Biacore™ S200 Software

XML export format description

Contents

1 Introduction

This document describes the format of the XML-data exported from Biacore™ S200 Software version 1.1.3 and newer. The exported data can be generated manually from the File: Export menu.

The XML files are generated using "iso8859-1" encoding.

All numerical values are exported with full precision and with dot (.) as decimal separator.

All date-time values are exported in the format "YYYY-MM-DD HH:MM:SS". For some values the time part is omitted.

All tables are exported as an XML structure that defines the row and column headers. The table itself is exported as a CDATA section containing tab-separated text with each table row on a separate line.

Description	The generic table format used in the export.

Format	```<Table Name="$(Table name)">```
	 ```<Column$(Column index)>$(Column name)</Column$(Column index)>```
	 ```<Row$(Row index)>$(Row name)</Row$(Row index)>```
	 ```<Data><![CDATA[$(Table data)]]>```
	 ```</Data>```
	```</Table>```
	
	where
	
	$(Table name) – the name of the table.
	
	$(Column index) – an index value indicating the number of the data column. First index is 1. All column# nodes should be read until the sought node doesn't exist.
	
	$(Column name) – the name of the column.
	
	$(Row index) – an index value indicating the number of the data row. First index is 1. All row# nodes should be read until the sought node doesn't exist. *Note: Is omitted in tables that have no specific name of the rows.*
	
	$(Row name) – the name of the row. *Note: Is omitted in tables that have no specific name of the rows.*
	
	$(Table data) – is the list of table data values. Each table row is on a separate line with the column values as tab separated data.

Example	```<Table Name="Report">```
	```<Column1>ka (1/Ms)</Column1>```
	```<Column2>kd (1/s)</Column2>```
	```<Column3>KD (M)</Column3>```
	```<Row1>Cycle: 7 2 nM</Row1>```
	```<Row2>Cycle: 9 8 nM</Row2>```
	```<Row3>Cycle: 11 32 nM</Row3>```
	```<Data><![CDATA[79294.04 0.000114 1.4404175E-09```
	``` 78843.23 0.000192 1.417225E-09```
	``` 80634.28 0.000125 1.450221E-09```
	```]]></Data>```
	```</Table>```

29161055AC

2 Biacore™ S200 Control Software

Description	The root node that contains all other data carrying nodes.
Format	<LIMSInformation> <FileInformation> … </FileInformation> <Table> … </Table> </LIMSInformation> **where** <LIMSInformation> – is the root node. <FileInformation> – is the node where the file information is stored, see detailed format of node in chapter 2.1. <Table> – is the node where the exported report point table is stored, see detailed format of node in chapter 2.2.

2.1 /LIMSInformation/FileInformation

Description	This node contains the file information data. The data exported into this node is equivalent to the data displayed in the File Properties dialog box in the Biacore™ S200 Control Software.
Format	<FileInformation> <FileProperties> … </FileProperties> <RunInformation> … </RunInformation> <Instrument> … </Instrument> <UserInformation> … </UserInformation> <CreatedWithSoftware> … </CreatedWithSoftware> <CurrentSoftware> … </CurrentSoftware> <ChipInformation> … </ChipInformation> <Immobilization> … </Immobilization> </FileInformation> **where** The sub nodes listed here are described in the chapters below.

2.1.1 /LIMSInformation/FileInformation/FileProperties

Description	This node contains BLR-file related properties.
Format	<FileProperties> <Name>$(File name)</Name> <Path>$(Path)</Path> <Size>$(File size)</Size> </FileProperties> **where** $(File name) – is the name of the file including file extension. $(Path) – is the path where the result file is stored. $(File size) – is the size of the file in bytes including the text "bytes"
Example	<FileProperties> <Name>Dummy.blr</Name> <Path>C:\Bia Users\Results</Path> <Size>100 000 bytes</Size> </FileProperties>

2.1.2 /LIMSInformation/FileInformation/RunInformation

Description	This node contains information regarding the run.

Format	`<RunInformation>`
	
	`<Type>$(Run type)</Type>`
	
	`<Method>$(Method)</Method>`
	
	`<Template>$(Template)</Template>`
	
	`<Cycles>$(Cycle count)</Cycles>`
	
	`<Start>$(Start date)</Start>`
	
	`<Reasonforunexpectedendofrun>$(UnexpectedRunEndReason)</Reasonforunexpectedendofrun>`
	
	`<End>$(End date)</End>`
	
	`</RunInformation>`
	
	where
	
	$(Run type) – is a string that identifies the type of the run. It is the name of the wizard that created the run.
	
	$(Method) – the name of the method used to create this file.
	Note! Only available for runs created with a method.
	
	$(Template) – the name of the wizard template used to create this file.
	Note! Only available for runs created with a wizard template.
	
	$(Cycle count) – is the number of cycles present in this file.
	
