General Improvements in GainSeeker versions 8.5 and 8.5.2

General Improvements in GainSeeker version 8.5.2

Multiple Modules

- A new **Search** field makes it easier to find a specific defect in a long list.
- A new System-wide setting lets you configure e-mail with SSL (Secure Sockets Layer). This makes it possible for GainSeeker to send e-mail via systems such as Gmail and Yahoo mail.

GainSeeker Charts

- On Pareto charts of defect counts (*Defects as % of total Defects*, *Defects as % of total Sample size*, and *Defects per Unit*), the top axis displayed one or more decimal places. These decimal places are no longer displayed on the top axis for these three charts, because this axis will always represent a count (which can only be a whole number).
- On a dashboard with the Dashboard Designer **Caption** property left blank, the dashboard name is now displayed (on the dashboard tab, at the top of the dashboard window, etc.) instead of the generic label "Dashboard".
- When using command line parameters to automatically email a desktop, the email subject line will be the desktop name.
  When using command line parameters to automatically email a dashboard, the email subject line will be the dashboard caption.
  When using command line parameters to automatically email charts, the email subject line will be the chart window caption for the last chart drawn.

GainSeeker Inspections

- For a Numeric Input test, a new **NoValue** property (and equivalent Python command `inspect.cursubi.ni(`*`idx`*`).novaluemode`) lets you configure the test to be required or optional.
- Step 2 of the Planned Inspection editor has been is now more intuitive when choosing the DMS Process and Part Numbers. The Process is now displayed in all cases, even when using the same Process as the underlying Inspection, and a **Reset** button makes it easy to change back to the Process from the underlying Inspection.
- A new **Search** field makes it easier to find a specific Inspection or Planned Inspection in a long list.
- The Planned Inspection editor no longer allows you to set traceability values for the two fields designated in the Inspection as "sub-inspection" and "test", because the Inspection stores information in these fields.

PC Collect

- For the Numeric Input test property **Label**, PC Collect will now display only the first 43 characters of this Label. This ensures that the displayed Label will be legible to users. To display longer text or instructions, add a Text test above the Numeric Input test.
- A new **Search** field makes it easier to find a specific Inspection or Planned Inspection in a long list.
- When using Active Directory Integration, a Checkbox test with Electronic Signature set to the user name and password will require the Active Directory password instead of the GainSeeker password.

GainSeeker Utility

- In the SQL Utility, you can now right-click the results grid to export its contents to a text file.
- Exporting data to XML now encodes the date format from your Windows Regional Settings, which allows someone with a different date format to import this data correctly.

Python

- The new `inspect.cursubi.ni(idx).label` command gets or sets the **Label** property of a Numeric Input test.
- The new `inspect.cursubi.selectsidepanelitem(`*`item`*`)` command lets you bring a particular document or image to the front.
- Three new commands make it easy for one Python script to loop through all of the tests in a sub-inspection and perform an action (such as sending e-mail) if there were failures:
  - `inspect.cursubi.testidlist` returns a list of the TestIDs in the current sub-inspection.
  - `inspect.cursubi.gettesttype(`*`idx`*`)` returns the type of test that corresponds to the specified TestID.
  - `inspect.cursubi.ni(`*`idx`*`).rtfvalue` identifies all real-time failures that occurred for the specified Numeric Input test.
- The `inspect.cursubi.ni(`*`idx`*`).visible` command now allows you to display or hide a numeric input test.
- The `inspect.setnextinspection` command now lets you jump to a Planned Inspection after the current sub-inspection is Submitted.
- For Inspections, a **Used by** button has been added to the **Python Design Selector** window. You can now select a Python Design and then click **Used by** to find out whether that Design is used by any Inspections.
- For Dashboards, a **Used by** button has been added to the **Python Design Selector** window. You can now select a Python Design and then click **Used by** to find out whether that Design is used by any Dashboards.
- Help topics for these command groups were updated for clarity: `datadms`, `dataspc`, `stdspc`, `inspect`

SPC Designer

- A new TRACE_OPT 5 command lets you configure a traceability column to display a list of values from a string array instead of displaying the full traceability list or the traceability priority list.

Online User Guide

- The topic Data entered with different subgroup sizes for one standard now describes how different GainSeeker modules will analyze a data record where the count of numeric data values is greater than its subgroup size.

