Waters
THE SCIENCE OF WHAT'S POSSIBLE.®

Empower 3

Data Acquisition and Processing Theory Guide

715005481
Revision A

General information

Copyright notice

The information in this document is subject to change without notice and should not be construed as a commitment by Waters Corporation. Waters Corporation assumes no responsibility for any errors that may appear in this document. This document is believed to be complete and accurate at the time of publication. In no event shall Waters Corporation be liable for incidental or consequential damages in connection with, or arising from, its use. For the most recent revision of this document, consult the Waters website (waters.com).

Trademarks

ApexTrack™ is a trademark of Waters Corporation.

e-SAT/IN™ is a trademark of Waters Corporation.

Empower® is a registered trademark of Waters Corporation.

Millennium® is a registered trademark of Waters Corporation.

SAT/IN™ is a trademark of Waters Corporation.

THE SCIENCE OF WHAT'S POSSIBLE® is a registered trademark of Waters Corporation.

Waters® is a registered trademark of Waters Corporation.

All other trademarks are property of their respective owners.

Customer comments

Waters' Technical Communications organization invites you to report any errors that you encounter in this document or to suggest ideas for otherwise improving it. Help us better understand what you expect from our documentation so that we can continuously improve its accuracy and usability.

We seriously consider every customer comment we receive. You can reach us at tech_comm@waters.com.

Contacting Waters

Contact Waters with enhancement requests or technical questions regarding the use, transportation, removal, or disposal of any Waters product. You can reach us via the Internet, telephone, fax, or conventional mail.

Waters contact information

Contacting medium	Information
Internet	The Waters Web site includes contact information for Waters locations worldwide. Visit www.waters.com
Telephone and fax	From the USA or Canada, phone 800-252-4752, or fax 508-872-1990. For other locations worldwide, phone and fax numbers appear in the Waters Web site.
Conventional mail	Waters Corporation Global Support Services 34 Maple Street Milford, MA 01757 USA

Empower 3 software

Intended use

Use the Waters Empower 3 software for acquiring, processing, reporting, and managing your chromatographic information.

Safety information

See the operator's guides of the instruments or devices associated with this software product for information on how to safely operate and maintain them.

Table of contents

1 Data acquisition

This chapter describes how data are acquired and how analog data are converted to digital data.

1.1 Overview

1.1.1 What is data acquisition?

A chromatogram is a series of detector responses sampled across a length of time. The elution of a compound results in a characteristic chromatographic peak profile.

1.2 Detection sampling rates

Empower software sets data collection frequency to the sampling rate you specify in the associated instrument tab of the Instrument Method Editor. The sampling rate must be high enough to provide a good representation of the chromatogram, but not so high that you are collecting more data than you need.

The optimum sampling rate for current instrumentation, such as UPLC, is a minimum of 25 to 50 data points from peak start to peak end for the narrowest peak of interest detected. For older chromatographic systems and qualitative work, 15 data points per peak is adequate. To determine the optimum number of data points, consider typical signal-to-noise ratios and the frequency content of an exponentially modified Gaussian peak.

Tip: Increasing the sampling rate can increase noise.

The amount of disk space required during data acquisition depends on the sampling rate and run time. Higher sampling rates require more storage space. For additional information on the theory of data acquisition, see Reference.

1.2.1 Determining the optimum sampling rate

You can use the following equation to determine the optimum sampling rate:

$$SR = \frac{N}{W}$$

where:

SR = Sampling rate (points/second)

N = Optimum number of data points from peak start to peak end, recommended values: 15 for LC; 25 for UPLC

W = Measured width (in seconds) of the narrowest peak you want to detect

For example, for a measured peak width of three seconds, a sampling rate of 5 ensures data collection of 15 raw data points (where 15/3 = 5).

Tip: If the number of data points across the narrowest peak of interest is less than 15, specify a faster sampling rate. Faster sampling rates produce more data points and require a greater amount of disk space for data storage (see Effects of data acquisition rate on disk space).

Recommendation: If the calculated sampling rate is not available, select the next available higher rate.

1.2.2 Displaying the data points

The **Peaks** tab in Review displays the **Start Time**, **End Time**, and **Points Across Peak** for each integrated peak in the chromatogram. These are reportable fields you can display in any report group.

1.2.3 Points Across Peak field

Empower software calculates a Points Across Peak value for each integrated peak in the chromatogram. The calculation uses the index of the data point closest to the end time minus the index of the data point closest to the start time. The calculation is accurate for peaks that have nonuniform data rates.

1.3 Effects of data acquisition rate on disk space

The amount of disk space required during data acquisition depends on the sampling rate and run time. The table below illustrates the amount of disk space needed to store a single channel of data collected at different sampling rates and run times.

Table 1–1: Effects of sampling rate and run time on disk space:

Sampling rate (points/sec)	Data points acquired per minute run time	Kilobytes per minute run time (1024 bytes)	Run Time (min)	Approx. space used (kilobytes)
1	60	0.23	10	2.3
5	300	1.17	10	12.0
20	1200	4.69	10	47.0

When you start data acquisition in Run Samples, the software determines the current amount of disk space available. If disk space is insufficient, the software warns you and does not start acquisition. If space becomes limited during Run and Process or Run and Report modes, processing stops and acquisition continues until all remaining disk space is used.

1.4 Analog-to-digital conversion

The detector analog output signal must be converted to a digital representation before Empower software can acquire and process data. This section describes the sequential processes of data conversion and data transfer and storage.

Note: The recommendations in this guide apply to Waters instrumentation. If you are using instrumentation from another vendor, check their product literature for recommendations.

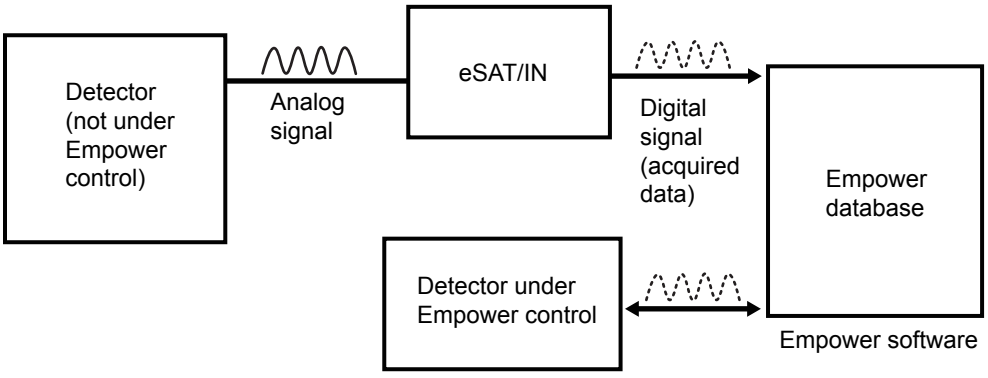
1.4.1 Data conversion

Analog-to-digital (A/D) conversion of detector data is performed in one of the following ways:

- A detector controlled by Empower software performs the conversion.
- A detector not controlled by Empower software transmits an analog output signal to a chromatographic interface (Waters e-SAT/IN module). The magnitude of the transmitted signal (in microvolts) corresponds to the amount of sample detected at a constant rate.

The voltage range over which the incoming analog signal can vary is -0.25 V to +2.25 V. Each millivolt of signal represents 1,000 height counts (where 1 height count is equal to 1 µV). For example, with a detector set so that 1 AU is equal to 1 V, a 1 AU peak is equal to a peak height of 1,000,000 height counts (from the baseline, at 0 V).

The e-SAT/IN module converts analog signals to digital signals at a specified number of times per second (sampling rate).



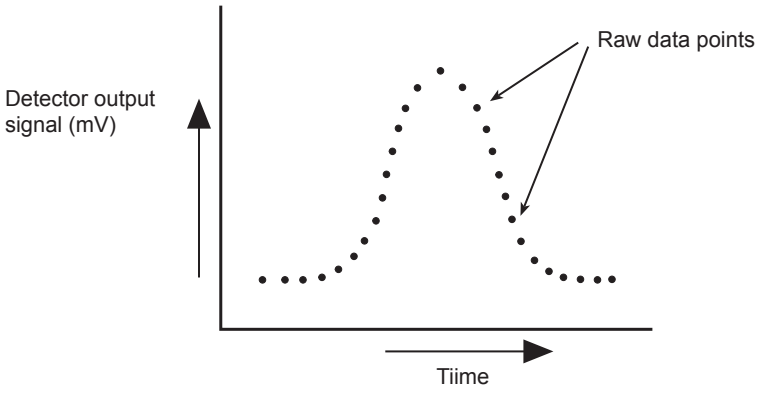
1.4.2 Data transfer and storage

This sequence of events describes how the software transfers and stores data:

1. The digital signal is transmitted to one of the following communication devices:
   - BusLAC/E card
   - Serial hub
   - COM port on the Empower computer
   - Ethernet port on the Empower computer

   Tip: You can use more than one type of communication device in an Empower workstation, LAC/E^{32} Acquisition Server, or acquisition client.
2. The collected data are transmitted from the communication device to the computer's hard drive.
3. The digital voltage values are stored as acquired, unprocessed data. The stored digital values are the raw data points of the chromatogram. Raw data can be viewed in the Run Samples window as it is being acquired. The Sample Sets, Injections, and Channels views of the Project window represent the raw data in the current project.

Figure 1–1: Plot of acquired data points



1.5 Reference

For further information on the theory of data acquisition, see the following reference guide.

Dyson, Norman, *Chromatographic Integration Methods*, The Royal Society of Chemistry, Thomas Graham House, Cambridge, 1998.

2 ApexTrack integration

This chapter describes ApexTrack peak detection and integration theory.

2.1 Features and capabilities

ApexTrack peak detection and integration by Empower software includes the following functions:

- Automatically determines appropriate peak width and detection threshold values for the chromatogram unless already set in the processing method.
- Detects peak apices in the chromatogram to determine the location of peaks and shoulders.
- Integrates peaks to determine their retention times, areas, and heights.

The processing method defines the parameters the software uses to detect and integrate the peaks within the raw data file (channel).

Functionality in traditional processing that is identical to that in ApexTrack processing is discussed in Commonalities between ApexTrack and traditional integration.

2.1.1 ApexTrack features

Empower software supports both traditional integration and ApexTrack integration. The term “traditional integration” refers to the technique that detects peaks by measuring the change in slope of the baseline. ApexTrack processes data differently than traditional integration:

- ApexTrack detects a peak at its apex rather than at its liftoff point, detecting the apex by its curvature (second derivative). In contrast, traditional integration detects a peak at its liftoff by its slope (first derivative).
- ApexTrack can reliably detect shouldered peaks because it uses a curvature criterion.
- The ApexTrack algorithm finds baselines by starting at each peak’s inflection points, expanding a trial baseline downward and outward.
- ApexTrack determines the end-points of peak and cluster baselines by internal slope comparisons. As a result, the location of the baseline is independent of detector drift, and ApexTrack can reliably integrate small peaks on sloped baselines.

ApexTrack offers the following major features:

- Shoulder detection – Detects shoulders and round peak pairs.
- Gaussian skims – Skims individual or multiple peaks within a cluster using a Gaussian profile.
- Negative peak detection and integration – Supports shoulder detection and Gaussian skimming of negative peaks.

2.1.2 How ApexTrack integrates peaks

ApexTrack integration consists of three major processes:

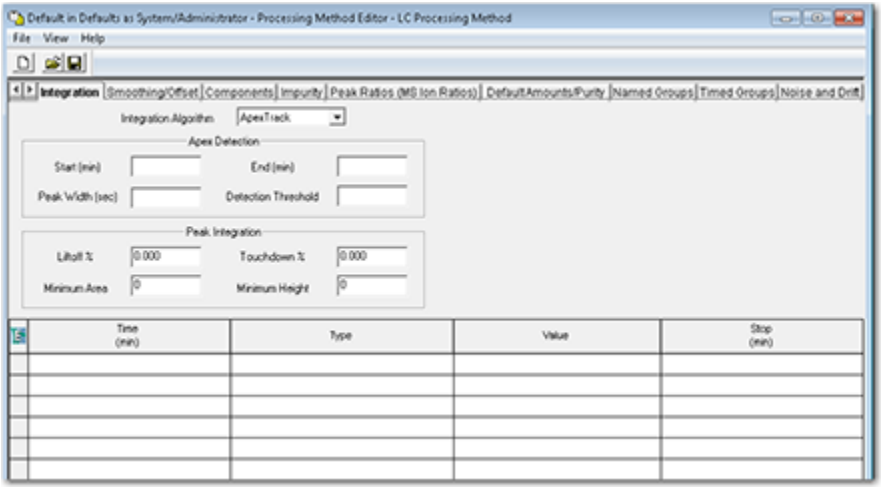
- Detects peaks – Detects a peak at its second derivative apex. The baseline slope does not affect peak detection. All apices above the detection threshold are detected. (Traditional integration detects a peak at its liftoff point.) (See Apex detection.)
- Determines baselines – Determines the baseline of each peak using the Liftoff % and Touchdown % parameters (see Baseline location and Determination of peak boundaries).

  Note: Peak detection and baseline determination are independent of each other.
- Calculates the peak area, height, and retention time (RT) – Integrates peaks and determines height and retention time by the quadratic fit method (5-point or 3-point fit), or by the time of the second derivative apex or the time of the highest point. The software uses the Trapezoidal rule to calculate peak area (see Computing integration results).

2.1.2.1 Summary of processing method parameters

The default processing method for ApexTrack integration differs from that for traditional integration.

Figure 2–1: ApexTrack default processing method:



The following processing parameters control peak detection:

- Integration Algorithm: Determines whether ApexTrack or traditional integration is used. You enable the use of ApexTrack by system policies and project properties. Once enabled, you can select the algorithm you want to use for the processing method.
- Start (min) and End (min): ApexTrack can only detect apices between the start and end times you specified. The effect of start, end, or both is similar to that of Inhibit Integration. (see Start and End).
- Peak Width (sec): This parameter, together with the Start, End, and Detection Threshold parameters, determines how many peaks are integrated. Reducing the Peak Width value, while keeping the other parameters constant, generally increases the number of integrated peaks and vice versa. The default value is blank. When the field is blank, the software uses the Auto-Peak Width algorithm to suggest an appropriate value for your chromatogram. (see Peak width parameter).
- Detection Threshold: With the other Apex Detection parameters, the Detection Threshold parameter also determines how many peaks are detected. Reducing the Detection Threshold value, while keeping the other parameters constant, increases the number of detected peaks and vice versa. The default value is blank. When the field is blank, the software uses the Auto-Detection Threshold algorithm to suggest an appropriate value for your chromatogram. (see Detection threshold parameter).

The following parameters control baseline location:

- Liftoff % and Touchdown %: These parameters control, respectively, the start and end points of integration. Higher values cause the integration to start, end, or both further up the sides of the peak or peak cluster. The default value for Liftoff % is 0.000. The default value for Touchdown %, as of Empower 3 FR 2, is 0.000. The values can range from 0.000% to 100.000%. (see How ApexTrack determines the slope difference threshold).

Restriction: The maximum value of Liftoff % and Touchdown % allowed in a GPC processing method is 5.000.

Two parameters reject peaks that fall below specified values:

- Minimum Area and Minimum Height: Both reject peaks based on integration results.

