HELIOSense USER INTERFACE
Software Reference Guide
# 1. TABLE OF CONTENTS

1. Document Conventions ........................................................................................................... 7

2. Important Safety Information .................................................................................................. 8
   2.1. General Safety Statement ................................................................................................ 8
   2.2. To Avoid Fire or Personal Injury .................................................................................. 8
   2.3. Operating Conditions: .............................................................................................. 9
   2.4. Lighting Hazard ........................................................................................................... 9

3. Use of Hyperlinks in Adobe Acrobat® ................................................................................ 10

4. Introduction ........................................................................................................................... 10

5. Installation and First Use ....................................................................................................... 11
   5.1. Installing HELIOSense .............................................................................................. 11
   5.2. First Use .................................................................................................................... 12

6. HELIOS Architecture ............................................................................................................. 14
   6.1. Architecture Overview ............................................................................................... 14
   6.2. HELIOSense Architecture - Local ............................................................................ 15
   6.3. HELIOSense Architecture - Remote ......................................................................... 16

7. Using HELIOSense ................................................................................................................. 17
   7.1. Basic User Flow .......................................................................................................... 17
   7.2. Using HELIOSense .................................................................................................... 17

8. Main User Interface ............................................................................................................... 18
   8.1. System Tree ............................................................................................................... 20
   8.2. Graph View ................................................................................................................. 21
   8.3. Graph Settings ............................................................................................................ 22
   8.4. Search Dialog (Found Devices) .................................................................................. 23
   8.5. Controlling Devices — Generic .................................................................................. 24
   8.6. Export Dialog ............................................................................................................. 25
   8.7. Export Data Settings ................................................................................................. 26

9. Isapi Location .......................................................................................................................... 27

10. State Files ............................................................................................................................. 29

11. Measurement DataBase ...................................................................................................... 30

12. Hunt and Seek .................................................................................................................... 31
   12.1.1. Simple H&S .......................................................................................................... 31
   12.1.2. Advanced H&S .................................................................................................... 31
   12.2. Simple Hunt and Seek Routine .................................................................................. 32
12.3. Advanced Hunt and Seek Routine................................................................. 34
12.3.1. Advanced H&S Targets, Steps, Detectors, Devices................................. 34
12.3.2. Advanced H&S Hierarchy and Flow ......................................................... 34
12.3.3. Advanced H&S Targets: Primary, Secondary, Tertiary............................ 35
12.3.4. Advanced H&S Steps .............................................................................. 36
12.3.5. Advanced H&S Detectors ....................................................................... 37
12.3.5.1. Special Advanced H&S Case: Spectrometer as detector .................... 37
12.3.6. Devices In Advanced H&S ................................................................. 39
12.4. Advanced Hunt and Seek: Use and Examples............................................. 39
12.4.1. Creating a new Advanced Hunt and Seek .............................................. 39
12.4.2. Adding Steps to a H&S Routing ............................................................ 40
12.4.3. Advanced H&S Example 1: Basic Lookup ........................................... 40
12.4.4. Advanced H&S Example 2: Multi-Target Seek ...................................... 42
13. Scripts ................................................................................................................. 44
13.1. Script Functions and Commands ................................................................. 44
13.2. Create or Modify a Script ........................................................................... 45
13.3. Understanding Parameters ......................................................................... 45
13.4. Variables (Variable Tags) .......................................................................... 45
14. Script Command Definitions ............................................................................. 47
14.1. LOAD ............................................................................................................ 47
14.2. VARIABLE .................................................................................................... 47
14.3. SET ............................................................................................................... 48
14.4. GET .............................................................................................................. 48
14.5. WAIT .......................................................................................................... 48
14.6. LOOP ......................................................................................................... 49

4
14.7. BREAK................................................................................................................. 49
    BREAK IF (var <|>|<=|>= value) .............................................................................. 49
    BREAK ......................................................................................................................... 49
14.8. IF....................................................................................................................... 49
    IF VAL (var <|>|<=|>= value) ................................................................................. 49
    END IF ......................................................................................................................... 49
14.9. WRITE................................................................................................................. 49
    WRITE USER (stuff) ............................................................................................... 49
    WRITE TCP (stuff) ................................................................................................. 49
    WRITE FILE (“path”, stuff) .................................................................................... 49
14.10. TCP .................................................................................................................... 50
    TCP CMD (timeout seconds, “message out”) ......................................................... 50
14.11. CATALOG........................................................................................................... 50
    CATALOG (No Parameters) ..................................................................................... 50
14.12. CONNECT .......................................................................................................... 50
    CONNECT (No Parameters) ..................................................................................... 50
14.13. QUERY.............................................................................................................. 50
    QUERY CONNECTED (No Parameters) ................................................................. 50
    QUERY TAGS (No Parameters) .............................................................................. 50
14.14. INSERT .............................................................................................................. 50
    INSERT FILE (“path”) ............................................................................................ 50
    INSERT TCP (timeout seconds, “message to send”) ............................................. 51
14.15. Special Commands .......................................................................................... 51
    Spectrometer Commands ....................................................................................... 51
    Detector Commands ............................................................................................... 51
15. Script Examples .................................................................................................... 52
15.1. Remote Control.................................................................................................. 52
15.2. Automated Test.................................................................................................. 52
16. Detectors .............................................................................................................. 54
16.1. Detector Calibrations ....................................................................................... 55
17. Filter Wheels ........................................................................................................ 56
18. Spectrometers ...................................................................................................... 57
1. DOCUMENT CONVENTIONS
This manual includes the following icon conventions:

The note icon is used to highlight important configuration information.

The caution icon is used to illustrate environments where an electrical shock hazard exists.

The heat icon indicates situations where hazardous temperatures exist.

The attention icon is used to highlight important configuration information where damage or system conflicts may occur.

The eyewear icon indicates important safety features related to the system light levels.

Table 1
SOFTWARE COMMAND CONVENTIONS

<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Bold Text</strong></td>
<td>Click commands and Buttons</td>
<td>“Click the Calibrate Now button”</td>
</tr>
<tr>
<td><strong>&gt;&gt;</strong></td>
<td>Menu paths</td>
<td>File &gt;&gt; Exit</td>
</tr>
<tr>
<td>“Quoted Italic”</td>
<td>Text from Message Windows</td>
<td>“Driver successfully communicated…”</td>
</tr>
<tr>
<td>Italic</td>
<td>Text to type into an entry location Or The name of a window or panel</td>
<td>Type, admin as the user name. Open the Create a Calibration window.</td>
</tr>
</tbody>
</table>
2. IMPORTANT SAFETY INFORMATION

Use this product only after reading the manual and the important safety information. This manual contains information and warnings that must be followed by the user for safe operation and to maintain the product in safe working condition.

2.1. General Safety Statement

Use this product only as specified. Review all safety information and precautions to avoid injury and prevent damage to the product. Carefully read all instructions and retain these instructions for future reference.

To operate this product correctly and safely, it is imperative that the operator comply with local and national safety codes. The operator should also follow generally-accepted safety procedures in addition to safety precautions specified in this manual.

This product is to be used by trained personnel only. Only operators who are aware of the hazards involved should operate this equipment. There are no serviceable parts inside the housings. Users should not open the housing to attempt any repairs.

Before use, always check that the product is installed correctly and that all electrical connections are in good working order. When incorporating this product into a larger system the safety of the system is the responsibility of the assembler of the system.

2.2. To Avoid Fire or Personal Injury

Use Proper Power Cord: Use only the power cord that is specified and certified for the country of use.

Ground the Product: This product is grounded through the grounding line in the power cord. To avoid electrical shock, the grounding conductor must be connected to earth ground. Before making connections to the input or output terminals of the product, make sure that the product is properly grounded. The system is grounded via an external ground cable that must be connected to the rear ground terminal stud on the electronics enclosure.

Power Disconnect: The power cord disconnects the product from the power source. Position the equipment so that the power cord is accessible to the user to allow for quick disconnect if possible.

Terminal Ratings: Observe all terminal ratings and markings on product. Using incorrect power/current supply may result in fire or shock hazard.

Exposing Internal Electronics: Do not remove the product cover or housing. Exposure to internal electronics may result in shock hazard. No user-serviceable parts are contained in the unit.
Proper Use and Operation Failure: Do not operate unit if it is suspected that there are electrical failures. Disable the product if it is damaged and return to Labsphere at the address provided in the user manual. The product should be clearly marked as defective after power is turned off and unit is disconnected from power source.

Electrical Fuse: The fuse rating is supplied on the rear of the unit. Only use a fuse rated for the instrument.

2.3. Operating Conditions:
- Do not operate in wet or damp conditions. This includes condensation that may occur from moving the unit from a cold to warm environment.
- Keep the product clean and dry.
- Provide proper ventilation around the unit. Be aware of the ventilation openings on the unit housing. Never cover or obstruct the ventilation openings. Do not plug or push objects into the ventilation openings.
- This product is intended to be used in a clean, indoor environment.

