# Overview

This document is an addendum to the Cornerstone Remote Control document that highlights the syntax of LECO’s remote control API for Cornerstone. This document contains additional commands that are exclusive to GDS instruments.

# Calculation Jobs Active

This returns the true if the software has background calculation jobs running.

## Syntax

<CalculationJobsActive />

## Reply

<CalculationJobsActive Value=”true”/>

## Possible Error Codes

6 – Exception

# CDP Method

This command returns detail data for the specified CDP method. The CDP method can be specified either by key or by name.

## Parameters

One of the following must be supplied to identify the CDP method. If both are provided, the Key parameter will be used to identify the method.

**Key**: The unique key that identifies the CDP method. Leading zeroes may be omitted.

**Name**: The unique name that identifies the CDP method.

## Syntax

<CdpMethod Key="D3"/>

<CdpMethod Name="CDP"/>

## Reply

<CdpMethod ErrorCode=”0” ErrorMessage=”Success”>

<Key>00000000000000D3</Key>

<Name Label="Name">CDP</Name>

<Description Label="Description"></Description>

<LastUsed Label="Last Used">03/19/2014 15:28:29</LastUsed>

<LastModified Label="Last Modified">06/03/2013 11:36:00</LastModified>

<Excluded Label="Excluded">False</Excluded>

<BulkMethod Label="Bulk Method">Bulk</BulkMethod>

<QuantifierParameters>

<CalibrationFactor Label="Calibration Factor">1</CalibrationFactor>

<QuantificationElements Label="Elements To Quantify">C Al Cr Fe Ni Cu Zn Sn</QuantificationElements>

</QuantifierParameters>

<SourceSettings>

<CalibrationFactor Label="Calibration Factor">1</CalibrationFactor>

</SourceSettings>

<CdpCalculations>

<CdpCalculation>

<Name Label="Name">Ct Wt Zn</Name>

<Description Label="Description"></Description>

<LastUsed Label="Last Used">01/01/0001 00:00:00</LastUsed>

<LastModified Label="Last Modified">01/01/0001 00:00:00</LastModified>

<Excluded Label="Excluded">False</Excluded>

<TraceDomain Label="Trace Domain">Composition vs. Depth</TraceDomain>

<OutputUnits Label="Units">Coating Weight</OutputUnits>

<CalculationExpression Label="Calculation">CoatingWeight(Zn[..])</CalculationExpression>