	$(Start date) – is the date (as an invariant string) when this run was started.
	
	$(End date) – is the date (as an invariant string) when this file was ended.
	
	$(UnexpectedRunEndReason) – if the run was stopped due to an unexpected event the reason is written here as text. This node is not present as default.

Example	`<RunInformation>`
	
	`<Type>Immobilization</Type>`
	
	`<Cycles>4</Cycles>`
	
	`<Start>2006-02-01 09:28:09</Start>`
	
	`<End>2006-02-01 11:28:38</End>`
	
	`</RunInformation>`

29161055AC

2.1.3 /LIMSInformation/FileInformation/Instrument

Description	This node contains information about the instrument that was used to create this run.

Format	`<Instrument>`
	`<InstrumentType>$(Instrument)</InstrumentType>`
	`<InstrumentId>$(Instrument id)</InstrumentId>`
	`<IFC>$(IFC)</IFC>`
	`<VacuumUnit>$(Vaccum status)</ VacuumUnit>`
	`</Instrument>`
	
	where
	
	$(Instrument) – is the instrument type that created this run. This value is always "BiacoreS200".
	
	$(Instrument id) – is the serial number of the instrument that was used to create this run.
	
	$(IFC) – is the name of the IFC type that was inserted in the instrument.
	
	$(Vacuum status) – has the value 'Yes' if the used instrument had an active vacuum unit.

Example	`<Instrument>`
	`<InstrumentType>BiacoreS200</InstrumentType>`
	`<InstrumentId>12001</InstrumentId>`
	`<IFC>TYPE105</IFC>`
	`</Instrument>`

2.1.4 /LIMSInformation/FileInformation/UserInformation

Description	This node contains information about the user that created the run and the user that exported the file.

Format	```<UserInformation>```
	
	```<RunPerformedBy>$(Run by user)</RunPerformedBy>```
	
	```<CurrentUser>$(Current user)</CurrentUser>```
	
	```</UserInformation>```
	
	where
	
	$(Run by user) – is the username of the user that was logged into Windows™ when the run was started.
	
	$(Current user) – is the name of the username that was logged into Windows™ when the exported file was created.

Example	```<UserInformation>```
	
	```<RunPerformedBy>user</RunPerformedBy>```
	
	```<CurrentUser>user2</CurrentUser>```
	
	```</UserInformation>```

2.1.5   /LIMSInformation/FileInformation/CreatedWithSoftware

Description	This node contains information about the software that was used to create this file.

Format	`<CreatedWithSoftware>`
	`<Name>$(Software name)</Name>`
	`<Version>$(Software version)</Version>`
	`<Module>$(Software module)</Module>`
	`<Update>$(Update)</Update>`
	`</CreatedWithSoftware>`
	where
	$(Software name) – is the name of the software that was used to create the run.
	$(Software version) – is the version of the software that was used to create the run.
	$(Software module) – contains the name of optional modules installed. This node can occur more than once in this `<CreatedWithSoftware>` node.
	$(Update) – contains the name of optional installed software updates. This node can occur more than once in this `<CreatedWithSoftware>` node.
	Note! This type of update is usually not used in Biacore™ S200.

Example	`<CreatedWithSoftware>`
	`<Name>Biacore S200 Control Software</Name>`
	`<Version>1.1</Version>`
	`<Module>Some installed module</Module>`
	`<Update>New MS maldi plate 2006-01-01</Update>`
	`</CreatedWithSoftware>`

2.1.6 /LIMSInformation/FileInformation/CurrentSoftware

Description	This node contains information about the software that was used to create this exported meta data file. This can be a different software than the file was created with.

Format	`<CurrentSoftware>`
	`<Name>$(Software name)</Name>`
	`<Version>$(Software version)</Version>`
	`<Module>$(Software module)</Module>`
	`<Update>$(Update)</Update>`
	`</CurrentSoftware>`
	
	where
	
	$(Software name) – is the name of the software that was used to create this exported meta data file.
	
	$(Software version) – is the version of the software that was used to create this exported meta data file.
	
	$(Software module) – contains the name of optional modules installed. This node can occur more than once in this `<CurrentSoftware>` node.
	
	$(Update) – contains the name of optional installed software updates. This node can occur more than once in this `<CurrentSoftware>` node.
	
	Note! This type of update is usually not used in Biacore™ S200.

Example	`<CurrentSoftware>`
	`<Name>Biacore S200 Control Software</Name>`
	`<Version>1.1</Version>`
	`<Module>Some installed module</Module>`
	`<Update>New MS maldi plate 2006-01-01</Update>`
	`</CurrentSoftware>`

29161055AC

2.1.7 /LIMSInformation/FileInformation/ChipInformation

Description	This node contains information about the currently inserted sensor chip.