2.4. Lighting Hazard
This class of devices emit light at wavelengths from 350nm – 940 nm with a maximum radiance of 24 W/m²-sr. Per IEC-6247-1, this product is in the Exempt Group (no hazard) LR ≤ 28000/a [W/m²-sr] within 10s, acc. (8) LIR (low vis stimulus) ≤ 6000/a [W/m²-sr] within 1000s for retina EIR ≤ 100 [W/m²] within 1000s – for cornea.

Although this product falls into the exempt family of products for lighting hazard, it is not recommended that user look directly into the source for extended periods of time.
3. USE OF HYPERLINKS IN ADOBE ACROBAT®
There are numerous hyperlinked shortcuts in place throughout this manual that can be taken advantage of when viewing the manual as a PDF file in Adobe Acrobat. The Table of Contents and every text reference to a figure title, chapter name, heading, sub-heading, page number, etc. has been authored using cross-referenced hypertext that contains a link to the section of the manual being referenced. Clicking on this hypertext will instantly switch the manual’s page to the referred section.

1. Locate the cross-referenced hypertext by noting any sentence or call-out that contains a reference to a chapter name, section heading, figure title, etc. (i.e. “See…or Refer to…”).
2. Roll the mouse over the text and observe if the cursor changes to a finger-pointing hand as shown on the left. If it does, then the text contains a hyperlink.
3. Click on the hypertext with the finger-pointing hand cursor. The manual will instantly advance to the section that is referenced.
4. Press and hold the “Alt” key and then press the “left arrow” key to return to the original source location.

4. INTRODUCTION
This manual covers the operation of the “HELIOSense” software that is used to control Labsphere’s HELIOS hardware systems. HELIOS has been designed to be a platform for a wide variety of uniform source applications and can include a wide variety of components (detectors, power supplies, spectrometers, etc.). HELIOSense is designed to control any configuration of HELIOS hardware with a single universal interface. This manual will cover the use of this interface from a “general use” perspective as well as provide specific examples of hardware configuration and use for illustration.

At the highest level, there are two modes of operation for HELIOSense; local and remote. In local mode operation, the user application HELIOSense.exe targets the hardware that is “locally” connected (i.e., the USB connections are established directly to the PC running HELIOSense). In remote mode operation, HELIOSense controls the hardware that is physically connected to a different PC. The remote PC must be running the HELIOS APIOTW server (refer to “HELIOSense Architecture - Remote” on page 16). Remote mode communicates using standard Ethernet network hardware and protocols. The Ethernet network configuration, security, or other network administration is not addressed in this manual.

Once connected, either “locally” or “remotely”, HELIOSense is operated in the same way.
5. INSTALLATION AND FIRST USE

5.1. Installing HELIOSense

Run the supplied installer, accepting all default settings.

Figure 1: HELIOS Installer

HELIOSense uses two main locations on disk for all application files:

1. `C:\Program Files (x86)\Labsphere\HELIOSense`
   This location holds the application itself and any permanent application support files, including HELIOS-specific device drivers.

Figure 2: HELIOS Application Directory
2. C:\ProgramData\Labsphere\HELIOS
This location holds user and application data that can change. This includes the default measurement data base, HELIOS.db which can be set by the user to any location. It also includes the default location for the StateFile, HuntAndSeekFile, and ScriptFile, directories.

Figure 3: HELIOS User Data

5.2. First Use

When operating in local mode, with a device USB connection directly to a customer PC, Windows must be able to recognize all system hardware before HELIOSense will perform correctly. Some device drivers are provided during installation here:
C:\Program Files (x86)\Labsphere\HELIOSense\DeviceDrivers

All HELIOS systems come pre-configured from Labsphere. This configuration is transferred in the form of a “State File”. Refer to Section 10 “State Files” on page 29 for more information.

New HELIOSense installations may require the user to find and move or copy the system-specific state file(s).
Click the HELIOSense shortcut to launch the application. The following Windows Firewall warning may display, click all check boxes and continue.

![Windows Firewall Warning]

**Figure 4: Windows Firewall Warning**

On the first launch, HELIOSense will open a default state file and the default locations for “State Files”, “Hunt and Seek Files”, and “Script Files”.

**To open the configuration supplied by Labsphere:**

1) Go to *File >> Open State* or right-click on the device root in the item tree (showing “Default” in the figure above) and select “Open State.”

2) Navigate to the State File (configuration) provided with the HELIOS of the HELIOSense Installer and open that state.

   It is highly recommended that the core configuration provided by Labsphere is copied to the default State File location:

   `C:\ProgramData\Labsphere\HELIOS\StateFiles`

3) Use the “Save State As” function (available via *File >> Save State As* or right-click device root) to create a copy of the state in a new location.
6. HELIOS ARCHITECTURE

6.1. Architecture Overview
All HELIOS have the same basic structure as shown in the diagram below.

The HELIOSense Hardware/Software Basic Structure

Figure 6: HELIOS Architecture
6.2. HELIOSense Architecture - Local

When operated locally, the HELIOSense application is running on a PC that is directly connected to the HELIOS hardware. There are two basic ways this is achieved:

1) Running directly on an embedded “Cube” (Windows mini PC) that is inside a HELIOS rack
2) Running on a third-party (customer) PC that is connected to the HELIOS hardware with USB connections.

---

**Figure 7: HELIOSense Local on Cube**

**Figure 8: HELIOSense Local on Customer PC**
6.3. **HELIOSense Architecture - Remote**

HELIOSense can target a remote HELIOS using a system called “API Over The Wire” (APIOTW). APIOTW uses efficient and robust network protocols to create bi-direction data sockets connecting the PC running HELIOSense and a remote PC that is connected to the HELIOS hardware. The remote PC must be running the HELIOS APIOTW Server to be available for remote connection. In practice, HELIOSense remote is available on HELIOSS that come with the embedded Cube PC.

By default, the Cube runs the APIOTW Server on startup. Refer to “Appendix 1: API Over the Wire” for additional details on APIOTW.

The embedded Cube can run APIOTW Server and HELIOSense locally. In this way, a user can choose to run the HELIOS hardware using monitor and peripherals connected directly to the HELIOS rack or “Over the Wire” with a remote PC connected to the same network.

APIOTW can only be accessed by PCs that can navigate to the Cube’s IP address. This requires network connectivity and in some cases, a network administrator may require special network configurations.

While APIOTW Server and HELIOSense can run simultaneously on the same system, individual hardware components can only be utilized by one application at a time. Further, hardware cannot be transferred from one application instance to another without disconnecting (in software) and reconnecting in the new application.

![Figure 9: HELIOSense Remote](image)

The IP address of the Cube may change. This is controlled by a company/facility network administrator. In some cases, the administrator can assign a “static IP” such that the Cube IP address will always be the same value.
7. USING HELIOSENSE

7.1. Basic User Flow

All HELIOSense versions follow the same basic user flow:

1. Launch the HELIOSense application – in a typical installation HELIOSense is installed in Program Files/Labsphere/HELIOSense and by default a shortcut is created on the public desktop. If HELIOSense has already been configured (operated previously), the application will load a saved device state configuration.

2. Next the system will try to establish a local or remote connection based on the last-used location of the core HELIOS DLL (Isapi.dll). If the connection to the previously-used location is not available, the “Isapi Location Dialog” will appear. Refer to Section 9 “Isapi Location” on page 27.

3. The first action typically required is to search for devices. Refer to Section 8.4 “Search Dialog” on page 23. This is to ensure that all devices that are expected to be available are, in fact available and their individual communication methods are verified.

4. Next, connect to devices. Generally, found devices will match those in the loaded state configuration. Once connected, devices will be initialized based on the state configuration or pre-determined defaults.

5. Once connected, the system and listed devices are ready to use. Users will be able to modify device settings, change data views, and export data as needed.

7.2. Using HELIOSense

- Refer to Section 8 “Main User Interface” on page 18 for information on the main user interface and most common user interactions.
- Refer to Section 9 “Isapi Location” on page 27 for details on targeting Isapi either locally (HELIOSense Local) or remotely (HELIOSense Remote).
- Refer to Section 11 “Measurement DataBase” on page 30 for details on the measurement data database.
8. MAIN USER INTERFACE

1. Ribbon Bar
   a. Search for new devices
      i. Note: Any time a session is started or a new device plugged/powered a new “search” must be initiated!
   b. Export data from the database to .CSV file.
   c. Help view window launcher.

2. System Tree
   a. Devices in the system “state.”

   Note: Click on a device in the tree to activate device-specific user interface in the panel to the right. Use the device-specific interface for device control and settings.
Note: Double-click on a device to “pop out” the device interface in a new window!

b. State Files system/device configurations stored on disk  
   i. Double-click a state file to load and active the device configuration

c. Hunt and Seek Routines  
   i.  Settings for active feed-back control of system output

d. Scripts  
   i.  Run a set of steps (States, Hunt and Seek routines, Wait Actions, etc.)

