

MIB Designer Error Messages

Error Number	Error Text <i>Description & Hints for Error Recovery</i>
0000	<p>File open error: <X>.</p> <p><i>The file <X> could not be read, please check access rights.</i></p>
0010	<p>The length of identifier <X> exceeds 64 characters (RFC2578 §3.1, §7.1.1, §7.1.4).</p> <p><i>It is recommended to use only identifiers with a length of less than 32 characters for interoperability issues. Identifiers that exceed 64 characters in length must be avoided.</i></p>
0050	<p>Encountered lexical error at ...</p> <p><i>The encountered character is not allowed in a SMI MIB module.</i></p>
1000	<p>Syntax error: Encountered "token1" at row <i>r</i>, column <i>c</i>, expected one of the following: ...</p> <p><i>The parser encountered a string it did not expect. Please look at the list of expected tokens carefully in order to determine the trouble cause. If the parser complains about a SMIV2 keyword like MAX-ACCESS, please check whether the first statement after the IMPORTS clause is a MODULE-IDENTITY definition. This is a requirement for a SMIV2 MIB module (RFC2578 §3).</i></p>
1020	<p>Identifier <X> is ambiguous (RFC2578 §3.1).</p> <p>The identifiers (descriptors) in a MIB module must be unique.</p>
1050	<p>The clause <X> is not allowed within this context.</p> <p><i>There are several clauses in SMI that are optional, but if specified those clauses need to be consistent with other clauses in the object definition. Examples for such clauses are the ACCESS, MIN-ACCESS, and SYNTAX clauses in MODULE-COMPLIANCE constructs, which must not be present for variations of NOTIFICATION-TYPES.</i></p>
1100	<p>Imported MIB module <X> unknown.</p> <p><i>The MIB module <X> could not be found in the MIB repository and neither in the MIB modules being compiled. Check whether the MIB module name is not misspelled (this is often the case for older RFC MIBs).</i></p>
1101	<p>Imported MIB module <X> contains a circular import.</p> <p><i>The MIB module <X> imports from a module that either imports itself from <X> or any other module in the import chain imports from a preceding module.</i></p>
1102	<p>MIB module <X> is imported more than once.</p> <p><i>The ASN.1 rules about IMPORTS that SMI is based on require that an</i></p>

	<i>import source is defined not more than once in a module.</i>
1110	<p><X> imported from MIB module <Y> must be imported from <Z> instead.</p> <p><i>For historical reasons, SMI requires to import the MACRO definitions SMI is based on from some ASN.1 modules. For SMIV1 and SMIV2 it is defined which MACRO (construct) is imported from which ASN.1 module. Since those ASN.1 modules (e.g. SNMPv2-SMI) are not SMI themselves, the MACRO definitions have to be removed in order to be able to compile them.</i></p>
1111	<p>Missing import statement for <X> (RFC2578 §3.2).</p> <p><i>To reference an external object, the IMPORTS statement must be used to identify both the descriptor and the module in which the descriptor is defined, where the module is identified by its ASN.1 module name.</i></p>
1112	<p>Imported object <X> is not defined in MIB module <Y>.</p> <p>Use the Edit>Search MIB Repository to search for the MIB module that defines <X>.</p>
1113	<p>Object <X> is imported twice from MIB module <Y>.</p> <p><i>An object definition shall only be imported once from a MIB module.</i></p>
1114	<p><X> cannot be imported (RFC2578 §3.2).</p> <p><i>Notification and trap type definitions as well as SEQUENCE constructs cannot be imported by other MIB modules.</i></p>
1150	<p>Wrong module order within file.</p> <p><i>The MIB file that failed to compile contains more than one MIB module and the order of those MIB modules does not correspond with their import dependencies.</i></p>
1200	<p>The SYNTAX clause of the columnar OBJECT-TYPE definition <X> does not match with the SYNTAX clause of the corresponding SEQUENCE definition.</p> <p><i>The object <X>'s syntax differs in a SEQUENCE definition from its OBJECT-TYPE definition.</i></p>
1202	<p>The OBJECT-TYPE <X> has inconsistent maximum access (RFC2578 §7.3).</p> <p><i>An object <X> has a MAX-ACCESS or ACCESS clause that does not match its context (RFC2578 §7.3). For example, a columnar object must not have a MAX-ACCESS value of "read-write" if any other columnar object in the table has a MAX-ACCESS value of "read-create".</i></p>
1210	<p>The conditionally GROUP clause <X> must be absent from the corresponding MANDATORY-GROUPS clause (RFC2580 §5.4.2).</p> <p><i>A conditionally group cannot be mandatory at the same time!</i></p>
1211	<p>OBJECT variation <X> must be included in a GROUP or MANDATORY-GROUPS reference (RFC2580 §5.4.2).</p> <p><i>The object reference <X> must be part of any object group specified as conditionally or mandatory for this compliance module.</i></p>