2.1.3 Commonalities between ApexTrack and traditional integration

Although the algorithms ApexTrack uses to detect peaks and to determine baselines differ from those used by traditional processing, significant functionality is the same for both traditional and ApexTrack integration:

- Ability to automatically determine peak width and threshold values. (Disabled by default with traditional integration.)
- Timed events such as Inhibit Integration, Tangent Skim, Set Peak Width, Set Minimum Height, Set Maximum Height, Set Minimum Area, and Set Maximum Peak Width are supported.
- Peaks can be manually added or deleted.

- Peak start and stop markers can be manually changed.
- Allow Negative Peaks and Valley to Valley are supported.

2.1.4 Summary of timed events

Empower software supports timed events for peak detection and integration.

Tip: Refer to the *Empower Online Information System* for an overview and description of timed event.

2.1.5 Integration peak labels in ApexTrack

Each identified peak in a chromatogram is given a two-letter label that describes the peak's start and end boundaries. The boundaries of a peak are described by a pair of letters. These letters appear in the **Int Type** column of the Peaks table, in the **Results** and **Main** windows of **Review**.

If no integration events are enabled, each peak starts or ends on the baseline (B) or in a valley (V) above the baseline. Each peak is labeled as follows:

- BB indicates a baseline-resolved peak.
- BV indicates a peak that starts a cluster.
- VV indicates a peak within a cluster.
- VB indicates a peak that ends a cluster.

ApexTrack integration supports four Int Type letters specific to ApexTrack: Shoulder, Round, Gaussian Skim, and Crossover.

To review how the software determines peak boundaries, see Determination of peak boundaries.

Table 2–1: Boundary identifications for peaks integrated by ApexTrack:

Name of peak start or end	Letter	Description
Baseline	B	The boundary of the peak is a baseline.
Valley	V	The boundary of the peak is a valley.
Shoulder[a]	S	The boundary of the peak is a shoulder (a mathematically derived valley point).
Round[a]	R	The boundary of the peak is a round.
Gaussian Skim[a]	G	The boundary of the peak is a Gaussian skim.
Crossover[a]	X	The boundary of the peak occurs where the signal intersects the baseline. The peaks on either side of this point have opposite signs (one is positive, one is negative).

Table 2–1: Boundary identifications for peaks integrated by ApexTrack: (continued)

Name of peak start or end	Letter	Description
Tangential Skim[a]	T	The boundary of the peak is a tangent skim.

a. You must enable the appropriate timed event.

Capitalization of each letter indicates that ApexTrack performed the integration automatically. Lowercase letters indicate manual integration.

For instance, a baseline label of Bb indicates that, while the peak start and peak end are both baseline-resolved, the peak start was automatically integrated by the software and you manually adjusted the peak end.

Tip: Refer to the *Empower Online Information System* for information on manual integration.

2.2 Apex detection

When ApexTrack processing is invoked, the first process applied to the data is apex detection. The apex of a peak is the point of maximum curvature. Apex detection is based on measuring the curvature (the rate of change of the slope, or second derivative) of the peak. ApexTrack uses the curvature at the peak apex to detect peaks, and the algorithm associates a peak with each detected apex. After detecting peak apices, ApexTrack locates the baselines (see Baseline location).

Tip: In describing or plotting the curvature of a chromatogram, this guide adopts a negative curvature convention. The second derivative chromatogram measures the chromatogram's curvature at each point, and it is scaled (multiplied) by -1 before plotting. Given this convention, the apex of a positive peak has positive curvature, and the apex of a negative peak has negative curvature.

2.2.1 Detecting apices

ApexTrack software detects peaks as follows:

1. Obtains the peak width parameter.
2. Uses the peak width to obtain the second derivative smoothing filter.
3. Uses the second derivative filter to obtain the chromatogram's second derivative (curvature) plot.
4. Within the second derivative plot, locates the times of each maximum (for positive peaks) or minimum (for negative peaks, when Allow Negative Peaks is enabled). Only apices between the start and end times are located.

5. Obtains the detection threshold parameter.
6. Applies the detection threshold parameter to the maximum (for positive peaks) or minimum (for negative peaks), and retains only the apices whose curvatures are above the detection threshold (for positive peaks) or below the detection threshold (for negative peaks). For negative peaks, the detection threshold parameter is multiplied by -1.

2.2.2 Apex detection parameters

The following parameters control apex detection:

- Start
- End
- Peak width
- Detection threshold

You can manually specify peak width and detection threshold values in the processing method, or Auto-Peak Width and Auto-Detection Threshold can automatically determine the values. The operation of Auto-Peak Width is summarized in Peak width parameter. The operation of Auto-Detection Threshold is summarized in Detection threshold parameter. The start and end parameters are optional.

2.2.2.1 Start and End

ApexTrack can detect only apices that are between the Start and End times. The effect of Start and End is similar to that of Inhibit Integration. You can specify Start and End values manually, or leave them blank (default). A blank entry in Start means start of data, and a blank entry in End means end of data. The range is 0 min to 655 min. If neither Start nor End value is blank, the value in Start must be less than the value in End.

Tip: Generally, it is good practice to ensure that an Inhibit Integration event, and Start and End times, occur within the baseline region of a chromatogram. When you specify Start and End times, the software calculates the parameters using the baseline in the region of the chromatogram you want to integrate.

Recommendation: For best results, enter values for the Start and End parameters when using the Auto Peak width, Auto Detection Threshold, or both features. The software calculates the auto parameter values based on the region of the chromatogram between the Start and End times.

2.2.3 Obtaining the second derivative plot

ApexTrack detects peaks by calculating the second derivative of the chromatogram. The top plot shows an ideal Gaussian peak. The bottom plot shows its second derivative profile.

Tip: The baseline in the second derivative chromatogram is at zero, even if the baseline in the chromatogram is not at zero.

Figure 2–2: Gaussian peak

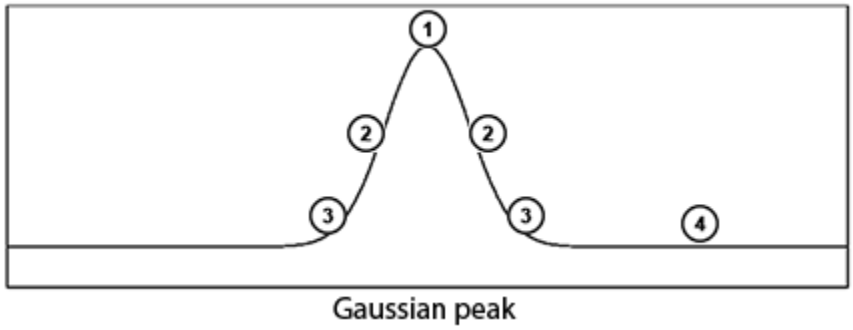
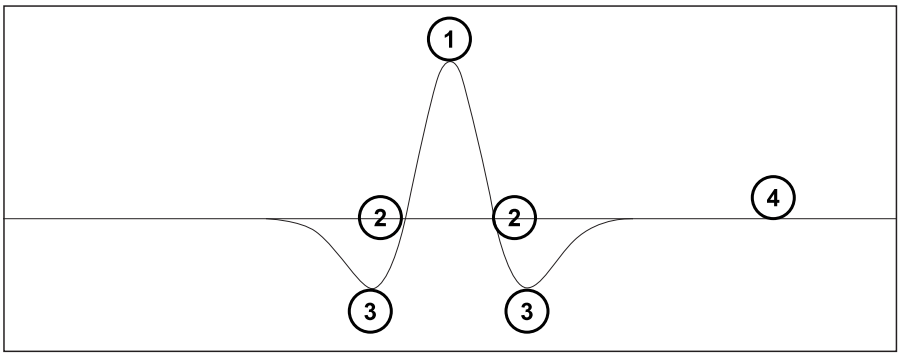


Figure 2–3: Second derivative plot of Gaussian peak



1 The maximum of the second derivative is the highest point in both plots.

2 In the bottom plot, the inflection points are where the second derivative crosses 0. The times of these points are carried up to the top plot.

3 The upslope points in the top plot are curvature minima in the bottom plot.

4 The second derivative of the baseline is always zero.

Tip: All second derivative plots in this guide are multiplied by -1, so the apex of a positive peak appears as a positive second derivative.

2.2.4 Detecting the peak

A positive peak in the chromatogram has a single maximum point of curvature in the second derivative plot. The time of that maximum identifies the peak's *apex*.

On either side of the apex are the *inflection points*, which straddle the apex and have zero curvature (pass through the zero line on the second derivative plot).

Continuing down the chromatographic peak are the *upslope points*, which have a minimum of curvature in the second derivative plot.

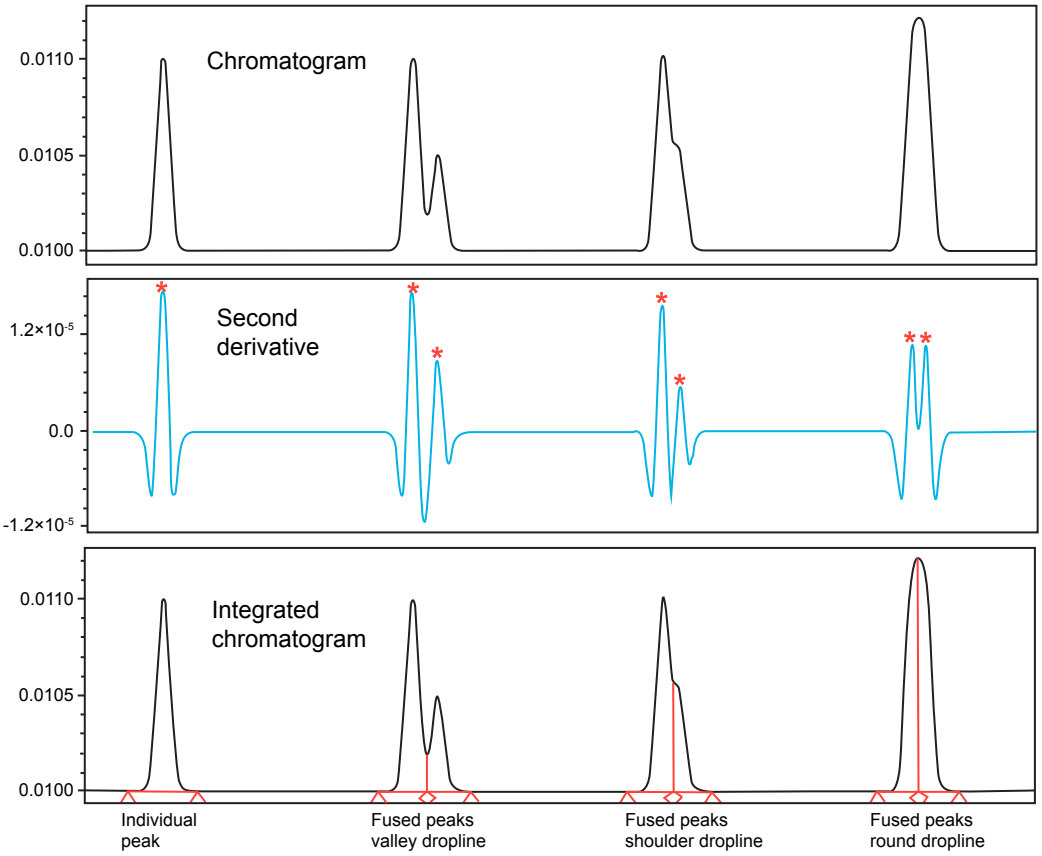
Finally, ApexTrack reaches the *baseline*, which has zero curvature. Even if the chromatographic baseline has significant drift or is not at zero, the baseline of the second derivative plot always has zero curvature because the curvature of a straight line is zero.

2.2.4.1 Resolved peaks and fused peaks

The following figure shows a simulated chromatogram along with its corresponding second derivative and integrated chromatogram. There is an isolated peak on the left followed by three pairs of fused peaks, each successive pair showing less separation between the two peaks.

Tip: All second derivative plots in this guide are multiplied by -1, so the apex of a positive peak appears as a positive second derivative.

Figure 2–4: Baseline resolved peak, valley boundary, shoulder boundary, and round boundary:



The asterisks in the figure identify the second derivative apices. Because ApexTrack detects peaks by locating maxima in the second derivative of the chromatographic signal, two apices (and therefore two peaks) are identified in each pair of fused peaks.

When integrated, each pair of fused peaks is separated by a dropline. The type of dropline between two fused peaks is determined as follows:

- A valley dropline (V) occurs when both the chromatogram and the second derivative contain a minimum between the two identified apices. Each peak has unique inflection points, located on either side of its apex where the second derivative passes through the zero line.
- A shoulder dropline (S) occurs when the chromatogram does not have a minimum between the two identified apices and the minimum in the second derivative between the two identified apices is less than zero. Each peak has unique inflection points, located on either side of its apex where the second derivative passes through the zero line.
- A round dropline (R) occurs when the chromatogram does not have a minimum between the two identified apices and the minimum in the second derivative between the two identified apices is greater than zero. These two peaks share inflection points.

Shoulder and Round droplines are only seen when the Detect Shoulders event is enabled. When this event is not enabled, all Shoulder or Round droplines that would have been present are omitted from the integrated chromatogram.

In the chromatogram, the apex of a pair of round peaks appears rounded. A pair of round peaks occurs with fused peaks of similar heights; a shoulder peak occurs with fused peaks that have differing heights.

Tip: The "Int Type" (integration type) field in the peaks table records the peaks' boundaries and includes Shoulder (S) and Round (R) boundaries. The seven peaks in the previous figure would have the following integration types: BB, BV, VB, BS, SB, BR, and RB.

2.2.4.2 Sequence of operations

The determination of peaks, baselines, and boundaries involves three processes (apex detection, baseline location, and boundary determination) with the processing of timed events (Allow Negative Peaks, Detect Shoulders, Valley-to-Valley, Gaussian Skim, and Tangential Skim).

The order of operation of these processes is as follows:

1. Detect apices.
2. Process Allow Negative Peak events.
3. Locate baselines.
4. Locate peak boundaries.
5. Process Valley-to-Valley events.
6. Process Detect Shoulders events.
7. Process Gaussian Skim events.
8. Process Tangential Skim events.

2.2.5 Second derivative apex and inflection point data in the Peaks table

The time of the second derivative apex for each peak appears in the **Peaks** table in the column labeled **2nd Derivative Apex**. This time is generally not the same as the peak retention time. If the time of the second derivative apex is used for the retention time, a peak code of I20 appears. For tailed peaks, the second derivative apex time generally precedes the retention time.

The displayed value for the second derivative apex of a particular peak does not change with changes to the processing method unless the Peak Width parameter is changed. The only exception is when there are shoulder or round boundaries that have been removed from the peak because the Detect Shoulders event is not enabled. If a peak has hidden round or shoulder boundaries, the second derivative apex displayed for it may change with changes to the processing method.

Inflection points straddle the apex and have zero curvature (pass through the zero line on the second derivative plot). The time between a peak's inflection points appears in the **Peaks** table in the column labeled **Inflection Point Width (sec)**.

2.2.6 Peak width parameter

Different separations can produce peaks whose widths vary over a wide range, from seconds to minutes. ApexTrack requires a peak width as input into the processing method. The peak width sets the widths of the digital filters, which are used internally to obtain the smoothed, first and second derivative chromatograms. In ApexTrack, the role of the peak width parameter is solely to determine these filter widths.