3. Dynamic User Interface Area  
   a.  Device, State File, Hunt and Seek, and Script views

4. Graph View of Database Data

   Note: Data viewed is controlled by “Graph Settings”, results are narrowed by session and further by the “View Window” control

5. Graph and Data Control  
   a. Double click “Graph Settings” to “pop out” control in a new window  
   b. Use the “View Window” to narrow view and rescale the graph!
8.1. **System Tree**

1. **System and Device**
   a. Top Level (Root)
      Right-click for menu including changing target location (ip address).
   b. Devices (defined by “state” configuration)
      
      *Note: Click to activate device-specific interface for settings and view.*
      - Grey = NOT CONNECTED
      - Green = Connected, no error
      - Yellow = Caution (see device UI for details)
      - RED = Error (see device UI for details)

2. **State Files**
   a. Top Level (Root)
      Targets a folder accessible from the PC running HELIOSense (Windows/network settings may limit selection).
   b. State Files
      State Files hold specific system configurations (device and device settings).

3. **Hunt and Seek**
   a. Top Level (Root)
      Same limitations as State File Root.
   b. Hunt and Seek Files
      These files hold specific Hunt and Seek Routines including devise to use and flux and targets.

4. **Real Time Item or Device-Specific Data**
5. **Item-Specific Interface Panel**
8.2. **Graph View**

1. **Legend or Graph Settings View**
   - When showing plot legend click on the trace key for color and other options

2. **Graph Display**
   - Right-click for more options
   - Time Window Data (grey text)
     i. Display to indicate session and time representing data currently viewed graph

3. **View Settings**
   - Graph Settings
     i. See below

4. **View Window**
   - Min and Max are determined by the seconds of available data as determined by the “Sessions Window” setting in Graph Settings
   - When a slider is fully at min or max it will “stick”, otherwise the slider to maintain relative time position in the view (i.e. it will move as new data is acquired)

5. **Graph Axis**
   - Right-click for more options
8.3. Graph Settings

The Graph Control window may be launched into a panel in the main window just to the left of the graph display, or it may be launched into its own pop-out window (as shown below) by double-clicking the Graph Settings Button.

Figure 14: Graph and Session Settings

1. Database Items
   - Each HELIOS Device has a table in the main database.
     *Note: Right-click tables to “collapse” or “expand” all.*
   - Each recorded data element is listed under the device table.
     *Note: Generally, “timestamp” is not displayed in this view but is used when plotting data elements in the graph view.*

2. Plot Selector
   - Clicking in the “N” or “Y” in the “Plot” column toggles the selection, a “Y” signifies that the data element will be displayed in the main graph view.
   - Clicking the “P” or “S” in the “Axis” column toggle the selection, a “P” will display the plot on the Primary Y axis, an “S” will plot against the Secondary Y axis.

3. Session Window Selection
   - Each time the HELIOSense application is started a “Session ID” is created and this ID is tracked with all data collected during that session.
Note: A new “Session” may be initiated at any time, even during system operation, by right-clicking on the “DB Settings” button in the main UI.

- Session window may be selected using the slider or by typing in the “Start” and “End” fields directly.

Note: To use only the last (current) session slide both pointers to the right.

8.4. Search Dialog (Found Devices)

![Image of the found devices dialog box]

Figure 15: Found Devices

Note: When a device appears in the above list in RED this suggests an expected device was not found, is not available, or has an error present – this must be resolved and another search must be initiated to achieve consistent HELIOSense operation.
8.5. **Controlling Devices — Generic**

Most HELIOS devices have a device-specific user interface that includes a generic “Config” tab and a setting for “Polling Rate”, as detailed here using an “LPS” as an example.

![Figure 16: Common Device Configuration Settings](image)

1. **Device Name**
   - This is the name to display in the Main System Tree.
   - Name is user-settable.

2. **Device Serial Number**
   - This is generally obtained via software from the device, and is not user-settable.
   - Note: Some “dumb” devices (e.g., halogen lamps) do not have a software method to get the SN.
   - For halogen lamps the SN must be set by the user here (or will come pre-configured from Labsphere).

3. **Device Errors**
   - Device will be displayed in red in the Main System Tree.
   - Errors may be cleared by clicking the “Clear Error” button.

![Figure 17: Device Polling Rate](image)
4. **Polling Rate (ms)**
   - Setting is often located in the main “Control” tab of the device user-interface.
   - The system will try to “poll” or measure device-specific data/settings/measurement, and write these to the database.

   *There may be technical or physical limitations on how fast a device can be polled. In systems with many polling devices, Labsphere recommends setting the polling to be fast enough but not too fast for each device to conserve CPU needs.*

8.6. **Export Dialog**

Only data included in currently loaded “Sessions” are available for export. Sessions are defined by “Graph Settings” in the main interface.

![Figure 18: Export Data Dialog](image)
1. **Export Settings**  
   - Export Settings are further described in “Export Data Settings”

2. **File Path**  
   - File Path – default: `user\documents\HELIOS\HELIOS_Export`

3. **Data Output**  
   - Data viewed here will be exported exactly as shown.  
   - These data may be manually modified by the user before export.

   *Modified data are not checked for “correctness” before writing to file.*

### 8.7. **Export Data Settings**

1. **Export only the data items (plots) currently visible in the Graph View.**  
   The drop-down is populated with the items to review.

2. **Export all data items available in the database.**  
   The drop-down is populated with the items to review.  
   *Note: Using this setting may create a large data file.*

3. **Use only data that falls within the current time window in the Graph View.**

4. **Export all data that is included in the selected session.**
9. LSAPI LOCATION
This window will be shown if the last location used for lsapi is no longer available, or it may be activated anytime by tools >> Change lsapi.dll Location or right-click the system root item in tree view.

![Figure 21: lsapi Location Dialog]

1. **Local DLL Path**
   - Application and DLL are on the same machine.
   - Not required for when targeting remote APIOTW.

2. **APIOTW Server IP**
   - Not required when targeting Local lsapi.
   - Type directly into this field or select from any “found” servers in the drop-down menu.

3. **Load .h**
   - Load one or both .h files associated with local and remote DLLs.
   - Local and remote DLLs are not required to be the same version.
4. **Use the Local or Remote DLL**
   - These buttons will not be available if DLL not found or .h version mismatch is observed.

![Figure 22: Isapi Location DLL and .h Settings](image)

5. **Red Indicator - connection error**
   - Use drop-down to see available APIOTW servers.
   - Use “Refresh” option to listen for any active servers.

![Figure 23: Isapi APIOTW (Remote) Server Discovery](image)

6. **Available APIOTW Servers**
   
   *Note: If desired the server is not shown, there may be a network connection issue or that specific APIOTW server is already in use by another system.*
10. **STATE FILES**
StateFiles are xml files that define a particular state (device configuration). StateFiles may be viewed by any xml or text editor but cannot be modified outside of the HELIOSense application.

*Attempting to modify a StateFile outside of HELIOSense will result in a corrupted configuration and will not be recognized by HELIOSense.*

StateFiles have several levels of data which have different permissions for editing.

1. **Core Data - cannot be directly edited**
   Items like “device serial number” cannot be modified.

2. **Device Limits**
   Items like “max voltage” are set by Labsphere and are locked to prevent user modification and unsafe levels.

3. **Running Parameters**
   Items like “Polling Rate” and other run-time parameters can be modified at anytime.

Device Limits and some other aspects of StateFiles are locked by default. This is to help ensure that any device controlled by HELIOSense is configured correctly, either by Labsphere or with direct support from Labsphere.
11. MEASUREMENT DATABASE
HELIOSense is continuously streaming data to a measurement data database from connected devices (based on polling rate and other factors). See Section 8.5 “Controlling Devices — Generic” on page 24.

By default, this database is located here: “C:\ProgramData\Labsphere\HELIOS\HELIOS.db”. However, users may choose to create a database at any available disk or network location. The measurement database location is StateFile specific.

Different StateFiles can direct data to different databases. Changing StateFiles during runtime could change database locations without obvious indication to the user.

Figure 24: Measurement DataBase Setting
To change the location of the measurement data database, type a path in the “DB Path” field or use the browse button . The database location will be saved with the StateFile and used whenever that StateFile is active.

12. **HUNT AND SEEK**

Hunt and Seek routines may be saved and recalled. The default location for these files is: *C:\ProgramData\Labsphere\HELIOS\HuntAndSeekFiles*, or as designated by the Hunt and Seek “Root” in the item tree.

Hunt and Seek routines provide a mechanism for automated *closed-loop* feedback control. In the early version of HELIOSense Hunt and Seek, this was only available to target overall flux as understood by detector readings. Later versions of HELIOS provide a mechanism for targeting flux and CCT. CCT hunt and seek is only available for systems with more than one variable attenuator (VA) on light sources with different CCT values.

To create a new Hunt and Seek Routine, right-click the Hunt and Seek root in the View Tree and select *Add >> New H&S Routine*. Then click on the new item in the Hunt and Seek section of the tree to activate the H&S UI.

