t r u c t          CSUTESM1
Aug 14                       S i n g l e   M o d u l e   T e s t   P r o g r a m      04:54 PM

Code Function                 *Model: _____
-----
R   Release Variables        |
*   Execute All Subp.       V
1-9 Execute One Subp.       _ Clear : _____ V
E   Edit source              _ Mod 1: _____ _ Mod 6: _____
C   Clear Edit Buffer        _ Mod 2: _____ _ Mod 7: _____
?   Help                    _ Mod 3: _____ _ Mod 8: _____
.   Terminate               _ Mod 4: _____ _ Mod 9: _____
-----
_                               _ Mod 5: _____ _ Mod 10: _____
                               _ Prgen: _____ _ Save : _____
                               _ Documt: _____ _ Postgn: _____

                               Source
                               Lines
                               Total: 133

                               Frame Parameter or Exit Name
                               _ Other : _____
                               _ Other : _____
                               _ Other : _____
                               _ Other : _____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
help retrn quit                                                    mai
    
```

CSUTLATE Subprogram

CSUTLATE	Description
What it does	<p>Translates message text at runtime. This subprogram receives a message number and position value and retrieves the appropriate text. If the message text contains multiple items delimited by a slash (/), the position value identifies which text is translated.</p> <p>This subprogram is invoked from the CSUCURS and CSUCURS1 subprograms.</p>
PDA's used	<ul style="list-style-type: none"> ● CSATLATE ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSTEM-FUSER

CSUTRANS Subprogram

CSUTRANS	Description
What it does	<p>Translates screen prompts before they are displayed. This subprogram receives a defined data structure (typically a translation LDA) containing SYSERR message numbers and translates them into the appropriate text.</p> <p>CSUTRANS reads the supplied data structure, searching for one of two message number patterns: *NNNN or *NNNN.A, where *NNNN identifies the message number and .A identifies a position within the message number. If a message number of the type *NNNN is located, the entire SYSERR message is retrieved. If a message number of type *NNNN.A is located, the portion of the message corresponding to the .A notation is retrieved. A message number can have up to 15 positions: the values 1 to 9 represent the first nine positions, and the values A to F represent the 10th to 15th positions.</p> <p>To locate the text corresponding to a message number, specify the library in which the SYSERR message numbers and text reside. By default, CSUTRANS checks the SYSERR message CSTLDA library. In most cases, you will create your own SYSERR message library. When you do, enter the library name in the #MESSAGE-LIBRARY field.</p> <p>In addition to retrieving the appropriate language message text, CSUTRANS searches for any formatting characters and formats the text as appropriate.</p> <p>CSUTRANS requires a specific data structure. The following example shows the translation LDA for the Standard Parameters panel for the Batch model:</p> <pre data-bbox="302 716 873 1129"> ** **SAG TRANSLATION LDA ** ** used by CTETRANS. 1 CTE-MAL 2 TEXT /* Corresponds to syserr message 3 #GEN-PROGRAM A 20 INIT<'*2000.1,.'> 3 #SYSTEM A 20 INIT<'*2000.2,+'> 3 #GDA A 20 INIT<'*2000.3,>'> 3 #TITLE A 20 INIT<'*2001.1,+/16'> 3 #DESCS A 20 INIT<'*2001.2,.'> 3 #GDA-BLOCK A 20 INIT<'*2001.3,>'> 3 #MAP-HEADER1 A 20 INIT<'*2049.1,./18'> 3 #MAP-HEADER2 A 20 INIT<'*2049.2,>/18'> 3 #USE-MSG-NR A 20 INIT<'*2050.1,.'> 3 #PASSWORD-CHECK A 20 INIT<'*2050.2,./20'> 2 TEXT 3 TRANSLATION-TEXT 4 TEXT-ARRAY A 1 (1:200) 2 ADDITIONAL-PARMS 3 #MESSAGE-LIBRARY A 8 INIT<'CSTLDA'> 3 #LDA-NAME A 8 INIT<'CTE-MAL'> 3 #TEXT-REQUIRED L INIT<TRUE> 3 #LENGTH-OVERRIDE I 4 /* Length to translate </pre> <p>Other details about the structural elements include:</p> <ul style="list-style-type: none"> • The first comment line (**SAG TRANSLATION LDA) indicates that this is a translation LDA. During a static install, Natural Construct scans for this comment line and replaces the SYSERR numbers with the appropriate text. • The CTE-MAL level 1 structure name is typically the LDA name; use this qualifier whenever the variables are accessed. • The level 3 variables (#GEN-PROGRAM, #SYSTEM, #GDA, etc.) are screen prompts that are initialized with a valid SYSERR number. All SYSERR numbers use the *NNNN.A notation and are listed in sequential order. <p>Note: This sequence does not apply to positions after the period within the *NNNN.A notation. For example, you can list *2000.2 before *2001.1.</p> <ul style="list-style-type: none"> • The TEXT-ARRAY value must match the total number of bytes in all prompt variables to be translated. • The #MESSAGE-LIBRARY value indicates the SYSERR library in which the text is stored. • The #TEXT-REQUIRED logical indicates whether translation is required. If it is, this field ensures that translation is performed only once. <p>Notes:</p> <ol style="list-style-type: none"> 1. For more information about SYSERR message numbers, see Use SYSERR References. 2. For more information about formatting the message text, see Format SYSERR Message Text.
PDAs used	<ul style="list-style-type: none"> • CSATRANS • CSASTD
Files accessed	<ul style="list-style-type: none"> • SYSTEM-FUSER

Drivers Menu Option