Major new features in GainSeeker version 8.5

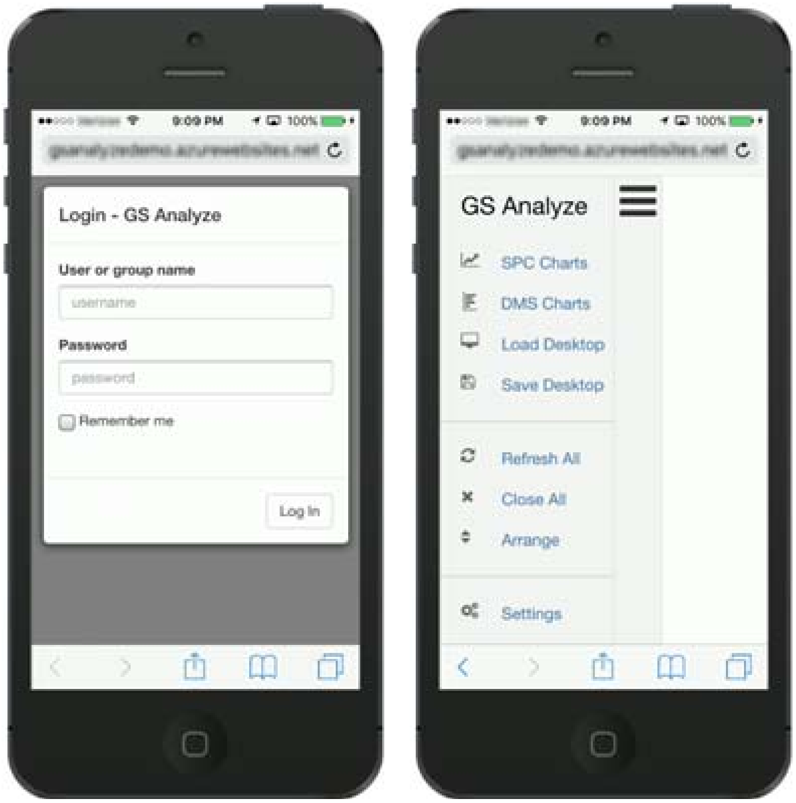
GS Analyze: The new face of Mobile GainSeeker Analysis

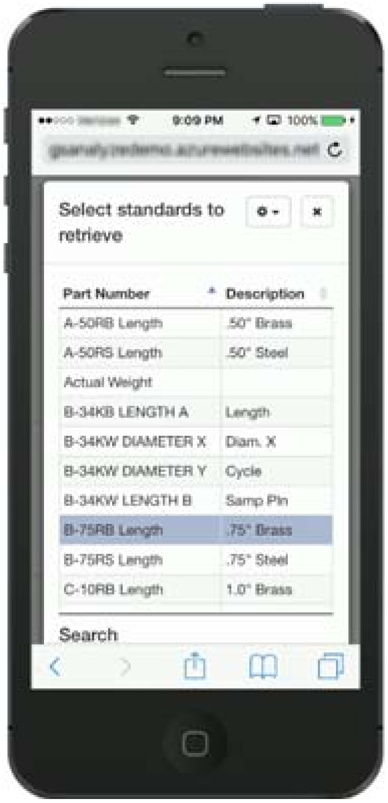
- Are you ready to use the browser on your smartphone, tablet, or PC to analyze your GainSeeker data?
- Are you ready to give colleagues, suppliers, or customers a web link to the live charts and dashboards that are important to them?

- Are you ready to use fewer IT resources installing the GainSeeker client on workstations or setting up remote access to your GainSeeker system?

  Meet **GS Analyze** – a browser-based analysis tool that performs many functions of the GainSeeker Charts module securely over the Internet or your company intranet.

  Because it is browser-based, no GainSeeker installation is required on the client computers. Simply install GS Analyze on a web server with Internet Information Services (IIS) and access it from any device with a web browser. GS Analyze lets you share charts, analytics, and dashboards with suppliers, customers, or colleagues over the Internet.