<StartExpression Label="Start">Start</StartExpression>

<EndExpression Label="End">End</EndExpression>

</CdpCalculation>

…

</CdpCalculations>

<Plots>

<GdsPlot>

<Name Label="Name">Basic Intensities</Name>

<Description Label="Description"></Description>

<LastUsed Label="Last Used">01/01/0001 00:00:00</LastUsed>

<LastModified Label="Last Modified">01/01/0001 00:00:00</LastModified>

<Excluded Label="Excluded">False</Excluded>

<TraceDomain Label="Trace Domain">Intensity vs. Time</TraceDomain>

<CompositionUnits Label="Composition">Percent</CompositionUnits>

<DepthUnits Label="Depth">Micrometers</DepthUnits>

<IsLogXAxis Label="X Axis Log Scale">False</IsLogXAxis>

<IsLogYAxis Label="Y Axis Log Scale">False</IsLogYAxis>

<IsAutoScalingTraces Label="Auto Scale">True</IsAutoScalingTraces>

<MinimumX Label="X Axis Minimum">NaN</MinimumX>

<MaximumX Label="X Axis Maximum">NaN</MaximumX>

<MinimumLeftY Label="Left Y Axis Minimum">NaN</MinimumLeftY>

<MaximumLeftY Label="Left Y Axis Maximum">NaN</MaximumLeftY>

<MinimumRightY Label="Right Y Axis Minimum">NaN</MinimumRightY>

<MaximumRightY Label="Right Y Axis Maximum">NaN</MaximumRightY>

<Traces>

<GdsPlotTrace>

<Name Label="Name">C</Name>

<Analyte Label="Analyte">C</Analyte>

<IsIncluded Label="Included">True</IsIncluded>

<IsMarkerVisible Label="Markers">False</IsMarkerVisible>

<IsLeftAxis Label="Axis">False</IsLeftAxis>

<Color Label="Color">#FF7B4F24</Color>

<Style Label="Style">Solid</Style>

</GdsPlotTrace>

…

</Traces>

<Calculations>

<GdsPlotCalculation>

<Name Label="Name">Avg Fe Intensity</Name>

<Key Label="Key">0000000000002AFF</Key>

<IsIncluded Label="Included">False</IsIncluded>

<IsMarkerVisible Label="Markers">False</IsMarkerVisible>

<Color Label="Color">#FFFF6700</Color>

<Style Label="Style">SolidLine</Style>

</GdsPlotCalculation>

…

</Calculations>

</GdsPlot>

…

</Plots>

</CdpMethod>

## Possible Error Codes

6 – Exception

11 - RequestedItemNotFound

# CDP Methods

This command returns all CDP methods.

## Syntax

<CdpMethods/>

## Reply

<CdpMethods ErrorCode=”0” ErrorMessage=”Success”>

<CdpMethod>

<Key>00000000000000D3</Key>

<Name Label="Name">SWV</Name>

<Description Label="Description"></Description>

<LastUsed Label="Last Used">03/19/2014 15:28:29</LastUsed>

<LastModified Label="Last Modified">06/03/2013 11:36:00</LastModified>

<Excluded Label="Excluded">False</Excluded>

</CdpMethod>

…

</CdpMethods>

## Possible Error Codes

6 – Exception

# CdpMethodKey

This command returns the key for the CDP method whose name matches the name provided.

## Parameters

**Name**: The unique name that identifies the CDP method.

## Syntax

<CdpMethodKey Name=”…”/>

## Reply

<CdpMethodKey Name="…" ErrorCode=”0” ErrorMessage=”Success”>

<Key>0000000000000010</Key>

</CdpMethodKey>

## Possible Error Codes

6 – Exception

11 - RequestedItemNotFound

# Conditioning Required

This returns whether conditioning is required. Conditioning would be required if an analyzed sample’s spectra is not a good match for the method. If true, add and analyze a conditioner sample.

## Syntax

<ConditioningRequired />

## Reply

<ConditioningRequired Value=”false”/>

## Possible Error Codes

6 – Exception

# Last Conditioner

This returns the date and time of the last successful conditioner calculation.

## Syntax

<LastConditioner />

## Reply

<LastConditioner Value=”06/27/2019 15:23:35”/>

## Possible Error Codes

6 – Exception

# Limit

This command returns detail data for the specified limit. The limit can be specified either by key or by name.

## Parameters

One of the following must be supplied to identify the limit. If both are provided, the Key parameter will be used to identify the limit.

**Key**: The unique key that identifies the limit. Leading zeroes may be omitted.

**Name**: The unique name that identifies the limit.

## Syntax

<Limit Key="D3"/>

<Limit Name="SWV"/>

## Reply

<Limit ErrorCode=”0” ErrorMessage=”Success”>

<Key>00000000000000D3</Key>

<Name Label="Name">SWV</Name>

<Description Label="Description"></Description>

<LastUsed Label="Last Used">03/19/2014 15:28:29</LastUsed>

<LastModified Label="Last Modified">06/03/2013 11:36:00</LastModified>

<Excluded Label="Excluded">False</Excluded>

<Elements>

<Element>

<Analyte>Fe</Analyte>

<LowerControlLimit>90</LowerControlLimit>

<UpperControlLimit>100</UpperControlLimit>

</Element>

…

</Elements>

</Limit>

## Possible Error Codes

6 – Exception

11 - RequestedItemNotFound

# Limits

This command returns all limits.

