

Format of ACD/ChemSketch Document (*.SK2).

Contents

Format of ACD/ChemSketch Document (*.SK2)	1
Contents	2
Fundamental Types Format.....	4
Numeric Data Types.	4
Logic Data Type	4
String Data	4
Additional Data Types	4
TColorRef	4
TPoint	4
TRect.....	4
TRPoint.....	4
TRRect.....	4
SmallSizeRec	5
General Description of ACD/ChemSketch Document	6
ACD_ChemSketch_2_0_DocumentFormat	6
DocumentDescriptionFormat	6
EnvironmentDescriptionFormat	6
PageDescriptionFormat.....	6
PageInfoFormat.....	7
PictureFormat.....	7
StylesListFormat	7
ObjectsListFormat.....	7
Format of Object Style Description	8
BaseGraphStyleFormat	8
PenStyleFormat.....	8
BrushStyleFormat	9
FontStyleFormat.....	9
ArrowStyleFormat.....	10
ParagraphStyleFormat.....	10
AtomStyleFormat.....	11
BondStyleFormat	11
Format of Graphic Object Description.....	11
GeomShapeFormat.....	12
GeomNodeFormat.....	12
LineFormat, PolylineFormat	12
RectangleFormat, EllipseFormat, PolygonFormat.	13
RoundedRectangleFormat.....	13
ArcFormat	13
BracketsFormat	14
CalloutFormat	14
GroupFormat.....	14
TextFormat.....	14
TextDataFormat	14
ParagraphDataFormat	15
TextLineDataFormat	15
TextBlockDataFormat.....	15
OleObjectFormat.....	15
ExternObjectFormat.....	15
MoleculeFormat	16
AtomDataFormat	16
BondDataFormat.....	17
AromaRingDataFormat.....	18
Table 1. Periodic Table of Elements.....	19
Table 2. Bonds types.....	19
Examples.....	20
Example of AtomStyleDataFormat	20

Example of BondStyleDataFormat	20
--------------------------------------	----

Fundamental Types Format

Before proceeding to the description of the ACD/Chem Sketch document format, it is necessary to determine the list of fundamental types used for data representation and the formats of those types.

Numeric Data Types.

Type name	Length in bytes	Comments
Byte	1	
Integer	2	
Word	2	
LongInt	4	MaxLongInt = 2 147 483 647
Single	4	
ShortInt	1	signed byte

Logic Data Type

Type name	Length in bytes	Comments
Boolean	1	False = 0; True = 1

OR {}: A, B, ..., Z - bitwise combination of A, B, ..., Z

String Data

Type name	Length in bytes	Comments
Char	1	
PChar	2 + StrLen	the first 2 bytes determines the number of characters in string
String	1 + StrLen	the first byte determines the number of characters in string

Additional Data Types

TColorRef

Red	Byte	Red component of Color
Green	Byte	Green component of Color
Blue	Byte	Blue component of Color
Reserved	Byte	should be 0

TPoint

X	Integer	
Y	Integer	

TRect

left	Integer	
top	Integer	
right	Integer	
bottom	Integer	

TRPoint

X	Single	
Y	Single	

TRRect

left	Single	
top	Single	
right	Single	
bottom	Single	

SmallSizeRec

SmallSizeRec format is used for recording the value of type LongInt when the value is most probably in the range from 1 to 255.

BValue	Byte	
if bValue = 0 then		
IValue	LongInt	
endif		

General Description of ACD/ChemSketch Document

ACD_ChemSketch_2_0_DocumentFormat

ACD_File_Signature	PChar	'(C) ACD \$\$'
ACD_ChemSketch_2_0_Signature	PChar	'.RPT.(V 2.0)'
DocumentDescriptionSize	LongInt	size of the document description block
DocumentDescription	<i>DocumentDescriptionFormat</i>	
EnvironmentDescriptionSize	LongInt	size of the editing environment parameters block
Environment	<i>EnvironmentDescriptionFormat</i>	editing environment block

Comments:

- If the **ACD_File_Signature** has '\$' in its last position, entire Document will not contain **VerError_1**.
- If the **ACD_File_Signature** has '\$' in its last but one position, entire Document will not contain **VerError_2**.