Tip: The number of points in the filtered chromatograms is the same as in the original chromatogram. ApexTrack, unlike traditional integration, does not bunch data points for peak detection. ApexTrack processing uses the filtered chromatograms for peak detection. The width of a filter determines how much smoothing is done to the filtered chromatograms. Wider filters produce increased smoothing in the smoothed, first, or second derivative chromatogram. Smoothing removes high frequency components (noise), leaving frequencies that correspond to actual chromatographic features (peaks). The ApexTrack processing method expects as input the width of the largest peak measured at 5% of its height in the second derivative, expressed in seconds.

Tip: In ApexTrack, you can measure the peak width at 5% height by visual inspection and specify this value in the method, or you can use Auto-Peak Width to determine the peak width automatically. The Peak Width value used during processing is listed in the Results table, in the field "Peak Width".
The most universal measure of the width of a distribution is its standard deviation (SD).

Table 2–2: Peak width values of a Gaussian Peak expressed in standard deviation units

Height % where peak width is measured	Peak width expressed in standard deviation units	Name
60.7	2	Inflection point width
50	2.355	Peak width at half height
13.5	4	Chromatographic peak width
5	4.895	
1	6.070	

2.2.6.1 ApexTrack integration with System Suitability

Accurate location of a peak's inflection points is important when using System Suitability, because the Width @ Tangent value is calculated by drawing tangent lines at the inflection points, and Width @ Tangent is used to calculate USP Resolution, USP Plate Count, and sometimes Relative Resolution. The value of the Peak Width parameter selected in the method affects the location of the peak's inflection points. The selected Peak Width value in the processing method might not be optimal for calculating inflection points for all peaks in the chromatogram. To eliminate this problem, when processing using a method in which System Suitability calculations are selected, the software calculates an optimized peak width value for each baseline resolved peak (integration type BB or bb), which it then uses to determine optimized locations of each peak's inflection points. The Width @ Tangent value is then calculated using these optimized inflection points. When an optimized peak width is used, the peak has an S53 peak code and the Optimized Peak Width value is seen in the Peaks Table. An S29 peak code is used to indicate that the tangent lines were drawn at the ApexTrack inflection points, rather than at the height percentage listed in the processing method.

2.2.6.2 Auto-Peak Width

The Auto-Peak Width algorithm determines an appropriate value for the processing method's Peak Width parameter. It is invoked when the Peak Width field in the processing method is blank. Because the ApexTrack algorithm is expecting a Peak Width value at 5% height, the Auto-Peak Width algorithm selects the peak with the largest magnitude second derivative in the selected region and calculates its width at 5% height by multiplying its inflection width by a factor of 4.89549 (assuming a Gaussian peak). That calculated value is the Peak Width value used during processing and it is copied to the Peak Width field in the **Results** Table.

You can select the region used to calculate Auto-Peak Width two ways:

- In Review, zoom in on a region of the chromatogram and click  (**Set Processing Method Peak Width**). The AutoPeak Width for this region is entered into the processing method.

When you process the data, ApexTrack reports and applies this value to the entire chromatogram.

- Leave the **Peak Width** field in the processing method blank. When you process the data, Auto-Peak Width uses the regions between the Start and End times, and the region between any Inhibit Integration events at the start and end of the chromatogram, to determine the peak width. ApexTrack reports and applies this value to the entire chromatogram. The software determines the optimum peak width value for each chromatogram.

2.2.6.3 Using Auto-Peak Width

If the **Peak Width** field is blank, Auto-Peak Width uses the data between start and end, and the region between any Inhibit Integration events at the start and end of the chromatogram, to determine the peak width. Choose the start and end times to exclude injection artifacts and artifacts associated with the return-to-initial conditions. For example, if the void volume is included, ApexTrack might select an injection artifact as the highest peak, which generally gives a width that is too small.

Even if start and end times are properly chosen, Auto-Peak Width can give inaccurate results under the following circumstances:

- If the largest peak is saturated, the peak width value can be too large.
- If the largest peak is co-eluting with another peak, the peak width value can be too large.
- If the largest peak is noisy, Auto-Peak Width can measure the width of a noise artifact and produce a width that is too small.

To address these problems, zoom in on a region of the chromatogram that contains a valid peak with a valid width, then click  (**Set Processing Method Peak Width**). The peak width is copied to the processing method. When you process the data with this method, ApexTrack reports and uses the specified value. Generally, the peak width obtained from a reference separation is relevant for subsequent separations.

2.2.6.4 Effect of varying the peak width parameter

On baseline-resolved peaks, the variation of width about the Auto-Peak Width value by a factor of up to 1.5 should have little effect on peak detection or baseline placement.

For complex chromatograms with co-eluted peaks that span a range of peak heights, changing the width by a factor of 1.5 (and redetermining the Detection Threshold with Auto-Detection Threshold) can change which low-level peaks are detected.

Increasing the width by about a factor of 2 (and redetermining the Detection Threshold with Auto-Detection Threshold) should result in detection of peaks near the baseline that might otherwise be missed. However, this level of increase also reduces the number of detected shoulders.

Consider the following guidelines using Auto-Peak Width:

- For most chromatograms, using the Peak Width value generated using the Auto-Peak Width algorithm detects the desired peaks. Setting the Detection Threshold to zero while using the

Auto-Peak Width value integrates all peaks available with that Peak Width setting. Changing the automatically determined value of Peak Width is recommended when a desired peak is not integrated when using the Auto-Peak Width value along with a Detection Threshold setting of zero.

- You can zoom in on various regions in the chromatogram and determine the Auto-Peak Width value for that region to see if a single Peak Width value is appropriate throughout the chromatogram. Note that a factor of two change in the Auto-Peak Width value is considered negligible. When desired peaks are not integrated due to changes in peak width across a chromatogram, you can use the Set Peak Width event to change the Peak Width value at a desired time.

2.2.7 Detection threshold parameter

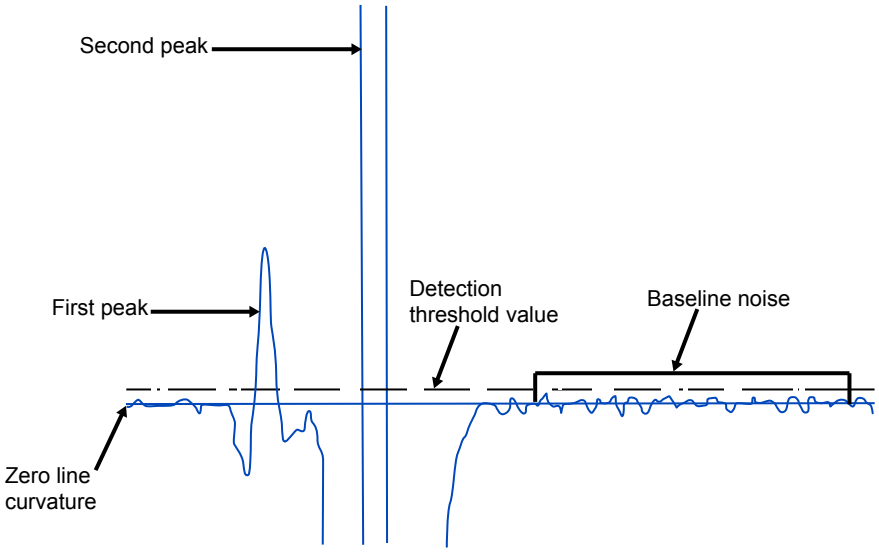
To distinguish chromatogram peaks from noise peaks, ApexTrack requires a detection threshold as input into the processing method. The software expects as input a value of the baseline's peak-to-peak noise, expressed in microvolts.

Two sources of noise can add fluctuations to the chromatographic signal:

- The irreducible statistical fluctuations inherent in any detection process.
- The detector's response to the solvent stream.

In ApexTrack, the baseline noise in the second derivative chromatogram is relevant in distinguishing peaks from baseline artifacts. The following figure shows the second derivative of a chromatogram with two chromatographic peaks and the baseline noise.

Figure 2–5: Second derivative showing baseline noise:



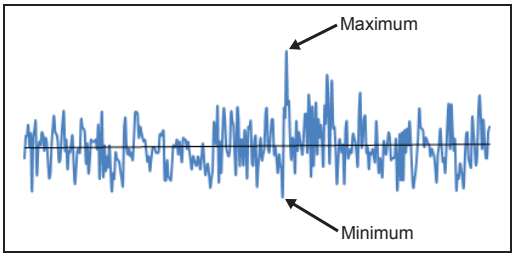
The noise in the baseline of the second derivative chromatogram is proportional to the noise in the baseline of the original chromatogram. Thus, the software can obtain a second derivative detection threshold value from the peak-to-peak noise in the original chromatogram's baseline.

Internally, the Detection Threshold entered into the processing method is converted to a value of curvature (microvolts/sec/sec). This converted value is applied to the second derivative chromatogram. Only peaks whose second derivative rises above the curvature threshold are accepted as valid detections of peak apices. A properly chosen threshold rejects all artifacts due to detector noise and accepts only valid peaks.

In ApexTrack, you can measure the baseline's peak-to-peak noise by visual inspection and specify the value in the method, or you can use Auto-Detection Threshold to measure the peak-to-peak noise automatically.

Tip: Increasing the Detection Threshold value causes fewer peaks to be detected; decreasing the Detection Threshold value causes more peaks to be detected.

Figure 2–6: Example of peak-to-peak baseline noise:



2.2.7.1 Auto-Detection Threshold

The Auto-Detection Threshold algorithm determines an appropriate value for the processing method's Detection Threshold parameter. The algorithm requires a region of the chromatogram and a value for the peak width parameter as input. Auto-Detection Threshold identifies sections of the selected region of the chromatogram that are free of peaks, and estimates the peak-to-peak noise in those sections. The selected region might contain one or more peaks, or no peaks. For a valid measurement, the selected region must contain a section with no peaks that is at least as wide as the peak width parameter. The Detection Threshold parameter is measured in uV.

As with Auto-Peak Width, you can select the region that is input to Auto-Detection Threshold in two ways:

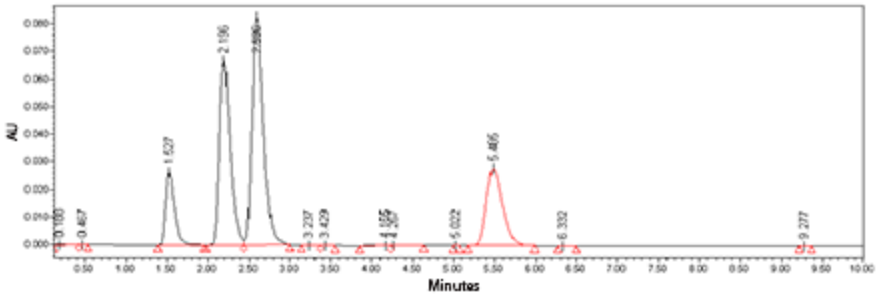
- Zoom in on a region of the chromatogram and click  (**Set Processing Method Threshold**) to set a detection threshold value in the processing method. (This button is active only if a peak width is specified in the processing method.) When you process the data, ApexTrack reports and uses this value.
- Leave the **Threshold** box in the processing method blank. When you process the data, Auto-Detection Threshold uses the regions between the start and end times, and between Inhibit Integration events at the beginning and end, to determine the detection threshold. ApexTrack reports and uses this value. The software determines the optimum detection threshold value for each chromatogram.

Auto-Detection Threshold determines the threshold by examining the noise regions in the second derivative chromatogram. It converts the noise in the second derivative chromatogram to the

equivalent peak-to-peak noise threshold that would be seen in the original baseline and reports this value.

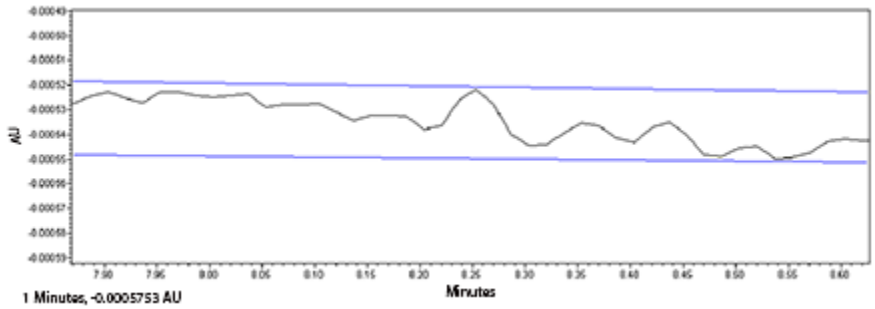
The following figure shows the integration of a sample chromatogram whose baseline is shown in the following figure. Auto-Detection Threshold reports the threshold as 23.00 µV.

Figure 2–7: Automatic measurement of baseline noise:



You can also manually zoom in on a baseline and estimate the peak-to-peak noise visually. A straight line with a positive or negative slope indicates drift, and you can estimate the peak-to-peak noise. The magnitude of peak-to-peak noise in this example is approximately 25 µV. You can specify this value in the ApexTrack method.

Figure 2–8: Manual measurement of baseline noise:



2.2.7.2 Using Auto-Detection Threshold

If the **Detection Threshold** box is blank, Auto-Detection Threshold uses the data between start and end times and between Inhibit Integration events to determine the detection threshold. Choose the start and end times to exclude regions that can have baseline noise that differs from the noise in the separation region.

When you enable the Inhibit Integration event at the beginning, end, or both of the chromatogram, Auto-Detection Threshold uses the data between the first and last data points outside the Inhibit Integration event. For example, if you enter an Inhibit Integration start time at zero minutes and end time at one minute, then another Inhibit Integration event starting at five minutes to end, Auto-Detection Threshold is calculated using the first data point after one minute and the last data point before five minutes.

In a chromatogram containing many components, there can be no region that is free of peaks. In this case, Auto-Detection Threshold can give a value that is too high, so that valid peaks are not detected.