```

CTETRANS          ***** Natural Related subprograms *****
Oct 21             - Driver for subprogram CSUTRANS -                1 of 1

Translation LDA ... CTE-MAL

Input Parameters ... #GEN-PROGRAM *2000.1, ._____
                   #SYSTEM *2000.2, +_____
                   #GDA *2000.3, >_____
                   #TITLE *2001.1, +/16_____
                   #DESCS *2001.2, ._____
                   #GDA-BLOCK *2001.3, >_____
                   #MAP-HEADER1 *2049.1, ./18_____
                   #MAP-HEADER2 *2049.2, >/18_____
                   #USE-MSG-NR *2050.1, ._____
                   #PASSWORD-CHECK *2050.2, ./20_____
                   #MESSAGE-LIBRARY CSTLDA__
                   #LDA-NAME CTE-MAL_
                   #TEXT-REQUIRED X
                   #LENGTH-OVERRIDE _____0
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help      quit      reset      bkwrdr frwrdr      right left
    
```

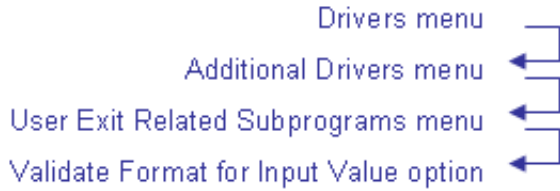
Note:

This driver program is provided as a sample only. Because the screen prompts translated by CSUTRANS vary depending on the application under development, the driver program must be tailored to the application.

CSUXCHK Subprogram

CSUXCHK	Description
What it does	Scans for the existence of a user exit in the edit buffer. This subprogram receives the name of a user exit and scans the edit buffer for that name.
PDA's used	● CSAXCHK
Files accessed	● None

Drivers Menu Option



```

CTEXCHK          N a t u r a l   C o n s t r u c t          CTEXCHK1
Aug 14           D r i v e r   f o r   s u b p r o g r a m   CSUXCHK          1   o f   1

User Exit Name....: _____
Found.....:

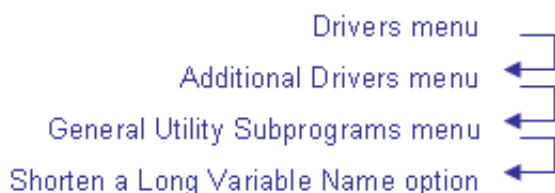
Read in New Source: _
*New Source Name...: _____
New Source Library: DEVPR____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
help retrn quit                                     mai
  
```

CSU2LONG Subprogram

CSU2LONG	Description
What it does	<p>Converts a long variable name to an abbreviation. This subprogram receives a long character string (32 characters) and a desired length, and returns the truncated string (abbreviation). The sixth position of the string is the first position truncated. If no length is given, the default is 30.</p> <p>If the long string is not longer than the desired length, the string is still truncated. For example, if the long string is “THIS-IS-A-LONG-VARIABLE” and the desired length is 20, the short string is “THIS-A-LONG-VARIABLE”.</p> <p>Tip: Use this subprogram when you add characters to a file or field name that is already 32 characters long.</p>
PDA's used	<ul style="list-style-type: none"> ● CSA2LONG
Files accessed	<ul style="list-style-type: none"> ● None

Drivers Menu Option



```

CTE2LONG          N a t u r a l   C o n s t r u c t          CTE2LNG1
Aug 14            D r i v e r   f o r   s u b p r o g r a m   C S U 2 L O N G          1   o f   1

Long Name.....: _____
Maximum Length:  ___

Short Name.....:

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                     mai
    
```

Predict-Related Subprograms (CPU*)

The subprograms described in this section retrieve information from the Predict data dictionary. While some of these subprograms generate code, most supply information to the calling subprogram and the calling subprogram generates the code.

Notes:

1. If you use Software AG’s Entire Net-work, the Predict data can reside on a platform other than the platform on which Natural Construct is running.
2. Driver programs for many of the supplied programs/subprograms are available through the Drivers menu option on the Administration main menu. If a driver program is available, its location is listed in the *Drivers Menu Option* section in the program/subprogram’s description. For more information about the supplied driver programs, see Drivers Menu Function.

This section covers the following topics:

- With Natural Security Installed
- CPU-OBJ Subprogram
- CPU-OBJ2 Subprogram
- CPU-OREL Subprogram
- CPU-VIEW Subprogram
- CPUEL Subprogram
- CPUELDE Subprogram
- CPUELKY Subprogram
- CPU-FREL Subprogram
- CPUELNX Subprogram
- CPUELRD Subprogram
- CPUELVE Subprogram

- CPUEXIST Subprogram
- CPUFI Subprogram
- CPUHOLD Subprogram
- CPUKY Subprogram
- CPUREDEF Subprogram
- CPURL Subprogram
- CPURLRD Subprogram
- CPUSUPER Subprogram
- CPUUNIQ Subprogram
- CPUVE Subprogram
- CPUVERUL Subprogram
- CPUXPAND Subprogram

With Natural Security Installed

If Natural Security is installed, the Predict-related subprograms restrict access to file and field information. Users can only retrieve information for files linked to the current application.

While generating a program, the program may access information about the same file many times. To avoid security checks each time, the access subprograms use the FILE-CODE field. This INPUT/OUTPUT field accesses file information and acts as a cipher code to avoid multiple security checks on the same file; it is available for all supplied subprograms.

If you are developing under Natural Security, include the FILE-CODE field in the model PDA for each file used multiple times during generation. The FILE-CODE field is passed in the PDA of the access subprogram and reassigned back to the model PDA after each call.

To avoid security checks for each access, the model subprogram that invokes CPUEL contains the following statements:

```
ASSIGN CPAEL.FILE-CODE = #PDA-FILE-CODE
CALLNAT 'CPUEL' CPAEL CSASTD
ASSIGN #PDA-FILE-CODE = CPAEL.FILE-CODE
```

Note:

For an example of using these subprograms to restrict access to file and field information, refer to the CUSCGPR program in the SYSCST library.

CPU-OBJ Subprogram

CPU-OBJ	Description
What it does	<p>Generates an external data area based on a Predict file view. This subprogram receives the view name and a set of logical variables that define the generation options. It generates an external data area structure to match the view. It can also generate the C# variables for each C* variable that corresponds to an MU or PE and/or includes the corresponding REDEFINE fields for redefined fields or superdescriptors.</p> <p>Note: For information about INPUT/OUTPUT parameters, refer to the CPA-OBJ data area in the SYSCST library.</p>
PDAs used	<ul style="list-style-type: none"> ● CPA-OBJ ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-EL ● SYSDIC-FI

Drivers Menu Option



```

CTE-OBJ          N a t u r a l   C o n s t r u c t          CTE-OBJ1
May 12          D r i v e r   f o r   s u b p r o g r a m   C P U - O B J          1   o f   1

*File: _____

Build Redefines..: _      Structure Level: _
SuperDe Redefines: _      Joined Fld Name: _____
Cstars.....: _      Joined Length..: ____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                     mai
    
```

CPU-OBJ2 Subprogram

CPU-OBJ2	Description
What it does	<p>Issues CALLNAT to the #CALLNAT subprogram and passes information about elements that make up an object. This subprogram receives:</p> <ul style="list-style-type: none"> ● a view name ● a key name ● a set of options ● the name of a passed subprogram to CALLNAT <p>An object is derived from view and key names. The view and key names are based on intra-object relationships defined in Predict (for example, ORDER-HEADER-HAS-ORDER-LINES).</p> <p>The elements of an object are the individual fields in the files that make up the object. This subprogram traverses the object tree and checks each element. For each element, it CALLNATs the #CALLNAT subprogram and passes it information about the element (for example, the format, length, and type).</p> <p>You can set options to limit or extend the number of elements to check (for example, whether to include all field redefinitions or just the lowest levels).</p> <p>Note: This subprogram replaces the CPU-OBJ subprogram; for all new development, use CPU-OBJ2.</p>
Parameters/PDAs used	<ul style="list-style-type: none"> ● CPA-OBJ2 ● CPA-ODAT ● CU--PDA ● #PASS(A1/1:V) ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC

CPU-OREL Subprogram

CPU-OREL	Description
What it does	<p>Adds entity information to a table. This subprogram receives the name of an object and information about each entity belonging to the object. It adds this information to a table. Optionally, it can display tracing information.</p> <p>Note: For more information, refer to CPA-OREL.ENTITY(*).</p>
PDAs used	<ul style="list-style-type: none"> ● CPA-OREL ● CU__PDA ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-RL ● SYSDIC-FI ● SYSDIC-EL

CPU-VIEW Subprogram

CPU-VIEW	Description
What it does	<p>Generates field definitions based on the contents of a Predict view. This subprogram receives the name of a Predict view and a set of logical parameters defining the options to be generated. It generates the view definition as it should appear in the DEFINE DATA END-DEFINE block of a Natural program, subprogram, or help routine.</p> <p>This subprogram can also generate the C# variables for each C* variable that corresponds to an MU (multiple-valued) or PE (periodic group), and/or includes the corresponding REDEFINE fields for redefined fields or superdescriptors.</p> <p>You can use this subprogram to define a structure based on a view in Predict. The format and length for each field is generated.</p> <p>Notes:</p> <ol style="list-style-type: none"> 1. This subprogram differs from CPU-OBJ in that it generates internal rather than external data structures. 2. For information about INPUT/OUTPUT parameters, refer to the CPA-VIEW data area in the SYSCST library.
PDAs used	<ul style="list-style-type: none"> ● CPA-VIEW ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-EL ● SYSDIC-FI

Drivers Menu Option



```

CTE-VIEW          N a t u r a l   C o n s t r u c t          CTE-VEW1
May 12            D r i v e r   f o r   s u b p r o g r a m   C P U - V I E W          1   o f   1

*File....: _____
View....:  _____          G e n   0 1   L e v e l.....:  _
Omit Fld:  _____