DocumentDescriptionFormat

Version	Byte	can equal 0 or 1
PageCount	Integer	the number of pages in the document
PageDataSize[1]	LongInt	size of the one page description
PageDescription[1]	<i>PageDescriptionFormat</i>	description of the 1 st page
.....	...	...
PageDataSize[PageCount]	LongInt	size of the last page description
PageDescription[PageCount]	<i>PageDescriptionFormat</i>	description of the last page

if (**Version** and 1 <> 0) and not **VerError_1** then

InternalBlock		skip when reading
----------------------	--	-------------------

endif

EnvironmentDescriptionFormat

ActivePage	Integer	
Zoom	Integer	
ViewOrigin	TPoint	

PageDescriptionFormat

Version	Byte	OR {}: 1 - Page Color 2 - Internal Block
PageNameFlag	Boolean	parameter determining the presence of the next data field PageName

if PageNameFlag then

PageName	PChar	
-----------------	-------	--

endif

PageInfoSize	LongInt	size of the PageInfo structure
PageInfo	<i>PageInfoFormat</i>	structure describing the metric characteristics of the page
PictureSize	LongInt	size of the picture description
Picture	<i>PictureFormat</i>	Picture description. Set Version variable of Picture to zero (but not 1).

if (**Version** and 1 <>0) and not **VerError_1** then

PageColor	TColorRef	background page color
------------------	-----------	-----------------------

endif

if (**Version** and 2 <>0) and not **VerError_1** then

InternalBlock		skip when reading
----------------------	--	-------------------

endif

PageInfoFormat

Reserved	Integer	should be -1
Reserved	Integer	should be -1
PageSize	TPoint	page size in logic units 1 log. unit = 1/300 inches
PrnPageSize	TPoint	print page size in log. units
PrnPages	TRPoint	page size expressed in the number of print pages along the length and width
Border	TRect	indentation from the page margins (log. units)
Orientation	Integer	1 - Portrait 2 - Landscape

PictureFormat

Version	Byte	OR {}: 1 - Internal Block 1 2 - Internal Block 2
DPI	Integer	the number of logical units in 1 inch; DPI should be = 300
StylesCount	Integer	the number of style descriptions in the StylesList list
ObjCount	Integer	the number of graphic objects in the ObjectsList list
StylesList	<i>StylesListFormat</i>	list of styles
ObjectsList	<i>ObjectsListFormat</i>	list of objects

if (**Version** and 1 <>0) and not **VerError_1** then

InternalBlock1		skip when reading
-----------------------	--	-------------------

endif

if (**Version** and 2 <>0) and not **VerError_1** and not **VerError_2** then

InternalBlock2		skip when reading
-----------------------	--	-------------------

endif

StylesListFormat

The style indexing in the list starts from 0. The number N of list units equals to StylesCount.

Style[0]	<i>BaseGraphStyleFormat</i>	
...	...	
Style[N-1]	<i>BaseGraphStyleFormat</i>	

ObjectsListFormat

The object indexing in the list begins from 0. The number (N) of list units equals ObjectsCount.

ObjectID[0]	Byte	identifier of object0
ObjectDescriptionSize[0]	SmallSizeRec	size of object0 description
ObjectDescription[0]	<i>depends on ObjectID[0]</i>	Description of object0
...	...	...
ObjectID[N-1]	Byte	identifier of object(N-1)
ObjectDescriptionSize[N-1]	SmallSizeRec	size of object(N-1) description
ObjectDescription[N-1]	<i>depends on ObjectID[N-1]</i>	description of object(N-1)

ObjectID can take the following values:

Value	Object	Format of ObjectDescription
1	Line	<i>LineFormat</i>
2	Polyline	<i>PolylineFormat</i>
3	Rectangle	<i>RectangleFormat</i>
4	Rounded Rectangle	<i>RoundedRectangleFormat</i>
5	Ellipse	<i>EllipseFormat</i>
6	Arc	<i>ArcFormat</i>
7	Polygon	<i>PolygonFormat</i>
8	Text	<i>TextFormat</i>
9	Brackets	<i>BracketsFormat</i>
10	Callout	<i>CalloutFormat</i>
32	Group of objects	<i>GroupFormat</i>
200 - 250	Extern objects	<i>ExternObjectFormat</i>
253	Molecule	<i>MoleculeFormat</i>
255	OLE object	<i>OleObjectFormat</i>