Format	`<ChipInformation>`
	`<ChipId>$(Chip id)</ChipId>`
	`<ChipLotNo>$(Chip lot no)</ChipLotNo>`
	`<ChipName>$(Chip name)</ChipName>`
	`<FirstDockDate>$(First dock date)</FirstDockDate>`
	`<LastModificationDate>$(Mod. date)</LastModificationDate>`
	`<LastUseDate>$(Last use date)</LastUseDate>`
	`</ChipInformation>`
	
	where
	
	$(Chip id) – is the id of the chip as entered when the chip was docked.
	
	$(Chip lot no) – is the lot number of the chip as entered when the chip was docked.
	
	$(Chip name) – is the name of the docked chip type.
	
	$(First dock date) – is the date when the chip was docked for the first time. Date as invariant string.
	
	$(Mod. date) – is the date when the chip was modified the last time. A chip is modified via the immobilization wizard. Date as invariant string.
	
	$(Last use date) – is the date when the chip was used the last time. Date as invariant string.

Example	`<ChipInformation>`
	`<ChipId>The id of the chip</ChipId>`
	`<ChipLotNo>1234567</ChipLotNo>`
	`<ChipName>CM5</ChipName>`
	`<FirstDockDate>2006-02-01</FirstDockDate>`
	`<LastModificationDate>2006-02-03</LastModificationDate>`
	`<LastUseDate>2006-02-05</LastUseDate>`
	`</ChipInformation>`

2.1.8 /LIMSInformation/FileInformation/Immobilization

Description	This node contains immobilization information for the flow cells on the chip. This node occurs one time for each flow cells that exists in the IFC. For Biacore™ S200 this means that there will be 4 Immobilization nodes.

Format	`<Immobilization>`
	`<Flowcell>$(Flow cell)</Flowcell>`
	`<ImmobilizationDate>$(Immob date)</ImmobilizationDate>`
	`<ImmobilizationResultFile>$(Immob result file) </ImmobilizationResultFile>`
	`<Ligand>$(Ligand)</Ligand>`
	`<LigandMolecularWeight>$(LigandMW)</LigandMolecularWeight>`
	`<FinalResponse>$(Final response)</FinalResponse>`
	`</Immobilization>`
	
	where
	
	$(Flow cell) – is the name of the flow cell that this node contains date for.
	
	$(Immob date) – is the date when this flow cell was immobilized. The date as invariant string. This value can be empty if the flow cell isn't immobilized.
	
	$(Immob result file) – is the name (including full path) of the result file that was created when this flow cell was immobilized. This value can be empty if the flow cell isn't immobilized.
	
	$(Ligand) – is the name of the immobilized ligand. This value can be empty if the flow cell isn't immobilized. If the flow cell is blank immoblized the value can be "[Blank]". If the run was aborted during this cycle and it wasn't possible to determine if the flow cell was successfully immobilized the value can be "[Incomplete results]".
	
	$(LigandMW) – is the molecular weight of the immobilized ligand. This node is only available if a ligand molecular weight has been entered.
	
	$(Final response) – is the id of the chip as entered when the chip was docked. This value only contains data if the flow cell has been successfully immobilized.

Example	```<Immobilization>```
	` <Flowcell>Fc=1</Flowcell>`
	` <ImmobilizationDate>2006-02-01</ImmobilizationDate>`
	` <ImmobilizationResultFile>C:\BIA Users\Results\Immob.blr</ImmobilizationResultFile>`
	` <Ligand>Test ligand</Ligand>`
	` <LigandMolecularWeight>150000</LigandMolecularWeight>`
	` <FinalResponse>1234.5</FinalResponse>`
	`</Immobilization>`

2.2   /LIMSInformation/Table

Description	This node contains the report point table data.
	This is all report points for all cycles and curves as a big tab separated table.
Format	Standard table format is used, see chapter 1. The name of the table is "ReportPointTable".
	Column headers ending with the character # can be treated as numeric values.
	The table data contains one initial row with the column headers. The column header in the data table does not contain the # character.
	Note: Row names are not used.

Example	`<Table Name="ReportPointTable">`

```
<Table Name="ReportPointTable">

 <Column1>Cycle</Column1>

 <Column2>Fc</Column2>

 <Column3>Aprog</Column3>

 <Column4>DiodeRow</Column4>

 <Column5>Time</Column5>

 <Column6>Window</Column6>

 <Column7>AbsResp</Column7>

 <Column8>SD</Column8>

 <Column9>Slope</Column9>

 <Column10>LRSD</Column10>

 <Column11>Quality</Column11>

 <Column12>Baseline</Column12>

 <Column13>RelResp</Column13>

 <Column14>Id</Column14>

 <Column15>Chip</Column15>

 <Column16>Ligand</Column16>

 <Column17>Method</Column17>

 <Column18>Procedure</Column18>

 <Column19>TargetLevel#</Column19>

 <Column20>ContactTime#</Column20>

 <Column21>FlowRate#</Column21>
```