Tips: Follow these guidelines when using auto-detection threshold:

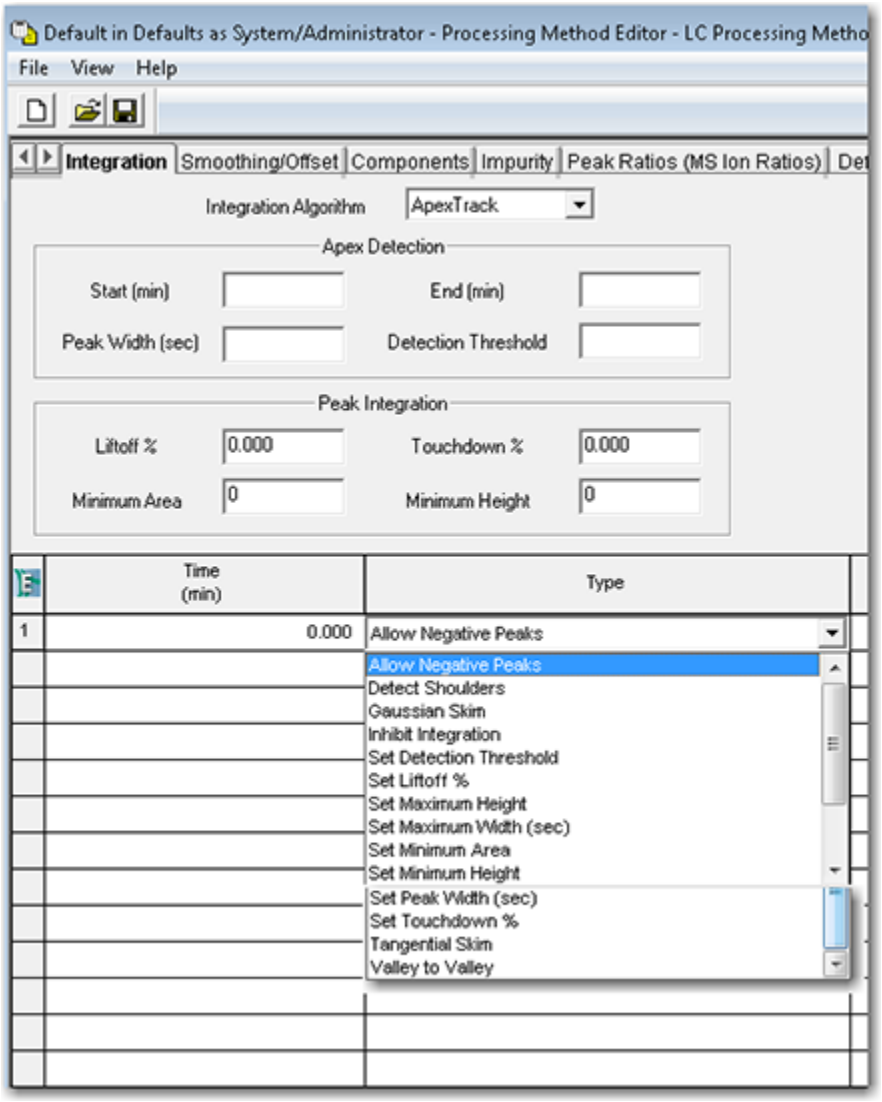
- Setting the Detection Threshold parameter to zero causes every peak apex that is detected with the selected peak width parameter to be integrated. If a desired peak is not integrated when the Detection Threshold is set to zero, change the peak width parameter to detect that peak's apex.
- If too many peaks are integrated in your chromatogram when using the Auto-Detection Threshold value, increase that value incrementally by a factor of 10 until only your desired peaks are integrated. Setting too high a value causes your peaks of interest to no longer be integrated.

2.2.8 Using timed events

ApexTrack peak detection and integration events are available in the **Type** column of the **Integration Events** table. ApexTrack events are available with all types of processing methods. The Merge Peaks event is also available with GPC processing methods.

Tip: Refer to the *Empower Online Information System* for an overview of all timed events and a description of each.

Figure 2–9: ApexTrack timed events



You can enable the following ApexTrack timed events within the time range specified in the table entry:

- Detect Shoulders – Enables the detection of shoulder and round peaks. The peak boundary and the integration results reflect the detection of the shoulder or round peak.
- Valley-to-Valley – Enables the replacement of cluster baselines with a separate baseline for each peak.
- Gaussian skim – Enables the replacement of vertical drop lines with Gaussian skims.
- Allow Negative Peaks – Enables negative peak detection.
- Tangential Skim – Enables the replacement of vertical drop lines by drawing a line tangentially to skim (front or rear) one or more rider peaks from the parent peak.

These events are unavailable by default. You can enable them in any combination and over any time range. However, these events cannot overlap themselves.

You can also enable the following events to modify the values already entered in the processing method:

- Inhibit Integration – Further delimits the time range within which peaks can be detected.
- Set Events – Modify the corresponding method values. Set events do not have a stop time.
    - Set Peak Width (sec)
    - Set Detection Threshold
    - Set Liftoff %
    - Set Touchdown %
    - Set Minimum Area
    - Set Minimum Height
    - Set Maximum Height
    - Set Maximum Width (sec)
- Merge Peaks Event – Available for GPC type processing methods.

All events require a start time; the default is 0.000 minutes. For events that have a stop time, you leave the field blank to indicate that the event is enabled until the end of the run, or you can specify a time in minutes. By default, the start and stop times have a precision of three significant figures, and the valid range of each parameter is 0.000 through 655.000 minutes. The start time must be less than the stop time, unless the stop time is blank.

2.2.8.1 Peak detection events

In ApexTrack, timed events can modify the detection of peaks.

ApexTrack events that affect peak detection are as follows:

- Inhibit Integration Event
- Detect Shoulders Event
- Allow Negative Peaks Event
- Set Events, as follows:
    - Set Peak Width (sec)
    - Set Detection Threshold

2.2.8.2 Peak integration events

Events can change the integration results associated with detected peaks.

These ApexTrack events affect peak integration:

- Valley-to-Valley Event
- Gaussian Skim Event
- Tangential Skim Event
- Merge Peaks Event (GPC, GPCV, GPC-LS, GPCV-LS only)
- Set Liftoff % Event
- Set Touchdown % Event
- Set Minimum Area
- Set Minimum Height
- Set Maximum Height
- Set Maximum Width (sec)

2.2.8.3 When timed events are active

These events affect peaks whose second derivative apex occurs during the time of the event:

- Inhibit Integration
- Allow Negative Peaks
- Set Detection Threshold
- Set Liftoff %
- Set Touchdown %

These events affect peaks whose retention time occurs during the time of the event:

- Set Minimum Area
- Set Minimum Height
- Set Max Height
- Set Max Width

These events affect vertical drop lines that occur during the time of the event:

- Detect Shoulders
- Gaussian Skim
- Tangential Skim
- Valley-to-Valley
- Merge Peaks (for GPC only)

This event affects the data point immediately after the timed event is enabled:

- Set Peak Width (sec)

2.3 Baseline location

After valid apices are found, ApexTrack determines the baselines associated with them.

To determine the baseline for positive peaks, ApexTrack completes these steps:

1. Initially draws a baseline between the inflection points of each peak.
2. Draws lines tangent to the inflection points.
3. Determines the slope differences between the inflection point baseline and a tangent line at each inflection point (upslope and downslope).
4. Determines the slope difference thresholds. The peak start slope difference threshold is defined as (Liftoff % × slope difference)/100. The peak end slope difference threshold is defined as (Touchdown % × slope difference)/100.
   - A Liftoff % or Touchdown % of 100% is at the inflection point.
   - A Liftoff % or Touchdown % of 0% merges with baseline noise.
5. Expands the baselines until the slope difference threshold criteria are met for peak start and peak end.

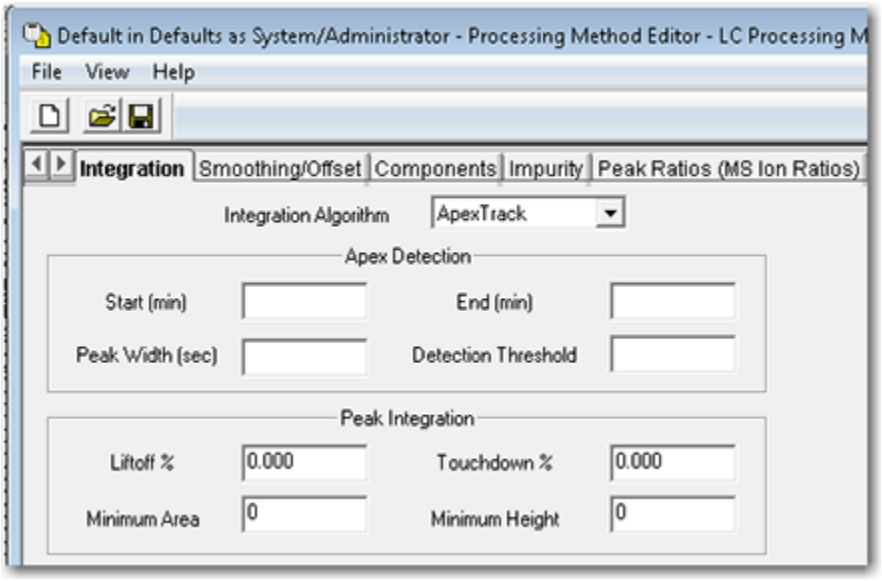
2.3.1 How ApexTrack determines the slope difference threshold

ApexTrack uses the Liftoff % and Touchdown % parameters when it determines the slope difference threshold to identify the start and end of peaks and peak clusters. Liftoff % is used for peak start and Touchdown % for peak end. The values can range from 0.000% to 100.000%. Increasing a value raises the point on the peak at which liftoff or touchdown occurs.

The processing method requires values for the following peak integration parameters:

- Liftoff % (default value is 0.000)
- Touchdown % (as of Empower 3 FR 2, is 0.000)

Figure 2–10: Baseline parameters:

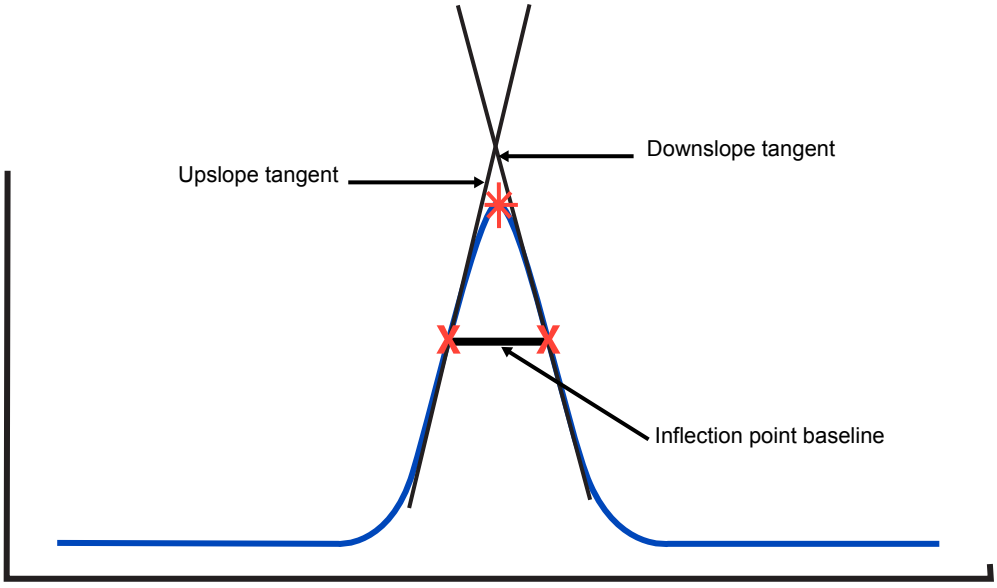


The algorithm then computes two slope difference thresholds for each peak based on the Liftoff % and Touchdown %.

To calculate the slope difference thresholds, ApexTrack completes these steps:

1. Identifies the inflection points that straddle a peak apex.
2. Draws a tangent at each inflection point and draws a baseline that connects the inflection points.

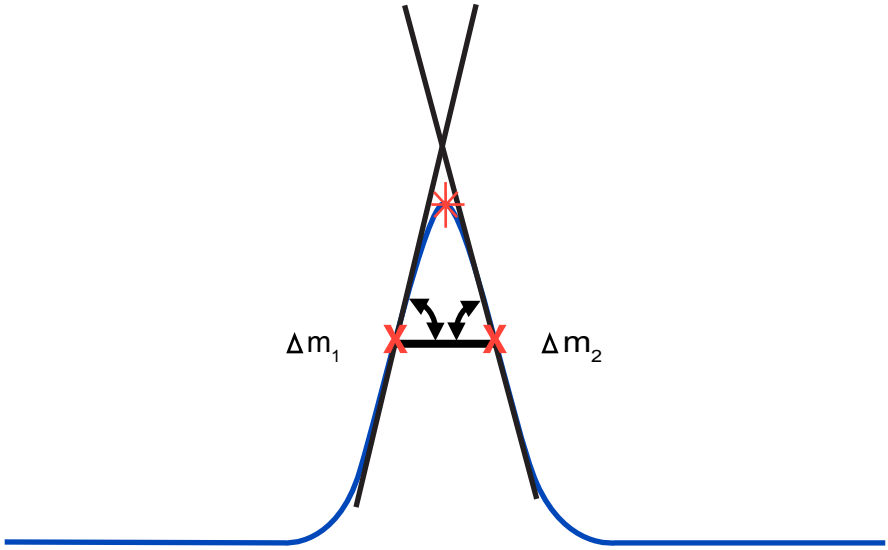
   Figure 2–11: Inflection point baseline:



3. Computes two slope differences:

- Δm_1, between the slope of the tangent at the upslope inflection point and the inflection point baseline.
- Δm_2, between the tangent at the downslope inflection point and the inflection point baseline.

Figure 2–12: Computing slope differences:



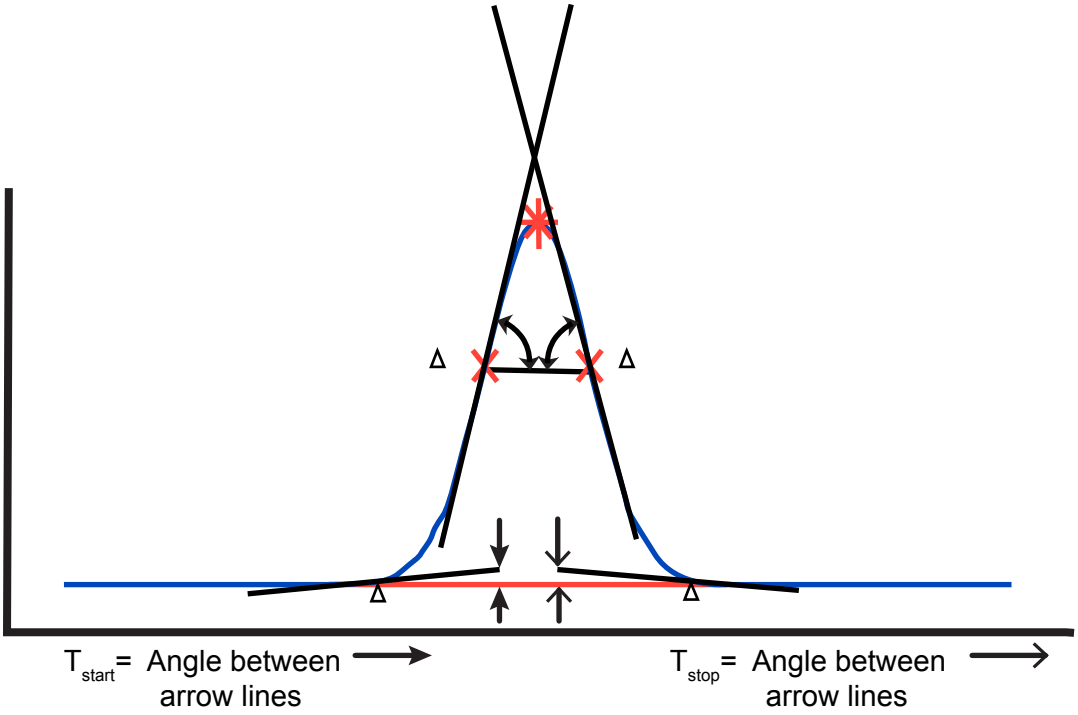
ApexTrack computes two slope difference thresholds (T_{start} and T_{end}) using Baseline % Thresholds from the method:

- $T_{start} = (\Delta m_1 \times \text{Liftoff \%})/100$
- $T_{end} = (\Delta m_2 \times \text{Touchdown \%})/100$

2.3.2 How ApexTrack locates the baseline for an isolated peak

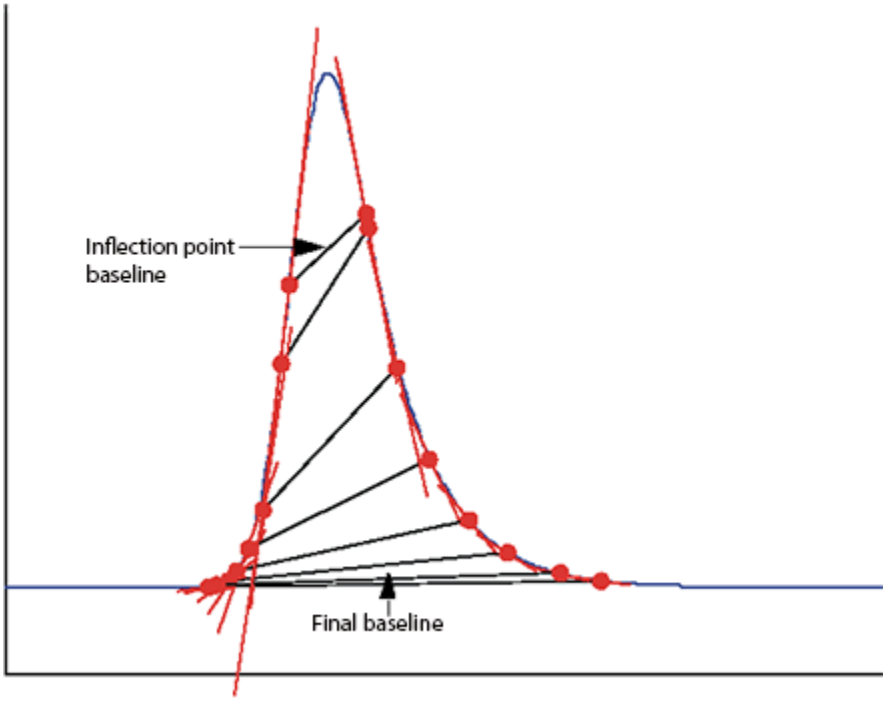
For each peak, ApexTrack expands the baselines downward and outward until the slope difference threshold criteria are met. At each point in the expansion, the slope difference threshold criteria are tested.