## Syntax

<Limits/>

## Reply

<Limits ErrorCode=”0” ErrorMessage=”Success”>

<Limit>

<Key>00000000000000D3</Key>

<Name Label="Name">SWV</Name>

<Description Label="Description"></Description>

<LastUsed Label="Last Used">03/19/2014 15:28:29</LastUsed>

<LastModified Label="Last Modified">06/03/2013 11:36:00</LastModified>

<Excluded Label="Excluded">False</Excluded>

</Limit>

…

</Limits>

## Possible Error Codes

6 – Exception

# LimitKey

This command returns the key for the limit whose name matches the name provided.

## Parameters

**Name**: The unique name that identifies the limit.

## Syntax

<LimitKey Name=”…”/>

## Reply

<LimitKey Name="…" ErrorCode=”0” ErrorMessage=”Success”>

<Key>0000000000000010</Key>

</LimitKey>

## Possible Error Codes

6 – Exception

11 - RequestedItemNotFound

# StandardsForElements

This command retrieves the standards for the specified elements.

## Parameters

**MethodKey**: The unique key that identifies the specific method of interest. This parameter can be replaced with the name of method in a parameter called MethodName.

**Usage:** Indicates which types of standards to return. Valid values are CALIBRATION, CHECK, DRIFT, TYPE and TRANSFER.

## Syntax

<StandardsForElements MethodKey=”…” Usage=”…”>

<Elements>

<Element>Fe</Element>

…

</Elements>

</StandardsForElements>

<StandardsForElements MethodName=”…”>

<Elements>

<Element>Fe</Element>

…

</Elements>

</StandardsForElements>

## Reply

<StandardsForElements ErrorCode=”0” ErrorMessage=”Success”>

<Standards>

<Standard Key="0000000000000462" Name="1761" Manufacturer="NIST" />

<Standard Key="0000000000000463" Name="1762" Manufacturer="NIST" />

<Standard Key="0000000000000464" Name="1763" Manufacturer="NIST" />

<Standard Key="0000000000000465" Name="1764" Manufacturer="NIST" />

<Standard Key="0000000000000466" Name="1765" Manufacturer="NIST" />

<Standard Key="0000000000000467" Name="1766" Manufacturer="NIST" />

<Standard Key="0000000000000468" Name="1767" Manufacturer="NIST" />

<Standard Key="0000000000004799" Name="RN 19-65" Manufacturer="SUS" />

<Standard Key="000000000000575E" Name="3Q13L1" Manufacturer="ASTM" />

<Standard Key="0000000000005850" Name="BS17A" Manufacturer="BRAMMER" />

<Standard Key="0000000000005EE1" Name="CFe2-5" Manufacturer="SUS" />

</Standards>

</StandardsForElements>

## Possible Error Codes

6 – Exception

11 - RequestedItemNotFound

# StandardManufacturers

This command retrieves the list of all standard manufacturers.

## Syntax

<StandardManufacturers/>

## Reply

<StandardManufacturers ErrorCode=”0” ErrorMessage=”Success”>

<Manufacturer Name="ALCAN" />

<Manufacturer Name="ALCOA" />

<Manufacturer Name="ALPHA" />

…

</StandardManufacturers>

## Possible Error Codes

6 – Exception

# StandardsForManufacturers

This command returns the list of standards for the specified manufacturer.

## Parameters

**Manufacturer:** Name of manufacturer as returned in the StandardManufacturers command.

## Syntax

<StandardsForManufacturers Manufacturer=”…”/>

## Reply

<StandardsForManufacturers ErrorCode=”0” ErrorMessage=”Success”>

<Standard>

<Key>0000000000000462</Key>

<Name Label="Name">1761</Name>

<Description Label="Description"></Description>

<Manufacturer Label="Manufacturer">NIST</Manufacturer>

<LastUsed Label="Last Used">08/10/2018 17:59:31</LastUsed>

<LastModified Label="Last Modified">08/02/2017 15:27:08</LastModified>

<Excluded Label="Excluded">False</Excluded>

<SputterRateMode Label="Sputter Rate Mode">Determined</SputterRateMode>

<SputterRate Label="Sputter Rate Mode">0.273</SputterRate>

</Standard>

…

</StandardsForManufacturer>

## Possible Error Codes

6 – Exception

11 – RequestedItemNotFound

# StringValue

This lists the specific replies for loading and unloading a sample through remote control, which is through the string value of “Sample Load State”.