![Create a New H&S Routine](image)

**12.1.1. Simple H&S**
- Basic H&S seek routine
- Use one or more variable attenuators (VA) to modify the system output
- Based on simple proportional algorithm
- Uses other devices in their current state (i.e. will not turn a lamp on or off)

**12.1.2. Advanced H&S**
- Advanced “Lookup Table” H&S routine
- May simultaneously seek multiple, different, targets (i.e. can seek output level and color)
- Defines different “steps” with each targeting a specific output range of the primary target
- Each step defines devices and devices states (such as “Lamp A is ON”)
- When the initiated or transitioning between steps the system will set devices to their defined settings
12.2. Simple Hunt and Seek Routine

1. Detector and VA Selections
   - Detectors and VAs that are “connected” are listed in black or green and available to use
     i. Note: if a device was previously configured for the H&S routine but not currently found it will be listed in red
   - Check or uncheck devices as needed

2. Activate/Register
   - Hunt and Seek works by “registering” itself with the relevant detector
Whenever that detector is measured the H&S routine is executed (which in turn writes to the VAs)

Note: because the H&S is executed every detector measurement, cycle time is controlled by polling rate of the detector

3. **Hunt and Seek Parameters**
   - Target (same units as detector measurements)
   - Proportional Settings
     i. Note: VA moves are based on target-actual error (%) – thus a bigger proportional setting moves the VA more but could introduce overshoot!

4. **Graph Display**
   - Display here is by “record” (no timestamp)
   - Data are also saved in device records in the database
12.3. Advanced Hunt and Seek Routine

The advanced hunt and seek routine as defined in the HELIOSense application is designed to be extremely flexible. Such flexibility can have the consequence of making the advanced routine difficult to understand at first. Please review the theory and definitions described below and see the example use-cases provided in sections *******

12.3.1. Advanced H&S Targets, Steps, Detectors, Devices

1. Targets
   a. The advanced H&S routine may have up to three individual targets: values the system will try to match using the devices available at any given time.

2. Steps
   a. The advanced H&S routine may any number of steps: definition and settings for detectors and devices to operate when the primary target falls in the step’s defined range.

3. Detectors
   a. The advance H&S routine may have up to three individual detectors defined for each step: each detector provides measurement feedback that is evaluated against each target.

4. Devices
   a. The advance H&S routine may utilize most system devices (like variable attenuators, lamps, and filter wheels) and each device can have settings (such as ON/OFF or position) defined for each step.

12.3.2. Advanced H&S Hierarchy and Flow

Figure 28: Advanced H&S Interface
1. The primary target defines what step is in use
2. Each step covers a range of values
3. The step in-use defines what detectors are providing feedback
4. The feedback of each detector is evaluate against the targets to calculate error
5. The gains for each device are used along with the error calculation to determine the move of each device, within device setting ranges defined

12.3.3. **Advanced H&S Targets: Primary, Secondary, Tertiary**

There are three (3) target fields available. The top-most is the primary target and is the only target required for the H&S routine to operate. One every iteration of the H&S routine the primary target value is evaluated, if the target falls outside the given range of the current step, or if the H&S routine has just been activated, the system will evaluate all steps in the routine and chose the step that best covers the specific target value.

Target values may be changed “on-the-fly” by user interface, TCP command, or Script command.

Steps may have overlapping ranges, if the H&S routine is operating in a given step that step will be used as long as the target is still within the given range, only switching steps when the target is moved outside the step range.
Targets by themselves do not have units but by extension have the units defined by the associate detector in use.

### 12.3.4. Advanced H&S Steps

Steps are defined at the highest level by the range of the primary target covered. In many situations steps are defined in some logical order where step “1” covers the first range, step “2” the next range, and so on.

There may be any number of steps defined within the advanced H&S routine.

Steps are like entries in a lookup table: for a given range of operation (a step) the system should use devices X, Y, Z…

Steps define what detectors are used to provide the feedback against the given targets. In a typically routine the detectors defined will be the same for all steps but this is not a rule.

Note: Like targets, only one detector is required: the primary detector (top-most).

When an active H&S is executing within a given step, the primary detector is defining the routine cycle time. For example, if the SiHi detector is the primary detector for a routine, the routine will execute every time the SiHi detector has a new value.

System devices (other than detectors) are also defined for each step. Each device has basic settings available:

- **On / In Use**
  - When selected the device will be used
  - For VA devices, this allows the device to move based on the H&S error calculation(s)
  - For Lamps, this allows the lamp to be turned on (if not already) when the step is initiated

- **Gain(s)**
  - Gains are used to define how much a device might “move” based on the error calculation of the associated measurement and target
  - No entry or “0” will result in no movement based on the calculation of associated measurement and target (if any)
Gain may always be understood to be “more=more”
- For example: if the target is luminance and the device is a VA on a lamp the gain would be some positive number; more of this device leads to more luminance (positive gain)
- For example, if the target is CCT and the device is a VA on a low color temperature lamp the gain would be a negative number; more of this device generally reduces color temperature of the system (negative gain)

- Start, Min, Max
  - Defines the starting “position” of the device when initialized by a transition into that step
  - Defines the limits of “move” for the device where gain not equal to “0”

12.3.5. Advanced H&S Detectors
Detectors are generally either photodiodes or spectrometers.

Detector calibrations (both photodiode and spectrometer) are not defined by the advanced hunt and seek routine at this time and must be set outside of the execution of the hunt and seek routine.

When a selected detector is a photodiode the measurement is provided in scaled units, providing a calibration has been selected for that detector in the detector UI or by script/TCP command.

**Hunt and seek routine will execute each time the primary detector is read.** Therefore the cycle time of the H&S routine is defined by the polling rate of the primary detector.

Where more than one detector (and target) are defined each detector is evaluated on every execution of the h&S routine. If new data is available for that detector it will be used in the move calculation. When a secondary or tertiary detector does not have new data, it is not used in the calculation.

### 12.3.5.1. Special Advanced H&S Case: Spectrometer as detector
When a detector is a spectrometer, the hunt and seek routine may use one of several measurement values.
For spectrometers, the feedback item may be Luminance (cd/m²), Color Temperature (CCT) or any previously-defined “Band” data (defined using the spectrometer UI). In the example image above only one band was defined: “Test” with units of W/m²-sr defined over a range of 500-550nm.

**Special note:** basic spectrometer settings are defined by the spectrometer UI including “continuous”. Please verify settings such as “continuous” and auto integration are set before activating a h&S routine that uses a spectrometer for feedback.


1. Name
   a. Name defined by HELIOSense statefile

2. Gains
   a. Used in the proportional move algorithm defined in Figure 29: Advanced H&S Flow
   b. When “0” or undefined the gain will not be used in the calculation

3. “On in this step?”
   a. When set, this device will be used in the step
   b. For lamps, this setting will turn a lamp on or off when a step is first initialized
      i. Initialization occurs when the H&S routine is first started or target change forces a new step to be used

4. Start, Min, Max
   a. For devices “in-use” the “start” value is used on initialization of the step
   b. For devices “in-use”, where at least one gain is not equal to “0”, the min and max define the range the H&S is allowed to set the device

12.4. Advanced Hunt and Seek: Use and Examples

12.4.1. Creating a new Advanced Hunt and Seek
To create a new advanced hunt and seek select File >> New Advanced H&S or right-click on the hunt and seek root in the node tree as shown in the figure below:

![Figure 34: New Advanced H&S](image)

Click on the new H&S item created in the node tree to display the advanced h&s user interface. Observe the step editor section:
1. Define the range for the first step
2. Select the primary detector to use and any other doctors
3. Click the “Set Current Step to Current System Values”
   a. This will auto-populate the device array with all appropriate system devices and use the current settings for each
4. Modify device settings as needed

12.4.2. Adding Steps to a H&S Routing
To add new steps to a routine click the “Copy Step” button, a new step will be created with the same settings as the current step, the step editor selector will advance to the new step, edit settings as desired within the new step.

12.4.3. Advanced H&S Example 1: Basic Lookup
In this example, the advanced H&S routine will be defined similar to a basic h&s routine to use one target and will only utilize one detector for feedback. However, this advanced routine defines different lamp combination to be used for different target ranges and thus will act as a “lookup table”.
Figure 37: Advanced H&S Lookup Steps

Each step defines the same detector for feedback but over different ranges for the target different combinations of lamp are used (observe the circled settings). When this example is active setting a target will define the step that is initialized. The target may be changed while the routine is active which may, in turn, force a new step to be used and new device combinations used. See the resulting graph of measuring over time.
In this graph, all devices were off before the routine was activated at a given target (10,000 cd/m²). The Red plot shows the measurement feedback from the detector. The Blue and Yellow plots are the VA settings. The VAs react to the defined lamps warming up and quickly settle on settings the hold the output to close to the target.

In the middle of this example the target was changed “on the fly” to 20,000 cd/m². The system turned on an additional lamp, as defined in the device settings, and the VAs reacted to the changing system again to converge on the target.

12.4.4. **Advanced H&S Example 2: Multi-Target Seek**

The power of the advanced h&s routine as defined here allows for both lookup-table behavior as shown in the previous example as well as multi-dimensional “seek” wherein more than one target are simultaneously used in the error, gain, move calculation. When settings are defined appropriately this advanced functionality allows the system to control for two different things at the same time such as luminance and color.
This example is like first example (shown in Advanced H&S Example 1: Basic Lookup), however in this new example two detectors have been defined for each step and two targets are used: one for luminance (cd/m²) and one for color (CCT).

