Natural System Error Messages 0601-0650

NAT0601: "END-START" missing for AT START OF DATA in line :1:.

Text	"END-START" missing for AT START OF DATA in line ....
Expl.	In structured mode, an AT START OF DATA statement block must be terminated with a corresponding END-START statement. Example: READ ... AT START OF DATA MOVE ... DISPLAY ... END-START DISPLAY ... END-READ END
Actn.	Correct error in program.

NAT0602: "END-ENDDATA" missing for AT END OF DATA in line :1:.

Text	"END-ENDDATA" missing for AT END OF DATA in line ....
Expl.	In structured mode, an AT END OF DATA statement block must be terminated with a corresponding END-ENDDATA statement. Example: READ ... AT END OF DATA COMPUTE ... DISPLAY ... END-ENDDATA DISPLAY ... END-READ END
Actn.	Correct error in program.

NAT0603: "END-BREAK" missing for AT BREAK statement in line :1:.

Text	"END-BREAK" missing for AT BREAK statement in line ....
Expl.	In structured mode, an AT BREAK statement block must be terminated by a corresponding END-BREAK statement. Example: READ ... AT BREAK ... DISPLAY ... END-BREAK DISPLAY END-READ END
Actn.	Correct error in program.

NAT0604: "END-TOPPAGE" missing for AT TOP OF PAGE in line :1:.

Text	"END-TOPPAGE" missing for AT TOP OF PAGE in line ....
Expl.	In structured mode, an AT TOP OF PAGE statement block must be terminated by a corresponding END-TOPPAGE statement. Example: READ ... DISPLAY ... AT TOP OF PAGE WRITE ... END-TOPPAGE DISPLAY ... END-READ END
Actn.	Correct error in program.

NAT0605: "END-ENDPAGE" missing for AT END OF PAGE in line :1:.

Text	"END-ENDPAGE" missing for AT END OF PAGE in line ....
Expl.	In structured mode, an AT END OF PAGE statement block must be terminated by a corresponding END-ENDPAGE statement. Example: READ ... AT END OF PAGE DISPLAY ... WRITE TRAILER ... END-ENDPAGE DISPLAY ... END-READ END
Actn.	Correct error in program.

NAT0606: "END-BEFORE" missing for BEFORE BREAK in line :1:.

Text	"END-BEFORE" missing for BEFORE BREAK in line ....
Expl.	In structured mode, a BEFORE BREAK PROCESSING statement block must be terminated by a corresponding END-BEFORE statement.
Actn.	Correct error in program.

NAT0607: "END-NOREC" missing for IF NO RECORDS FOUND.

Text	"END-NOREC" missing for IF NO RECORDS FOUND.
Expl.	In structured mode, an IF NO RECORDS FOUND clause must be terminated with a corresponding END-NOREC statement. Example: FIND ... IF NO RECORDS FOUND DISPLAY ... END-NOREC DISPLAY END-FIND
Actn.	Correct error in program.

NAT0608: "END-ENDFILE" missing for AT END OF FILE in line :1:.

Text	"END-ENDFILE" missing for AT END OF FILE in line ....
Expl.	In structured mode, an AT END OF FILE statement block must be terminated by a corresponding END-ENDFILE statement. Example: READ WORK FILE AT END OF FILE COMPUTE ... DISPLAY ... END-ENDFILE DISPLAY ... END
Actn.	Correct error in program.

NAT0609: "END-ERROR" missing for ON ERROR statement in line :1:.

Text	"END-ERROR" missing for ON ERROR statement in line ....
Expl.	In structured mode, an ON ERROR statement block must be terminated by a corresponding END-ERROR statement. Example: FIND ... UPDATE ... ON ERROR RETRY END-ERROR END TRANSACTION END-FIND END
Actn.	Correct error in program.

NAT0610: This statement is not permitted in reporting mode.

Text	This statement is not permitted in reporting mode.
Expl.	An attempt was made to issue a statement which is valid in structured mode only.
Actn.	Make sure you are working in the correct programming mode.

NAT0611: Loop/condition not properly closed.