Cycle	Fc	Aprog	DiodeRow	Time	Window
AbsResp	SD	Slope	LRSD	Quality	Baseline
RelResp	Id	Chip	Ligand	Method	Procedure
TargetLevel	ContactTime		FlowRate		
1	1	Amine_1	10	273	5
36808.0709635417		0.124936659977557		0.0614955357142857	
0.0544657669044648	Ok	Yes	N/A		Baseline
CM5	Blank	Amine	Blank		
1	1	Amine_1	10	400	5
43519.3209635417		1.46559342795652		-0.783147321428571	
0.041003831478355	Ok	No	6711.25		usr rpt
CM5	Blank	Amine	Blank]]>		

(Data is enclosed in `<Data><![CDATA[ ... ]]></Data>`)

```
 </Data>

</Table>
```

3   Biacore™ S200 Evaluation Software

3.1   /EvaluationFile

Description	The exported file from Biacore™ S200 Evaluation is an XML text file. The XML structure contains file properties and a section for each evaluation (item).
| Format | ```
<EvaluationFile>
 <FileProperties>
  …
 </FileProperties>
<Evaluation>
 …
</Evaluation>
<Evaluation>
 …
</Evaluation>
 …
</EvaluationFile>
``` |

3.2   /EvaluationFile/FileProperties

Description	The file properties section contains one section for the evaluation file properties and one section for all result files included in the evaluation file. The result files section contains a sub-section for each result file.
| Format | ```
<FileProperties>
 <EvalFileProperties>
  …
 </EvalFileProperties>
 <ResultFiles>
 <ResultFile1>
  …
 </ResultFile1>
 <ResultFile2>
  …
 </ResultFile2>
 …
 </ResultFiles>
</FileProperties>
``` |

3.2.1 /EvaluationFile/FileProperties/EvalFileProperties

Description	The evaluation file properties section contains the properties of the evaluation file. The Notebook is saved as CDATA.

Format	`<EvalFileProperties>`
	`<EvaluationFile>`
	`…`
	`</EvaluationFile>`
	`<UserInformation>`
	`…`
	`</UserInformation>`
	`<CreatedWithSoftware>`
	`…`
	`</CreatedWithSoftware>`
	`<Notebook>`
	`<![CDATA[ … ]]>`
	`</Notebook>`
	`</EvalFileProperties>`

3.2.1.1 EvaluationFile/FileProperties/EvalFileProperties/EvaluationFile/

Description	The name, path and size of the evaluation file.

Format	`<EvaluationFile>`
	`<Name>$(Name)</Name>`
	`<Path>$(Path)</Path>`
	`<Size>$(Size)</Size>`
	`</EvaluationFile>`
	
	where
	
	$(Name) – is the name of the evaluation file.
	
	$(Path) – is the path to the evaluation file.
	
	$(Size) – is the size (in bytes) of the evaluation file.
	
	Note: This section is empty if the file has not yet been saved as an evaluation file.

Example	`<EvaluationFile>`
	`<Name>Kinetics.bme</Name>`
	`<Path>C:\Bia Users</Path>`
	`<Size>911356</Size>`
	`</EvaluationFile>`

29161055AC

3.2.1.2   EvaluationFile/FileProperties/EvalFileProperties/UserInformation/

Description	The username of the user that saved the file and the currently logged in user.

Format	`<UserInformation>`
	`  <PerformedBy>$(Performed by)</PerformedBy>`
	`  <CurrentUser>$(Current user)</CurrentUser>`
	`</UserInformation>`
	
	where
	
	$(Performed by) – is the user that last saved the evaluation file.
	
	$(Current user) – is the currently logged in user.

Example	`<UserInformation>`
	`  <PerformedBy>Wolfgang</PerformedBy>`
	`  <CurrentUser>Helga</CurrentUser>`
	`</UserInformation>`

3.2.1.3   EvaluationFile/FileProperties/EvalFileProperties/CreatedWithSoftware/

Description	Information about the software that created the file.

Format	`<CreatedWithSoftware>`
	`  <Name>$(Name)</Name>`
	`  <Version>$(Version)</Version>`
	`  <Modules>`
	`    <Module>$(Module)</Module>`
	`    <Module>…</Module>`
	`    …`
	`  </Modules>`
	`</CreatedWithSoftware>`
	
	where
	
	$(Name) – is the name of the software that created the file.
	
	$(Version) – is the version of the software that created the file.
	
	$(Module) – is an installed software module.