Note the gain settings, for the gains associated with color detector and target the gain sign is important! The device “VAA” in this example is associated with a HES (halogen lamp) while the device “VAD” is associated with a Plasma lamp. We know the halogen has a relatively lower color temperature and the plasma has a relatively higher. Therefore we set the halogen gain to be negative such that when our error calculation suggests the need to move the color temperature up the halogen lamp contribution is decreased. This can be observed in the following result graphs.

In this example, the system was started with all lamps off before activating the h&s routine with a target of 10,000 cd/m² and 4,000 CCT. At some point the target was changed to 20,000 cd/m² and 4,000 CCT.

Observe how the VAs act differently as they control for both luminance (in the same direction) and color (in opposite directions).

Figure 40: Advanced H&S Example 2, Luminance Results

Figure 41: Advanced H&S Example 2, CCT Results

Figure 42: Advanced H&S Example 2, Luminance and CCT
13. **SCRIPTS**

Scripts are simple commands (one per line) which can be used to control some aspects of a HELIOS system and can be used to create automated test routines. Scripts are case-insensitive but must otherwise be properly formatted to compile and execute correctly.

### 13.1. **Script Functions and Commands**

1. **LOAD** – Load and activate a system configuration or Hunt and Seek routine
2. **VAR** – Create or set a variable using an expression that can include math functions
3. **SET** – Set variable or specific device parameter (i.e. set the position of a VA)
4. **GET** – Get variable or specific device parameter (i.e. set the position of a VA)
5. **WAIT** – Wait for Time, User, or TCP input
6. **LOOP** – Define a Loop structure
7. **BREAK** – Break out of a loop or other Structure
8. **IF** – Define an If structure
9. **WRITE** – Write data to a file, user or TCP connection
10. **TCP** – Receive a command from a TCP connection
11. **CATALOG** – Catalog devices connected to the HELIOS
12. **CONNECT** – Connect to all devices in the state file and found in the connect
13. **INSERT** – Insert text of a script from file or TCP connection

A command in a script is a single line structured as follows:

```
function sub-function (parameters1, parameter2, ...)  
```

A single space is used to separate function, sub-function, and parameters. Parameters are contained in parenthesis and separated by commas.

Comments in scripts can be preceded by “//” or “\” but must be on a single line of their own.
13.2. **Create or Modify a Script**
To create a new script, right-click on the “Script Files” root in the View Tree and select *Add > New Script*. Then click on the new item to activate the Script UI.

To modify an existing script, click on the script in the View Tree to activate the script UI as shown below:

![Figure 43: Script View](image)

13.3. **Understanding Parameters**
Most functions expect specific parameters (such as “path” in a LOAD command). Some functions can have any number of parameters (such as text and variables in a WRITE command). Text parameters must be in quotes, variable tags do not have quotes. Variable tags must contain at least one character that is not a number. Numbers alone are interpreted as a static numeric value (double-precision float).

13.4. **Variables (Variable Tags)**
HELIOS devices are continuously creating and collecting data (based on their individual polling rates). Each data item is saved in the HELIOS database as well as stored in a temporary memory construct called a “current value table” (CVT). As the name implies, the CVT holds only the most recent (current) value for each data item.

Each data item in the CVT has an associated “tag” in the form `devicetype:serialnumber:item`. Whenever such a tag is part of a script command, the system will retrieve the value of the item when that command line is executed. Note, there is no way to “force” the system to update a data item before retrieving a value from the CVT; the value returned is the “most recent” but is otherwise not synchronized with the script.

To retrieve a tag from a list of currently available variables in the CVT:
1) Click the “List Vars” button.
2) Review the list of tags. Clicking on a tag automatically copies the tag to the clipboard.
3) Close the tag viewer and right-click on the script and select “Paste”.

45
Figure 44: System Variables and CVT Tags

Numeric variables can be created as part of the VAR SET or SET VAR commands. For example: “VAR SET (i=1)” has an “i” in the parameters but “i” doesn’t correlate with any known data value item. When this happens, the system will create a variable with the tag “i”. This is a global variable and will be available to any script as long as the HELIOS application doesn’t close.
14. SCRIPT COMMAND DEFINITIONS

14.1. LOAD

LOAD STATE ("path") – This is where the path specifies a pre-configured StateFile. The path can be a full path (i.e. C:\users\documents\somefile) or it can be a “relative” path where the root is understood to be (C:\ProgramData\Labsphere\HELIOS\StateFiles). When the desired file is in the default location as shown, the path is the file name only. StateFiles are always “.xml” files – this extension will be assumed when no extension is given.

“State” includes system hardware (devices) and instantaneous hardware setting. For example, if a state was saved with a lamp operating at 12V, 1.33A, these settings will be implemented when the state is loaded.

LOAD H&S ("path", Target, True/False) – This is where the path specifies an existing Hunt and Seek file. The path can be absolute or relative with root (C:\ProgramData\Labsphere\HELIOS\HuntAndSeekFiles). “Target” (optional) will override the target previously saved in the H&S file. When not specified, the original (file) target will be used. True/False (optional) will set the “active” state of the H&S routine, when not specified the system will automatically set the H&S routine to “Active”.

“Target” may be a variable as designated by a CVT variable tag. The tag must already exist in the CVT list either by creating a variable using the “VAR” command or by selecting from the CVT list of available device data.

14.2. VARIABLE

VAR SET (expression) – where expression is in the form x=stuff, where “x” is the variable to be set and “stuff” is any combination of numbers and variables with common operators (+, -, /, *). “x” should not be an existing device data item (i.e. devicetype:serialnumber:item) because that item will be overwritten by the device.

In addition to common operators, variables can be set using common math functions such as “x = sin(y)”. Please note that this is a simplistic implementation, and there can only be one number or one variable in the parenthesis of the function.

“VAR SET” and “SET VAR” are interchangeable and will both compile to VAR SET.
14.3. **SET**

**SET DEVICE (device item tag, setting)** – The device item tag is in the same form as described above and can be located using the CVT table. See Variables (Variable Tags). Script also accepts “SET DVC”, “SET ITEM”, “DEVICE SET”, “DVC SET”

Not all devices and device items can be set. The CVT list is for all variables in the system, most of which are read-only. See “Settable Device Settings” for each device type and specific items that can be set.

**SET VAR (expression)** – See “VAR SET” in Section 14.2 on page 47.

14.4. **GET**

**GET ITEM (item tag)** – where an item or variable tag is in the same form as described above and can be located using the CVT table. See Variables (Variable Tags). This also accepts “GET DEVICE”, “GET DVC”, “GET VAR”.

14.5. **WAIT**

**WAIT SECONDS (#)** – where “#” is the number of seconds to wait. A dialog will show seconds remaining. The user may skip the wait function using the red “X” (cancel) button in upper right.

**WAIT USER (timeout seconds, “Message”)** – where “Message” can be text or variables. Text is always in quotes “”. Multiple parameters are separated by commas. A dialog is presented to the user with the evaluated message. The user must press “Continue” or “Terminate Script” before any further action is taken.

**WAIT TCP (timeout seconds, “Message to Match”)** – where “Message to Match” is some text sting (no variables allowed in WAIT TCP). When this step is executed a dialog is presented to show the user this function is waiting for the specified command to be observed on the TCP connection. An active TCP connection must already be established. The user can skip this function using the “Continue” or “Terminate” buttons provided.

“timeout seconds” must not be “0” but can be set to “-1” to signify never timeout.
14.6. **LOOP**

**LOOP BEGIN** – no parameters evaluated. LOOP BEGIN is a simple placeholder for the starting point of a loop.

**LOOP END** – no parameters evaluated. LOOP END is a placeholder for the end of a loop. If the script reaches LOOP END it will return to the next previous LOOP BEGIN line. The functions “BEAK” or “BREAK IF” are the only way to “skip” ahead of the LOOP END line.

Nested loops are supported.

14.7. **BREAK**

**BREAK IF (var <|>|<=|>= value)** – where “var” can be any variable and “value” can be static number or a variable. The only operators this function understands are “>”, “<”, “>=”, “<=”, or “=”. When the condition is met this function skips all subsequent lines until it reaches the “LOOP END” line associated with the current loop structure. Note: “END IF” is interchangeable.

**BREAK** – when no sub-function is specified “break” will act as a direct break without a test.

14.8. **IF**

**IF VAL (var <|>|<=|>= value)** – where “var” can be any variable and “value” can be static number or a variable. The only operators this function understands are “>”, “<”, “>=”, “<=”, or “=”.

**END IF** – no parameters evaluated. END IF is a placeholder for the end of an IF structure.

14.9. **WRITE**

**WRITE USER (stuff)** – Where “stuff” can be any text in “”, and/or variables designated by tag, all separated by comma “,”.

**WRITE TCP (stuff)** – Where “stuff” can be any text in “”, and/or variables designated by tag, all separated by comma “,”.

**WRITE FILE (“path”, stuff)** – Where “path” can be an absolute or relative path with default data location as root: C:\Users\user\Documents\HELIOS\ScriptOutputFiles. Where “stuff” can be any text in “”, and/or variables designated by tag, all separated by comma “,”.
14.10. **TCP**  
**TCP CMD (timeout seconds, “message out”)** – When executed, the string in “message out” will be sent to the TCP connection and a listener will launch to monitor the TCP connection for any incoming message starting with “HELIOS” (case insensitive).

