

General Improvements with GainSeeker version 8.6

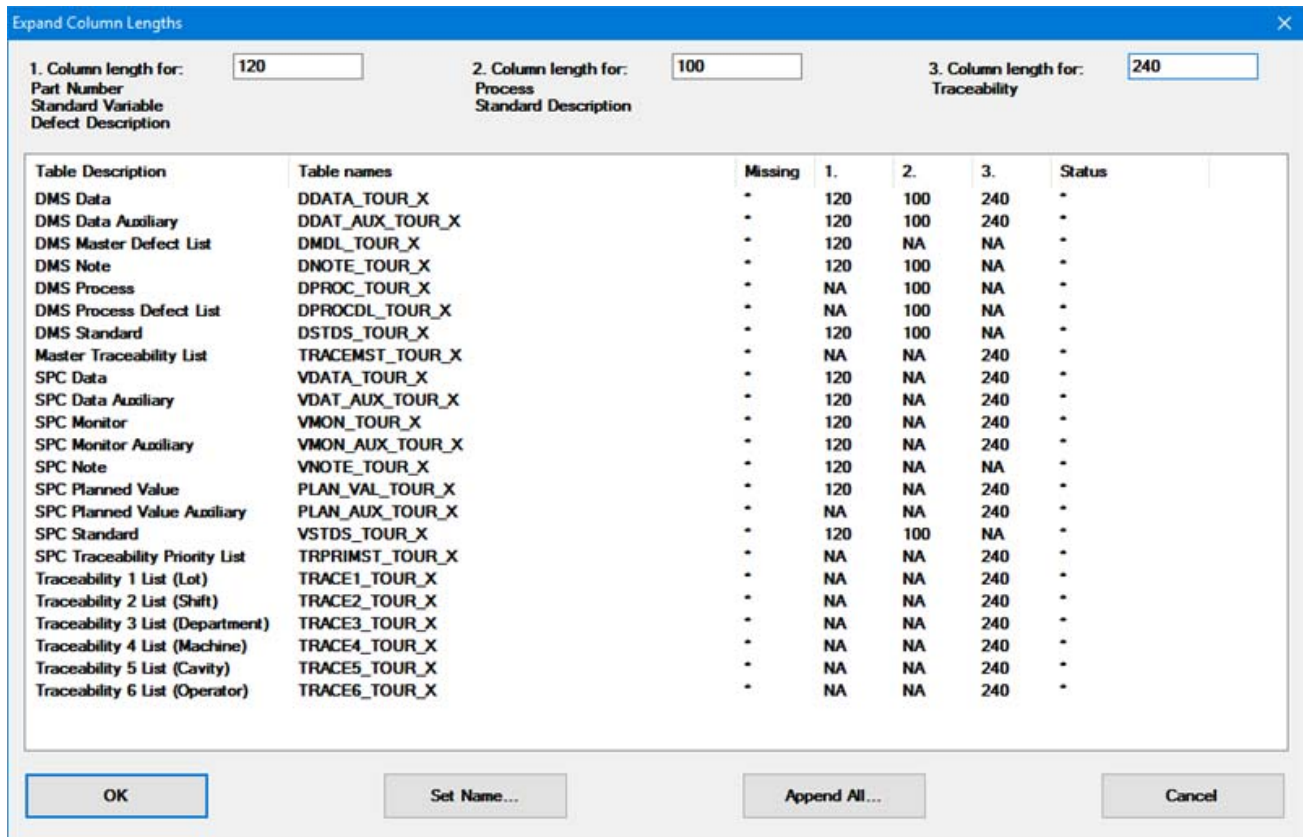
Major new features in GainSeeker version 8.6:

Expanded data fields

If you have ever struggled to shorten crucial information to fit GainSeeker's field length limits, then you'll be glad to hear about our new options for expanded data fields!

You can now define a maximum field length of up to 240 characters for the following fields:

- Part Number (for SPC and DMS)
- SPC Standard Variable
- DMS Defect Description
- DMS Process
- SPC Standard Description
- Traceability (for SPC and DMS)



MariaDB database support

GainSeeker version 8.6 has been tested on MariaDB - a free, open-source database that can be used in place of the MySQL database. MariaDB is now supported as one of five GainSeeker database options available.