Example	`<CreatedWithSoftware>`
	`  <Name>Biacore S200 Evaluation Software</Name>`
	`  <Version>1.1</Version>`
	`  <Modules>`
	`    <Module>Some installed module</Module>`
	`  </Modules>`
	`</CreatedWithSoftware>`

29161055AC

29161055AC

3.2.2 EvaluationFile/FileProperties/ResultFileX

Description	The result file properties are child sections in the ResultFiles sections. The first result file is in the section ResultFile1, the second is in the ResultFile2 section and so on. The Immobilization sections contain immobilization information for the four flow cells.
Format	``` <ResultFile1> <ResultFile> … </ResultFile> <RunInformation> … </RunInformation> <Instrument> … </Instrument> <UserInformation> … </UserInformation> <CreatedWithSoftware> … </CreatedWithSoftware> <ChipInformation> … </ChipInformation> <Immobilization> … </Immobilization> <Immobilization> … </Immobilization> … </ResultFile1> ```

3.2.2.1 EvaluationFile/FileProperties/ResultFileX/ResultFile/

Description	The result file properties.
Format	This section has the same format as the corresponding section in the Biacore™ S200 Evaluation file properties (see chapter 3.2.1.3).

29161055AC

3.2.2.2   EvaluationFile/FileProperties/ResultFileX/RunInformation/

Description	The information about the run.
Format	This section has the same format as the corresponding section in the Biacore™ S200 Control Software export format (see chapter 2.1.2).

3.2.2.3   EvaluationFile/FileProperties/ResultFileX/Instrument/

Description	The information about the instrument.
Format	This section has the same format as the corresponding section in the Biacore™ S200 Control Software export format (see chapter 2.1.3).

3.2.2.4   EvaluationFile/FileProperties/ResultFileX/UserInformation/

Description	This section contains information about the user that created the run.
Format	This section has the same format as the corresponding section in the Biacore™ S200 Control Software export format (see chapter 0), except that the CurrentUser section is not present.

3.2.2.5   EvaluationFile/FileProperties/ResultFileX/CreatedWithSoftware/

Description	This section contains information about the Control Software that created the run.
Format	This section has the same format as the corresponding section in the Biacore™ S200 Control Software export format (see chapter 0).

3.2.2.6   EvaluationFile/FileProperties/ResultFileX/ChipInformation/

Description	This section contains information about the chip that was used in the run.
Format	This section has the same format as the corresponding section in the Biacore™ S200 Control Software export format (see chapter 2.1.7).

3.2.2.7   EvaluationFile/FileProperties/ResultFileX/Immobilization/

Description	This section contains information about the immobilization in one flow cell on the chip.

Format	This section has the same format as the corresponding section in the Biacore™ S200 Control Software export format (see chapter 2.1.8).

3.3 /EvaluationFile/Evaluation

Description	Each evaluation is exported as its own section. The Type section contains the type of evaluation, e.g. "Plot", and the Name section contains the name of the evaluation, e.g. "Binding level". The rest of the section is different for the different evaluations.

Format	```<Evaluation>```
	 ```<Type>…</Type>```
	 ```<Name>…</Name>```
	 ```…```
	 ```</Evaluation>```

3.3.1 Sensorgram

Sensorgram is not exported.

3.3.2 QCPlot

Description	The QCPlot evaluation item.

Format	```<Evaluation>```
	 ```<Type>Plot</Type>```
	 ```<Name>$(Name)</Name>```
	 ```<XAxis>```
	 ```<Variable>$(Variable)</Variable>```
	 or
	 ```<ReportPoint>$(ReportPoint)</ReportPoint>```
	 ```<ResponseType>$(ResponseType)<ResponseType>```
	 ```<ReportPointAdjustment>```
	 ```<Adjustment>```
	 ```<AdjustmentName>$(AdjustmentName)</AdjustmentName>```
	 ```</Adjustment>```
	 ```</ReportPointAdjustment>```
	 ```</XAxis>```
	 ```<YAxis>```
	 ```<ReportPoint>$(ReportPoint)</ReportPoint>```
	 ```<ResponseType>$(ResponseType)<ResponseType>```
	 ```<ReportPointAdjustment>```
	 ```<Adjustment>```
	 ```<AdjustmentName>$(AdjustmentName)</AdjustmentName>```
	 ```</Adjustment>```

	`</ReportPointAdjustment>`
	
	`</YAxis>`
	
	`<Table Name="Table">`
	
	` $(Table)`
	
	`<Table>`
	
	`</Evaluation>`
	
	where
	
	$(Name) - is the name of the plot.
	
	$(Variable) - is the variable name.
	
	$(ReportPoint) - is the report point name.
	
	$(ResponseType) - is the response type. Possible values are "Absolute response", "Relative response" and "Slope".
	
	$(AdjustmentName) – is the name of a report point adjustment. Possible values are "Molecular Weight Adjustment" and "Capture Adjustment".
	
	$(Table) – contains the plot table in standard table format, see chapter 1. Note: The ReportPointAdjustment section is available only if report point adjustment is performed.
	
	The XAxis section contents are different if the plot is a "Report point vs variable" compared to a "Report point vs report point".

3.3.3   Result plot

Description	The Result plot evaluation item.

Format	`<Evaluation>`
	
	`<Type>Plot</Type>`
	
	`<Name>$(Name)</Name>`
	
	`<XAxis>`
	
	`<Variable>$(Variable)</Variable>`
	
	`</XAxis>`
	
	`<YAxis>`
	
	`<ReportPoint>$(ReportPoint)</ReportPoint>`
	
	`<ResponseType>$(ResponseType)<ResponseType>`
	
	`<ReportPointAdjustment>`
	
	`<Adjustment>`
	
	`<AdjustmentName>$(AdjustmentName)</AdjustmentName>`
	
	`<BlankSample>$(BlankSampleName)</BlankSample>`
	
	`<BlankSubtractionType>$(SubtractionSettings)</BlankSubtractionType>`
	
	`<PositiveControl>$(PosControlName)</PositiveControl>`
	
	`<NegativeControl>$(NegControlName)</NegativeControl>`
	
	`<ResponseSetAsZeroLevel>$(ResponseSetAsZeroLevel)</ResponseSetAsZeroLevel>`