Format of Object Style Description

BaseGraphStyleFormat

StyleID	Byte	identifier of style
StyleDescriptionSize	SmallSizeRec	size of style description
StyleDescription	<i>depends on StyleID</i>	Description of style

StyleID can take the following values:

Value	Style	Format of StyleDescription
1	PenStyle	<i>PenStyleFormat</i>
2	BrushStyle	<i>BrushStyleFormat</i>
3	FontStyle	<i>FontStyleFormat</i>
4	ArrowStyle	<i>ArrowStyleFormat</i>
5	ParagraphStyle	<i>ParagraphStyleFormat</i>
129	AtomStyle	<i>AtomStyleFormat</i>
130	BondStyle	<i>BondStyleFormat</i>
131	IntersectionStyle	<i>IntersectionStyleFormat</i>

PenStyleFormat

Version	Byte	should be 0
PenWidth	Integer	Pen width in log. units
PenStyle	Byte	Pen style
PenColor	TColorRef	color

PenStyle can take the following values:

0	NULL	
1	SOLID	
2	DASH	
3	DOT	
4	DASHDOT	
5	DASHDOTDOT	

BrushStyleFormat

Version	Byte	should be 0
BrushStyle	Byte	
if BrushStyle <> NULL then		
BrushColor	TColorRef	
if BrushStyle = PATTERN then		
BrushPattern	Byte	
else if BrushStyle = SHADE then		
BrushShade	Byte	
BrushShadeLevel	Integer	parameter determines the shading gradient level -128 (to black) 128 (to white)
endif		
endif		

BrushStyle can take the following values:

0	NULL	
1	SOLID	
2	PATTERN	
3	SHADE	

BrushPattern can take the following values:

0	BDIAGONAL	
1	CROSS	
2	DIAGCROSS	
3	FDIAGONAL	
4	HORIZONTAL	
5	VERTICAL	

BrushShade can take the following values:

0	LEFT2RIGHT	
1	RIGHT2LEFT	
2	TOP2BOTTOM	
3	BOTTOM2TOP	
4	DIAGLT2RB	
5	DIAGRT2LB	
6	DIAGLB2RT	
7	DIAGRB2LT	

FontStyleFormat

Version	Byte	should be 0
Size	Integer	font size in 1/2 pt
FontStyle	Word	
Color	TColorRef	
Name	String	font name
DPI	Integer	font DPI

FontStyle parameter can take the following values:

\$0000	NORMAL
\$0001	BOLD
\$0004	ITALIC
\$0010	UNDERLINE
\$0040	STRIKEOUT
\$0100	SUPERScript
\$0200	SUBSCRIPT

ArrowStyleFormat

Version	Byte	should be 0
DPI	Integer	DPI for recalculating the Double and Size parameters into logic coordinates
Heads	array[0..3] of Byte	arrowhead styles for each of the ends of two arrow lines.
Double	Integer	distance (in 1/72") between two lines of two-line arrow. If Double = 0, the arrow consists of one line.
Size	TPoint	arrow size in 1/72"

Elements of the Heads array can take the following values:

\$00	NULL	
\$01	UP	
\$02	UPDOWN	
\$03	DOWN	
\$04	DOUBLE	
\$10	FILLED	

ParagraphStyleFormat

Version	Byte	should be 0
Justification	Byte	
IndentLeft	Integer	Indentation of paragraph from the left margin of text box
IndentRight	Integer	Indentation of paragraph from the right margin of text box
IndentFirstLine	Integer	First line indentation
ParSpacingBefore	Integer	
ParSpacingAfter	Integer	
LineSpacingMode	Byte	
LineSpacingAt	Integer	
TabStep	Integer	

Justification can take one of the following values:

1	LEFT
2	RIGHT
3	CENTER
4	JUSTIFY

LineSpacingMode can take one of the following values:

1	Single
2	1_5Lines
3	Double
4	AtLeast
5	Exactly
6	Percent

AtomStyleFormat

Version	Byte	should be 0
New Properties	Byte	should be 0. (not 1)
Base Font	<i>BaseGraphStyleFormat</i>	
Hydrogen Font	<i>BaseGraphStyleFormat</i>	
Index Font	<i>BaseGraphStyleFormat</i>	
Valence Font	<i>BaseGraphStyleFormat</i>	
Charge Font	<i>BaseGraphStyleFormat</i>	
Isotope Font	<i>BaseGraphStyleFormat</i>	
Numbering Font	<i>BaseGraphStyleFormat</i>	
Index Offset (vertical)	Integer	(in log. units)
Valence Offset (vertical)	Integer	(in log. units)
Isotope Offset (vertical)	Integer	(in log. units)
Charge Offset (vertical)	Integer	(in log. units)
Numbering Offset (horz., vert.)	TPoint	(in log. units)
Valence After Hydrogen	Boolean	
Charge After Hydrogen	Boolean	
Show Carbons	Boolean	
Show Terminal Carbons	Boolean	
Show Hydrogens	Boolean	
Show Valence	Boolean	
Show Zero Charge	Boolean	
Show Isotope	Boolean	
Show Numbering	Boolean	
Cross Out Invalid Atom	Boolean	

BondStyleFormat

Version	Byte	should be 0
New Properties	Byte	should be 0 (not 1)
Single Pen	<i>BaseGraphStyleFormat</i>	
Double Pen	<i>BaseGraphStyleFormat</i>	
Triple Pen	<i>BaseGraphStyleFormat</i>	
Up Stereo Pen	<i>BaseGraphStyleFormat</i>	
Down Stereo Pen	<i>BaseGraphStyleFormat</i>	
Coordinating Pen	<i>BaseGraphStyleFormat</i>	
Undefined Pen	<i>BaseGraphStyleFormat</i>	
Double Between	Integer	(in log. units)
Double Shift	Integer	(in log. units)
Triple Between	Integer	(in log. units)
Triple Shift	Integer	(in log. units)
Up Stereo Width	Integer	(in log. units)
Down Stereo Width	Integer	(in log. units)
Down Stereo Step	Integer	(in log. units)
Coord.Head Size (length, width)	TPoint	(in log. units)
Undefined Width	Integer	(in log. units)
Undefined Step	Integer	(in log. units)

IntersectionStyleFormat

Version	Byte	should be 0
New Properties	Byte	should be 0 (not 1)
Intersections Enable	Boolean	
White Space	Integer	

Format of Graphic Object Description

The basic element of graphic objects is the object **GeomShape** which describes the graphic shape consisting of the sequence of points (dots) which are connected by line or curve segments.

The curve segments are described by the following cubic equations:

$$X(t) = A_x * t^3 + B_x * t^2 + C_x * t + D_x,$$

$$Y(t) = A_y * t^3 + B_y * t^2 + C_y * t + D_y,$$

where t varies from 0 to 1

GeomShapeFormat

NodesCount	Integer	The number of nodes
IsShapeClosed	Boolean	determines whether the shape is closed, i.e. whether Node[0] and Node[N-1] are connected.
Node[0]	<i>GeomNodeFormat</i>	
...	...	array of shape node descriptions
Node[N-1]	<i>GeomNodeFormat</i>	N = NodesCount

GeomNodeFormat

NodeType	Byte	Node style
Pos	TRPoint	Node coordinates

if **NodeType** and \$80 <> 0 then { curved connection with the previous node }

PrevV	TRPoint	Parameters for calculating the curved connection with the previous node
--------------	---------	---

endif

if **NodeType** and \$02 <> 0 then { curved connection with the next node }

NextV	TRPoint	Parameters for calculating the curved connection with the subsequent node.
--------------	---------	--

endif

The algorithm for calculating the parameters of the curved connection equations with the use of **PrevV** and **NextV** parameters:

$$A.X := 3*(Node.NextV.X + NextNode.PrevV.X) - 2*(NextNode.Pos.X - Node.Pos.X);$$

$$B.X := 3*(NextNode.Pos.X - Node.Pos.X) - 3*(2*Node.NextV.X + NextNode.PrevV.X);$$

$$C.X := Node.NextV.X*3;$$

$$D.X := Node.Pos.X;$$

$$A.Y := 3*(Node.NextV.Y + NextNode.PrevV.Y) - 2*(NextNode.Pos.Y - Node.Pos.Y);$$

$$B.Y := 3*(NextNode.Pos.Y - Node.Pos.Y) - 3*(2*Node.NextV.Y + NextNode.PrevV.Y);$$

$$C.Y := Node.NextV.Y*3;$$

$$D.Y := Node.Pos.Y;$$

where

Node	the current node
NextNode	the node following the (Node). If the shape has a closed contour, Node[N-1] is followed by Node[0]

LineFormat, PolylineFormat

Version	Byte	can equal 0 or 1
PenStyleIndex	Integer	index of Pen style in the list of graphical styles of the picture (Picture).
BrushStyleIndex	Integer	should be -1
ArrowStyleIndex	Integer	index of Arrow style in the list of graphical styles of the picture.