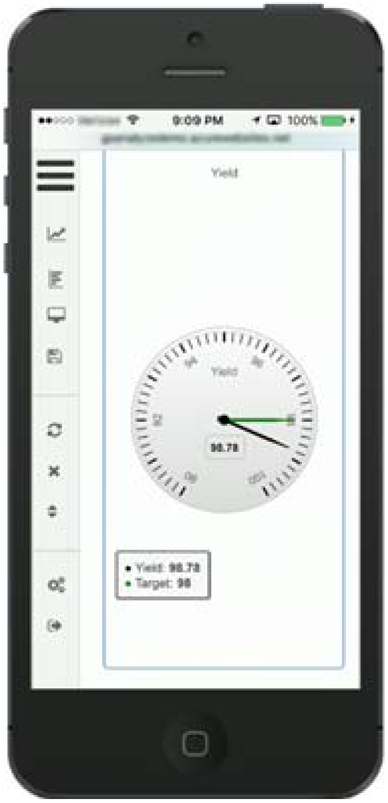


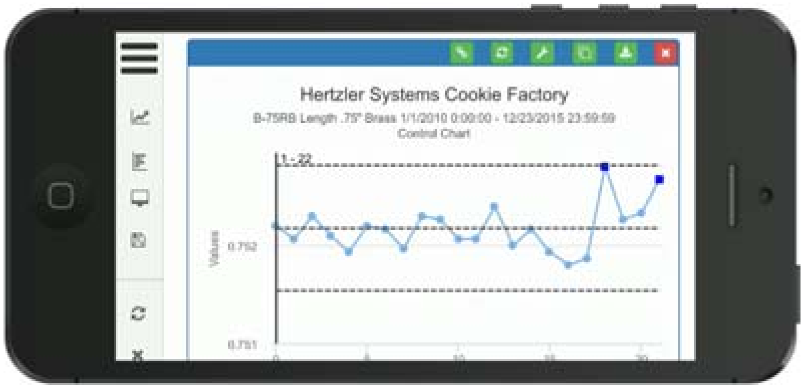


Select standards to retrieve

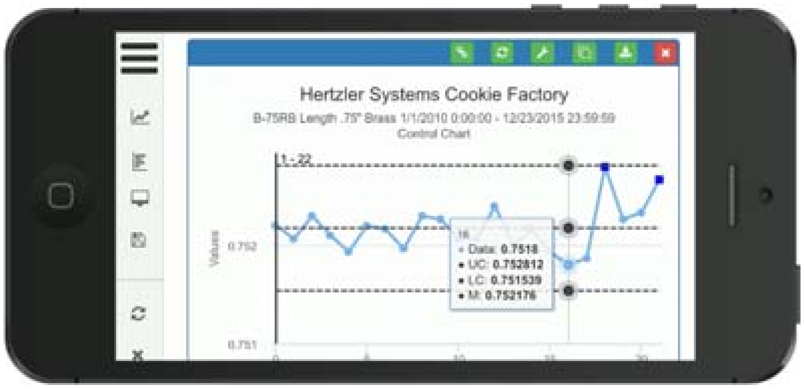
Part Number	Description
A-50RB Length	.50" Brass
A-50RS Length	.50" Steel
Actual Weight	
B-34KB LENGTH A	Length
B-34KW DIAMETER X	Diam. X
B-34KW DIAMETER Y	Cycle
B-34KW LENGTH B	Samp Pln
B-75RB Length	.75" Brass
B-75RS Length	.75" Steel
C-10RB Length	1.0" Brass

Search





Hertzler Systems Cookie Factory
B-75RB Length .75" Brass 1/1/2010 0:00:00 - 12/23/2015 23:59:59
Control Chart
1 - 22
Values
0.752
0.751
0
5
10
15
20



Hertzler Systems Cookie Factory
B-75RB Length .75" Brass 1/1/2010 0:00:00 - 12/23/2015 23:59:59
Control Chart
1 - 22
Values
0.752
0.751
16
Data: 0.7518
UC: 0.752812
LC: 0.751539
M: 0.752176
0
5
10
15
20

GS Analyze intended to replace legacy WebViz program

With the release of GS Analyze, no new features will be added to the legacy WebViz program, and the installation disks for GainSeeker versions 8.5 and newer will not contain an installation program for WebViz.

If you have WebViz already installed on an IIS server, you can continue to use it – although you will likely find GS Analyze to be more powerful and user-friendly.