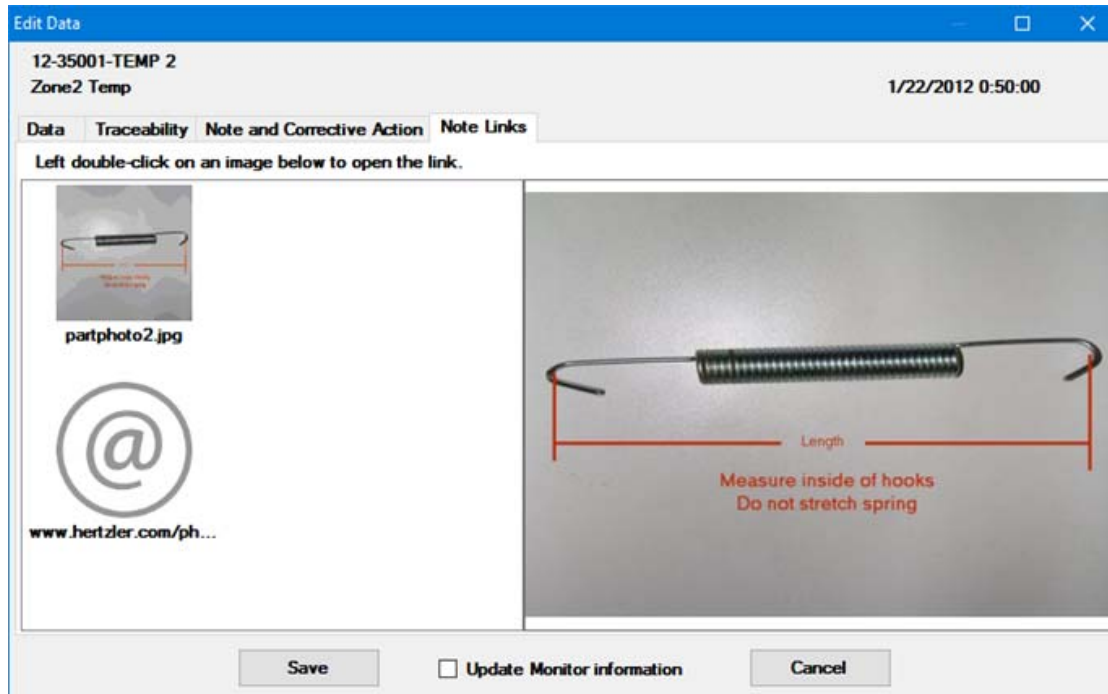
Reduced database size on Microsoft SQL Server, MySQL, and MariaDB

GainSeeker now creates all new tables with variable-length character (`varchar`) columns, instead of the fixed-length character (`char`) columns required by previous versions of GainSeeker, on Microsoft SQL Server, MySQL, and MariaDB. The resulting GainSeeker tables can be significantly smaller than their previous counterparts - particularly for the SPC and DMS auxiliary tables such as `VDAT_AUX` and `DDAT_AUX`.

If you are upgrading from a GainSeeker version prior to 8.6 and use one of these databases, you can reduce the size of your GainSeeker database by moving your data into these newer tables. This is purely optional, as GainSeeker continues to support the larger tables created by previous versions.

Multiple Modules

- When you view or edit a Note with an image attached, newer modules such as GainSeeker Charts, PC Collect, and GainSeeker Utility now display those images. You can double-click the smaller thumbnail image to open the linked file (or URL) in its default application on your computer.



- Performance charts now use the Y-axis scaling options in the SPC standard and in the Standard Overrides settings. This scaling is overridden if the Show entire chart element outside limits setting is checked.
- When displaying Specification limits on a Performance chart, a new **Display Spec limit values** setting can be used to display the values of those limits.
- The GainSeeker Charts and PC Collect modules have been optimized to retrieve and chart data much faster in certain situations. Better performance was particularly noticeable for queries where a Maximum Count was set, and where the database contained a large number of data records for each standard. These optimizations apply to GainSeeker databases on Microsoft SQL Server, Oracle, and MySQL.
- Multiple modules have been optimized for speed when retrieving a very long list of standards - for example, when drawing a new chart or editing standards - with an average speed improvement of 83%.
- In GainSeeker Charts, Dynamic Reports, and GainSeeker Inspections: On the list of multi-retrieval SPC charts, the **Box and Whisker** option has been renamed to **Process Capability**. None of the charting functionality has changed, but this label more clearly describes the category of chart types that provides a visual comparison of multiple retrievals - including the histogram curve chart (the default type), the standard Box and Whisker chart, and others. As with previous versions, you can use the System Administration module to set the default chart type generated for the Process Capability chart.
- Control charts with a large number of data points now display fewer tick marks on the x-axis.
- Newer modules such as GainSeeker Charts, GainSeeker Inspections, and PC Collect now sort Priority Lists in alphabetical order - for standards, desktops, and traceability.
- The Hertzler Systems Data Collection Policy has been modified to include the collection of Operating System, .NET version, and GainSeeker Database type (Oracle, Microsoft SQL Server, etc.).