Incoming messages will be evaluated as a single command/function and must conform to the structure and syntax required for the desired command/function. For example, an incoming message could be: “HELIOS var set (i=1)” – when received the system would close the current TCP listener, set a variable “i” to “1”, and then move to the next line in the operating script.

14.11. **CATALOG**  
**CATALOG (No Parameters)** – This command will start a system device catalog, devices that are found in the search will be checked against the devices that are in the currently-loaded state file, missing devices will be listed in the log. Script also accepts “CAT”.

14.12. **CONNECT**  
**CONNECT (No Parameters)** – This command will attempt to connect (or reconnect) to all devices in the currently-loaded state file. Devices must have been previously “found” in a catalog either by the above command or through the UI.

14.13. **QUERY**  
**QUERY CONNECTED (No Parameters)** – This command will return to log and TCP a CSV list of connected devices.

**QUERY TAGS (No Parameters)** – This command will return to log and TCP a CSV list of device item tags.

14.14. **INSERT**

When executed, “insert” takes a string (from file or TCP) and evaluates it as a script. The steps created from the evaluated string are inserted into the currently operating script just after the “insert” line.

**INSERT FILE (“path”)** – where path specifies a script text file. Path can be a full path (i.e. C:\users\documents\somefile) or it can be a “relative” path where the root is understood to be (C:\ProgramData\Labsphere\HELIOS\ScriptFiles). When the desired file is in the default location as shown, path is the file name only. ScriptFiles are always “.txt” files – this extension will be assumed when no extension is given.
**INSERT TCP (timeout seconds, “message to send”)** – where the message to send is text and/or variables similar to “write” commands. When executed, the INSERT TCP command will evaluate and send the “message to send” to the TCP port then wait for an appropriate incoming message (string) from the TCP port.

INSERT TCP will wait for a message with the text “script” as the first characters in the message. The rest of the message is evaluated as a script and must conform to script syntax.

### 14.15. Special Commands

Some commands are implemented that fall outside the above definitions and format.

**Spectrometer Commands** – certain items for spectrometer control can be set using the “SET DVC” format but are not listed as items/variables in the standard way.

Use the same format for the item (*devicetype:serialnumber:item*) with these special items:
- **scan** – set an item scan to a “1” to start an immediate scan
- **continuous** – set to “1” to start continuous scanning, “0” to stop

**Detector Commands** – certain items for detector control can be set using the “SET DVC” format
- **last_zero** – sending anything to this item initiates a zeroing operation
15. SCRIPT EXAMPLES

15.1. Remote Control

```
1 Loop Begin
2   TCP CMD (30, "Waiting For Command")
3 Loop End
```

Review the following explanation of each line shown:

1) The starting point of a loop.
2) Send “Waiting For Command” to TCP connection and launch a listener with a timeout of 30 seconds. If a valid command is received it will be evaluated and executed immediately.
3) The end point of a loop. If the script is still running, this will loop back to the beginning and another command can be received and executed.

15.2. Automated Test

Review the following code block for example:

```
Load State ("PlasmaOn")
Var Set (i=0)
Loop Begin
   Wait Seconds (2)
   Var Set (i=i+1)
   End If (i >= 50)
Loop End
Wait TCP ("Start Test")
Load State ("PlasmaStable")
Load HSS ("PlasmaStabilized", 0.0001, T)
Write User ("Now in Stable Mode")
```

Figure 45: Script Example Code Block

Review the following explanation of each line shown:

1) Loads a state called “PlasmaOn”. This is clearly not a full path so the system converts it from a relative path to a full path as follows: C:\ProgramData\Labsphere\HELIOS\StateFiles\PlasmaOn.xml
2) Set a variable “i” to “0”. The system will first check to see if there is a variable “i” already and if not, create it before setting the value to “0”.
3) The starting point of a loop.
5) Wait for 2 seconds.
7) Write a line in the file “TestData.txt”, this is a relative path so the system converts to the full path: C:\Users\user\Documents\HELIOS\ScriptOutputFiles\TestData.txt. The message written will be something like: “2015-07-27 -- 16:04:14.603 --- Detector Value: 1.234 Timestamp 123465897”.
9) Write a message to the TCP port, if the port is not active, nothing happens, if active the message will be written. The message is the same basic message as written to file.
11) Index the variable “i” by 1. Note that the system does not understand the commonly used short-hand “i++”.
12) Evaluate the statement is “i” greater than or equal to 50?
   a. If TRUE skip past the Loop End
   b. If FALSE continue to next line
13) The end point of a loop, if reached (i.e. not skipped by an END IF statement) go back to the beginning of the loop.
15) Observe ALL incoming messages on the TCP port, if a message is received that matches the designated text, move on. This function allows a user to skip forward using a “Continue” button.
17) Load a state “PlasmaStable”. Note that this state was pre-configured to move the plasma lamp into “stable mode”.
18) Load and start a Hunt and Seek routine with a target of 0.0001 amps. Note that the H&S file already exists and configured properly, the only control scripts have over H&S is dynamically setting a target.
20) Write a final message to the user.
16. DETECTORS

Users can define calibrations to scale detector measurements. When used both the raw detector values (amps) and the scaled values (units) will be stored in the HELIOS database.

Note: Filter Wheels can be configured to “notify” system detectors when a filter is changed. If filter wheel position name matches a detector calibration that value will be used.

Note: Some detectors have mechanical slides with pinhole or photopic filters installed. The user must manually choose the correct calibration when this position is changed.

Figure 46: Detector Control

1. Detector Control
   - Averages – set the value, in seconds, of data that will be averaged for each detector measurement
   - Zero Detector – start a detector zero routine, depending on the detector this can take >1min

2. Calibration Results and Control
   - Calibration Name, Factor, and Units
   - Available Cals – provides a list of calibrations associated with the detector set in the calibrations tab

3. Calibrations in System (all detectors are shown)
   C:\ProgramData\Labsphere\HELIOS\HELIOS_Detector_Cals.xml
4. **Create or Modify Calibrations**
   a. Click on an existing cal to view, modify, or delete a calibration.
   b. Simple Scalar: Enter a single value to be used for all detector amps measured.
   c. Lookup Table: Enter a set of amp:factor values – the system will interpolate values as needed (out of bound values will not be interpolated).

![Figure 47: Scalar and Lookup Table Calibrations Examples](image)

16.1. **Detector Calibrations**

To add or modify a calibration for a specific detector perform the following steps:

1. Click on the “Calibrations” tab to see currently loaded calibrations.

   ![Note: calibrations are given a name – there may be more than one calibration with the same name for the same detector, but only the most recent will be used](image)

2. Click on an existing calibration in the list to modify, or click “New” to create a new line.
3. Create new calibration entries for each filter setting as defined by the system and calibration certifications provided by Labsphere. Name each calibration entry based on the filter associated with the calibration.

   ![Note: Filter name and calibration name must match exactly when the system includes a filter wheel](image)

4. Click “Save” when finished adding or modifying calibration entries.
5. Click “Delete” to remove unwanted entries. Calibrations are saved here: “ProgramData\Labsphere\HELIOS\HELIOS_Detector_Cals.xml”
6. Calibrations will be loaded whenever the filter wheel position is set.
7. Currently loaded calibration will be shown in the detector UI.
8. When there is a valid calibration, scaled detector data will be saved in the DB.
17. FILTER WHEELS

Figure 48: Filter Wheel Control and Settings

Filter Wheels can be configured to “notify” system detectors when a filter is changed. If the filter wheel position name matches a detector calibration, that value will be used.

1. **Filter Wheel Control**
   - Position (1-indexed)
   - Reset and Home – reset the filter wheel
   - Current Filter Name

2. **Filter/Calibration Names**
   - Name Filters – according to position

3. **Associated Devices**
   - System Detectors are listed. When checked, each will be “notified” when the filter position is set. If the position name matches a calibration, that calibration will be automatically loaded.
18. SPECTROMETERS

Spectrometer operation requires significant CPU resources, polling/continuous scanning should be used carefully. Consider a long “Polling Rate” if using continuous scanning.

Auto Integration acts like a blocking call to the API – Auto Integration can be used with continuous scanning but such use is not recommended.

18.1. Dark Correction

All photo-sensitive sensors have some inherent “dark current” that creates a real and measurable signal even when no light is hitting the sensor. Each pixel on a sensor array may have a different dark current. “Dark” readings change based on sensor integration time and is also related to the temperature of the sensor. Generally speaking, dark readings increase with the temperature of the sensor.

To get an accurate measurement during operation, the dark current must be understood and subtracted from the spectrometer’s signal before other analysis can be performed (candela/meter², color, etc.).

Some spectrometers have an internal (mechanical) shutter while others do not.

18.1.1. Spectrometers with a Shutter

When a spectrometer has an internal shutter this, shutter may be programmatically closed and a measurement performed with no light reaching the sensor. This measurement is used to understand and offset the dark current across the sensor. Typically this measurement is performed periodically (not every measurement) and whenever the integration time of the spectrometer changes.