Figure 2–13: Computing the final baseline:



The following figure shows the simulation of a tailed peak and uses it to illustrate how ApexTrack locates the baseline of a baseline-resolved peak. The initial baseline is the inflection point baseline. The baseline expands as it moves down the peak and the slope difference thresholds are tested. With each step, the ends of the baseline become more tangent to the peak. The expansion stops when the slope difference thresholds are met at both ends.

Figure 2–14: Example of search for baseline in a tailed peak:



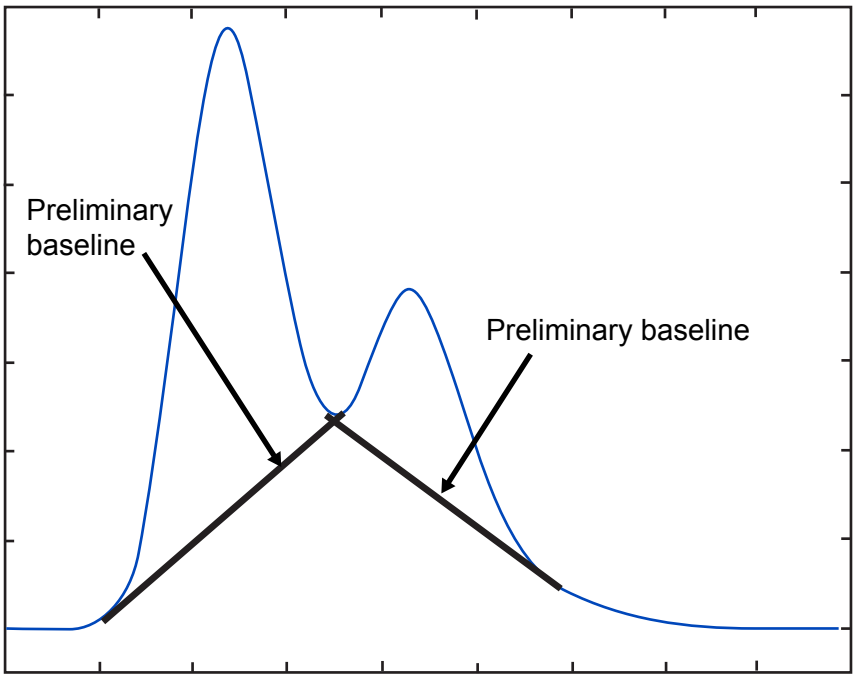
2.3.3 How ApexTrack locates the preliminary baseline for a cluster

ApexTrack uses a slightly different method to determine the baseline for cluster peaks.

To determine the preliminary baseline for cluster peaks:

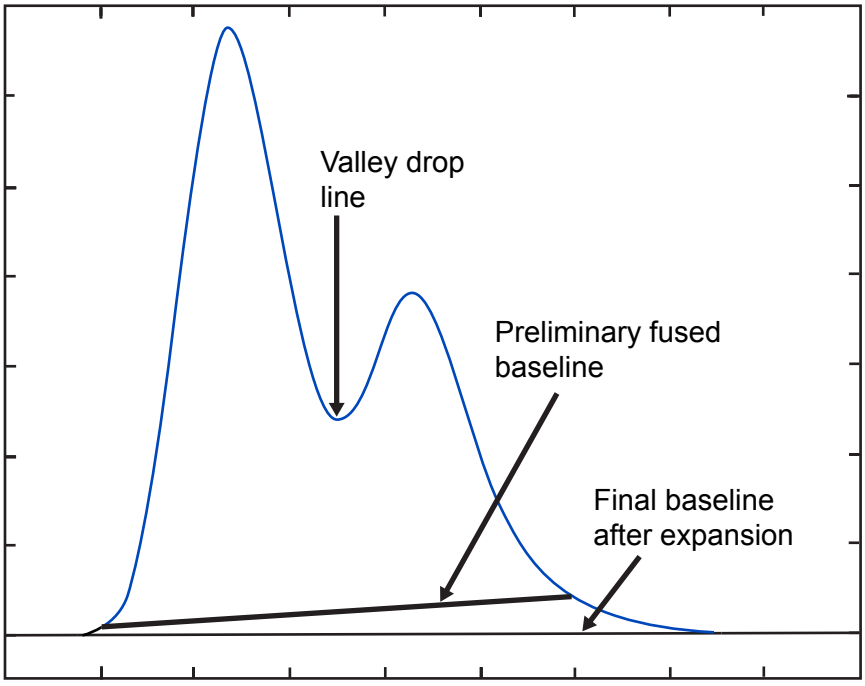
1. ApexTrack expands each peak's baseline until its ends meet the slope difference threshold criteria. If peaks are not resolved as the baselines expand, they overlap. The following figure shows the preliminary overlapped baselines for a two-peak cluster.

Figure 2–15: Preliminary baselines in cluster peaks:



2. Identifies the valleys by the overlap of the expanded baselines.
3. Replaces the two overlapped preliminary baselines with one fused baseline that starts at the beginning of the first preliminary baseline and ends at the last preliminary baseline on the cluster.

Figure 2–16: Cluster baselines:



2.3.4 How ApexTrack determines the final cluster baseline

After a baseline is fused, the slope difference thresholds are tested at the beginning and end of the cluster baseline.

If the slope difference thresholds have not been met, ApexTrack behaves as follows:

1. Expands the cluster baseline as before.
2. Stops the expansion when the slope difference thresholds are met:
   - Slope difference threshold $T_{start} = (\Delta m_1 \times \text{Liftoff \%})/100$
   - Slope difference threshold $T_{end} = (\Delta m_2 \times \text{Touchdown \%})/100$
3. Positions valley drop lines at the point of minimum height above the final baseline.

2.3.5 Effect of Liftoff % and Touchdown % on baseline location

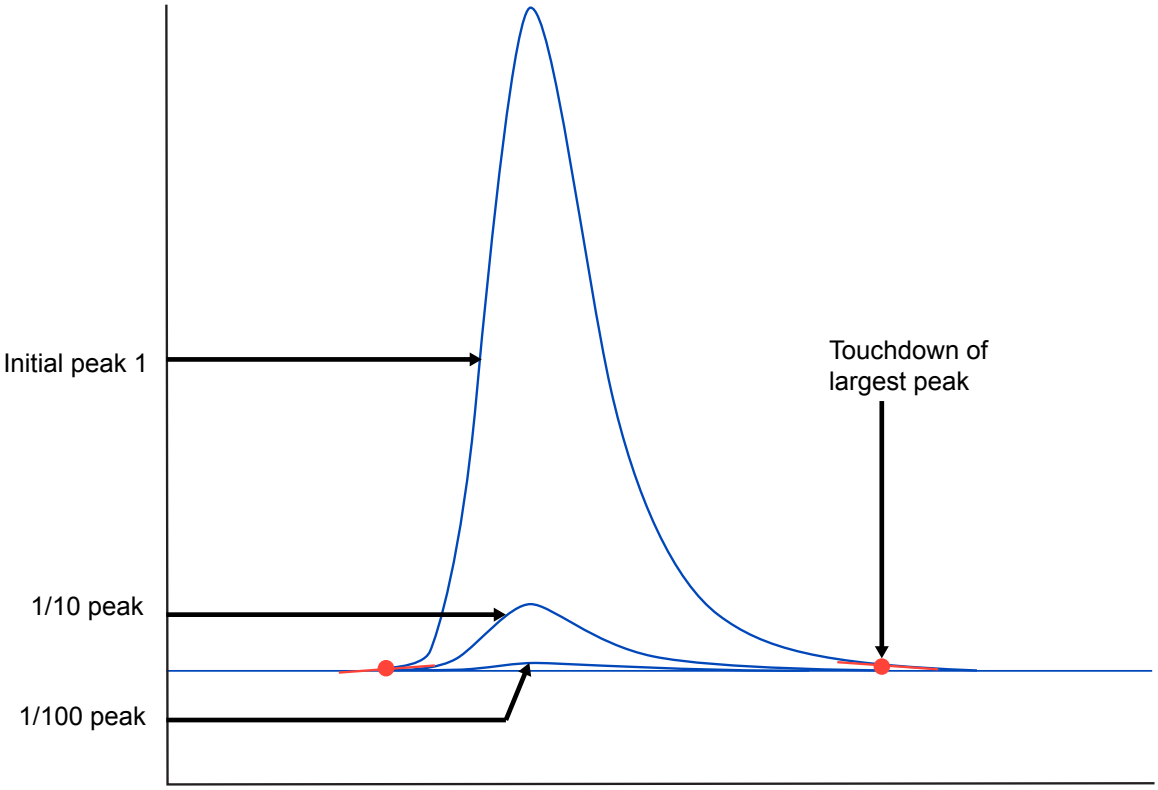
The location of the baseline is controlled by Liftoff % and Touchdown %. If both are set to 0.000%, the resulting baselines are tangent to the detector baseline. If both are set to 1.000%, then the slope difference thresholds are 1.000% of the inflection points' slope differences (Δm_1 and Δm_2). The resulting baselines terminate at about 1.000% of the peak's height. If both are set to 100%, the baseline used for each peak is its inflection point baseline.

Because ApexTrack uses a percentage to calculate the slope difference threshold, the threshold computed for a series of peaks is proportional to peak height. Thus, big peaks have big slope difference thresholds and small peaks have small slope difference thresholds, which allows a single method to successfully integrate peaks of varying sizes using the same Liftoff % and Touchdown % values.

The following figure shows an example of peaks whose height ratios – 1, 1/10, and 1/100 – use the default values Liftoff % = 0.000 and Touchdown % = 0.5000.

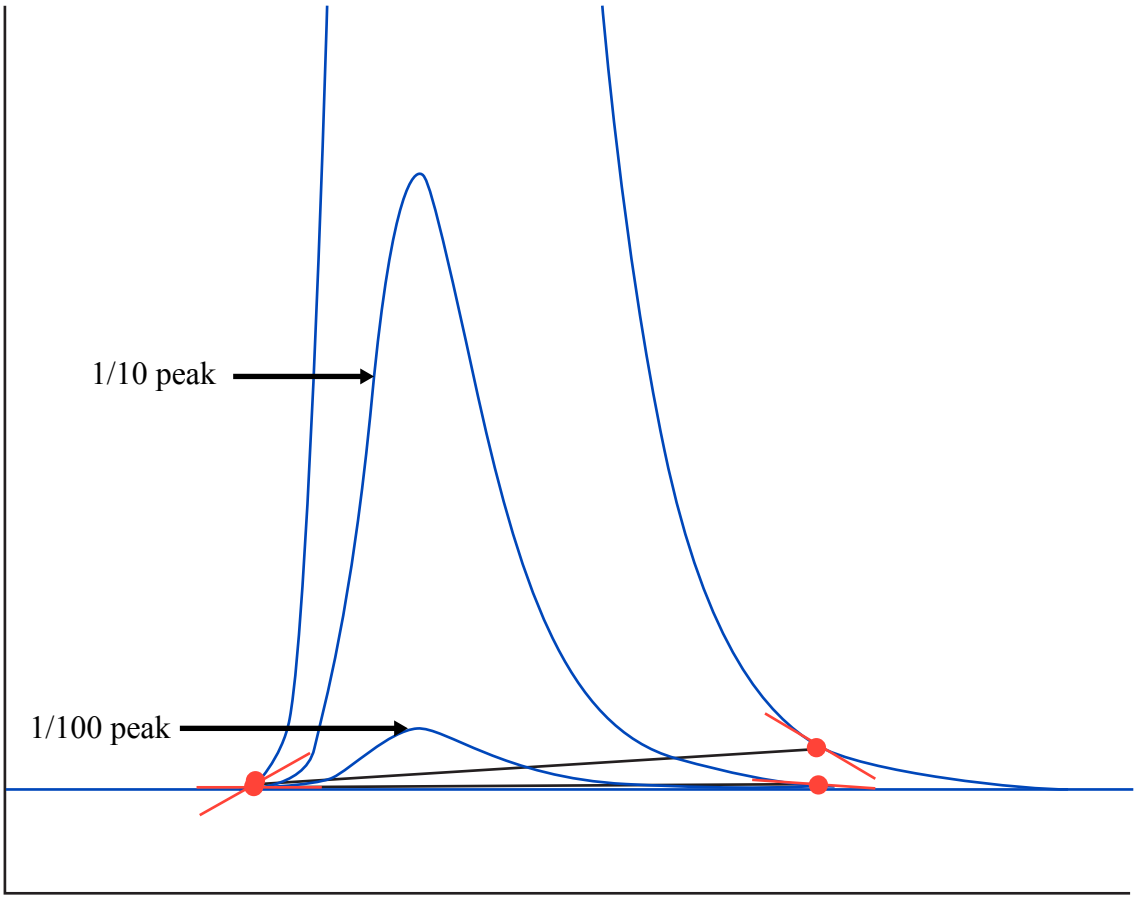
- Liftoff is the same for each peak.
- Touchdown is the same for each peak.
- Touchdown is well positioned for each peak.