1212	<p>Only 'not-implemented' is applicable for the ACCESS clause of the notification type variation <X> (RFC2580 §6.5.2.3).</p> <p><i>If the notification has to be implemented, then the ACCESS clause should be removed.</i></p>
1220	<p>The CREATION-REQUIRES clause of variation <X> must only be present for conceptual row definitions (RFC2580 §6.5.2.4).</p> <p><i>The CREATION-REQUIRES clause must not be present unless the object named in the correspondent VARIATION clause is a conceptual row, i.e., has a syntax which resolves to a SEQUENCE containing columnar objects.</i></p>
1221	<p>Only columnar object type definitions with access read-create may be present in the CREATION REQUIRES clause of variation <X> (RFC2580 §6.5.2.4).</p> <p><i>Other objects and columns cannot be created and thus they cannot participate in a row creation.</i></p>
1500	<p>Undefined syntax(es): <X>[,...]</p> <p><i>The syntax (data type) <X> is not defined in the parsed MIB module and it is not imported from another MIB module. Use the Edit>Search MIB Repository function to search the MIB repository for object name <X> and add the corresponding IMPORT FROM clause for <X>.</i></p>
1501	<p>Undefined object(s): <X>[,...]</p> <p><i>The object name <X> is not defined in the parsed MIB module and it is not imported from another MIB module. Use the Edit>Search MIB Repository function to search the MIB repository for object name <X> and add the corresponding IMPORT FROM clause for <X>.</i></p>
1502	<p>The object <X> must be defined or imported (RFC2578 §3.2).</p> <p><i>The object <X> is not defined in the parsed MIB module and it is not imported from another MIB module. Use the Edit>Search MIB Repository function to search the MIB repository for object name <X> and add the corresponding IMPORT FROM clause for <X>.</i></p>
1600	<p>The object definition <X> references a <Y> definition, expected a reference to an OBJECT-TYPE conceptual row definition instead.</p> <p><i>The AUGMENTS clause, for example, requires that the referenced object definition is a conceptual table definition, i.e., has a syntax which resolves to a SEQUENCE containing columnar objects.</i></p>
1601	<p>The GROUP clause <X> references a <Y> definition, expected a reference to an OBJECT-GROUP or NOTIFICATION-GROUP instead (RFC2580 §5.4.2).</p> <p><i>The GROUP clause requires a reference to an object group definition.</i></p>
1602	<p>The object reference <X> points to a <Y> definition, expected a reference to an OBJECT-TYPE or NOTIFICATION-TYPE definition instead.</p> <p><i>The VARIATION clause, for example, requires a reference to an OBJECT-TYPE or a NOTIFICATION-TYPE definition.</i></p>
1700	<p>Object reference(s) with wrong type: <X> (expected <Y> but found <Z>) [...]</p>

	<i>The referenced to object <X> must be of type <Y> but it is of type <Z>.</i>
1800	The SEQUENCE clause of the table entry definition <X> does not match the order or number of objects registered for that table at entry <Y>. <i>The column references in the SEQUENCE definition of a table must be lexicographically ordered by their object-identifiers. The object name Y is the name of the first object reference in the SEQUENCE definition that does not match the order of columnar objects of that table.</i>
1810	The OBJECT-TYPE <X> has an invalid index definition (RFC2578 §7.7). <i>The OBJECT-TYPE <X> has an invalid INDEX clause, i.e., an empty clause.</i>
1811	The OBJECT-TYPE <X> has invalid index definition because <Y> may be negative (RFC2578 §7.7). <i>Index values have to be encoded as OID suffixes on the wire. Since OID sub-identifiers are 32-bit unsigned integer values, negative values cannot be encoded over the wire. See RFC2578 §7.7 for more details.</i>
1850	The OBJECT-TYPE <X> has invalid index definition, because <Y> is not a columnar object (RFC2578 §7.7). <i>The OBJECT-TYPE <X> has an invalid INDEX clause, because <Y> does not refer to a columnar OBJECT-TYPE definition. An OBJECT-TYPE is columnar object, if it is part of a table definition. See RFC2578 §7.7 for more details.</i>
1851	OBJECT-TYPE definition <X> is a scalar and therefore it must not have an INDEX clause (RFC2578 §7.7). <i>Scalar objects have a fixed instance identifier ("index") of '0', thus an INDEX clause must not be specified.</i>
2000	Duplicate object registration of <X> after <Y> for the object ID <Z> (RFC2578 §3.6). <i>Once an object identifier has been registered¹ it must not be reregistered.</i>
2010	Illegal object registration of <X> under <Y> for the object ID <Z>. <i>For example, it is not legal to register objects in the sub-tree of an OBJECT-TYPE registration.</i>
3000	The default value of OBJECT-TYPE <X> is out of range (RFC2578 §7.9). <i>The values specified in a DEFVAL clause have to be valid values for the corresponding data type syntax.</i>
3001	The size of the default value of OBJECT-TYPE <X> is out of range (RFC2578 §7.9). <i>The length of the specified octet string exceeds the SIZE constraints defined for the corresponding data type syntax.</i>
3002	The format of the default value of OBJECT-TYPE <X> does not match its syntax (RFC2578 §7.9). <i>The value <X> is not properly defined for the corresponding syntax.</i>

¹ An object registration is any object definition other than OBJECT-IDENTIFIER.

3003	A DEFVAL clause is not allowed for OBJECT-TYPE <X> which has a base syntax of Counter (Counter32 or Counter64) (RFC2578 §7.9).
4000	The syntax definition of the object <X> is not a valid refinement of its base syntax (RFC2578 §9). <i>A refinement must not extend the range of valid values for a data type.</i>
4010	The range restriction is invalid because ... <i>The lower bound (first value) of range restriction must be less or equal than the corresponding upper bound (second value). In addition, bounds for unsigned values cannot be negative.</i>
5000	The object definition <X> must be included in an OBJECT-GROUP or a NOTIFICATION-GROUP definition respectively (RFC2580 §3.1 and §4.1). <i>This requirement ensures that compliance statements for a MIB module can be written.</i>
5100	Object group <X> must not reference OBJECT-TYPE <Y> which has a MAX-ACCESS clause of not-accessible (RFC2580 §3.1). <i>Only accessible objects and notifications may be included in object groups.</i>