18.1.2. Spectrometers without a Shutter

For the “shutter-less” spectrometers the dark offset correction becomes more involved. For these spectrometers, Labsphere employs a technique called “Baseline Offset Correction” (BOC).

18.1.3. Baseline Offset Correction (BOC)

A BOC is a technique that starts by creating an array of measurements that are performed when the equipment is in a “dark state.” For example, BOC data may be acquired when the integrating sphere is closed and no light source is present in the sphere.

BOC data is an array of measurements performed at various representative integration times. These times are chosen by Labsphere and are picked to cover the entire range of integration times of which a specific spectrometer is capable.
When the system performs the dark offset correction using a shutter-less spectrometer, the BOC data array is analyzed and interpolated as needed (to match the actual integration time), and a “virtual” dark signal is determined from those data.

Because dark current is also dependent on sensor temperature, an additional correction is required. This correction utilizes a set of sensor pixels that have been permanently (internally) masked. These “dark pixels” are not typically returned in normal measurements but may be evaluated to understand the “scale” of the dark current while the spectrometer is operating. Using data from these dark pixels the BOC function can scale virtual dark offset data as needed.

Given that shutter-less spectrometers must employ this BOC calculation and use predetermined data, the resulting dark offset correction is inherently less accurate as compared to corrections made with internally shuttered spectrometer.

![BOC Check](image)

**Figure 49: Baseline Offset Correction Dialog**

18.1.4. **Maximize BOC Accuracy**

One way to maximize the accuracy of a BOC is to take readings for the baseline offset data array just before important measurements and to make sure that these readings are performed when the system is in a normal operating state and is fully temperature-stabilized.

One way to ensure stability is to turn on the system and wait for 30-60 minutes. This will allow the electrical components in the spectrometer to thermally stabilize.

HELIOS can collect new baseline offset data on-demand by clicking the update button. On-screen instructions will direct the user to ensure no light is present in the system for the entire time the data are being recorded.
18.2. Spectrometer Controls
The checkbox Take Dark Scans with DUT Scans should generally be used. This setting will force the system to take a new “dark scan” whenever the integration time changes to ensure a recent and valid dark correction can be performed. For spectrometers that have internal shutters such as the CDS30x0, this will force a scan with the shutter closed. For spectrometers without an internal shutter such as the CDS610, the system will use a calculation based on the Baseline Offset Correction data.

![Figure 50: Spectrometer Controls]

Single and Continuous Scanning – Clicking “Single” will initiate a single scan, the SPD and calculations fields will update when the scan returns. “Cont.” will latch “on” and will try to take scans based on the spectrometer polling rate set in the “Config” tab.

The indicator between “Sinlge” and “Cont.” will green when new data have been returned. When performing a Single scan the indicator will go dark while the scan is being acquired. When continuous scanning the indicator will flash whenever new data are returned.

Record Scans – When latched, all scan data including full spectral power data will be saved to the measurement database. When not latched, the spectral data will not be saved but other measurement data (i.e. CCT) will still be saved for each scan.

Integration Time – This is how long the spectrometer “integrates” the light hitting its sensor. This can be set manually, or the user may click the Auto-exposure button to have Integral calculate the best integration time.

Averages – This is the number of scans that are averaged for the final data set. This is useful for reducing noise in spectral results.
**Filter** – This is only available for spectrometers with integrated filters such as the CDS-30x0 family. This may be set manually, or can be determined from the auto-exposure feature.

**Auto Integrate** – To achieve accurate measurements, it is important that the spectrometer operates in a range that best utilizes the dynamic range of its internal sensor. The auto-exposure feature runs the spectrometer through a range of integration times to achieve maximum saturation of 70-85%, which is optimal for most spectrometers.

**Display Units** – Select the desired radiometric display units for the “Y” axis of the graph. This setting will be stored in the StateFile and used during export functions.

Display Units will also define the units of scan data stored in the measurement DB.

### 18.3. Spectrometer Calibration

Spectrometer calibrations are stored in the database associated with the core lsapi.dll (i.e., calibrations are not stored or managed by the HELIOS-specific database).

Calibrations involve several steps to correctly and accurately acquire the data required. Therefore, a semi-automated “Calibration Wizard” is provided to lead the user through the step of creating a calibration. To get started, click on the UI of an active and connected spectrometer, and then select “Create New” from the calibration drop-down as shown below.

![Figure 51: Create a New Calibration](image)

Follow the on-screen instructions. In the current revision of HELIOSense, the calibration wizard only controls the spectrometer itself. This is different from other Labsphere product platforms, such as Integral LM, where the calibration wizard may control power supplies or other devices in addition to the spectrometer.

In other Labsphere platforms (such as Integral LM), there is a concept of “Auxiliary Lamp Correction” which does not apply to the more remote-sensing applications. In the early implementation of the calibration wizard in HELIOSense, there are some references to “Auxiliary Lamp” which should be ignored.
1. **Power Supplies** – Use “Manual” for HELIOS Calibrations  
2. **Install a New Calibration Lamp File**  

   The file must be provided by Labsphere and in the correct format and data units for use by HELIOS systems.

3. **Run Settings (explained)**  
   - Stability Threshold - used to evaluate lamp stability after warmup time – will automatically advance calibration when met.  
   - Time for Averages – Scan integration time is automatically determined, use the “Time” setting to specific total integration in cal scan.  
   - View Window – time for the data history as displayed in chart on cal running screen.

4. **Calibration Running**  
   - Calibration will automatically proceed to take calibration scan averages when the warmup time is complete and the “Stability Threshold” is met.  
   - Users may press “It’s Stable – Take Readings NOW” at any time to proceed to the next step – taking averages for calibration scan.
18.4. Stray Light Factors

Labsphere spectrometers can include correction for “stray light”. Stray light is light that is “misplaced” inside the spectrometer (for example some “red” light ends up in “blue” pixel bins). Labsphere can characterize an individual spectrometer to correct for these errors. Note that the CDS30x0 family of spectrometer handle stray light correction internally and do not need an additional correction factor applied.

Spectrometers that include stray light correction factors are:
- CDS-600
- CDS-610
- CDS-1100
- CDS-2100
- CDS-2600

The stray light correction factor is included with the spectrometer when shipped from the Labsphere factory. Please contact a Labsphere sales representative for help locating the stray light correction factor.

This factor must be set for each spectrometer in the HELIOSense UI by going to: *System >> Install New StrayLight*
and following the steps provided.

![Figure 53: Adding Stray Light Factors](image)

1. **Available Spectrometers**
   - A list of currently connected spectrometers (in main application instance).
   - New Stray Light Factors can only be added to connected spectrometers.

2. **Stray Light Factor to Add**
   a. Must be a number between 0 and 1.
   b. Enter the number and press “Add Factor”.
   c. The factor should now appear at the top of the list.
3. **Factors for Selected Spectrometer**
   Spectrometers can have more than one factor in the database but **only the most recent factor will be used** - the most recent factor is listed on top.

   The required Stray Light Factors is often included in paper or electronic documentation included with Labsphere spectrometers. This factor may be included in the TOCS.ini or SMS.ini file.

18.5. **Install CDS30x0 Wavecal File**
   The CDS30x0 family of spectrometers cannot operate without an individually-created “wavecal” file. This file may be obtained from Labsphere and is shipped with any CDS30x0 spectrometer.

   To install a wavecal file navigate to:
   *System >> Install MCPD WaveCal.*
   This will open a Windows file browser. Navigate to the Labsphere-provided wavecal file associated with the specific spectrometer serial number. The file will have a format similar to: “MCPD-9800_0029.txt.”
19. **POWER SUPPLIES**

Power supplies like the LPS-400, can be both stand-alone devices as well as a “parent” device to “child” lamps (lamps may be halogen, plasma, or xenon). In most configurations, the power-supply acts as a slave to the child lamp. For example, a halogen lamp is turned ON – this command is carried up to the parent LPS which is instructed to provide the appropriate ramp/power output.

Power supplies, when acting without a child lamp, have limited control and do not have a mechanism for automated ramping.
20. **HALOGEN LAMPS**

Halogen lamps are not a “smart” device – meaning that they cannot self-identify.

*Given that halogen lamps cannot track their own operating time, this value (total operation time) is evaluated from data in the HELIOS.db which is local to the PC running HELIOSense – if a lamp is operated from more than one PC, total time will be inaccurate (accrued in two different databases).*

---

**Figure 54: Halogen Lamp Operator Control**

**Figure 55: Halogen Lamp Configuration Limits**
1. **Output Enable (On/Off)**
   - Transition from Off to On will always ramp.
   - Transition from On to Off will not ramp.

2. **Settings**
   - Settings cannot exceed the limits set in Configuration (item 4 below).
   - Current (A) value changes will ramp.
   - Voltage (V) value changes will not ramp.

3. **Measured Values**
   - Measured values are passed “down” from the parent LPS.

4. **Limits**

   ![Notebook](image)

   Limits are set by Labsphere and locked in normal use by default.
21. LED LAMPS

LED Lamps are very similar in operation to Halogen Lamps. LED lamps are not a “smart” device – meaning that they cannot self-identify.