```
      <FittingFunction>$(FittingFunctionName)</FittingFunction>

<UpperRangeLimit>$(UpperRangeLimit)</UpperRangeLimit>

<LowerRangeLimit>$(LowerRangeLimit)</LowerRangeLimit>

<WindowWidth>$(WindowWidth)</WindowWidth>

</Adjustment>

</ReportPointAdjustment>

</YAxis>

 <Ranking>

   <RankingItem>

    <RankingName>$(RankingName)</RankingName>

    <RankingResponse>$(RankingResponse)</RankingResponse>

    <RankingValue>$(RankingValue)</RankingValue>

   </RankingItem>

 </Ranking>
```

```
<CutOffs>

<CutOff>

<CutOffControlSampleName>$(CutOffControlSampleName)</CutOffControlSampleName>

  <CutOffControlSampleSeriesAvgResponse>$(CutOffCSSeriesAvgResp)
</CutOffControlSampleSeriesAvgResponse>

<CutOffControlSampleSeriesSD>$(CutOffCSSeriesSD)</CutOffControlSampleSeriesSD>

<NoOfSD>$(CutOffNoOfSD)</NoOfSD>

  <CutOffResponse>$(CutOffResponse)</CutOffResponse>

</CutOff>

</CutOffs>

<curveFitting>

   <curve>

    <name>$(Curve name)</name>

    <function>$(Fitting function)</function>

    <Rhi>$(Rhi)</Rhi>

    <Rlo>$(Rlo)</Rlo>

    <A1>$(A1)</A1>

    <A2>$(A2)</A2>

    <Chi2>$(Chi2)</Chi2>

   </curve>

   <curve> .....

   </curve>

    Or

   <curve>

    <name>$(Curve name)</</name>

    <function>$(Fitting function)</function>

    <slope>$(Slope)</slope>

    <intercept>$(Intercept)</intercept>
```

29161055AC

```
        <R2>$(R2)</R2>

      </curve>

      <curve> .....

      </curve>

</curveFitting>


<Table Name="Table">

$(Table)