## Syntax

<StringValue Key="Sample Load State”/>

## Reply

<StringValue Key=”Sample Load State” Value=”…”/>

|  |  |
| --- | --- |
| **Value** | **Meaning** |
| Unclamping | Start of unload step 1. Retracting the reamer from the sample. |
| Unclamped | Unload step 1 sequence complete. Finished retracting the reamer. |
| Releasing | Start of unload step 2. Cleaning burst to free sample from lamp. |
| Released | Unload step 2 complete. Cleaning burst finished. |
| Cleaning Anode | Start of unload step 3. Reaming and cleaning anode. |
| No Sample | Unload step 3 complete. Anode cleaning complete, no sample on lamp, door unlocked. |
| Evacuating | Start of load step 1. Drawing to vacuum on lamp to remove gases. |
| Evacuated | Load step 1 complete. Achieved lamp pressure < 1 torr. |
| Error: pressure evacuation timeout | Unable to achieve < 1 torr, when evacuating, in desired time. |
| Clamped – Low Pressure | Start of load step 2. Clamping sample to lamp with reamer assembly at low pressure. |
| Clamped – High Pressure | Clamping sample to lamp with reamer assembly at high pressure. |
| Depressurizing to 0.1 torr | Evacuating lamp to < 0.1 torr. |
| Preparing | Pressurizing lamp to lamp seed pressure for method. |
| Loaded | Load step 2 complete. Sample loaded and ready for analysis. |
| Error: door close timeout | Door was not closed fully within 60 seconds of sample being clamped at low pressure. |
| Error: depressurizing to 0.1 torr timeout | Lamp pressure did not get below 0.1 torr in 60 seconds, when trying to depressurize. |
| Error: prepare sample | Either operator aborted or timeout while pressurizing or depressurizing, during lamp preparation cycles. |
| Operator Abort: xxx | Operator intervened in load or unload of sample by manually stopping with abort button. |

# AddConditioningSamples

This command will add conditioning samples to the list of samples to analyze and return the key values for each sample set added.

## Parameters

**RepCount**: The number of replicates to add to each conditioning set. This parameter is optional. If not specified, the number of replicates added to each set will be 1.

**Operator:** The value to place into the operator field for the conditioning replicates. This parameter is optional. If not specified, the operator field will be empty.

## Syntax

<AddConditioningSamples/>

<AddConditioningSamples RepCount=”2”/>

<AddConditioningSamples Operator=”User”/>

## Reply

<AddConditioningSamples ErrorCode=”0” ErrorMessage=”Success”>

<Keys>

<Key>…</Key>

…

</Keys>

</AddConditioningSamples>

## Possible Error Codes

6 – Exception

# AddSamples

For GDS, this command has additional options in addition to the base.

## Parameters: Add Replicates to a New Set

**SampleType**: The sample type of the set. Valid values are "Sample", “CDP”, “Check”, and "Drift".

**CdpMethodKey**: The unique key that identifies the specific CDP method to assign to the set of sample type CDP. Leading zeros may be omitted.

**CdpMethodName**: The unique name that identifies the specific CDP method to assign to the set of sample type CDP.

Note: When adding a CDP set, you should only specify the CDP Method. There is no need to specify the method, as that will be set along with the CDP Method.