Text	Loop/condition not properly closed.
Expl.	An active loop or condition was not properly closed. For example, in structured mode, a FIND statement was issued without a corresponding END-FIND statement.
Actn.	Check program and correct error.

NAT0612: Specified close loop only permitted in structured mode.

Text	Specified close loop only permitted in structured mode.
Actn.	Check programming mode, program, and correct error.

NAT0613: END-FIND/END-SELECT missing for FIND/SELECT in line :1:.

Text	END-FIND/END-SELECT missing for FIND/SELECT in line ....
Expl.	In structured mode, a loop initiated with a FIND/SELECT statement must be closed with a corresponding END-FIND/END-SELECT statement. Examples: FIND ... * SELECT ... UPDATE * DISPLAY ... END TRANSACTION * END-SELECT END-FIND *
Actn.	Supply END-FIND/END-SELECT to close the FIND/SELECT loop.

NAT0614: END-READ missing for READ in line :1:.

Text	END-READ missing for READ in line ....
Expl.	In structured mode, a READ statement must be followed by a corresponding END-READ statement. Example: READ ... DISPLAY COMPUTE ... END-READ
Actn.	Supply END-READ to close the READ loop.

NAT0615: END-HISTOGRAM missing for HISTOGRAM in line :1:.

Text	END-HISTOGRAM missing for HISTOGRAM in line ....
Expl.	In structured mode, a HISTOGRAM statement must be followed by a corresponding END-HISTOGRAM statement. Example: HISTOGRAM ... DISPLAY ... END-HISTOGRAM
Actn.	Supply END-HISTOGRAM to close the HISTOGRAM loop.

NAT0616: END-SORT missing for SORT in line :1:.

Text	END-SORT missing for SORT in line ....
Expl.	In structured mode, a SORT statement must be followed by a corresponding END-SORT statement. Example: FIND ... END-ALL SORT ... DISPLAY ... END-SORT
Actn.	Supply END-SORT to close the SORT loop.

NAT0617: END-REPEAT missing for REPEAT in line :1:.

Text	END-REPEAT missing for REPEAT in line ....
Expl.	In structured mode, a REPEAT statement must be followed by a corresponding END-REPEAT statement. Example: REPEAT ... ... END-REPEAT
Actn.	Supply END-REPEAT to close the REPEAT loop.

NAT0618: END-FOR missing for FOR in line :1:.

Text	END-FOR missing for FOR in line ....
Expl.	In structured mode, a FOR statement must be followed by a corresponding END-FOR statement. Example: FOR ... ... END-FOR
Actn.	Supply END-FOR to close the FOR loop.

NAT0619: END-WORK missing for READ WORK FILE in line :1:.

Text	END-WORK missing for READ WORK FILE in line ....
Expl.	In structured mode, a READ WORK FILE statement must be followed by a corresponding END-WORK statement. Example: READ WORK FILE ... ... END-WORK
Actn.	Supply END-WORK to close the READ WORK FILE loop.

NAT0620: END-FILE missing for CALL FILE in line :1:.

Text	END-FILE missing for CALL FILE in line ....
Expl.	In structured mode, a CALL FILE statement must be followed by a corresponding END-FILE statement. Example: CALL FILE ... ... END-FILE
Actn.	Supply END-FILE to close the CALL FILE loop.

NAT0621: END-LOOP missing for CALL LOOP in line :1:.

Text	END-LOOP missing for CALL LOOP in line ....
Expl.	In structured mode, a CALL LOOP statement must be followed by a corresponding END-LOOP statement. Example: CALL LOOP ... ... END-LOOP
Actn.	Supply END-LOOP to close the CALL LOOP loop.

NAT0622: RETURN/END-SUBROUTINE missing for DEFINE SUBROUTINE in line :1:.