Figure 2–17: Touchdown of largest peak:



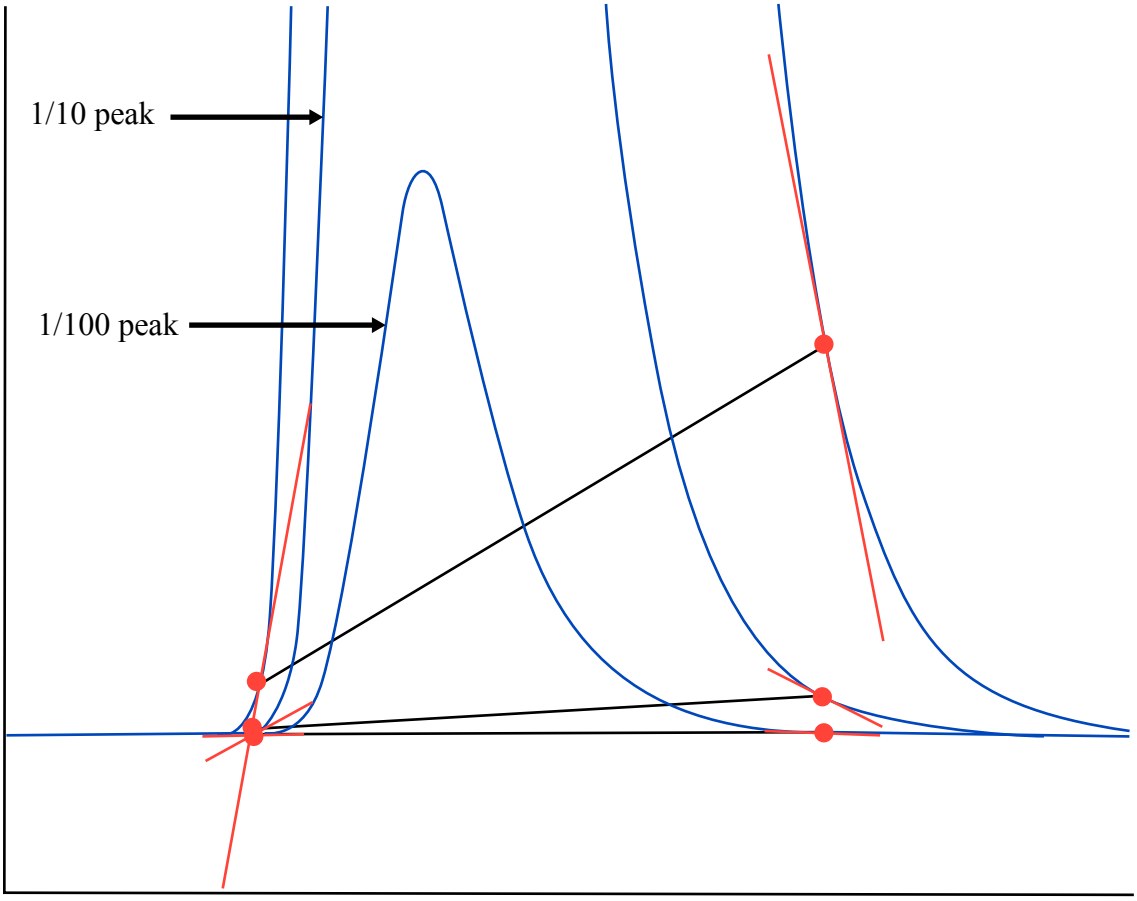
Zooming in to focus on the middle (1/10) peak, touchdown is well positioned, although the slopes are different. The end point for the middle peak occurs at the same relative point to the peak tail.

Figure 2–18: Touchdown of 1/10 peak:



Zooming in to focus on the smallest (1/100) peak, touchdown is well positioned, although the slopes are different. The end point for the third peak occurs at the same relative point to the peak tail. The slope difference threshold for the largest peak is 100 times that of the smallest peak.

Figure 2–19: Touchdown of 1/100 peak:



2.3.6 Effect of changing Liftoff % and Touchdown % on cluster peaks

Changing the **Liftoff %** and **Touchdown %** changes the baseline location and can cause the time of a valley drop line to change. This is because the drop line is set at the lowest point relative to the current baseline.

Changing **Liftoff %** and **Touchdown %** can change shoulder drop lines to valley drop lines and the opposite way. Determining whether a drop line is a shoulder or valley depends on the slope of the baseline. If there is a minimum between the two adjoining peak apices with respect to the current baseline, the drop line is a valley boundary (V). If there is no minimum, the drop line is a shoulder boundary (S). Shoulder drop lines can become valley drop lines (and the opposite way) when changes in **Liftoff %** and **Touchdown %** change the slope of the baseline.

If **Detect Shoulders** is not enabled, peaks can be added or deleted as you change **Liftoff %** and **Touchdown %**. The determination of whether a peak is a shoulder is made using the current baseline. If **Detect Shoulders** is not enabled, a shoulder peak will not appear. However, if you change **Liftoff %** and **Touchdown %**, the slope of the baseline changes. The boundary of that peak may appear as a valley with respect to the new baseline and the peak will appear. In

general, the disappearance of a shoulder boundary has the effect of combining the two adjoining peaks into one.

2.4 Determination of peak boundaries

After detecting the apices and locating the baselines, ApexTrack identifies the start and stop of each peak. The default boundaries are baseline and valley. If a peak is baseline-resolved, the start and stop are the baseline's end points and are labeled by a B. If a peak is in a cluster, the boundaries between peaks are vertical drop lines placed at valley points and are labeled "V" (see Integration peak labels in ApexTrack).

Timed events enable the following additional types of boundaries:

- Detect Shoulders – In regions where shoulder detection is enabled, the boundaries for shoulder and round peaks are vertical drop lines labeled "S" and "R".
- Gaussian Skim – In regions where Gaussian skimming is enabled, a Gaussian profile replaces vertical drop lines and the new peak boundary is labeled "G".
- Tangential Skim – In regions where tangential skimming is enabled, a line drawn tangentially to skim (front or rear) the rider peaks or peaks from the parent peak replaces vertical drop lines, and the new peak boundary is labeled "T".
- Negative Peaks – If a peak cluster contains only negative peaks and **Allow Negative Peaks** is enabled, peak start and end boundaries are labeled "B" and "V". If **Detect Shoulders** or **Gaussian Skim** is enabled in these regions, the S, R, and G boundaries are also allowed.
- Crossover – If a cluster contains positive and negative peaks, the chromatographic signal will intersect the baseline between these adjoining peaks. In this case, the boundary is a crossing point and is labeled "X".

2.5 Computing integration results

Once the apices are detected, the baselines are placed, and the boundaries are identified, ApexTrack obtains the integration results for each peak. Peak area, retention time, and height are all computed using the baseline-corrected signal.

If one or both of a peak's boundaries is a Gaussian skim, then a portion of the peak profile or baseline is replaced by the skim before baseline correction. For example, if a peak generates a skim, the skim profile replaces the responses for the larger (parent) peak between the start and stop time of the skim. This same profile becomes the baseline of the adjoining, smaller (child) peak that is being skimmed.

2.5.1 Peak area

The software uses a trapezoidal calculation to determine peak area. The contribution to peak area from each adjoining pair of sample points is the average of the baseline-corrected responses at those sample points, multiplied by the sample period (the time between the adjoining sample points, Delta T).

2.5.1.1 Calculating peak area

Data can be acquired with or without time values. The software calculates peak area using the following guidelines:

Table 2–3: Guidelines for calculating peak area

Data	Delta T calculation
No time (X) values	ApexTrack and Traditional: Delta T is the inverse of the sampling rate.
Both X and Y values	ApexTrack: Delta T is calculated for each pair of data points and is used to calculate area for those data points. Traditional: The average Delta T across the entire chromatogram is used to calculate area for each pair of data points.

2.5.2 Peak height

Peak height is the value of the baseline-corrected response at the retention time.

2.5.3 Retention time

ApexTrack determines the retention time and height in one of four ways, depending on the peak boundaries and the properties of the peak shape:

- 5-point fit of a quadratic curve to the points at the peak apex.
- 3-point fit of a quadratic curve to the points at the peak apex.
- Time of the second derivative apex.
- Time of the highest point.

Tip:

- Under most conditions, the 5-point quadratic fit is used to determine a peak's height and retention time and no processing code is reported.
- The 3-point fit and the time of the second derivative are unique to ApexTrack and are not implemented in traditional integration.
- In ApexTrack integration, the 3-point and 5-point fit is to the baseline-corrected signal.

2.5.4 Rules that determine which retention time method is used

For each peak, ApexTrack carries out a hierarchy of tests to determine the retention time of each identified peak apex.

The tests and the order in which they are done is as follows:

1. The time of the second derivative apex is used as the retention time as follows:
   - When either boundary of the peak is a round (R).
   - When the highest point on the baseline-corrected signal is at a peak boundary. In general, this approach ensures that the retention time of a shoulder peak is obtained from the second derivative apex.
     The processing code I20 is included in the integration result of the peak whenever the retention time and height reported in the result are calculated at the second derivative apex.
     If the retention time of the second derivative apex cannot be used because it falls outside the peak boundary, the retention time of the highest point is used instead. The processing code I23 is included in the integration result, signifying that the attempt at using the second derivative apex retention time failed.
2. If the peak does not fit the criteria for the first test, the 3-point fit is used to determine the retention time as follows:
   - When there are fewer than four sample points within the inflection point width of the peak.
     The processing code I19 is included in the integration result of the peak whenever the retention time and height reported in the result are calculated from a 3-point fit.
     If the retention time falls outside the 3 points used for the fit, the second derivative apex value is used, and the processing code I22 is included in the integration result to indicate that a 3-point fit was attempted but failed. The processing code I20 is included in the integration result of the peak whenever the retention time and height reported in the result are calculated at the second derivative apex.
     If the retention time of the second derivative apex falls outside the peak boundary, the retention time of the highest point is used instead. The processing code I23 is included

in the integration result, signifying that the attempt at using the second derivative retention time failed.

3. If the peak does not fit the criteria for either test, the 5-point fit is used to determine retention time. No processing code is reported. The 5-point fit might fail if these conditions apply:

   - The first or last point of the 5 points used for the fit lies outside the start and stop times of the peak.
   - The retention time from the fit falls outside the 5 points used for the fit.
     The processing code I21 is included in the integration result signifying that a 5-point fit was attempted but failed. In either case, a 3-point fit is then attempted. The processing code I19 is included in the integration result of the peak whenever the retention time and height reported in the result are calculated from a 3-point fit.
     If the retention time from the 3-point fit falls outside the 3 points used for the fit, then the processing code I22 is included in the integration result, signifying that a 3-point fit was attempted but failed. In this case, the second derivative value is attempted. The processing code I20 is included in the integration result of the peak whenever the retention time and height reported in the result are calculated at the second derivative apex.
     If the retention time of the second derivative apex falls outside the peak boundary, the retention time of the highest point is used instead. The processing code I23 is included in the integration result, signifying that the second derivative retention time was attempted but failed.

2.5.5 Retention time and height values for a manually adjusted peak

The software uses the baseline location and peak boundaries to determine a peak's retention time, height, and area. When a peak is manually integrated, its manual baseline and peak boundaries are used to determine retention time, area, and height. When a manually determined baseline location and peak boundaries coincide with the automatically determined values, the manually determined retention time, height, and area values are identical to the automatically determined values.

Tip: If retention time and height are computed using the second derivative apex, an I20 peak code is reported. If an I20 peak is manually adjusted, the second derivative value is unavailable, so the software uses a different set of rules to determine how to calculate retention time and height. As a result, the retention time and height is determined by a 5-point fit, a 3-point fit, or by time of the highest point.
If **Detect Shoulders** is enabled, an I20 peak code can occur if a peak is determined to be a shoulder or round. If **Detect Shoulders** is not enabled, an I20 peak code can occur for a narrow, low-level peak.

2.6 References

For further information on the theory of ApexTrack peak detection and integration, see:

ApexTrack Integration: Theory and Application, Waters Corp., Milford, MA, 2016, posted on www.waters.com.

3 Traditional integration

This section describes traditional peak detection and integration theory.

3.1 Features and capabilities

Traditional peak detection and integration by Empower software includes these functions:

- Automatically determining appropriate peak width and threshold values for the chromatogram, unless already set in the processing method (the system policy to "Use V3.0 Style Peak Width and Threshold Determination" must be disabled).
- Detecting peaks in the chromatogram to determine their location.
- Integrating peaks to determine their retention times, areas, and heights.

The processing method defines the parameters (including detection and integration events) that the software uses to detect and integrate the peaks within the raw data file (channel).

3.1.1 Commonalities between ApexTrack and traditional integration

Although the algorithms ApexTrack uses to detect peaks and to determine baselines differ from those used by traditional processing, significant functionality is the same for both traditional and ApexTrack integration:

- Ability to automatically determine peak width and threshold values. (Disabled by default with traditional integration.)
- Timed events such as Inhibit Integration, Tangent Skim, Set Peak Width, Set Minimum Height, Set Maximum Height, Set Minimum Area, and Set Maximum Peak Width are supported.
- Peaks can be manually added or deleted.
- Peak start and stop markers can be manually changed.
- Allow Negative Peaks and Valley to Valley are supported.

3.1.2 Peak detection

The peak detection processes include:

1. Performing data bunching
2. Determining peak start
3. Determining preliminary peak apex
4. Determining peak end
5. Determining peak width and threshold values in the processing method
6. Inhibiting integration

The detection algorithm first determines the presence of peaks by comparing the rate of change of the signal to specific acceptance criteria, determining where peaks in the acquired raw data file start and end. The software must perform these peak detection tests before it can integrate the peaks.

You determine the peak detection test criteria several ways:

- Peak width and threshold selections in the **Integration** tab of the Processing Method window in Review
- Integration toolbar of the Review Main window or the Processing Method Editor
- Processing Method wizard in Review

See also: For additional information on peak detection theory, see References.

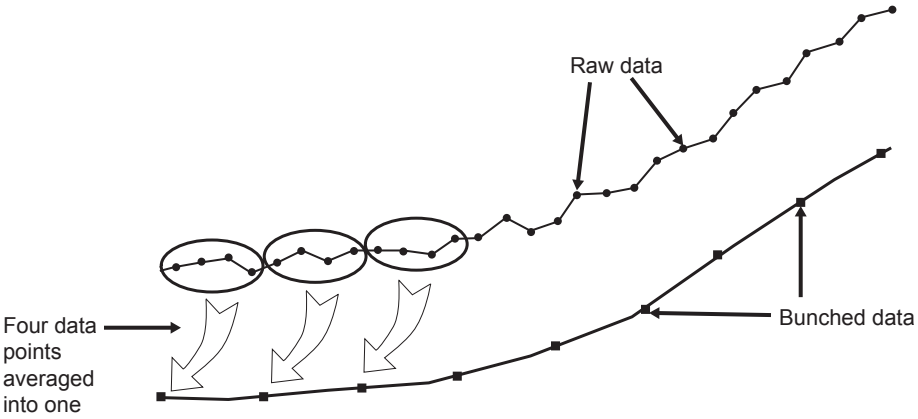
3.1.2.1 Performing data bunching

As the detection algorithm tests the data for a peak, the software averages individual raw data points into discrete groups, or bunches, to produce a single point. The peak width parameter determines the number of data points in a bunch.

In most instances, each bunch of data contains one point when the sampling rate is optimized. Data bunching has no effect on the acquired raw data. It is an internal calculation used to enhance the process of determining peak start and peak end. When the peaks contain more data points than necessary, the bunched data points are used only to detect peaks; all raw data points are used for integration. The software uses all the points across a peak during integration.