GeomShape	<i>GeomShapeFormat</i>	description of the picture shape
if Version and 1 <> 0 then		
Reserved	Integer	should be -1
endif		

RectangleFormat, EllipseFormat, PolygonFormat.

Version	Byte	can equal 0 or 1
PenStyleIndex	Integer	Index of Pen style in the list of graphical styles of the picture (Picture).
BrushStyleIndex	Integer	index of Brush style in the list of graphical styles of the picture.
ArrowStyleIndex	Integer	should be -1
GeomShape	<i>GeomShapeFormat</i>	description of the picture shape
if Version and 1 <> 0 then		
ShadowBrushStyleIndex	Integer	Index of Brush style in the list of graphical styles of the picture. This style is used for picture shadow.
if ShadowBrushStyleIndex >=0 then		
ShadowOffset	TPoint	Offset of the shadow relative to the object
endif		
endif		

RoundedRectangleFormat

Version	Byte	can equal 0 or 1
PenStyleIndex	Integer	Index of Pen style in the list of graphical styles of the picture (Picture).
BrushStyleIndex	Integer	Index of Brush style in the list of graphical styles of the picture.
ArrowStyleIndex	Integer	should be -1
GeomShape	<i>GeomShapeFormat</i>	description of the picture shape
if Version and 1 <> 0 then		
ShadowBrushStyleIndex	Integer	Index of Brush style in the list of graphical styles of the picture. This style is used for picture shadow.
if ShadowBrushStyleIndex >=0 then		
ShadowOffset	TPoint	Offset of the shadow relative to the object
endif		
endif		
Rmult	Single	Parameter determining the ratio of radius of curvature to minimal side of the rectangle.

ArcFormat

Version	Byte	can equal 0 or 1
PenStyleIndex	Integer	Index of Pen style in the list of graphical styles of the picture (Picture).
BrushStyleIndex	Integer	should be -1
ArrowStyleIndex	Integer	Index of Arrow style in the list of graphical styles of the picture.
GeomShape	<i>GeomShapeFormat</i>	description of the picture shape
if Version and 1 <> 0 then		
Reserved	Integer	should be -1
endif		
EllipseShape	<i>GeomShapeFormat</i>	Shape making up an ellipse.

BracketsFormat

Version	Byte	should be 0
PenStyleIndex	Integer	
BrushStyleIndex	Integer	
ShadowBrushStyleIndex	Integer	
if ShadowBrushStyleIndex <> -1 then		
ShadowOffset	TPoint	
endif		
FrameStyle	Byte	
TemplateShape	<i>GeomShapeFormat</i>	
Width	Single	
IsSingle	Boolean	

CalloutFormat

Version	Byte	should be 0
PenStyleIndex	Integer	
BrushStyleIndex	Integer	
ShadowBrushStyleIndex	Integer	
if ShadowBrushStyleIndex <> -1 then		
ShadowOffset	TPoint	
endif		
FrameStyle	Byte	
TemplateShape	<i>GeomShapeFormat</i>	

GroupFormat

Version	Byte	can equal 0 or 1
ObjectsCount	Integer	The number of objects in a group
ObjectsList	<i>ObjectsListFormat</i>	Objects list
if Version and 1 <> 0 then		
InternalBlock		skip when reading
endif		

TextFormat

Version	Byte	should be 0
StyleCount	Integer	
StyleIndex[0]	Integer	
...		
StyleIndex[N-1]	Integer	N = StyleCount
IsArtisticText	Boolean	
if IsArtisticText then		
ScaleX	Single	
ScaleY	Single	
endif		
Rect	TRRect	
Rcent	TRPoint	
Ang1	TRPoint	
Ang2	TRPoint	
Rang1	TRPoint	
Rang2	TRPoint	
TextData	<i>TextDataFormat</i>	