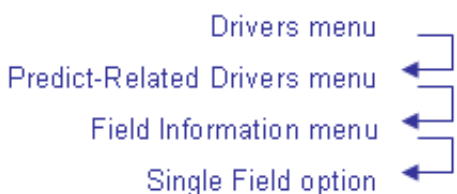
Variable Indexes : _      Include Hyper DE...: _      Include MU Counter: _
Build Redefines..: _      Include Phonetic DE: _      Include PE Counter: _
SuperDe Redefines: _      Include Sub DE.....: _      Include MU Hyper...: _
Specify Formats..: _      Include Super DE...: _      Include PE Hyper...: _
Cstars.....: _      Redefine Cstars....: _

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                          mai
    
```

CPUEL Subprogram

CPUEL	Description
What it does	Returns Predict information about a field in a file. This subprogram finds a field in a Predict file and returns information about the field.
PDA's used	<ul style="list-style-type: none"> ● CPAEL ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-EL

Drivers Menu Option



```

CTEEL                      N a t u r a l   C o n s t r u c t          CTEEL11
Aug 14                      D r i v e r   f o r   s u b p r o g r a m   C P U E L          1 o f   2

*File Name..: _____ DDM Prefix: _____
*Field Name : _____
Simple Outputs: _

Fld Found...:      Adabas Fld Name:      Fld Format....:      Field Uq :
Ver Found...:      Fld Length.....:      Predict Format:      De Type..:
Lvl Number...:     Sign.....:      Suppression...:     Gr Struct:
Occurrences.:      Fld Type.....:      A/Descend.....:     Pe Ind...:
                   Fld Redefined   :      Rank.....:

Edit Mask...:      Field Headings:
DDM Fld Name:
Index Name..:
Fld Sequence:

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                left  right mai
    
```

Press Enter to display the second panel. For example:

```

CTEEL                      N a t u r a l   C o n s t r u c t          CTEEL21
Aug 14                      D r i v e r   f o r   s u b p r o g r a m   C P U E L          2 o f   2

File Name..:
Field Name :

LEVEL
-----
DDM Field Name      Field Type      Is Redefined

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                left  right mai
    
```

CPUELDE Subprogram

CPUELDE	Description
What it does	Returns a description attribute from a specified file. This subprogram receives the name of a file and finds a description attribute. It returns the names of all fields that have the DESCRIPTION keyword.
PDAAs used	<ul style="list-style-type: none"> ● CPAELDE ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-FI ● SYSDIC-EL ● SYSDIC-KY

CPUELKY Subprogram

CPUELKY	Description
What it does	Returns keywords linked to a field in a specified file. This subprogram receives the name of a file and field; it returns keywords linked to the field.
PDAAs used	<ul style="list-style-type: none"> ● CPAELKY ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-FI ● SYSDIC-EL ● SYSDIC-KY

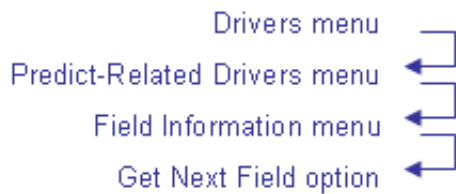
CPU-FREL Subprogram

CPU-FREL	Description
What it does	Retrieves information about a foreign relationship and CALLNATs a pass-through subprogram. This subprogram passes CPA-FREL, CU--PDA, and CSASTD to the pass-through subprogram.
PDAAs used	<ul style="list-style-type: none"> ● CPARLRD ● CU--PDA ● CPA-FREL ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-FI ● SYSDIC-EL

CPUELNX Subprogram

CPUELNX	Description
What it does	<p>Returns field-by-field information if it is called repeatedly. This subprogram receives the name of a Predict file, the CPAELNX data area (contains options for field types), and the CPRELNX data area (contains information about current processing), and logically reads through the fields in the file.</p> <p>Notes:</p> <ol style="list-style-type: none"> 1. CPRELNX contains reserved variables that keep track of the current position; it must not be modified by the calling program. 2. For information about INPUT/OUTPUT parameters, refer to the CPAELNX data area in the SYSCST library.
PDA's used	<ul style="list-style-type: none"> ● CPAELNX ● CPRELNX ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-EL ● SYSDIC-FI

Drivers Menu Option



```

CTEELNX          N a t u r a l   C o n s t r u c t          CTEENX11
Aug 14           D r i v e r   f o r   s u b p r o g r a m   C P U E L N X          1   o f   2

*File Name....: _____ First Time : X EOF.....:
DDM Prefix...: _____

Redef Base Fld: _ Super Subs: _ Mus.....: _ Nulls Only : _ Counters: _
First Redefine: _ Phonetics : _ Pe Groups : _ Seq Only...: _ Groups..: _
All Redefines : _ Hypers....: _ Pes.....: _ Uq Only....: _ Fillers : _
Max Rede Rank : _ Derived...: _ Mus in Pes: _ VE Only....: _ REDE STR: _

Fld Name.....:          Fld Type...:
Fld Format....:          Length.....:
Predict Format:          Sign.....:

Adabas Name...:  Fld Def...:  De Type...:  Fld Count..:  Rank..:
Level Number..:  Fld Uq....:  Pe Ind....:  Occurrences:

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help retrn quit                                left  right mai

```

Press Enter to display the second panel. For example:

```

CTEELNX          N a t u r a l   C o n s t r u c t          CTEENX21
Aug 14           D r i v e r   f o r   s u b p r o g r a m   C P U E L N X          2   o f   2

      F i e l d   H e a d i n g s
-----

                                IMS Offset....:          Access Lvl:
                                IMS Fld Name..:          Update Lvl:
                                IMS Fld Length:

Index Name..:
DDM Fld Name:

Edit Mask...:
Level Type Trail:  ->  ->  ->  ->  ->  ->  ->
Redefine Trail..:  ->  ->  ->  ->  ->  ->  ->

Fld is Redefined:  Redefine Cnt:

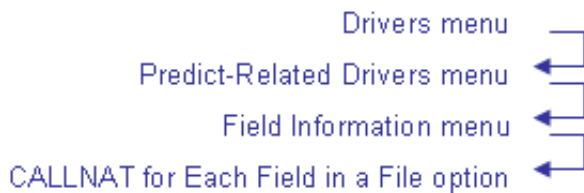
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help retrn quit                                left  right mai

```

CPUELRD Subprogram

CPUELRD	Description
What it does	<p>Reads through the fields in a Predict file, issues a CALLNAT for the specified subprogram for each field, and passes information about the field to the subprogram. It receives:</p> <ul style="list-style-type: none"> ● the name of a file ● the name of a subprogram to CALLNAT ● the selection criteria (in CPAELRD.INPUTS) <p>The subprogram traverses the specified file. For each selected field, it CALLNATs the passed subprogram to process the current field.</p>
PDAs used	<ul style="list-style-type: none"> ● CPAELRD ● CU--PDA (model PDA) ● CSAPASS (can be redefined as required and used to store additional information that must be preserved between CALLNATs) ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-EL

Drivers Menu Option



```

CTEELRD          N a t u r a l   C o n s t r u c t          CTEELRD1
Aug 14          D r i v e r   f o r   s u b p r o g r a m   C P U E L R D          1   o f   1

*File Name.....: _____          Fld Count.....:
*Key Name.....:  _____          Level.....:
CALLNAT.....:   CTEL RDSM          M a x   R e d e   R a n k . . :   _

ReDe Base Fld: _ S P s / S B s . . : _ P e s . . . : _ P e   G r o u p : _ O n l y   V E . . : _ F i l l e r s : _
First ReDe...: _ P h o n e t i c s : _ M u s . . . : _ M u   i n   P e : _ O n l y   U Q . . : _ D e r i v e d : _
All ReDe.....: _ H y p e r s . . . : _ G r o u p s : _ C o u n t e r s : _ O n l y   N u l l : _ R e d e   S t : _

Fld Name   :                               F o r m a t   :   P R D   F o r m a t   :
DDM Field  :                               Fld UQ   :   L e n g t h . . . . :
Index.... :                               T y p e . . . :   A d a b a s   N a m e :
Headings  :                               D e   T y p e :   O c c u r r e n c e s :
                                                P e   T y p e :                               :
Edit Mask :                               R a n k . . . :                               :
Type Trail:                               R e d e f . . :   R e D e   C o u n t   :
ReDe Trail:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                       bkwrđ frwrđ                               mai
    
```

Note:

If you change the name of the subprogram in the CALLNAT field, the specified subprogram must have the same parameters as those in the PDAs used by CPUEL RD.

CPUELVE Subprogram

CPUELVE	Description
What it does	Returns the verification rule names for a field in a file. This subprogram finds a field in Predict and returns the names of the verification rules of type N (Natural Construct).
PDAs used	<ul style="list-style-type: none"> ● CPAELVE ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-EL

Drivers Menu Option



```

CTEELVE          N a t u r a l   C o n s t r u c t          CTEELVE1
Aug 14           D r i v e r   f o r   s u b p r o g r a m   C P U E L V E          1   o f   1

*File Name : _____      Field Found.....:
*Field Name: _____      Num of Verifications:

+-----+
| 1__  VERIFICATION NAME |
|-----|
|                          |
|                          |
|                          |
+-----+

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                bkwrđ frwrđ                mai
    
```

CPUEXIST Subprogram

CPUEXIST	Description
What it does	Verifies the existence of a specified Predict object. This subprogram receives the name and type of an object and verifies its existence in Predict.
PDA's used	<ul style="list-style-type: none"> ● CPAEXIST ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-SY ● SYSDIC-PR ● SYSDIC-KY ● SYSDIC-DB ● SYSDIC-FI ● SYSDIC-RL ● SYSDIC-VE

Drivers Menu Option



```

CTEXIST          N a t u r a l   C o n s t r u c t          CTEXST1
Aug 14          D r i v e r   f o r   s u b p r o g r a m   C P U E X I S T          1   o f   1

Object Name: _____ Object Exists:
Object Type: __ (SY,PR,KY,FI,DB,RL,VE)

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help retn quit                                     mai
    
```

CPUFI Subprogram

CPUFI	Description
What it does	Returns Predict information about a file. This subprogram receives the name of a file and returns Predict information about that file.
PDA's used	<ul style="list-style-type: none"> ● CPAFI ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-FI

Drivers Menu Option



```

CTEFI          N a t u r a l   C o n s t r u c t          CTEFI1
Aug 14          D r i v e r   f o r   s u b p r o g r a m   C P U F I          1   o f   1

*File Name: _____ Ripp File Nr..:
File Type:              Ext File Nr..:

Master File Name..:
Primary Seq Field :