The following figure illustrates the effects of data bunching on a noisy signal. In this example, with the peak width set to 60 and sampling rate set to 1, the detection algorithm produces a bunched data point for each set of four raw data points. This combination of parameter values optimizes the number of bunches to 15 (across a 60-second peak containing 60 data points) and effectively smooths the data.

Figure 3–1: Data bunching example:



During detection, the software calculates the number of points in a bunch using the equation:

$$PB = \frac{(PW \times SR)}{15}$$

where:

PB = Points in a bunch

PW = Peak width (in seconds)

SR = Sampling rate (data points/second as specified in the instrument method used for acquisition)

Tip: The peak detection algorithm functions most effectively with 15 data points across each peak. For this reason, the software organizes the raw data into 15 discrete bunches when setting the peak-width value. When peak width is 15 and the sampling rate is 1, no data point bunching occurs; all data points are used to detect peak start and peak end.

3.1.2.2 Determining peak start

The threshold specified in the processing method defines the minimum slope of the signal in µV/sec, at or above which the start of a peak is detected.

Tip: By default, negative peaks are not detected. To activate the Allow Negative Peaks event, see Using timed events.

To determine peak start, the software's detection algorithm behaves as follows:

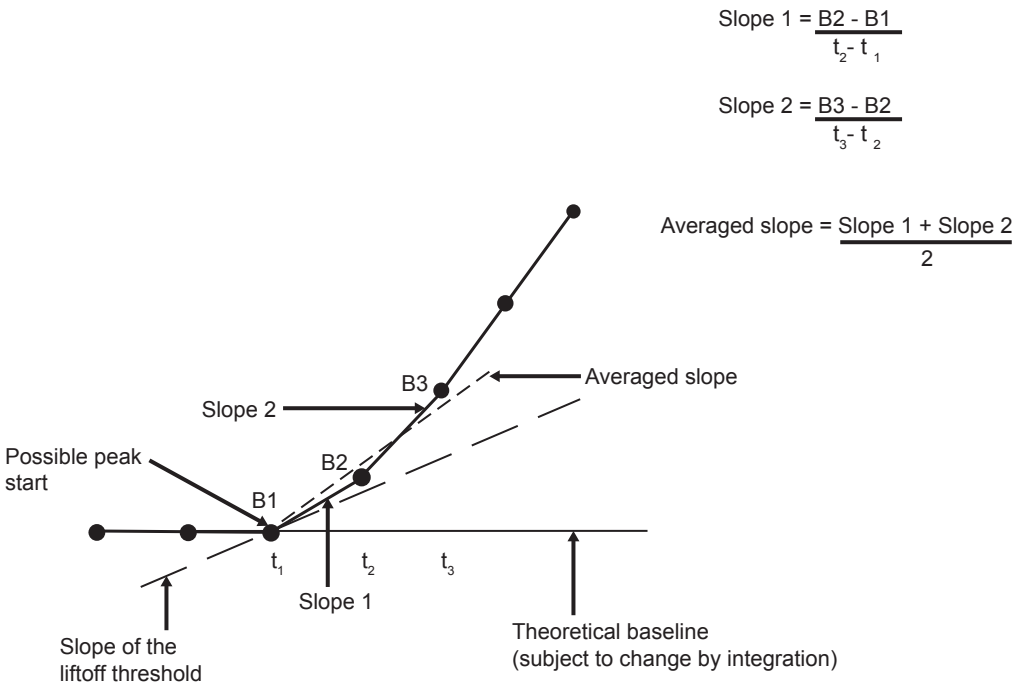
1. It performs the threshold test on the signal:

- The software averages the signal slope across two data bunch intervals and then compares it to the threshold.
- When the averaged slope of the signal between bunches B1 and B3 is greater than, or equal to, the threshold value, the software flags B1 as the possible peak start.

2. It examines the individual points in the B1 bunch to determine the actual start point. For positive peaks, this is the data point with the minimum Y-value. For negative peaks, this is the data point with the maximum Y-value.

The preliminary start point of a detected peak can be affected by an Inhibit Integration event that ends near the start of the peak. Because the bunched point that would have been selected as the preliminary start point of the peak is inside the Inhibit Integration event, a different bunched point is selected as the preliminary start point, even when the “bunch” contains only one data point.

Figure 3–2: Determining peak start:



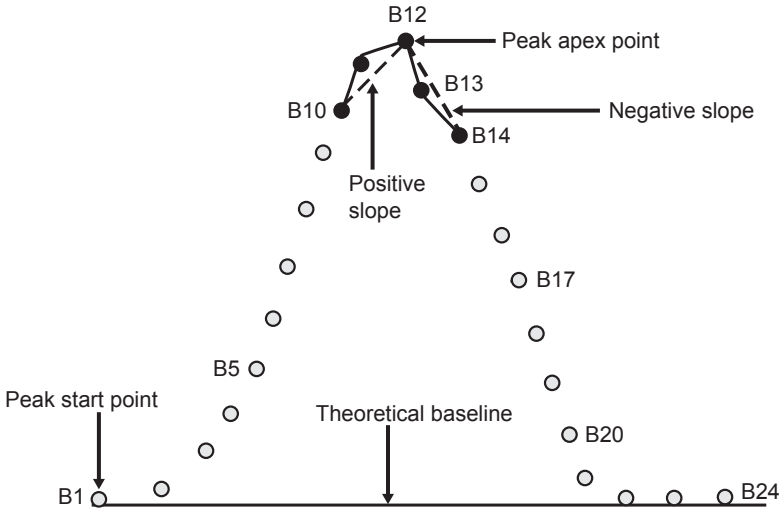
3.1.2.3 Determining preliminary peak apex

To determine the preliminary peak apex (peak maximum) after the peak start is confirmed, the software performs as follows:

1. It monitors the signal until the slope changes sign from positive to negative. For a negative peak, the slope changes sign from negative to positive.
2. It analyzes the bunch where the slope change occurs (bunch B12 in the next figure) and assigns a tentative peak apex to the data point within the bunch that is farthest away from the theoretical baseline.

Tip: This peak apex is preliminary because the software does not determine the actual peak apex until integration occurs and baselines are assigned.

Figure 3–3: Determining the preliminary peak apex:



The preliminary apex of a detected peak can be affected by an Inhibit Integration event placed too close to the peak apex. Because the bunched point that would have been selected as the preliminary peak apex is inside the Inhibit Integration event, the peak can go undetected.

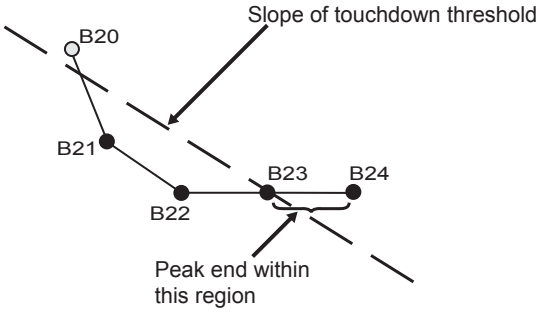
3.1.2.4 Determining peak end

To determine the peak end, the software performs as follows:

1. It compares the slope of the signal to the threshold. When two consecutive slopes are less than the threshold value, the algorithm flags the last data point in the last bunch as the possible peak end.
2. It examines the individual data points in the current and next bunch to determine the actual peak end. For positive peaks, this is the data point with the minimum Y-value. For negative peaks, this is the data point with the maximum Y-value.
3. During the peak-end test, it checks for a change in the sign of the slope. A change in sign before the touchdown indicates a preliminary peak valley (the end point of the current peak and the start point of the next peak).

   Tip: This peak end point and start point is preliminary because the software does not determine the actual data point for the end of a peak until the integration process occurs and baselines are assigned.
4. It proceeds from this peak start point to the peak apex test and continues until it successfully determines peak touchdown.

Figure 3–4: Determining peak end:



The preliminary end point of a detected peak can be affected by an Inhibit Integration event that starts near the end of the peak. Because the bunched point that would have been selected as the preliminary end point of the peak is inside the Inhibit Integration event, a different bunched point is selected as the preliminary end point, even when the “bunch” contains only one data point.

3.1.2.5 Determining peak width and threshold values

The software uses a second derivative to set the peak width and threshold values in the processing method automatically when you disable the system policy to "Use V3.0 Style Peak Width and Threshold Determination".

Default: The system policy "Use v3.0X Style Peak Width and Threshold Determination" is enabled by default.

Tip: You can use the Peak Width and Threshold determination method that was used in all Millennium^{32} software, by clicking **Configuration Manager** > **View** > **System Policies**, and then clicking **Use v3.0X Style Peak Width and Threshold Determination** in the **Data Processing** tab (see the Empower Online Information System). When this system policy is active, the **Peak Width** and **Threshold** buttons, Processing Method wizard, processing methods, and results all function as they do in all Millennium^{32} software.

To determine the peak width value (when the system policy to "Use V3.0 Style Peak Width and Threshold Determination" is enabled):

1. Zoom on the narrowest peak of interest.
2. Click and drag the mouse in the chromatogram to draw a baseline under the peak from liftoff to touchdown.
3. Click **Set Processing Method Peak Width**. The new value replaces the previous value.

To determine the threshold value (when the system policy to "Use V3.0 Style Peak Width and Threshold Determination" is enabled):

1. Zoom on a section of the baseline that contains only noise.
2. Click **Set Processing Method Threshold**. The new value replaces the previous value.

3.1.2.6 Peak width parameter

When the system policy use v3.0 style peak width and threshold determination is disabled, the software automatically determines the peak-width value (Auto-Peak Width) using the inflection points of the second derivative of the peak that exhibits the highest second derivative within a chromatographic region.

Since the software uses the peak-width value to determine a bunching factor during peak detection (see Performing data bunching), this value affects the sensitivity of peak detection. The guideline is to use a peak width value within a range of plus-or-minus two times the software-determined peak-width value.

If the signal-to-noise (S/N) ratio is acceptable, the peak-width value at the high end of this range can increase sensitivity and allow relatively small peaks to be properly integrated. However, shoulders on larger peaks, if present, might no longer be detected. Increasing the peak-width value above this range results in a decrease in sensitivity.

The valid range of the peak width setting is 0.01 through 9999.99. The default peak-width setting is blank.

There are several ways to set a peak-width value in Review:

- When using the Processing Method wizard in Review to create a new processing method or edit an existing one, the software automatically determines an appropriate peak width using the data contained within the zoomed region (in the Integration – Integration Region wizard page).
- When viewing data in the Review Main window, clicking the **Peak Width** button in the **Integration** toolbar automatically sets the peak-width value to the peak with the highest second derivative within the current zoom region (which may be the entire chromatographic region).
- With no peak width set in the active processing method, you can integrate the data by clicking **Process** > **Integrate** (or clicking the **Integrate** button in the toolbar). The peak width is automatically set according to the data in the entire chromatogram (unless there is an Inhibit Integration event at the start, end or both of the chromatogram).

The region of the chromatogram used to set peak width starts at the beginning of the chromatogram or the stop time of an Inhibit Integration event that starts at the beginning of the chromatogram. The region of the chromatogram used to set peak width ends at the end of the chromatogram or the start time of the Inhibit Integration event that stops at the end of the chromatogram.

Inhibit Integration events that do not overlap the beginning or end of the chromatogram are ignored when setting peak width.

Tip: When using this method, the peak width is placed in the **Result Peak Width** field only. The **Processing Method Peak Width** field remains blank. To copy the Result Peak Width value to the **Method Peak Width** field, click **Copy to Processing Method** from the right-click menu.

- Set the peak width value manually by entering a value in the **Integration** toolbar of the Main window of Review or in the **Integration** tab of the Processing Method window.

Tip: If the peak with the highest second derivative is fused, the peak width value might not be optimal. In such cases, zoom in on peaks other than the fused peak when setting the peak width parameter.

3.1.2.7 Threshold values

When the system policy use v3.0 style peak width and threshold determination is disabled, the software automatically determines the threshold value (Auto-Threshold) by first applying a median filter to the second derivative of the chromatographic data to determine the noise. The software then derives the threshold value by multiplying the second derivative noise by the current peak width value.

The threshold value is a slope measurement the software uses to determine peak start and end points during peak detection (as described in Determining peak start and Determining peak end). A relatively low threshold value increases sensitivity and may allow relatively small peaks to be properly integrated. If too many small, baseline noise peaks are being integrated, increasing the threshold value can prevent these small peaks from being integrated.

The software normally uses the global threshold value in the processing method, to determine both peak start (liftoff) and peak end (touchdown). If you need to use a different threshold value for peak starts or ends due to a tailing or a sloping baseline, use the Set Liftoff or Set Touchdown event.

The valid range of the threshold setting is 0.0 or greater. The default threshold setting is blank.

There are several ways to set a threshold value in Review:

- When using the Processing Method wizard to create a new processing method or edit an existing one, the software automatically determines an appropriate threshold using the data in the zoomed region in the Integration – Integration Region wizard page.
- When viewing data in the Review Main window, clicking the **Threshold** button in the **Integration** toolbar automatically sets the threshold value using data in the current zoom region (which can be the entire chromatographic region).

  Tip: **Set Processing Method Threshold** is disabled when the **Processing Method Peak Width** field is blank.
- With no threshold set in the active processing method, you can integrate the data by clicking **Process** > **Integrate** (or clicking **Integrate** in the toolbar). The software automatically sets the threshold according to the data in the entire chromatogram (unless there is an Inhibit Integration event at the start, end or both of the chromatogram).
  The region of the chromatogram used to set the threshold starts at the beginning of the chromatogram, or the stop time of an Inhibit Integration event that starts at the beginning of the chromatogram. The region of the chromatogram used to set the threshold ends at either the end of the chromatogram, or the start time of the Inhibit Integration event that stops at the end of the chromatogram.
  Inhibit Integration events that do not overlap the beginning or end of the chromatogram are ignored when setting threshold.

Tip: When using this method, the determined threshold is placed in the **Result Threshold** field only. The **Processing Method Threshold** field remains blank. To copy this value to the **Method Threshold** field, click **Copy to Processing Method** from the right-click menu.
A peak-width value is required before determining a threshold value. If the processing method does not include a peak-width value, the threshold button is not available. If you use this method to perform integration without a peak width value, the software first determines the peak width value and then the threshold value. Both values are automatically placed in their respective toolbar fields.

- You can set the threshold value manually by entering a value in the Integration toolbar of the Main window of Review, or in the **Integration** tab of the Processing Method window.

3.1.2.8 Peak Width and Threshold fields

The peak width and threshold values are reported as both method and result fields. These fields are in the **Integration** toolbar of the Review Main window, and the result fields are available for reports. The method fields report the peak width and threshold values from the processing method. The result fields report the peak width and threshold values used when the raw data was processed.

During processing, the software uses the values in the **Processing Method Peak Width** and the **Processing Method Threshold** fields. It then stores these values in the **Result Peak Width** and **Result Threshold** fields. In this case, the **Result Peak Width** and the **Result Threshold** fields are the same as the **Processing Method Peak Width** and the **Processing Method Threshold** fields. If the **Processing Method Peak Width**, the **Processing Method Threshold**, or both are blank, then the software determines the **Result Peak Width**, **Result Threshold**, or both fields during data processing.

When data is processed using a processing method that contains a blank **Processing Method Peak Width**, **Processing Method Threshold**, or both, each result can be produced using a different **Result Peak Width** and **Result Threshold**.

