

VFD File Syntax

1. XML Container's DTD

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<!ELEMENT featureMX_model (meta?,vfd_star_diagram)>
```

```
<!ATTLIST featureMX_model name CDATA #REQUIRED>
```

```
    <!ELEMENT meta (data+)>
```

```
        <!ELEMENT data (#PCDATA)>
```

```
        <!ATTLIST data name CDATA ID #REQUIRED>
```

```
    <!ELEMENT vfd_star_diagram (#PCDATA)>
```

2. «vfd_star_diagram» field Context-Free Grammar

terminal NON-TERMINAL GrammarDefiningSyntax

FIELD	=	:mand (:prim)? ID COMMENT? ATTRIBUTES? CONSTRAINTS? FEATURES?
FEATURES	=	<gr> CARDINALITY (FEATURE)+ </gr>
FEATURE	=	(:dag ID) ((:mand :opt) (:prim)? ID COMMENT? ATTRIBUTES? CONSTRAINTS? FEATURES?)
ID	=	[a-zA-Z0-9]+
COMMENT	=	<!-- [^<!-->]* -->
CARDINALITY	=	[(NATURAL *), (NATURAL *)]
ATTRIBUTES	=	<att> ATTRIBUTE+ </att>
ATTRIBUTE	=	:bool ID COMMENT? BOOL_VALUES :int ID COMMENT? INT_VALUES :real ID COMMENT? REAL_VALUES
BOOL_VALUES	=	<val> :sel BOOL_EXPRESSION :desel BOOL_EXPRESSION </val>
INT_VALUES	=	<val> :sel INT_EXPRESSION :desel INT_EXPRESSION </val>
REAL_VALUES	=	<val> :sel REAL_EXPRESSION :desel REAL_EXPRESSION </val>

```

BOOL_EXPRESSION      =  BOOL_EXPRESSION && BOOL_EXPRESSION
                       |  BOOL_EXPRESSION || BOOL_EXPRESSION
                       |  BOOL_EXPRESSION ==> BOOL_EXPRESSION
                       |  BOOL_EXPRESSION <=> BOOL_EXPRESSION
                       |  !(BOOL_EXPRESSION)
                       |  (BOOL_EXPRESSION)
                       |  .true.
                       |  .false.
                       |  ID
                       |  ID :booleanof ID
                       |  INT_EXPRESSION == INT_EXPRESSION
                       |  INT_EXPRESSION == REAL_EXPRESSION
                       |  REAL_EXPRESSION == INT_EXPRESSION
                       |  REAL_EXPRESSION == REAL_EXPRESSION
                       |  INT_EXPRESSION >= INT_EXPRESSION
                       |  INT_EXPRESSION <= INT_EXPRESSION
                       |  REAL_EXPRESSION >= REAL_EXPRESSION
                       |  REAL_EXPRESSION <= REAL_EXPRESSION

INT_EXPRESSION       =  INT_EXPRESSION + INT_EXPRESSION
                       |  INT_EXPRESSION - INT_EXPRESSION
                       |  INT_EXPRESSION * INT_EXPRESSION
                       |  INT_EXPRESSION / INT_EXPRESSION
                       |  -(INT_EXPRESSION)
                       |  (INT_EXPRESSION)
                       |  max(INT_EXPRESSION,INT_EXPRESSION)
                       |  min(INT_EXPRESSION,INT_EXPRESSION)
                       |  abs(INT_EXPRESSION)
                       |  ID :intof ID
                       |  ? BOOL_EXPRESSION : INT_EXPRESSION , INT_EXPRESSION ?
                       |  INTEGER

```

REAL_EXPRESSION = REAL_EXPRESSION + REAL_EXPRESSION
 | REAL_EXPRESSION - REAL_EXPRESSION
 | REAL_EXPRESSION * REAL_EXPRESSION
 | (REAL_EXPRESSION)
 | REAL
 | ID :realof ID
 | ? BOOL_EXPRESSION : REAL_EXPRESSION , REAL_EXPRESSION ?

CONSTRAINTS = <const> (:sel (CONSTRAINT)+
 | :desel (CONSTRAINT)+
 | :sel (CONSTRAINT)+ :desel (CONSTRAINT)+ </const>

CONSTRAINT = [COMMENT? (BOOLEAN_CONSTRAINT | EXTENDED_CONSTRAINT)]

BOOLEAN_CONSTRAINT = :cnf BOOLEAN_CONSTRAINT_EXPRESSION

BOOLEAN_CONSTRAINT_EXPRESSION = BOOLEAN_CONSTRAINT_EXPRESSION && BOOLEAN_CONSTRAINT_EXPRESSION
 | BOOLEAN_CONSTRAINT_EXPRESSION || BOOLEAN_CONSTRAINT_EXPRESSION
 | BOOLEAN_CONSTRAINT_EXPRESSION ==> BOOLEAN_CONSTRAINT_EXPRESSION
 | BOOLEAN_CONSTRAINT_EXPRESSION <=> BOOLEAN_CONSTRAINT_EXPRESSION
 | !(BOOLEAN_CONSTRAINT_EXPRESSION)
 | (BOOLEAN_CONSTRAINT_EXPRESSION)
 | .true.
 | .false.
 | ID

EXTENDED_CONSTRAINT = :ext BOOL_EXPRESSION

NATURAL = 0 | [1-9][0-9]*

INTEGER = 0 | (-)?[1-9][0-9]*

REAL = INTEGER.([0-9]*[1-9])?