Planned Inspections for PC Collect and the GS Collect Mobile App

New Planned Inspections shorten your setup time when you have many similar parts and/or processes. They also make it simpler to deploy changes to an Inspection across similar parts and processes.

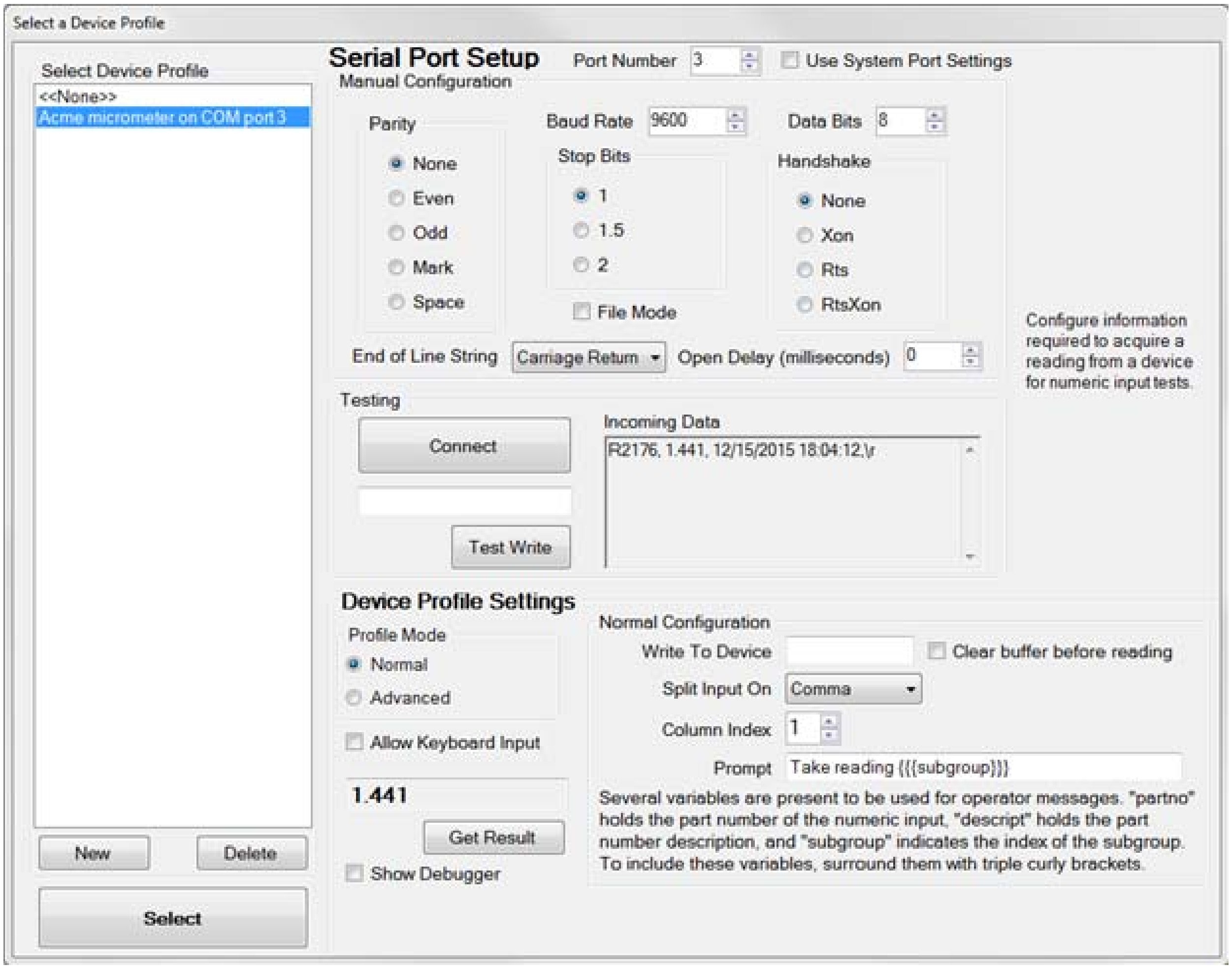
A *Planned Inspection* lets you choose an existing Inspection but specify different standards, processes, traceability values, and documents. So instead of creating and publishing multiple Inspections that are *almost* identical, you create one Inspection for the first part and process and use it as the basis for creating and publishing multiple Planned Inspections.