<Table>

</Evaluation>
```

where

$(Name) - is the name of the plot.

$(Variable) - is the variable name.

$(ReportPoint) - is the report point name.

$(ResponseType) - is the response type. Possible values are "Absolute response", "Relative response" and "Slope".

$(AdjustmentName) – is the name of a report point adjustment. Possible values are "Blank Subtraction", "Molecular Weight Adjustment", "Capture Adjustment", "Adjustment For Controls" and "Median Filtering".

$(BlankSampleName) – is the name of blank sample in Blank Subtraction.

$(SubtractionSettings) – is the Subtraction settings variable in Blank Subtraction. Possible values are "Nearest Blank", "Average Nearest Blanks", "Preceding Blank" and "Following Blank".

$(PosControlName) – is the name of positive control in Adjustment For Controls.

$(NegControlName) – is the name of negative control in Adjustment For Controls.

$(ResponseSetAsZeroLevel) – is the response set as zero level in Adjustment For Controls.

$(FittingFunctionName) – is the name of used fitting function in Adjustment For Controls. Possible values are "Linear" and "Polynomial".

$(UpperRangeLimit) – is the response limit for the upper range in Median Filtering.

$(LowerRangeLimit) – is the response limit for the lower range in Median Filtering.

$(WindowWidth) – is the x-axis window width in Median Filtering.

$(RankingName) – is the name of a ranking item.

$(RankingResponse) – is the response of a ranking item.

$(RankingValue) – is the value attached to a ranking item.

$(CutOffControlSampleName) – is the name of selected control sample when using automatic cut-off.

$(CutOffCSSeriesAvgResp) – is the average response for selected control sample series when using automatic cut-off.

$(CutOffCSSeriesSD) – is the standard deviation for selected control sample series when using automatic cut-off.

$(CutOffNoOfSD) – is the selected standard deviation offset of ranking line from mean value when using automatic cut-off.

$(CutOffResponse) – is the response level for a cut-off line.

$(Fitting function) - is the name of selected fitting function.

$(Curve name) - is the name of the curve for current curve section.

$(Rhi) – is the calculated Rhi value

$(Rlo) – is the calculated Rlo value.

$(A1) – is the calculated A1 value.

$(A2) – is the calculated A2 value.

$(Chi2) – is the calculated Chi2 value.

$(Slope) – is the calulated slope value.

$(Intercept) – is the calculated intercept value.

$(R2) – is the calculated R2 value.

$(Table) – contains the plot table in standard table format, see chapter 1.

Note: The Report point adjustment section is available only if report point adjustment is performed. The content in the section is also dependent on which report point adjustment that are used

The Ranking section is available only if ranking is performed.

The Cut-off section is available only if cut-off is performed and the content in the section is dependent on if automatic or manual cut-off mode is used.

The curveFitting section is available only if curve fitting is performed. One curve section is available for each curve. Reported parameters are dependent of selected fitting function.

The XAxis section contents are different if the plot is a "Report point vs variable" compared to a "Report point vs report point".

3.3.4 Clean screen

Description	The Clean screen evaluation item.

Format	`<Evaluation>`
	`<Type>Clean Screen</Type>`
	`<Name>$(Name)</Name>`
	`<XAxis>`
	`<Variable>$(Variable)</Variable>`
	`</XAxis>`
	`<YAxis>`
	`<ReportPoint>$(ReportPoint)</ReportPoint>`
	`<ResponseType>$(ResponseType)<ResponseType>`
	`</YAxis>`
	`<CutOffs>`
	`<CutOff>`
	`<CurveName>$(CutOffCurveName)</CurveName>`
	`<CutOffResponse>$(CutOffResponse)</CutOffResponse>`
	`</CutOff>`
	`</CutOffs>`
	`<Table Name="Table">`
	`$(Table)`
	`<Table>`
	`</Evaluation>`
	
	where
	
	$(Name) - is the name of the Clean Screen item.
	
	$(Variable) - is the variable name.
	
	$(ReportPoint) - is the report point name.
	
	$(ResponseType) - is the response type.
	
	$(CutOffCurveName) – is the curve name for a cut-off line.
	
	$(CutOffResponse) – is the response level for a cut-off line.
	
	$(Table) – contains the Clean screen table in standard table format, see chapter 1.

3.3.5 Binding level screen

Description	The Binding level screen evaluation item.
| Format | ```
<Evaluation>
 <Type>Binding Level Screen</Type>
 <Name>$(Name)</Name>
 <XAxis>
  <Variable>$(Variable)</Variable>
 </XAxis>
 <YAxis>
  <ReportPoint>$(ReportPoint)</ReportPoint>
  <ResponseType>$(ResponseType)<ResponseType>
  <ReportPointAdjustment>
   $(ReportPointAdjustment)
  </ReportPointAdjustment>
 </YAxis>
 <CutOffs>
  <CutOff>
   <CurveName>$(CutOffCurveName)</CurveName>
   <CutOffResponse>$(CutOffResponse)</CutOffResponse>
   <CutOffIncludedAbovePoints>
    $(CutOffIncludedAbovePoints)
   </CutOffIncludedAbovePoints>
   <CutOffTotalPoints>
    $(CutOffTotalPoints)
   </CutOffTotalPoints>
   <CutOffIncludedPercent>
    $(CutOffIncludedPercent)
   </CutOffIncludedPercent>
   <CutOffIncludedMarkers>
    $(CutOffIncludedMarkers)
   </CutOffIncludedMarkers>
  </CutOff>
 </CutOffs>
 <Table Name="Table">
  $(Table)
 <Table>
</Evaluation>
``` |

	where $(Name) - is the name of the Binding Level Screen item. $(Variable) - is the variable name. $(ReportPoint) - is the report point name. $(ResponseType) - is the response type. $(ReportPointAdjustment) - is the report point adjustment section as defined in chapter 3.3.3. $(CutOffCurveName) – is the curve name for a cut-off line. $(CutOffResponse) – is the response level for a cut-off line. $(CutOffIncludedAbovePoints) – is the number of sample points above the cut-off line. $(CutOffTotalPoints) – is the total number of sample points. $(CutOffIncludedPercent) – is the percent of sample points above the line. $(CutOffIncludedMarkers) – is the number of sample points above the line with markers. $(Table) – contains the Binding level screen table in standard table format, see chapter 1.

3.3.6   Report point table

Description	The Report point table evaluation item.