Text	RETURN/END-SUBROUTINE missing for DEFINE SUBROUTINE in line ...
Expl.	This error occurs under one of the following conditions: 1) A subroutine was started in a conditional statement block (e.g. IF, DECIDE) or in a loop-initiating statement (e.g. REPEAT, FIND), but it has not yet been closed at the end of this statement (e.g. END-IF, END-REPEAT).Examples: IF FIELD = "A" THEN * REPEAT DEFINE SUBROUTINE ABC * DEFINE SUBROUTINE ABC WRITE "HELLO" * WRITE "HELLO" END-IF * END-REPEAT 2) A subroutine was started, but not yet closed at program end. Example: DEFINE SUBROUTINE ABC END
Actn.	END Supply RETURN or END-SUBROUTINE to close the subroutine.

NAT0623: Variable/field/view must be defined in DEFINE DATA.

Text	Variable/field/view must be defined in DEFINE DATA.
Expl.	If a DEFINE DATA statement is present, all variables/fields to be used must be defined in the DEFINE DATA statement in an LDA, GDA or PDA, which are referenced with the keyword USING in the DEFINE DATA statement. If this error occurs for a DDM and a DEFINE DATA statement is present, a view of the DDM (and of any database field referenced in the program) must be defined in the DEFINE DATA statement.
Actn.	Check program and correct error.

NAT0624: Statement END-... either missing or misplaced.

Text	Statement END-... either missing or misplaced.
Expl.	Example: READ ... IF AGE > 65 THEN COMPUTE ... DISPLAY ... END-READ END-IF END The statement END-IF must be before the END-READ and after the last statement to be executed in the THEN-clause of the IF-statement. This could be after the COMPUTE or the DISPLAY statement, depending on the actual function the program is to perform. Statements with the syntax "xxx ... END-xxx" must be correctly paired.
Actn.	Make sure that all statement blocks are terminated with corresponding "END-..." statements, and that the "xxx" and "END-xxx" statements are correctly paired.

NAT0625: END-DECIDE missing for DECIDE statement.

Text	END-DECIDE missing for DECIDE statement.
Expl.	A statement block initiated with a DECIDE ON or DECIDE FOR statement must be terminated with an END-DECIDE statement. This error may also occur if an END-DECIDE statement is misplaced.
Actn.	Make sure that each DECIDE statement is correctly paired with a corresponding END-DECIDE statement.

NAT0626: Statement not supported with :1:.

Text	Statement not supported with ....
Expl.	The statement cannot be used with the database type you are using. For details as to which statements support which database types, see the Natural documentation.
Actn.	Check program and correct error.

NAT0627: Value range with keyword "TO" not supported for :1:.

Text	Value range with keyword "TO" not supported for ....
Expl.	The statement option "TO" cannot be used with the database type you are using. When "TO" is used to specify a value range in a READ or HISTOGRAM statement, the end-value check will not be done by Natural, but is performed by the database. However, this requires a corresponding capability of the accessed database. Therefore, option "TO" can only be used if the underlying database is Adabas Version 7 (or above), VSAM, DL/I or DB2. For details as to which statements support which database types, see the Natural documentation.
Actn.	Check program and correct error.

NAT0628: Invalid usage of MULTI-FETCH option for :1:.

Text	Invalid usage of MULTI-FETCH option for ....
Expl.	The MULTI-FETCH option can only be used if the underlying database is Adabas Version 5.3 (or above). Moreover, if a READ or HISTOGRAM statement contains a "DYNAMIC .. SEQUENCE" clause, the use of a MULTI-FETCH option is not permitted. For further details, see the Natural documentation.
Actn.	Check program and correct error.

NAT0629: Incompatible range combinations in relational expression.

Text	Incompatible range combinations in relational expression.
Expl.	When two arrays are compared in a relational expression, the number of occurrences in each dimension of the one array must match the number of occurrences in the corresponding dimension of the other array. If no corresponding dimension is defined for the other array, the dimension is assumed to be 1. A variable number of occurrences in one dimension must not be compared with a fixed number of occurrences in the corresponding dimension of the other array.
Actn.	Correct error in program.

NAT0630: Error in RESET INITIAL statement.

Text	Error in RESET INITIAL statement.
Expl.	See the Natural documentation for information on the correct usage of the RESET INITIAL statement. Note in particular that a RESET INITIAL of fields resulting from a redefinition is not possible, and RESET INITIAL cannot be applied to database fields.
Actn.	Check program and correct error.