TextDataFormat

TextDataFlag	array[0..2] of Char	should be 'AFT'
ParagraphCount	Longint	

TextWidth	Integer	
WordWrap	Integer	if 1 then TRUE else False
Paragraph[0]	<i>ParagraphDataFormat</i>	
...		
Paragraph[N-1]	<i>ParagraphDataFormat</i>	N = ParagraphCount

ParagraphDataFormat

ParagraphFlag	Integer	should be 1
LinesCount	Longint	
ParagraphStyle	Integer	
Line[0]	<i>TextLineDataFormat</i>	
...		
Line[N-1]	<i>TextLineDataFormat</i>	

TextLineDataFormat

LineFlag	Integer	should be 2
TextBlockCount	Longint	
Spaces	LongInt	
SpaceSize	Integer	
TextBlock[0]	<i>TextBlockDataFormat</i>	
...		
TextBlock[N-1]	<i>TextBlockDataFormat</i>	N = TextBlockCount

TextBlockDataFormat

BlockFlag	Integer	should be 3
BlockLen	Longint	
FontStyleIndex	Integer	
BlockString	array[0.. BlockLen-1] of Char	

OleObjectFormat

Version	Byte	should be 0
Rect	TRRect	
NativeDataSize	Longint	
NativeData	array[0.. NativeDataSize-1] of Byte	

ExternObjectFormat

Version	Byte	should be 0
ExternObjectFlag	Byte	should be \$1F
VersionInfo	Byte	any value is possible
StylesCount	Integer	
StyleIndex[0]	Integer	
...		
StyleIndex[N-1]	Integer	N = StylesCount
BasePenIndex	Integer	
BaseBrushIndex	Integer	
BaseFontIndex	Integer	
Rect	TRRect	
NameFlag	Boolean	

if NameFlag then

ObjectName	PChar	
-------------------	-------	--

endif

ImageSize	Longint	
ObjectImageData	array[0..ImageSize-1] of Byte	

NativeDataSize	Longint	
NativeData	array[0..NativeDataSize-1] of Byte	

MoleculeFormat

Version	Byte	should be 0
New Properties	Byte	should be 0 (not 1)
Atoms Count	Integer	
AtomRecSize[0]	Integer	
AtomRec[0]	<i>AtomDataFormat</i>	
.....		
AtomRecSize[AtomsCount - 1]	Integer	
AtomRec[AtomsCount - 1]	<i>AtomDataFormat</i>	
Bonds Count	Integer	
BondRecSize[0]	Integer	
BondRec[0]	<i>BondDataFormat</i>	
.....		
BondRecSize[BondsCount - 1]	Integer	
BondRec[BondsCount - 1]	<i>BondDataFormat</i>	
Number of Radical Atom	Integer	<0 for independent Structure; >=0 for SubStructure - the number of an atom which is the binding point of the given structure to the main structure. For the top-level structure it should be less than 0 (e.g. -1)
AromaRings Count	Integer	
AromaRingRecSize[0]	Integer	
AromaRingRec[0]	<i>AromaRingDataFormat</i>	
.....		
AromaRingRecSize[AromaRingsCount - 1]	Integer	
AromaRingRec[AromaRingsCount - 1]	<i>AromaRingDataFormat</i>	

AtomDataFormat

AutoScale	Single	all of the atom style properties, such as Font Sizes, Offsets, should be multiplied by this variable to get the true value; default =1. Should not be equal to zero.
NewProperties	Byte	OR {}: 1 - Aroma Charge 2 - Atom Label Attach Pos stored below (sets only if length of Atom Label field <>0 (see below))
Style	Integer	Index in StylesList of Picture (from 0)
ElementNum	Integer	-1 if there is a string instead of an element; -2 if there is no atom ("bent" bond); >=0 - element number in Periodic Table (see Table 1 below)
Valence	ShortInt	>=0
Charge	ShortInt	
Isotope	Integer	
NumStr	PChar	
X	Single	(log. Units)