## Parameters: Replicates

**Mass**: Not applicable.

**Location**: Not applicable.

## Syntax: Add Replicates to a New Set

<AddSamples>

<PromptOperatorForEntry>True</PromptOperatorForEntry>

<Set>

<Field Id=”SampleType”></Field>

<Field Id=”Name”></Field>

<Field Id=”Description”></Field>

<Field Id=”CdpMethodKey”>0</Field>

<Field Id=”StandardKey”>0</Field>

<Field Id=”UDF: 1111”>User defined field value here</Field>

<Field Id=”SetId”>Set 1</Field>

</Set>

<Replicates>

<Replicate>

<Field Id=”Comments”></Field>

<Field Id=”UDF: 1234”>User defined field value here</Field>

<Field Id=”RepId”>Rep 1</Field>

</Replicate>

<Replicate>

<Field Id=”Comments”></Field>

<Field Id=”UDF: 1234”>User defined field value here</Field>

<Field Id=”RepId”>Rep 2</Field>

</Replicate>

</Replicates>

</AddSamples>

# AutoAnalyze

This command changes the auto analyze on sample load state of the instrument.

## Parameters

**State**: Auto analyze on sample load state to set on instrument. Valid values are "ENABLED" and "DISABLED".

## Syntax

<AutoAnalyze State=”ENABLED”/>

## Reply

<AutoAnalyze ErrorCode=”0” ErrorMessage=”Success”/>

## Possible Error Codes

6 – Exception

8 - CommandCurrentlyUnavailable

9 – UnknownParameterValue

10 – MissingAttribute

13 - UserDoesNotHavePermissionToExecuteCommand

# Condition

This command performs a conditioning operation for the specified sample set.

## Parameters

**SetKey**: The unique key that identifies the specific sample set of interest.

## Syntax

<Condition SetKey=”…”/>

## Reply

<Condition ErrorCode=”0” ErrorMessage=”Success”/>

## Possible Error Codes

6 – Exception

11 – RequestedItemNotFound

12 – GeneralError

101 – GdsUnexpectedConditioningResult

# Drift

This command performs a drift operation for the specified method on the specified sets.

## Parameters

**MethodKey**: The unique key that identifies the specific method of interest. This parameter can be replaced with the name of method in a parameter called MethodName.

## Syntax

<Drift MethodKey=”…”>

<Sets>

<Set Key=”…” />

…

</Sets>

</Drift>

<Drift MethodName=”…”>

<Sets>

<Set Key=”…” />

…

</Sets>

</Drift>

## Reply

<Drift ErrorCode=”0” ErrorMessage=”Success”/>

## Possible Error Codes

6 – Exception

11 – RequestedItemNotFound

12 - GeneralError

# Analyze a Sample

This walks through the basic series of steps to analyze a sample. The instrument must be ready to analyze.

Before accessing calculated values of the just analyzed replicate, you will want to ensure the background calculation job is complete. This may be queried through the CalculationJobActive command.

Once the sample has been analyzed and calculated, you may check with the ConditioningRequired command to see if the spectra did not match what was expected for the method.

## Steps

1. Prepare for analysis
   1. Using remote sample login command, add the new sample.
   2. Mark the replicate as the next to analyze.
2. Load Sample
   1. Follow the steps for loading a sample per Appendix B’s load sample.
   2. Note: If Remote Control AutoAnalyze is true, analysis will start when the sample is loaded.
3. Analyze
   1. Use Remote Control commands to start the analysis.
   2. Monitor until complete.
4. Unload Sample
   1. Follow the steps for unloading a sample per Appendix B’s unload sample.

## Key Internal Signals

## Switches

|  |  |  |
| --- | --- | --- |
| **Name** | **Key** | **Value** |
| Door Switch | 9151327912035622919 | Set = door fully closed |
| Door Interlock | 9151327912069185536 | Set = door mostly closed, ready to be locked and latched |
| Door Safety Interlock | 9151327912069185537 | Set = door safety interlock engaged |

## Solenoids

|  |  |  |
| --- | --- | --- |
| **Name** | **Key** | **Value** |
| Door Lock | 9151327912052400141 | Set = door lock engaged |
| Latch Release | 9151327912052400129 | Set = door unlatched |

## Ambients

|  |  |  |
| --- | --- | --- |
| **Name** | **Key** | **Value** |
| Lamp Pressure (PT21) | 9079270317897056800 | Pressure, in torr, of lamp area |

# Instrument Conditioning

This walks through the basic series of steps to condition the instrument. The instrument must be ready to analyze.