When changing the Inspection for these parts and/or processes, you only need to change and publish the single Inspection underlying the Planned Inspections. (For the GS Collect mobile app, you also need to re-save the Planned Inspections.) This is much simpler than editing many Inspections that are *almost* identical.

Your users will run Planned Inspections in the same PC Collect module and/or GS Collect mobile app that you currently use for Inspection data entry.

Device Profiles simplify RS-232 setup for PC Collect

New Device Profiles for Numeric Input tests let you configure a PC Collect Inspection for RS-232 data entry without needing to write Python code:



Select Device Profile: <<None>>, Acme micrometer on COM port 3

Port Number: 3

□ Use System Port Settings

Parity: ● None ○ Even ○ Odd ○ Mark ○ Space

Baud Rate: 9600

Stop Bits: ● 1 ○ 1.5 ○ 2

□ File Mode

Data Bits: 8

Handshake: ● None ○ Xon ○ Rts ○ RtsXon

End of Line String: Carriage Return

Open Delay (milliseconds): 0

Configure information required to acquire a reading from a device for numeric input tests.

Incoming Data: R2176, 1.441, 12/15/2015 18:04:12,\r

Profile Mode: ● Normal ○ Advanced

□ Allow Keyboard Input

1.441

□ Show Debugger

Write To Device:

□ Clear buffer before reading

Split Input On: Comma

Column Index: 1

Prompt: Take reading {{{subgroup}}}

Several variables are present to be used for operator messages. "partno" holds the part number of the numeric input, "descript" holds the part number description, and "subgroup" indicates the index of the subgroup. To include these variables, surround them with triple curly brackets.

Also related to this new feature:

- A new `inspect.cursubi.ni(idx).deviceprofile` Python command lets you retrieve or set the Device Profile for a Numeric Input test.
- It is now possible to configure a PC Collect Inspection in which the user can interrupt RS-232 data entry – for example, if the device stops transmitting data – and enter data via keyboard instead.

- ◦ A new TabDirection property for Numeric Input tests lets you change the direction in which the Inspection automatically advances after a measurement is entered.

File Import Wizard simplifies file entry for PC Collect

The new File Import Wizard helps you set up and run different types of file imports without needing to write Python code:

This can be used to import standards, data, or both. Information can be imported from text files and Excel files (*.xls and *.xlsx).

OPC for PC Collect

The PC Collect module can now interface with OPC servers:

- ◦ You can use the new OPC Action to generate a Python script for basic communication with an OPC Classic or OPC DA server.
- ◦ New Python OPC commands let you communicate with an OPC UA server. You can also use these commands to further customize your communication with an OPC Classic or OPC DA server.

> ❗ **Important:** To interface with OPC servers via Python, you must install additional components that are not included in the GainSeeker installation.
>
> Instructions for installing the OPC components can be found in an **OPCSetup** sub-folder of the GainSeeker program folder – typically C:\Program files (x86)\Hertzler Systems\GainSeeker.

Display a Slideshow of charts and dashboards

You can display a Slideshow of charts and dashboards in full-screen mode on your computer.

If you have GainSeeker Server Edition, you can also publish the Slideshow to HTML files. This allows an unlimited number of users to access these files, without requiring additional GainSeeker seats.

Other new features and improvements in GainSeeker 8.5

Multiple Modules

- Google Docs are now supported for data entry display in Inspections (the URL/HTML/File test) and in Templates (Template Wizard Step 6).
- For SPC standards, the Limits tab has been updated to let GainSeeker calculate limits that are equidistant from the Target X value.
- The Windows login name was added as a new statistic available in both SPC and DMS retrievals.
- The Dashboard Retrieval name was added as a new statistic available in both SPC and DMS retrievals.
- In the GainSeeker Charts module, the names for the three Bypass options for SPC data are now the same whether editing data or brushing data. The Note Manager module in GainSeeker Utility has also updated the names of these settings.

GainSeeker Charts

- The Dashboard Grid control now allows for **Region Colors** to allow for ranges of data to use the same colors.
- The improved Search function for Chart Settings makes it easier to navigate the results of your search.