DDM Prefix.....:      IMS DB Number.: 00
DDM File Name.....:   IMS File Level:
IMS Parent File...:   IMS File Nr...: 00
IMS Root File Name:   IMS Seg Type..:
IMS DBD Name.....:    IMS DDM Suffix:
IMS Seg Name.....:    DDM Matches...:
IMS Root Seg Name :

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help retn quit                                     mai
    
```

CPUHOLD Subprogram

CPUHOLD	Description
What it does	Determines the hold field for a file. This subprogram receives the name of a file and determines the hold field for the file. To define a hold field, attach the HOLD-FIELD keyword to the field in Predict.
PDAAs used	<ul style="list-style-type: none"> ● CPAHOLD ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-FI ● SYSDIC-EL

CPUKY Subprogram

CPUKY	Description
What it does	Retrieves information related to a Predict keyword. You can use the keyword comments to store attribute values that can be returned by this subprogram.
PDAAs used	<ul style="list-style-type: none"> ● CPAKY ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-KY ● SYSDIC-EL

CPUREDEF Subprogram

CPUREDEF	Description
What it does	Generates redefinitions for compound keys, superdescriptors, or redefined fields in Predict. This subprogram invokes the CPUXPAND subprogram, which retrieves the components of the field to be redefined. Redefinitions can be generated in either inline or external data area format.
PDAAs used	<ul style="list-style-type: none"> ● CPAREDEF ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-EL

Drivers Menu Option



```

CTEREDEF          N a t u r a l   C o n s t r u c t          CTERDEF1
Aug 14            Driver for subprogram CPUREDEF          1 of 1

*File : _____          Redef Level.....: _
*Field: _____          Change Format N to A: _

Super Options
-----
Include Deriv Level: _   Inside Histogram: _
Include Redef Level: _   Omit Format.....: _

Resets Required:

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                     mai
  
```

CPURL Subprogram

CPURL	Description
What it does	Returns information about a relationship in Predict. This subprogram receives a Predict relationship name and returns information about the relationship.
PDA's used	<ul style="list-style-type: none"> ● CPARL ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-RL

Drivers Menu Option



```

CTERL          N a t u r a l   C o n s t r u c t          CTERL1
Aug 14         D r i v e r   f o r   s u b p r o g r a m   C P U R L       1   o f   1

*Relationship Name: _____ Relationship Found:
                                   Relationship Type :

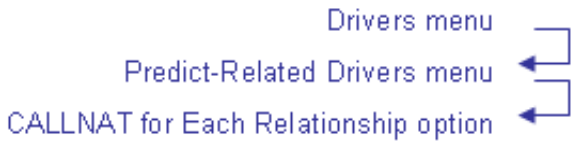
      Relationship File           Relationship Field           Card
      -----
      Ddm Relationship Field      Minimum           Average           Maximum
      -----

Constraint Type Upd:                Db2 Constraint Name:
Constraint Type Del:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                     mai
    
```

CPURLRD Subprogram

CPURLRD	Description
What it does	<p>Retrieves the Predict relationships for a specified file, and optionally a specified type. This subprogram receives:</p> <ul style="list-style-type: none"> ● the name of a file ● a relationship type (optional) ● the name of a subprogram (in CPARLRD.INPUTS) <p>It finds relationships for the specified file, issues a CALLNAT to the specified subprogram, and passes the information about the relationship to the subprogram for processing.</p>
PDA's used	<ul style="list-style-type: none"> ● CPARLRD ● CU--SYSLIBSPDA (model PDA) ● CSAPASS (can be redefined as required and used to store additional information that must be preserved between CALLNATs) ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-FI ● SYSDIC-KL

Drivers Menu Option



```

CTERLRD      N a t u r a l   C o n s t r u c t      CTERLRD1
Aug 14      Driver for subprogram CPURLRD      1 of 1

*File Name.....: _____
Relationship Type.....: _
CALLNAT.....: CTERLRDSM
Relationship Count....:
Relationship Name.....:
Relationship File ....:
Relationship Field....:
DDM Relationship Field:
Cardinality.....:
Minimum.....:
Average.....:
Maximum.....:
DB2 Constraint Name...:
Constraint Type Upd...:
Constraint Type Del...:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10-
      help  retrn  quit
  
```

CPUSUPER Subprogram

CPUSUPER	Description
What it does	Returns the definition for a super/subdescriptor (or DB2 compound key). This subprogram receives the name of a superdescriptor or subdescriptor (or DB2 compound key) and the name of the Predict file or table to which it belongs. It returns information about the derived fields.
PDA's used	<ul style="list-style-type: none"> ● CPASUPER ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-EL

Drivers Menu Option




```

CTESUPER          ***** Predict Related Subprograms *****          CTESUPR1
Oct 09            - Driver for subprogram CPUSUPER -                    03:08 PM

*File Name : _____ Superde Length....:
*Field Name: _____ Superde Format....:

  Contains Repeating Fields:                                C#Derivation Group:
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1__          Source Field Name          Start End  A/ Fld Sup PE  Dimension |
|          Char Char D Typ Opt Ind 1 2 3 |
+-----+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+-----+
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                    bkwrdr frwrdr                    mai

```

CPUUNIQ Subprogram

CPUUNIQ	Description
What it does	Determines the unique description field (primary key). This subprogram receives the name of a file and determines the unique description field (primary key) for the file.
PDA's used	<ul style="list-style-type: none"> ● CPAUNIQ ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-FI ● SYSDIC-EL

CPUVE Subprogram

CPUVE	Description
What it does	Prints verification rules to the source buffer. This subprogram prints either the code or the data definition for a type N (Natural Construct) verification rule to the source buffer.
PDA's used	<ul style="list-style-type: none"> ● CPAVE ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-VE-ACT

Drivers Menu Option