Y	Single	(log. Units)
Z¹	Single	(log. Units)
HydrogenPosition	Byte	0 - Auto 1 - Right 2 - Left 3 - Bottom 4 - Top
PersonalShowState	Byte	0 - Show 1 - Hide 2 - Undefined
FixStates	Byte	OR {}: 1 - Valence fixed 2 - Charge fixed 4 - Charge fixed but "soft" <u>Note:</u> 2 and 4 cannot be used simultaneously
ShowStates	Byte	OR {}: 1 - Show Hydrogens 2 - Show Valence 4 - Show Zero Charge 8 - Show Isotope 16 - Show Numbering 32 - Cross Out Invalid Atom
SubStructureExist	Boolean	

if SubStructureExist then

SubStructure	<i>MoleculeFormat</i>	
---------------------	-----------------------	--

endif

AtomLabel	PChar	if length of Atom Label <>0, it uses instead of name of element in Periodic Table
------------------	-------	---

if AtomLabel <> 0 then

Reserved	Integer	should be -1
AtomLabelReversed	Boolean	

endif

if NewProperties and \$02 <> 0 then

AtomLabelAttachPos	Integer	number of a character to which atom bonds should be pointed
---------------------------	---------	---

endif

Comments:

- If the **AromaCharge** bit is defined in **NewProperties**, the charge is not displayed at the atom but it is added to the charge in the center of the aromatic ring to which the atom belongs.
- When mapping the atom, **ShowStates** variables belonging to the atom are used. The similar variables in the atom style are used for initialize this parameter at the draw stage.
- Personal Show State in a position On or Off has a priority over Show Carbons, Show Terminal Carbons when displaying hydrogen. It can be also used to hide the non-hydrogen atoms.

BondDataFormat

AutoScale	Single	all the bond style properties, such as PenWidths, Shifts, Steps, Between should be multiplied by this variable to get the true value; default = 1. Should not be equal to zero.
NewProperties	Byte	OR {}: 1 - BondLabel stored below

¹ Make sure the **Z** variables of Atoms are particular values (set as zero if this kind of information is inaccessible). If an atom in a molecule has a bad Z-coordinate, trying to 3D-rotate this molecule will fail.

		2 - IntersectOrder stored below
Style	Integer	Index in StylesList of Picture (from 0)
IndexOffFirstAtom	Integer	Index in AtomsList of Structure (from 0)
IndexOffSecondAtom	Integer	Index in AtomsList of Structure (from 0)
BondType	Byte	TBondType (see Table 2 below)
Reserved	TPoint	should be (0, 0)
Reserved	TPoint	should be (0, 0)
StereoView	Byte	see Table 2 below
SymmetryOfDoubleBond	Byte	0 - Auto 1 - Symmetric 2 - Asymmetric 1 3 - Asymmetric 2
Reserved	Byte	should be (1)
RingState	Byte	OR {}: 1 - Ring bond (default = 0) 2 - Bond belongs to aromatic ring (displayed as an ellipse)

if **NewProperties** and \$01 <> 0 then

BondLabel	String	
------------------	--------	--

endif

if **NewProperties** and \$02 <> 0 then

IntersectOrder	LongInt	default = MaxLongInt; (see comments below)
-----------------------	---------	--

endif

Comments:

- On default all the bonds are intersected in order of their drawing, i.e. in the Picture the first bond of the first structure will be background. However the order can be changed if the non-default values are assigned to **IntersectOrder**. If this is a case, the bond having the greater value will be foreground.

AromaRingDataFormat

NewProperties	Byte	OR {}: 1 - Ring Charge stored below
EllipseStored	Boolean	Ellipse stored below

if EllipseStored then

Ellipse	<i>EllipseFormat</i>	this field is present only if the field 'Ellipse Stored' = true
----------------	----------------------	---

endif

BondsCount	Integer	the number of bonds in a ring
Bonds[0]	Integer	numbers of bonds in a ring
.....		
Bonds[BondsCount - 1]	Integer	

if **NewProperties** and \$01 <> 0 then

RingCharge	ShortInt	charge in the center of a ring
-------------------	----------	--------------------------------

endif

Document may contains objects and styles that are not mentioned in this description. When reading such objects and styles, you should skip them (to be able to proceed in a file, use the sizes that stand before such objects and styles).

Table 1. Periodic Table of Elements.