NAT0631: Invalid index specified in arithmetic/ASSIGN statement.

Text	Invalid index specified in arithmetic/ASSIGN statement.
Actn.	Check program and correct error.

NAT0632: Invalid view structure defined for HISTOGRAM statement.

Text	Invalid view structure defined for HISTOGRAM statement.
Expl.	When using the HISTOGRAM statement with a view, it must be defined with only one variable: the search variable.
Actn.	Check program and correct error.

NAT0633: Invalid use of view which contains super-/subdescriptor.

Text	Invalid use of view which contains super-/subdescriptor.
Expl.	Depending on the Adabas version, a sub- or superdescriptor may be read with a HISTOGRAM statement only or with all database access statements. However, an UPDATE or STORE statement must not be applied to a sub- or superdescriptor.
Actn.	Correct error in program.

NAT0634: Access criterion not supported for this type of database.

Text	Access criterion not supported for this type of database.
Expl.	The DBID of this DDM identifies a non-Adabas database. Non-Adabas databases introduce certain restrictions for Natural"s database-access statements.
Actn.	See the Natural documentation for restrictions which apply to the type of database you wish to access.

NAT0635: Incorrect use of "BY VALUE" or "OPTIONAL" in DEFINE DATA.

Text	Incorrect use of "BY VALUE" or "OPTIONAL" in DEFINE DATA.
Expl.	The keyword "BY VALUE" or "OPTIONAL" may only be supplied for fields defined in a DEFINE DATA PARAMETER clause. However, this is not permitted for REDEFINE fields or groups.
Actn.	Correct error in program.

NAT0636: Incorrect usage of STARTING WITH ISN clause for :1:.

Text	Incorrect usage of STARTING WITH ISN clause for ....
Expl.	A STARTING WITH ISN clause can only be used for - Adabas access : FIND READ logical READ physical - VSAM access : READ physical
Actn.	Correct error in program.

NAT0637: Invalid USING clause specified.

Text	Invalid USING clause specified.
Expl.	Invalid USING clause specified in OPEN CONVERSATION statement. Possible reasons for this error are: 1. Keyword USING missing. 2. Incorrect object name specified. Either a constant or a variable of type alpha and length <= 8 is expected.
Actn.	Correct error in program.

NAT0638: Invalid clause in CLOSE CONVERSATION statement.

Text	Invalid clause in CLOSE CONVERSATION statement.
Expl.	The operand in a CLOSE CONVERSATION statement is incorrect. Please specify keyword ALL or *CONVID or a variable of type (I4).
Actn.	Correct error in program.

NAT0639: Referenced GDA cannot be converted to the current version.

Text	Referenced GDA cannot be converted to the current version.
Expl.	When a GDA cataloged with Version 2.2, 2.3 or 3.1 is referenced in a DEFINE DATA GLOBAL statement, the GDA contents will automatically be converted to the layout required by the current Natural version. However, for internal reasons, this does not work for this GDA object.
Actn.	Recatalog the GDA.

NAT0640: Label reference not permitted for this statement type.

Text	Label reference not permitted for this statement type.
Expl.	Labels may be used to reference the following statements only: FIND, READ, GET, HISTOGRAM, SORT, FOR, CALL FILE, CALL LOOP, STORE and REPEAT.
Actn.	Check program and correct error.

NAT0641: Invalid DEFINE PRINTER statement syntax.

Text	Invalid DEFINE PRINTER statement syntax.
Expl.	The correct syntax is: DEFINE PRINTER (logical-name = n) OUTPUT operand1 where "logical-name" is the name allocated to printer, "n" is the printer number in range from 1 to 31, "operand1" is the destination within the online spooling system. Additional reports can be assigned for default with the following names: SOURCE - Output in the Natural source area. CONNECT - Output into a Con-nect folder. DUMMY - Output to be deleted. HARDCOPY - Output to the current hardcopy device.
Actn.	Check program and correct error.

NAT0642: Invalid definition of OUTPUT variable/constant.

Text	Invalid definition of OUTPUT variable/constant.
Expl.	This field must be specified with format A8.
Actn.	Check program and correct error.