- You can now configure a Dashboard Button control with a **ClickAction** property to open a Desktop in Edit/Open mode.
- The right-click menu and **Options** menu for most SPC and DMS charts now includes shortcuts to change the Filter and Date Period.
- New Send to HTML option for charts, dashboards, and dashboard controls.
- New right-click menu with Send to and print options for dashboard Python controls that display a linechart or barchart.
- Exporting a dashboard will now export the dashboard design and referenced items - retrievals, data filters, statistical filters, data table column reports, statistics list reports, and Python scripts - to a single .dbz file.

  In previous versions of GainSeeker, exporting a dashboard would only export its design as a .dbx file - so if you needed to export the referenced items, you had to do so manually. The current version of GainSeeker still allows you to import these .dbx files and manually import the referenced items like you did in previous versions.
- Previously, there was no way to set the back color, fore color, or font information for a Statistics List control or a Data Table control of a Dashboard. Properties for **BackColor**, **ForeColor**, and **Font** were displayed for these dashboard controls, but these properties were ignored when the controls were displayed on the dashboard. Starting in version 8.5, those properties are removed and a **Chart Skin** property has be added to the Statistics List and Data Table controls of a dashboard. You can now modify the font and background color of these controls by choosing a Chart Skin.
- The number of retrievals that can be displayed in a Multiple Charts window has increased from 49 (7x7) to 70 (10x7).
- The dashboard Grid has been improved to run faster, allow larger queries, and accommodate more data on the same Grid.

GainSeeker Inspections

- In previous versions, the **TestID** property of any test was a read-only number that was assigned by the system and designed to be referenced by Python script.

  You can now name the **TestID** property as desired and write Python scripts that reference that name (between quotation marks).
- You can now double-click directly on any Formula test to edit its Python Design. In previous versions, it was necessary to look in the Properties grid for the Formula test and find the appropriate property to edit.
- The GainSeeker Inspections module now exports and imports any Inspection as a zipped file with a .cpz filename extension. This .cpz file contains not only the Inspection, but also any Device Profiles or Python Designs that are referenced by that Inspection. (Inspections exported from a previous GainSeeker version as .cpx files can still be imported.)
- The default PictureMode property for a sub-inspection has changed from **Low** to **Off**.
- A new **Startup** menu now lets you choose which window to open by default when you launch the GainSeeker Inspections module on this computer – **Manage Inspections**, **Manage Planned Inspections**, or **None**.
- A Custom Password has been added as another Electronic Signature type for the **Checkbox** test.
- The new Inspection property **PreFormula** for the GS Collect mobile app lets you run LiveCode before the Inspection is loaded. This is useful for setting standards, traceability values, and other values that can be set without needing a specific sub-inspection to be loaded and submitted.
- When you add a Defect List test to an Inspection, the prompt for choosing defects has changed.

  In an inspection designed for Mobile devices, this test can now specify an *Override Process* for displaying defects and storing data. In previous versions, this option was only available on inspections designed for PC.

  The properties **DefectList** and **OverrideProperty** have been combined into one **Defects** property that applies to both PC and Mobile inspections.
- The maximum length of the Inspection property **IDLabel** has expanded from 14 to 20 characters.

PC Collect

- Real-time failure colors and priorities are now applied during data entry in PC Collect.
- A new **Startup** menu now lets you choose which window to open by default when you launch the PC Collect module on this computer – **Inspection**, **Planned Inspection**, **Start Python**, **Manage Python**, or **None**.
- Images stored with inspection data are now filed in sub-folders based on the year and month when the corresponding data was collected.

GS Collect mobile app

- Images stored with inspection data are now filed in sub-folders based on the year and month when the corresponding data was collected.
- The Login page on each device now lists the User or Group names that have successfully logged in to GS Collect on that device.
- Steps to print a Data Grid have been updated to reflect GS Collect mobile app release version 1.0.18.
- Due to larger customer interest and more stable platform features, the GS Collect mobile app is now available exclusively on the iOS platform and is no longer being offered on the Android platform.