22. XENON LAMPS

Xenon Lamps are identified by serial number. The only control for Xenon Lamps is on/off.

23. EXPORTING DATA

Data may be exported from the database at any time by pressing the Export button in the main user interface screen. See Sections 8.6 “Export Dialog” and 8.7 “Export Data Settings” on page 26 for more details on using the export dialog.

Selected data are copied out of the HELIOS database into a text file in “.csv” format (Note: the core HELIOS database is located here: C:\ProgramData\Labsphere\HELIOS\HELIOS.db). The files are saved by default to the user “My Documents” location with a date-coded file name, for example: C:\Users\user\Documents\HELIOS\HELIOS_Export\20150622_001HELIOS_Export.csv.

These export files can typically be opened by a spreadsheet application like Microsoft Excel.

Data are time-stamped in the database using “Unix seconds” for convenience. These are converted to “excel days” in the export. To get Excel to display the timestamp, those data will need to be formatted by right-clicking on the column and selecting “Format Cells.” A custom format may be required such as “d/m/yyyy h:mm:ss.000” (displays date with decimal seconds included).
Figure 56: Format Cells for Excel Viewing
24. FILE MENU

Figure 57: HELIOSense File Menu

24.1. Unlock StateFile
StateFiles, as described in Section 10 “State Files” on page 29, contain important device limit settings that are “locked” by default. To unlock a StateFile, a password is required from Labsphere.

Adding devices to a StateFile also requires the unlock password given that adding a new device suggests that new device needs initial device limit settings to be implemented.

Do not add new devices to a HELIOS system without direct support from Labsphere.

25. TOOLS MENU

Figure 58: HELIOSense Tools Menu

25.1. System Snap Shot
Creates or appends to a “snap shot” file:

user documents\HELIOS\HELIOS_SnapShots\date_HELIOS_SnapShot.csv

Each line in this file is a time-stamped record of all system settings and measurement values available (data elements are similar to those view in “List Available Device Item Names”).
25.2. **List Available Device Item Names**
Generates the following dialog listing all data item tags and current variables. Clicking on an item stores that tag on the user clipboard. “Refresh” will refresh item values.

![HELIOSense Data Item List](image)

25.3. **New Session**
Starts a new “Session”. All data items stored in the HELIOSense measurement DB are tagged with a “session” value to allow for data base analysis and grouping across different measurement types collected during the same “session”. Starting a new session can be useful when starting a new test as a way of easily differentiating the data that will be stored during the new test.

25.4. **Turn Off All Lamps**
The item will instruct all HELIOS devices to immediately transition to a “safe” or “de-energized” state.
26. SYSTEM MENU

26.1. Change Isapi.dll Location
See Section 9 “Isapi Location” on page 27.

26.2. Change Measurement Data Database Location
HELIOSense is continuously streaming data to a measurement data database from connected devices based on polling rate and other factors. See Section 8.5 “Controlling Devices — Generic” on page 24.

By default, this database is located here:
C:\ProgramData\Labsphere\HELIOS\HELIOS.db
However, users may choose to use a database at any available disk or network location. The measurement database location is StateFile specific and may be changed here through the “System” menu if desired.

See Section 11 “Measurement DataBase” on page 30 for more information.

Different StateFiles can direct data to different databases. Changing StateFiles during runtime could change database locations without obvious indication to the user.

26.3. Install New StrayLight Factor
Stray light factors are provided by Labsphere for every Labsphere spectrometer. A stray light factor must be installed on any PC directly connected to the spectrometer.

26.4. Install MCPD Wavecal
A special “wavecal” file is required to operate the CDS-30x0 series spectrometers. This file is available from Labsphere and provided with the CDS-30x0 spectrometer. The wavecal file must be installed into the system files of any PC that is directly connected to the CDS-30x0 spectrometer.
26.5. **Demo Hardware**

Demo Hardware – Install the “demo hardware” (set of virtual devices that can be found and connected to for demonstration purposes)

Setting is persistent (when checked the demo hardware will be automatically installed on each startup)

26.6. **TCP Listener Active on Startup**

The TCP Listener can be activated through the scripting interface or may be set to be active anytime HELIOSense is operating. The TCP interface provides remote control into the HELIOSense application as described in Section 13 “Scripts” on page 44.
27. TROUBLESHOOTING

Complete troubleshooting directions are not yet determined. Most issues can be resolved by restarting the system.

For issues regarding a specific device please try to “connect” to the device then click on the device and the device UI “config” tab. There is a field for “error” in the config tab that may provide more information.
APPENDIX 1:  API OVER THE WIRE

APIOTW Implementation
API Over the Wire (APIOTW) is implemented in HELIOS systems in a LabVIEW layer of code. This is important to note because APIOTW is available in HELIOSense applications as well as in the HELIOS API/SDK in LabVIEW, but is not provided by default in any other programming platform. However, the technique described below is likely possible in any number of programming environments.

In its most basic form APIOTW is implemented as a part of a generalized call to the core of all Labsphere systems; the lsapi.dll. This generalized call takes all calls to lsapi and turns these calls into a byte stream. This is true for both local and APIOTW calls.

For example, the call “lsapi_ApiCall_Api_Open” is a defined/populated C-Structure which in memory is the following bytes: “18, 0, 0, 0, 1, 0, 0, 0, C, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0”. A similar understanding is true for most calls. By understanding that calls are just bytes, these bytes can be directed to a local lsapi.dll or to a remote lsapi.dll by way of bi-directional network sockets.

The APIOTW feature essentially creates a network link “in between” the formation of the byte stream and the API call itself. Instead of passing these bytes directly into the LabVIEW Call Library Function Node (CLFN), the bytes are passed into an “outgoing” LabVIEW Network Stream. The other end of that link is a LabVIEW-based “server” operating on the computer. The server computer must also be the machine that is connected to the system hardware. The server receives the message, passes the API function call bytes into the local DLL (via a LV CLFN), and returns the results.
APIOTW is further described in the diagram below:

Header File Required
Under the hood, using the lsapi in LabVIEW requires a complete understanding of the function calls provided by lsapi_win32.dll. This is true in both the “local” and “remote” (APIOTW) conditions. This is required for the LabVIEW Wrappers to be able to properly format the data needed for the many lsapi function calls.

The only way the LabVIEW layer can do this is by parsing and “caching” the information provided in the header file provided with the DLL. This header file is “lsapi.h”. The header file used and the target DLL (local or remote) must be matched or some function calls could fail. In principle, this means that the header and DLL are the same “version” and were created at the same time by Labsphere.

APIOTW Server: Network Streams
As previously stated the APIOTW is built on LabVIEW Network Streams. Network Streams are one-way and therefore, the APIOTW must have both a “reader” and a “writer” stream. These are established by the server. In practice, the fact that there are two streams is abstracted to the user by a pre-defined naming convention.

However, before any network stream is usable one side of the link must be able to find the other.
When launched, the server creates a stream endpoint which is then available to a client at the server IP address and server name. Now, the client must create the other end of the link by specifying this exact address.

**APIOTW Server: UDP Broadcast for Notification**

While the server is idle (not actively connected to a client) the server broadcasts its IP and Name via a UDP message to the entire network every second or so. Client code operating on the same network can receive this UDP message when searching for a server. Once connected, the server stops broadcasting UDP and any other clients will not be able to see that server. In practice, there is usually only one server or client operating.

**APIOTW Server**

The PC hosting the API targeted by APIOTW must be running the APIOTW Server.

![APIOTW Server](image)

**Figure 62: APIOTW Server**

1) The only required setting for the APIOTW server is the path to the local DLL.
   - This path will be remembered the next time APIOTW Server is launched.
2) Error and warning messages are displayed.
   - Error Message: “cannot find the dll” – the path to the local DLL is not correct.
   - Error Message: cannot “Create Network Stream” – the server is waiting for a connection by a client; this message is normal when the server is idle.
3) Additional Instructions
APIOTW: Windows Firewall Issues

Windows Firewall (if enabled) will likely block some or all of the network traffic created by both the APIOTW server and client code.

The first time a new application (client or server) is launched Windows will likely display a warning similar to below. Please enable on all network “types” that this connection will be used with (check all boxes if unsure).

![APIOTW Firewall Warning](image)

Figure 63: APIOTW Firewall Warnings

Windows Firewall Exceptions

There are several other services, ports, and applications that must also be allowed through the Firewall for the APIOTW architecture to operate. These can be enabled by creating exceptions for some additional programs using these steps:

1) Navigate to: Start Menu >> Control Panel >> Windows Firewall >> Allow a Program or Feature…

2) Change Settings >> Allow another program

3) Find and add the APIOTW Server executable's name

4) Add C:\Windows\System32\lkads.exe.
   Note that for Windows 7 64 bit, this file will be found in:
   C:\Windows\SysWOW64\lkads.exe.

Also add the following:
C:\Windows\system32\lkads.exe
C:\Windows\system32\lktsrv.exe
C:\Program Files\National Instruments\Shared\Tagger\tagsrv.exe