The date of the last successful conditioner may be checked with the LastConditioner command.

## Steps

1. Add and analyze warm up sets.
2. Add conditioner set using AddConditioningSamples command.
3. Analyze conditioner.
4. Calculate system registration using Condition command.
   1. Returns once the calculation complete.
   2. Check the return error code for any problems.

# Unload Sample

The series of commands to interactively unload a sample.

Step 1 of the unload process will retract the reamer assembly to allow access to the sample. Once successfully completed, the sample load state will be Unclamped.

Step 2 will release the sample from the lamp via positive pressure.

Step 3 will perform the cleaning of the lamp. The door must be closed to complete.

Capturing released sample: If you wish to capture the released sample, the mechanism can be in place prior to step 1 or 2, depending upon the mechanism and sample size. If large sample that will fall as soon as unclamped or you wish more control of the sample, prior to step 1 is probably the preferred course of action.

Notes: Use Remote Query StringValue to access Sample Load State.

## Variable to Monitor

Sample Load State – string

* Unload Sample Step 1 values
  + Unclamping
  + Unclamped
  + Operator Abort: retract reamer
* Unload Sample Step 2 values
  + Releasing
  + Released
  + Operator Abort: cleaning burst to release
* Unload Sample Step 3 values
  + Cleaning Anode
  + No Sample

## Commands and Actions

Open the door and be ready to catch the sample as it is released. (can either be done here, or prior to starting step 2)

<ExecuteSequence Sequence=”Unload Sample Step 1”/>

Monitor until unclamped or abort.

<ExecuteSequence Sequence=”Unload Sample Step 2”/>

Monitor until released or abort.

Ensure the lamp area is clear, before starting the next step. Start step 3 and close the lamp compartment door.

<ExecuteSequence Sequence=”Unload Sample Step 3”/>

Monitor until no sample.

Door latch will be released at the completion of step 3.

# Load Sample

The series of commands to interactively load a sample for analysis.

This assumes that the sample area door is open and the lamp ready to accept a sample.

Step 1 of the load process will draw the sample to the lamp by evacuating the lamp area. Once successfully completed, the sample load state will be Evacuated.

After step 1 and before step 2, remove any physical obstruction for the reamer assembly to move forward and clamp the sample.

Step 2 will clamp the sample to the lamp at low pressure.

Step 3 will wait for the door to be closed and locked, clamp the sample to the lamp at high pressure, then prepare the lamp area for sample analysis.

Note: Use Remote Query StringValue to access Sample Load State.

## Variable to Monitor

Sample Load State - string

* Load Sample Step 1 values
  + Evacuating
  + Evacuated
  + Error: pressure evacuation timeout
  + Operator Abort: evacuation
* Load Sample Step 2 values
  + Clamped – Low Pressure
* Load Sample Step 3 values
  + Clamped – High Pressure
  + Depressurizing to 0.1 torr
  + Preparing
  + Loaded
  + Error: door close timeout
  + Error: depressurizing to 0.1 torr timeout
  + Error: prepare sample
  + Operator Abort: wait for door close
  + Operator Abort: wait for door switch
  + Operator Abort: depressurizing to 0.1 torr

## Commands and Actions

Door is expected to be open and no sample on lamp.

Hold the sample in place on the lamp for loading.

<ExecuteSequence Sequence=”Load Sample Step 1”/>

Monitor until evacuated, error or aborted.

Ensure lamp area is clear, start step 2.

<ExecuteSequence Sequence=”Load Sample Step 2”/>

Close the lamp compartment door, start step 3.

<ExecuteSequence Sequence=”Load Sample Step 3”/>

Monitor until Loaded, error or aborted.

# GDS Error Code

|  |  |  |
| --- | --- | --- |
| **Error code** | NAME | DESCRIPTION |
| 101 | GdsUnexpectedConditioningResult | Unexpected conditioning result. Please confirm the conditioner is mounted and repeat analysis and conditioning calculation. |