```

CTEVE          N a t u r a l   C o n s t r u c t          CTEVE1
Aug 14         D r i v e r   f o r   s u b p r o g r a m   C P U V E   1   o f   1

Verification Name: _____ Verification Found:
*User View Name...: _____ Rule Generated....:
*DDM Field Name...: _____

Generate Data....: _
Occurrences.....: _____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                                     mai
  
```

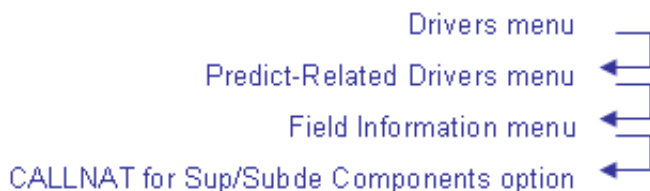
CPUVERUL Subprogram

CPUVERUL	Description
What it does	Returns information about Predict verification rules.
PDA's used	<ul style="list-style-type: none"> ● CPAVERUL ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-VE

CPUXPAND Subprogram

CPUXPAND	Description
<p>What it does</p>	<p>Expands a super/subdescriptor or redefined field. This subprogram receives:</p> <ul style="list-style-type: none"> ● the name of a super/subdescriptor (or DB2 compound key) ● the name of the Predict file (or table) to which the key belongs ● the expansion options ● the name of a subprogram to CALLNAT (in CPAXPAND.INPUTS) ● the parameters in the model PDA (CU--PDA) ● an additional A1/1:v parameter (CSAPASS) <p>It expands the specified super/subdescriptor (or DB2 compound key) into its underlying components. For each component, it CALLNATs the specified subprogram.</p> <p>Note: When this subprogram expands a superdescriptor, redefinitions of the derived fields are included.</p>
<p>PDA's used</p>	<ul style="list-style-type: none"> ● CPAXPAND ● CU--PDA ● CSAPASS ● CSASTD
<p>Files accessed</p>	<ul style="list-style-type: none"> ● SYSDIC-EL

Drivers Menu Option



```

CTEXPAND          N a t u r a l   C o n s t r u c t          CTEXPN11
Aug 14            Driver for subprogram CPUXPAND            1 of 3

*File Name.....: _____ Phantom Bytes: _
*Base Field Name: _____ Fillers.....: _
CALLNAT.....: CTELRDSM      P

Base Field Information                                Field Headings
-----
Sequence:      Adabas Field Name:      :
Format..:      Field Definition :      :
Length..:      Field Type.....:      :

Edit Mask.....:
DDM Field Name :

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help retrn quit                                left right mai

```

Note:

If you change the name of the subprogram in the CALLNAT field, the specified subprogram must have the same parameters as those in the PDAs used by CPUXPAND.

Press Enter to display the second panel. For example:

```

CTEXPAND          N a t u r a l   C o n s t r u c t          CTEXPN21
Aug 14            Driver for subprogram CPUXPAND            2 of 3

Derived Field Information                            Field Headings
-----
First Showing.:      :
Field Count...:      :
Whole Field...:      :

Sequence.....:      Adabas Field Name:      Start Character:
Format.....:      Field Definition :      End   Character:
Length.....:      Field Type.....:

Edit Mask.....:
Field Name....:
DDM Field Name:

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help retrn quit                                left right mai
Scrolling performed

```

Press Enter to display the third panel. For example:

```

CTEXPAND          N a t u r a l   C o n s t r u c t          CTEXPN31
Aug 14            D r i v e r   f o r   s u b p r o g r a m   C P U X P A N D          3   o f   3

Ascending/Descending
Expanded Field Information          F i e l d   H e a d i n g s
-----
Field Count..:                    :
Offset Start..:                   :
Offset End....:                   :

Sequence.....:                    P r e d i c t   F o r m a t . . . :          S p e c i a l   c h a r a c t e r i s t i c :
Format.....:                      F i e l d   D e f i n i t i o n   :
Length.....:

Edit Mask.....:
Field Name....:
DDM Field Name:

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit                      left  right mai
Scrolling performed
    