GainSeeker Inspections

- A new **Sample Size** property has been added to the **Checkbox** and **Pass/Fail** tests. This gives you more flexibility when designing an inspection for a large batch of product and wish to pass, fail, or check off the entire batch at once..
- A new sub-inspection property **VisibleInList** makes it possible to hide a sub-inspection from the list of sub-inspections.
- A syntax checker has been added to the LiveCode editing window.
- When a Formula test contains a LiveCode script (and no Python script), you can double-click this test to edit the LiveCode script.

PC Collect

- For a Numeric Input test where a real-time failure occurs, the color displayed for the entire subgroup now corresponds with the highest priority color, whether that indicates a failure for an individual measurement or for the subgroup. In previous versions, the color in this area corresponded only with subgroup failures.
- PC Collect has now been optimized for users who frequently connect and disconnect their data entry computer with an external monitor. If the last-used location of the PC Collect window is no longer available - because the external monitor has been disconnected, for example - then the PC Collect window will be centered on the available screen.

Python

- New commands- such as `inspect.cursubi.checkbox(idx).settrace` and `inspect.cursubi.checkbox(idx).gettrace` - allow you to set (and get) a traceability value for a single test that will store a data record- checkbox, defect list, numeric input, or pass/fail - instead of for the entire sub-inspection.
- The new `inspect.cursubi.ni(idx).subgroupcomplete` command checks a numeric input test to determine whether it has been completed.
- The new `file.xls.getrow` command returns a list of cell values in the specified row of the spreadsheet.
- The new `inspect.storerecords` command disables or enables automatic storage of data records when the user Submits the current sub-inspection. This lets you create a sub-inspection where the user enters information on the sub-inspection, but you read the data they entered and then use other Python commands to create and store whatever records you need.
- Documentation has been expanded for the `disp` commands that are used to display forms and messages, including these changes:
 - The `disp.newlistselect` command has been updated so that `.value` can specify which items should be selected on the list that is displayed to the user. Previously, `.value` could only be used to read which items the user had selected.
 - A new `disp.showconsole` command makes it easy to display the Python console window without creating a form.
 - A new `disp.settablabel` command lets you set the label on any tab. Also, the `disp.addpane` command has been renamed to `disp.addtab`, and all previous references to "panes" on display forms have been changed to "tabs". Any existing scripts that use this command will need to be edited to reflect this change. A topic for Breaking Changes to Python commands has been added to track this and other situations where changes to Python commands will break backward-compatibility.
 - The `disp.newtextbox` command has been updated to support multiple lines. When displayed in multiple-line mode, the textbox expands vertically to use the entire area available.
 - The `disp.newbrowser` command has been modified as follows: after the user fills in an HTML form and clicks a button to close the form, the `.html` property which supplied the original HTML code to the new browser control will be updated to include the resulting user input. This reduces the programming work needed to re-display the form and include the user's data (e.g., during a validation step, if you need to prompt the user for more information).
- Suggestions for optimizing speed have been added to the topics for the `dataspc` and `datadms` commands that can be used to build and store GainSeeker records.
- Documentation has been added for the `retrspc.periodstr` and `retrdms.periodstr` commands.
- The `dataspc.store` commands have been optimized to avoid retrieving data if the only real-time checks being performed are for fixed values such as specification limits and gate limits. This improves performance when storing data - especially when storing large amounts of data quickly. Also, standards are now refreshed when data is re-queried.

System Administration

- The Table Creation Utility now checks for other configurations that point to the same tables for SPC Data and/or DMS Data but specify different table names for DMS Event, DMS Master Defect List, SPC Event, SPC Cause, and/or SPC Action Taken. These tables use an internal code or number to log these values, so having two configurations that point to the same data table but different corrective action tables and/or defect master list table will produce unexpected results. The reference topic for Tables and Files used in GainSeeker 8 has also been updated to reflect this practice.

.Launch

- Shortcuts for PC Collect now include options for opening Planned Inspections, debugging Inspections and Planned Inspections, and closing PC Collect after opening an Inspection, Planned Inspection or Python Script.

Online User Guide

- The installation instructions for GS Analyze have been updated. They now direct you to install the GainSeeker client software on the web server or intranet server before you install GS Analyze.

- The topic on designing Dynamic Reports has been updated to reflect the data types of all GainSeeker values.
- The topics Setting charts to display in inspections, SPC Chart Options, and DMS Chart Options have been updated for clarity.
- The topic for DMS standards was updated to clarify which modules can automatically perform real-time checking for DMS data, and which require the use of the `datadms.hadgatefailures` Python command. The topic for `datadms.hadgatefailures` was also updated.