0	'H'	21	'Ti'	42	'Tc'	63	'Au'	84	'Er'
1	'He'	22	'V'	43	'Ru'	64	'Hg'	85	'Tm'
2	'Li'	23	'Cr'	44	'Rh'	65	'Tl'	86	'Yb'
3	'Be'	24	'Mn'	45	'Pd'	66	'Pb'	87	'Lu'
4	'B'	25	'Fe'	46	'Ag'	67	'Bi'	88	'Ac'
5	'C'	26	'Co'	47	'Cd'	68	'Po'	89	'Th'
6	'N'	27	'Ni'	48	'In'	69	'At'	90	'Pa'
7	'O'	28	'Cu'	49	'Sn'	70	'Rn'	91	'U'
8	'F'	29	'Zn'	50	'Sb'	71	'Fr'	92	'Np'
9	'Ne'	30	'Ga'	51	'Te'	72	'Ra'	93	'Pu'
10	'Na'	31	'Ge'	52	'I'	73	'La'	94	'Am'
11	'Mg'	32	'As'	53	'Xe'	74	'Ce'	95	'Cm'
12	'Al'	33	'Se'	54	'Cs'	75	'Pr'	96	'Bk'
13	'Si'	34	'Br'	55	'Ba'	76	'Nd'	97	'Cf'
14	'P'	35	'Kr'	56	'Hf'	77	'Pm'	98	'Es'
15	'S'	36	'Rb'	57	'Ta'	78	'Sm'	99	'Fm'
16	'Cl'	37	'Sr'	58	'W'	79	'Eu'	100	'Md'
17	'Ar'	38	'Y'	59	'Re'	80	'Gd'	101	'No'
18	'K'	39	'Zr'	60	'Os'	81	'Tb'	102	'Lr'
19	'Ca'	40	'Nb'	61	'Ir'	82	'Dy'	103	'D'
20	'Sc'	41	'Mo'	62	'Pt'	83	'Ho'	104	'T'

Table 2. Bonds types.

	Bond	Bond Type	Stereo View
	—	1	0 or 1
	≡	2	0 or 1
	≡	3	0 or 1
*	◄, ►	4	0 or 1
	▬	4	2
*	⋯⋯⋯, ⋯⋯⋯	5	0 or 1
		5	2
	-----	5	3
*	←-----, -----→	6	0 or 1
	~~~~~	7	0 or 1

* The direction of these bonds depends on which atom in the bond is the first and which is the second.

## Examples

### *Example of AtomStyleDataFormat*

Version	0
New Properties	0 (not 1)
Base Font	(20, GFS_Normal, (0, 0, 0, 0), 'Arial')
Hydrogen Font	(20, GFS_Normal, (0, 0, 0, 0), 'Arial')
Index Font	(16, GFS_Normal, (0, 0, 0, 0), 'Arial')
Valence Font	(16, GFS_Normal, (0, 0, 0, 0), 'Arial')
Charge Font	(16, GFS_Normal, (0, 0, 0, 0), 'Arial')
Isotope Font	(16, GFS_Normal, (0, 0, 0, 0), 'Arial')
Numbering Font	(16, GFS_Normal, (0, 0, 0, 0), 'Arial')
Index Offset (vertical)	10
Valence Offset (vertical)	-30
Isotope Offset (vertical)	-30
Charge Offset (vertical)	-30
Numbering Offset (horz., vert.)	(0, 30)
Valence After Hydrogen	True
Charge After Hydrogen	True
Show Carbons	False
Show Terminal Carbons	True
Show Hydrogens	True
Show Valence	False
Show Zero Charge	False
Show Isotope	False
Show Numbering	False
Cross Out Invalid Atom	True

### *Example of BondStyleDataFormat*

Version	0
New Properties	0
Single Pen	(3, GPS_Solid, (0, 0, 0, 0))
Double Pen	(3, GPS_Solid, (0, 0, 0, 0))
Triple Pen	(3, GPS_Solid, (0, 0, 0, 0))
Up Stereo Pen	(3, GPS_Solid, (0, 0, 0, 0))
Down Stereo Pen	(3, GPS_Solid, (0, 0, 0, 0))
Coordinating Pen	(3, GPS_Solid, (0, 0, 0, 0))
Undefined Pen	(3, GPS_Solid, (0, 0, 0, 0))
Double Between	16
Double Shift	20
Triple Between	16
Triple Shift	20
Up Stereo Width	24
Down Stereo Width	24
Down Stereo Step	7
Coord.Head Size (length, width)	(16, 6)
Undefined Width	24
Undefined Step	12