```

Predict-Related Helproutines (CPH*)

You can attach the following helproutines to fields that require the input of Predict information. They are active helproutines that fill the field to which they are attached.

Note:

Some of the following routines provide help information, although they are coded as subprograms and not as helproutines. This provides greater flexibility to access help information.

This section covers the following topics:

- CPHEL Subprogram
- CPHEL B Subprogram
- CPHFI Helproutine
- CPHFIB Subprogram
- CPHPRED Helproutine
- CPHRL Helproutine
- CPHSET Helproutine

CPHEL Subprogram

CPHEL	Description
What it does	Browses the fields in a file for selection. This subprogram receives the name of a Predict file. (If no file name is specified, it provides file selection.) It browses all the fields in the specified file and returns the selected field.
Attached to	Input of a Predict field name.
PDA's used	<ul style="list-style-type: none"> ● CPAHEL ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-FI

CPHELB Subprogram

CPHELB	Description
What it does	Browses the fields in a file for selection. This subprogram receives the name of a file and browses all the fields in the file for selection. Optionally, this subprogram can browse only the descriptor fields. Note: For information about INPUT/OUTPUT parameters, refer to the CPHELBA data area in the SYSCST library.
PDA's used	<ul style="list-style-type: none"> ● CPAHEL ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-EL

CPHFI Helproutine

CPHFI	Description
What it does	Browses Predict views/files for selection. This helproutine browses all the views and files in Predict for selection.
Attached to	Input of a Predict file name.
Parameters used	<ul style="list-style-type: none"> ● #PDA-FILE(A32)
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-FI

CPHFIB Subprogram

CPHFIB	Description
What it does	Browses Predict views and files for selection.
Parameters/PDAs used	<ul style="list-style-type: none"> ● #PDA-KEY(A32) ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-FI

CPHPRED Helproutine

CPHPRED	Description
What it does	<p>Browses Predict objects (by object type) for selection. This helproutine receives an object type and browses the Predict objects of that type for selection. Valid object types are:</p> <ul style="list-style-type: none"> ● S (system) ● P (program) ● K (keyword) ● M (module) ● R (report)
Attached to	Input of a Predict object type.
Parameters used	<ul style="list-style-type: none"> ● #PDA-TYPE(A1) ● #PDA-KEY(A32)
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-SY ● SYSDIC-PR ● SYSDIC-KY ● SYSDIC-RE ● SYSDIC-MO

CPHRL Helproutine

CPHRL	Description
What it does	Browses the names of Predict relationships for selection. This helproutine receives the names of a Predict relationship and a file and returns the selected relationship. If a file name is specified, the helproutine browses only the Predict relationships for that file. If no file name is specified, it browses all existing relationships.
Attached to	Input of a Predict relationship name.
Parameters used	<ul style="list-style-type: none"> ● #PDA-FILE(A32) ● #PDA-RELATIONSHIP-NAME(A32)
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-FI ● SYSDIC-RL

CPHSET Helproutine

CPHSET	Description
What it does	<p>Sets a flag to indicate that help was requested for a field. This helproutine receives the name of a parameter and sets a flag to indicate help was requested. The parameter should be checked after the INPUT statement. If a flag is set, for example, reset the flag and issue CALLNATs to do the help processing.</p> <p>This technique allows the helproutine to access all data entered in a single panel transaction. When you generate a browse subprogram, for example, you can type the file name (without pressing Enter) on the Additional Parameters panel and request help for a field.</p>
Attached to	Any input field.
Parameters used	<ul style="list-style-type: none"> ● #PDA-SET-HELP(L)
Files accessed	<ul style="list-style-type: none"> ● SYSDIC-FI ● SYSDIC-RL

General Purpose Generation Subprograms (CU--*)

The subprograms described in this section are general purpose generation subprograms. These subprograms are identified by a CU-- prefix.

CU--EM Subprogram

CU--EM	Description
What it does	Returns edit masks used by the generated programs for displaying date and time fields. This subprogram can be changed to suit your standards. Changes to this routine should be made in a higher level steplib to minimize maintenance. Unless you modify your models, the date and time field edit masks should not be longer than nine characters.
PDAs used	<ul style="list-style-type: none"> ● CU--EMA

CU--LRP Subprogram

CU--LRP	Description
What it does	Returns the left and right prompt displayed on the Natural Construct panels. The left prompt displays the current month and day in *DATX (EM=LLL''DD), which can be language sensitive. The right prompt displays the "1 of 1" or "1 of 3" panel indicators, depending on the number of panels. This prompt uses several control record fields to build the prompt position indicators, which are compressed on both sides of the "of" indicator.
Parameters/PDAs used	<ul style="list-style-type: none"> ● #PDA-LEFT-PROMPT(A9) ● #PDA-LEFT-INDICATOR(A4) ● #PDA-RIGHT-PROMPT-OF(A4) ● #PDA-RIGHT-INDICATOR(A4) ● #PDA-RIGHT-PROMPT(A9) ● CSASTD

CU--MSG Subprogram

CU--MSG	Description
What it does	<p>Returns the text for an application error message. It receives a message number in #PDA-FRAME-PARM. After ensuring this literal is numeric, it retrieves the short message for the SYSTEM application and the *Language variable.</p> <p>The error message is written (left-justified and enclosed within single quotes) to the source buffer, thus substituting for the frame parameter. The usual search criteria and defaults (English) apply. The following example shows a code frame:</p> <pre> USE-MSG-NR 1 ASSIGN MSG-INFO.##MSG-NR = 8123 " ELSE 1 ASSIGN MSG-INFO.##MSG = " SUBPROGRAM:CU--MSG PARAM: 8123 N "</pre>
PDAs used	<ul style="list-style-type: none"> ● CU--PDA ● CSASTD
Files accessed	<ul style="list-style-type: none"> ● Application error message file

CU--UL Subprogram

CU--UL	Description
What it does	<p>Returns the underscore line used on Natural Construct panels. This subprogram receives an underscore character set and creates the underscore line. The character(s) specified on the control record (A4) is duplicated to fill the A80 length.</p>
Parameters/PDAs used	<ul style="list-style-type: none"> ● #PDA-UNDERSCORE(A4) ● #PDA-UNDERSCORE-LINE(A80) ● CSASTD