Format	`<Evaluation>`
	 `<Type>Report Point Table</Type>`
	 `<Name>Report Point Table</Name>`
	 `<Table Name="ReportTable">`
	  `$(Table)`
	 `</Table>`
	`</Evaluation>`
	
	where
	
	$(Table) – contains the report point table in standard table format, see chapter 1.

3.3.7   Solvent correction

Solvent correction is not exported.

3.3.8 Kinetics and Affinity

Description	The Kinetics and Affinity evaluation items.

Format	```<Evaluation>```
	 ```<Type>$(Type)</Type>```
	 ```<Name>$(Name)</Name>```
	 ```<Table Name="Table">```
	 ```$(Table table)```
	 ```</Table>```
	 ```<curveSets>```
	 ```<curveSet>```
	 ```<LigandName>$(Ligand)</LigandName>```
	 ```<SampleName>$(Sample)</SampleName>```
	 ```<Temperature>$(Temperature)</Temperature>```
	 ```<CurveName>$(CurveName)</CurveName>```
	 ```<Fits>```
	 ```<Fit>```
	 ```<Model>$(Model)</Model>```
	 ```<Description>$(Description)</Description>```
	 ```<Table Name="Report">```
	 ```$(Report table)```
	 ```</Table>```
	 ```<Table Name="Parameters">```
	 ```$(Parameters table)```
	 ```</Table>```
	 ```</Fit>```
	 ```<Fit>```
	 ```…```
	 ```</Fit>```
	 ```…```
	 ```</Fits>```
	 ```</curveSet>```
	 ```</curveSets>```
	 ```</Evaluation>```

	where $(Type)$ – is the type of the evaluation item, Affinity or Kinetics. $(Name)$ – is the name of the Affinity or Kinetics evaluation. $(Table table)$ – is the table configured for the evaluation item using standard table format, see chapter 1. $(Ligand)$ – is the ligand name. $(Sample)$ – is the sample name. $(Temperature)$ – is the temperature. $(CurveName)$ – is the name of selected curve. $(Model)$ – is the model used in the fit. $(Description)$ – is the description of the fit. $(Report table)$ – is the report table of the fit using standard table format, see chapter 1. $(Parameters table)$ – is the parameters table of the fit using standard table format, see chapter 1.

3.3.9 Thermodynamics

Description	The Thermodynamics evaluation item.

Format	`<Evaluation>`
	　`<Type>Thermodynamics</Type>`
	　`<Name>$(Name)</Name>`
	　`<LigandName>$(Ligand name)<LigandName>`
	　`<SampleName>$(Sample)</SampleName>`
	　`<Table Name="ImportTable">`
	　　`$(Import table)`
	　`</Table>`
	　`<Table Name="ResultTable">`
	　　`$(Result table)`
	　`</Table>`
	`</Evaluation>`
	
	where
	
	$(Name)$ – is the name of the Thermodynamics evaluation.
	
	$(Ligand name)$ – is the ligand name.
	
	$(Sample)$ – is the sample name.

	$(Import table) – is the import table of the fit using standard table format, see chapter 1. $(Result table) – is the result table of the fit using standard table format, see chapter 1.

3.3.10 Affinity in solution

Description	The Affinity in solution evaluation item.

Format	`<Evaluation>`
	 `<Type>Affinity in Solution</Type>`
	 `<Name>$(Name)</Name>`
	 `<CalibrationCurveName>$(Calibration)</CalibrationCurveName>`
	 `<Table Name="CalibrationTable">`
	 `$(Calibration table)`
	 `</Table>`
	 `<Sample>$(Sample)</Sample>`
	 `<ConstantConcentrationB>$(Const. B)</ConstantConcentrationB>`
	 `<Table Name="SampleTable">`
	 `$(Sample table)`
	 `</Table>`
	 `<KD>$(KD)</KD>`
	 `<SEKD>$(SE KD)</SEKD>` **Or** `<TVALUEKD>$(T-value KD)</TVALUEKD>`
	 `<Chi2>$(Chi2)</Chi2>`
	 `</Evaluation>`
	
	where
	
	$(Name) – is the name of the Affinity in solution evaluation.
	
	$(Calibration) – is the name of the selected calibration curve.
	
	$(Calibration table) – is the calibration table using standard format, see chapter 1.
	
	$(Sample) – is the selected sample name.
	
	$(Const. B) – is the constant concentration B.
	
	$(Sample table) – is the sample table using standard format, see chapter 1.
	
	$(KD) – is the calculated KD value.
	
	$(SE KD) is the standard error of the calculated KD value.
	
	$(T-value KD) – is T-value of the calculated KD value.
	
	$(Chi2) – is the Chi2 in the curve fit.

29161055AC

cytiva.com

For local office contact information, visit cytiva.com/contact

29870417 AA V:2 12/2024

29161055AC