NAT0643: END-PROCESS missing for PROCESS PAGE MODAL in line :1:.

Text	END-PROCESS missing for PROCESS PAGE MODAL in line ....
Expl.	This error occurs under one of the following conditions: 1) A PROCESS PAGE MODAL statement was started in a conditional statement block (e.g. IF, DECIDE) or in a loop-initiating statement (e.g. REPEAT, FIND), but it has not been closed yet at the end of this statement (e.g. END-IF or END-REPEAT). Examples: IF FIELD = "A" THEN * REPEAT PROCESS PAGE MODAL * PROCESS PAGE MODAL MOVE #A TO #B * MOVE #A TO #B END-IF * END-REPEAT 2) A PROCESS PAGE MODAL statement was started, but not yet closed at program end. Example: PROCESS PAGE MODAL END
Actn.	Supply END-PROCESS to close the PROCESS PAGE MODAL block.

NAT0644: Maximum number of parameters exceeded.

Text	Maximum number of parameters exceeded.
Expl.	The number of parameters permitted in a CALLNAT or PERFORM statement is limited to 4096.
Actn.	Reduce the number of parameters.

NAT0645: END-PARSE missing for PARSE XML in line :1:.

Text	END-PARSE missing for PARSE XML in line ....
Expl.	In structured mode, a PARSE XML statement must be followed by a corresponding END-PARSE statement. Example: PARSE XML #DOC DISPLAY .. END-PARSE
Actn.	Supply keyword END-PARSE to close the PARSE XML loop.

NAT0646: :1: missing for :2: in line :3:.

Text	... missing for ... in line ....
Expl.	In the case of flexible SQL syntax, the characters ">>" are not recognized for the matching "<<" characters. Probably a Natural comment sign "/*" is contained in the flexible SQL string between the characters "<<" and ">>".
Actn.	Check the flexible SQL syntax for the characters ">>".

NAT0647: Program code not compatible with Version 4.1; reason :1:.

Text	Program code not compatible with Version 4.1; reason ....
Expl.	Compiler parameter V41COMP=ON is set. The following syntax constructions are not accepted by Natural Version 4.1: Possible reason codes: 01: Unicode type not allowed. 07: Wrong system variable. 02: X-Array not allowed. 08: I/O field option (DL=..). 03: Constant length > 253 bytes. 09: INCLUDE parameter > 80 bytes. 04: Wrong compiler option. 10: DYNAMIC field used in data view. 05: Wrong statement clause. 06: Wrong use of SET GLOBALS. For a more detailed error description, see the compiler parameter V41COMP (NTCMPO macro) in the Natural documentation.
Actn.	Do not use the indicated programming feature, or set V41COMP=OFF.

NAT0648: Parameter :1: inconsistently specified. Reason code :2:.

Text	Parameter ... inconsistently specified. Reason code ....
Expl.	The parameter supplied in a CALLNAT statement is not compatible with the corresponding parameter definition in the subprogram. Reason codes: 04 : The CALLNAT parameter is specified as an empty parameter ("nX"), but the definition in the subprogram is not "OPTIONAL". 08 : Mandatory parameter in the subprogram is missing in the CALLNAT statement. 12 : Parameters do not match in format. 16 : Parameters do not match in length. 20 : Parameters do not match in array definitions.
Actn.	Supply compatible parameter specifications.

NAT0649: Database type :1: is not supported.

Text	Database type ... is not supported.
Expl.	The database type is not supported in the current Natural version.
Actn.	Use another database type.

NAT0650: Incorrect comparator used in READ/HISTOGRAM for :1:.

Text	Incorrect comparator used in READ/HISTOGRAM for ....
Expl.	The comparators LT, GT, LE and GE cannot be used with the database type you are using. When the comparator LT, GT, LE or GE is used in a READ/HISTOGRAM statement, the underlying database has to be Adabas Version 7 (or above), VSAM, DL/I or DB2. Moreover, when a Super Natural user file is accessed, the comparators LT, GT, LE and GE are not applicable. For details as to which statements support which database types, see the Natural documentation.
Actn.	Check program and correct error.