GainSeeker Utility

- The SQL Assistant function was added to the **SQL Utility**. The SQL Assistant makes it easy to change data without the need to write your own database statements in Structured Query Language (SQL). This is equivalent to the initial window ("Standard Mode") in the legacy SQL Utility module.
- A new Assign Inspections function was added to the **SQL Utility**. It is useful for assigning multiple Inspections and Planned Inspections to one user.
- A new function is available in the **SQL Utility** to help locate and delete orphaned data records.
- Report Transfer now exports and imports any Inspection as a zipped file with a .cpz file name extension. This .cpz file contains not only the Inspection, but also any Device Profiles or Python Designs that are referenced by that Inspection. (Inspections exported from a previous GainSeeker version as .cpx files can still be imported.)
- The Import/Export Data utility now lets you save and load **Settings** such as selected standards and processes, date period, and filters.
- When using the Note Manager to edit notes, any hyperlink that was stored in the note now displays as a clickable link.

Python

- Use the new MTConnect Action to generate a Python script for basic data collection from an MTConnect agent.
- Real-time checks can now be performed when storing SPC data with the `dataspc` Python commands. See `dataspc.store()`, `dataspc.store(mode)`, and `dataspc.rtf` for more details.
- The `testid` Python variable now returns special values "[PRE]" and "[POST]" when the a Formula test triggers the Python Design using the **FormulaMode** property of **Pre** or **Post**.
- Use the new `login` commands in Python to retrieve information specific to the current GainSeeker User.
- Use the new `hsidate.format_to_db` command to convert a formatted date/time (such as '`02-28-2015`') to the GainSeeker database format.
- The `sql.gettable` command has been updated to retrieve data from large text columns, such as the last column of the HSIRPT table. In previous versions, this type of column was simply left blank.
- The new command `inspect.cursubi.anchorpoint` lets you set anchor points via Python script.
- The new command `inspect.cursubi.cancel` lets you cancel the current sub-inspection via Python script.

- The new command `inspect.cursubi.ni(idx).visible` lets you check the visible status of a Numeric Input test.
- The `inspect.cursubi.ni(idx).subgroupsize` command has been updated so that you can set the subgroup size for a Numeric Input test with **SPCStandardSelection** = DoNotStore.

  In previous versions, this type of Numeric Input test was only allowed a subgroup size of 1.
- The new command `inspect.cursubi.trace(idx).label` lets you get or set the label displayed to the user for a Traceability test.
- The new command `traceability.getprioritylist(traceindex)` returns the traceability priority list for the specified traceability index that has been set up for the currently logged-in GainSeeker user.
- The new command `serial[x].allowkeyboard` lets you control whether data entry users can temporarily switch to Keyboard input by pressing CTRL+K on the serial input window.
- The new commands statspc.data.rtf and statspc.data.rtfstr return real-time failure information for each SPC record retrieved.
- The "OnChange" FormulaMode has been expanded for tests such as DMS Standard, Traceability, Defect List, and Numeric Input.

  In previous versions, a changed value in these tests would trigger an OnChange formula only if the user then clicked on another test. Now a changed value in these tests will also trigger an OnChange formula when the user clicks **Submit** or **Cancel**.
- The installation process now copies the file `TemplateToPythonCommands.txt` to the `PyLib` folder in the GainSeeker Installation folder. This file contains a list of all the template commands and the corresponding Python command to achieve the same results. This file is useful as templates are converted to Inspections or stand-alone Python scripts. This file will be updated with each new version of the GainSeeker Suite.
- When accessing the list of Python Designs, GainSeeker displayed a separate error message for each Python Design it couldn't open because the Design had been created by a newer version of GainSeeker. Now when GainSeeker can't open a "newer" version of a Python Design, it displays one message that includes all of the Python Designs that could not be opened.

SPC Data Entry

- When a real-time failure note window is displayed for an individual value or for the subgroup, you can now configure GainSeeker to automatically place the cursor in the **Operator notes**. To do so, add the line `rtfn_oper_focus=1` to the Convert.db file. This setting will apply to all SPC data entry users in all configurations.

SPC Designer

- Now you can clear formulas from cells on the grid (on Designer step 5) by selecting one or more cells and then pressing CTRL+L. Or you can right-click a cell and choose **Clear formula** from the right-click menu.
- The new DE_OPT 43 command sets the number of rows to display before scrolling on the DATAROWSOK, DATAROWSOK_POS, and DATAALLOK windows.
- The new DE_OPT 44 command can automatically select any existing text in the input box on the **Data Entry** tab, making it easier for Data Entry users to change long values (such as traceability) with minimal keystrokes.