Tip: You can disable the Auto-Peak Width and Auto-Threshold determinations by clicking **Configuration Manager** > **View** > **System Policies**, and then clicking **Use v3.0X Style Peak Width** and **Threshold Determination** in the **Data Processing** tab (see the *Empower Online Information System*). When this system policy is active, the **Peak Width** and **Threshold** buttons, the Processing Method wizard, processing methods, and results all function as they do in all Millennium^{32} software.

Default: The system policy "Use v3.0X Style Peak Width and Threshold Determination" is enabled by default.

3.1.3 Peak integration

This section describes the following peak integration processes:

- Determining fused peaks
- Constructing the baseline
- Calculating peak retention time, height, and area

Integration uses the peak start and peak end values identified during peak detection to determine baselines and integrate isolated and fused (clustered) peaks.

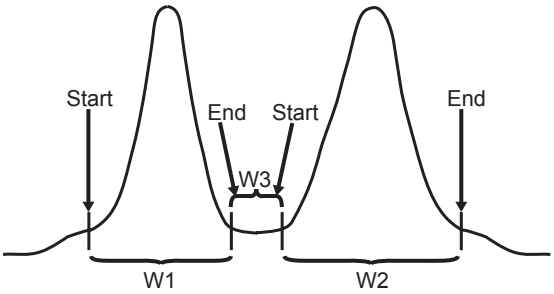
If your chromatograms are complicated, you can enable time-based integration events to refine peak integration.

See also: For in-depth information on peak integration theory, see References.

3.1.3.1 Determining fused peaks

The first process in integration is distinguishing any fused and isolated peaks in the chromatogram. The software checks the distance between adjacent peaks.

Figure 3–5: Adjacent peak width comparison:



3.1.3.1.1 Determining fused peaks

When determining fused peaks, the traditional integration algorithm operates as follows:

1. It compares the width of the space between the detected start and end points of adjacent peaks (W3) to the width of the wider adjacent peak (either W1 or W2).
2. It locates the wider adjacent peak (W2 > W1).
3. It computes the ratio of the wider adjacent peak to the space between the two peaks (W3) using the equation W2/W3. If the ratio is greater than or equal to 3.0, the peaks are considered fused. If the ratio is less than 3.0, the peaks are considered resolved.

Tip: The software uses the ratio of 3.0 to increase the chances of detecting peak overlap.

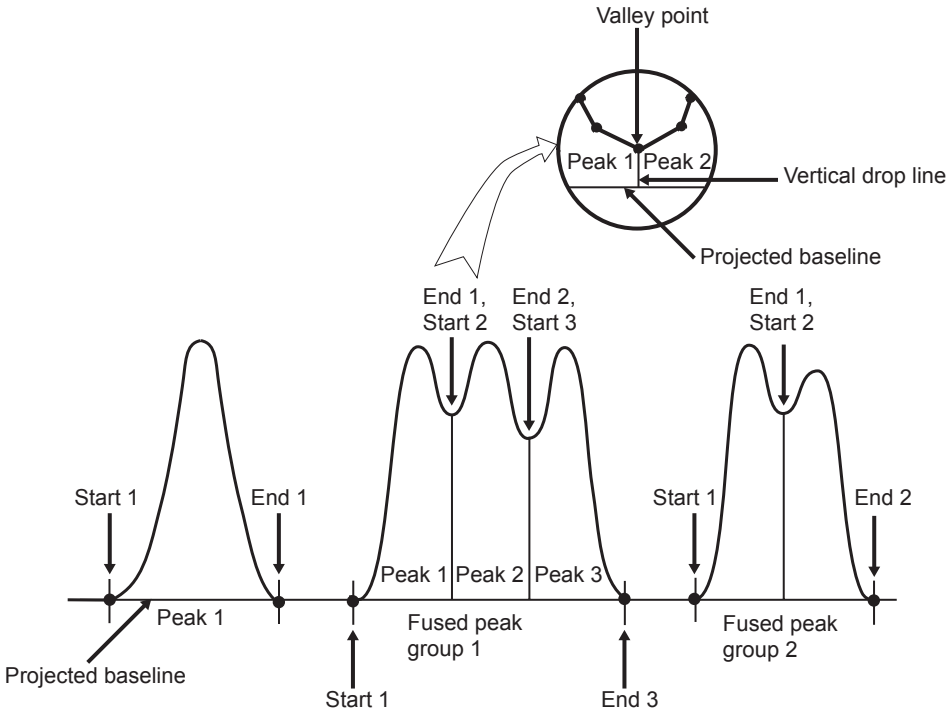
3.1.3.1.2 Setting the valley point

To set the valley point between fused peaks, the software operates as follows:

1. It draws a projected baseline from the start point of the first peak in the cluster to the end point of the last peak in the cluster.
2. It searches for the valley point between each pair of adjacent fused peaks, and chooses the raw data point closest to the projected baseline as the valley point. The software adjusts the end point of the peak preceding the valley to the time of the valley point. Similarly, it adjusts the start point of the peak following the valley to the time of the valley point.
3. It draws a vertical line from the valley point to the projected baseline, thereby separating the peaks.

In the next figure, for example, the integration algorithm locates two fused peak groups and a total of six peaks within the chromatogram.

Figure 3–6: Determination of resolved and fused peaks:



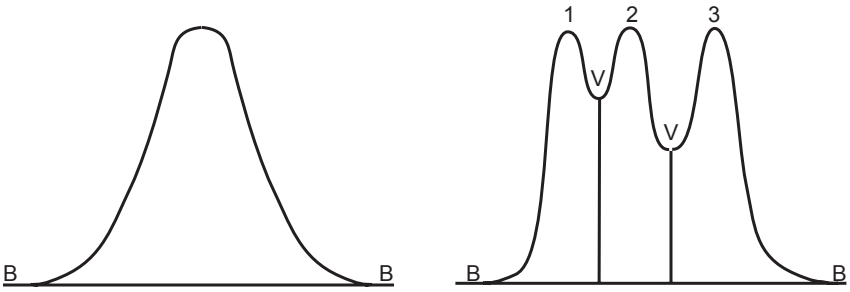
3.1.3.2 Constructing the baseline

Once resolved and fused peaks are identified within the chromatogram, the integration algorithm draws a baseline from the start to the end of each peak or fused peak group.

The fields **Start Time** and **End Time** display the start and end times of the peak calculated during peak integration. The fields **Baseline Start** and **Baseline End** display the start and end times of the baseline used to integrate a peak. The **Baseline Start** and **Baseline End** values are as follows:

- The same as the **Start Time** and **End Time** values when integration is for a baseline-resolved peak (baseline-to-baseline).
- Different from the **Start Time** and **End Time** values when integration is for a fused peak (peaks not baseline-resolved).

Figure 3–7: Baseline construction:



BB = Baseline-to-baseline

Peak 1 = BV (baseline-to-valley)
Peak 2 = VV (valley-to-valley)
Peak 3 = VB (valley-to-baseline)

When you use the default integration setting, each identified peak is given a two-character label that indicates whether the peak starts or ends at a point on the baseline (B) or in a valley (V) above the baseline. A peak can have four types of baseline construction. The label appears in the **Int Type** column of the **Peaks** tab of the Results and Main windows of Review.

Table 3–1: Default integration peak labels:

Peak start and end point	Label
Baseline-to-Baseline	BB
Baseline-to-Valley	BV
Valley-to-Baseline	VB
Valley-to-Valley	BB

Tip: When you use the Exponential Skim or Tangential Skim integration events, additional types of baseline construction can appear (see Integration peak labels).

Capitalization of the label indicates the following conditions:

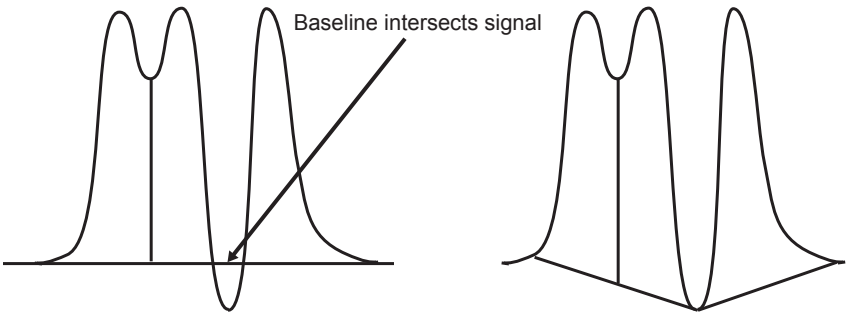
- Capital letters – The integration was performed automatically by the software.
- Lowercase letters – The integration was performed manually.

For instance, a baseline label of Bb indicates that while the peak start and peak end are both baseline-resolved, the peak start was automatically integrated by the software and the peak end was manually adjusted.

3.1.3.3 Baseline adjustment

If the projected baseline intersects the signal in the chromatogram, the software adjusts the baseline to the lowest point within the fused peak group, separating the peak group into individual or fused peaks, as appropriate. The software then rechecks the new baselines to ensure that they do not intersect the chromatographic signal except at peak start or end points, and it readjusts the baseline as necessary.

Figure 3–8: Baseline adjustment:



3.1.3.4 Calculating peak retention time, height, and area

Once actual baselines are constructed, the integration algorithm operates as follows:

1. It calculates the retention time, height, and area for each peak.
2. It compares each integrated peak to the minimum area and minimum height rejection criteria you specify in the processing method.

3.1.3.5 Retention time and height

To determine retention time and height, the software operates as follows:

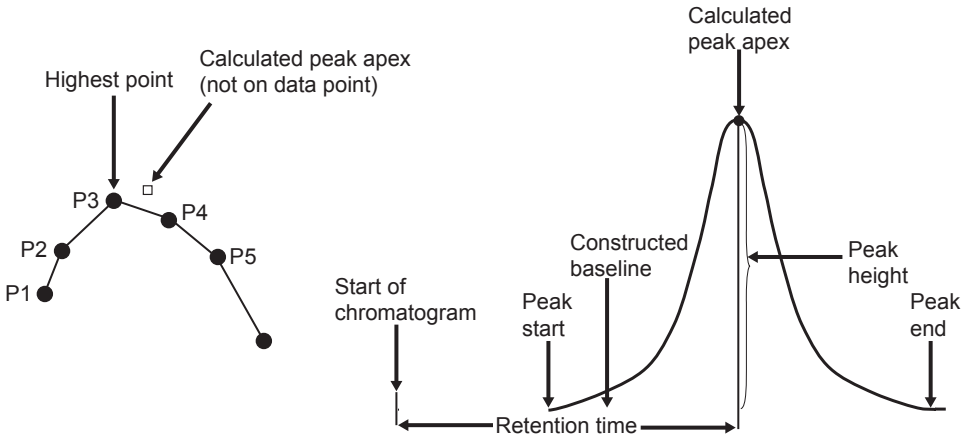
1. Locates the retention time of the data point in the peak farthest from the constructed baseline.
2. Fits a quadratic curve to the five points at the top of the peak (the highest data point and the two data points on either side of this point).
3. Sets the peak apex point to the inflection point of the fitted curve. The X-value of the peak apex is the retention time of the peak.
4. Calculates the peak height as the distance (in µV) from the constructed baseline to the Y-value of the calculated peak apex.

Tip: If the software fails to fit a curve to the top of the peak, it uses the apex point (the data point farthest from the baseline) to calculate retention time and height as in Millennium software

version 2.15 and earlier. The software adds a processing code (I05, I06, I07, or I08) to the **Codes** column in the **Peaks** tab in Review to explain why the curve fitting to the top of the peak failed.

The following figure illustrates the peak retention time and peak height calculation.

Figure 3–9: Peak retention time and peak height calculation:



Tip: You can disable fitting a quadratic curve to the top of the peak by clicking **Configuration Manager** > **View** > **System Policies** and implementing a system policy named *Use v2.XX Style Retention Time Calculations* (see the *Empower Online Information System*) in the **Data Processing** tab. When results are integrated with this system policy active, processing code I09 is added to the **Codes** field in the result. This field is visible in the chromatogram **Result** table, in the Result window of Review or in the Project window's **Results** tab.

3.1.3.6 Area

The software uses a trapezoidal calculation to determine peak area. The contribution to peak area from each adjoining pair of sample points is the average of the baseline-corrected responses at those sample points, multiplied by the sample period (the time between the adjoining sample points, Delta T).

3.1.3.7 Calculating peak area

Data can be acquired with or without time values. The software calculates peak area using the following guidelines:

Table 3–2: Guidelines for calculating peak area

Data	Delta T calculation
No time (X) values	ApexTrack and Traditional: Delta T is the inverse of the sampling rate.

Table 3–2: Guidelines for calculating peak area (continued)

Data	Delta T calculation
Both X and Y values	ApexTrack: Delta T is calculated for each pair of data points and is used to calculate area for those data points. Traditional: The average Delta T across the entire chromatogram is used to calculate area for each pair of data points.

3.1.3.8 Peak rejection criteria

When the software integrates a peak, the integration algorithm compares the peak with the integration rejection criteria you specify. You can set the rejection criteria in the **Integration** tab of the Processing Method window in Review by using the **Minimum Area** and **Minimum Height** buttons in the Review Main window, or by using the Processing Method wizard. Based on this comparison, the algorithm accepts or rejects the peak. Integration rejection criteria can include:

- Minimum area
- Minimum height
- Minimum of five points across a peak

3.1.3.9 Minimum area

The minimum area criterion determines the minimum area (in µV • sec) required for an integrated peak to be included in the peak list. If the area of the integrated peak falls below the set value, the peak is removed from the peak list. If the area is equal to or greater than the set value, the peak is accepted.

3.1.3.10 Minimum height

The minimum height criterion determines the minimum height (in µV) required for an integrated peak to be included in the peak list. If the height of the integrated peak falls below the set value, the peak is removed from the peak list. If the absolute value of the height is greater than or equal to the set value, the peak is accepted.

Tip: Minimum Area and Minimum Height parameters are useful for removing small integrated peaks from the result. A high value may cause integrated peaks to be rejected as noise; conversely, a low value may cause baseline noise to be integrated as peaks.

3.1.3.11 5-point peak rejection

The 5-point peak rejection criterion instructs the software to remove from the peak list any peak that contains fewer than five points.

Tip: The 5-point peak rejection criterion is built into the software and occurs automatically. It is not a parameter found in the processing method.

3.1.4 Using timed events

Empower software supports timed events for peak detection and integration.

Tip: Refer to the Empower Online Information System for an overview of all timed events and a description of each.

3.1.4.1 Peak detection events

The software supports the following time-based detection events to further refine peak detection:

- Allow Negative Peaks
- Set Liftoff
- Set Touchdown
- Set Peak Width

Tip: The Set Liftoff, Set Touchdown, and Set Peak Width events take effect only in the baseline regions outside detected single peaks or fused peak groups. If the event starts within an isolated peak or fused peak group, the event takes effect at the end of the isolated or fused peak group.

3.1.4.2 Peak integration events

The software supports the following time-based integration events to further refine peak integration:

- Force Baseline by Time
- Force Baseline by Peak
- Forward Horizontal by Time
- Forward Horizontal by Peak
- Reverse Horizontal by Time
- Reverse Horizontal by Peak
- Valley-to-Valley
- Force Drop Line
- Force Peak
- Exponential Skim
- Tangential Skim
- Set Minimum Height
- Set Minimum Area
- Set Maximum Height
- Set Maximum Width (sec)

3.1.5 Integration peak labels

When you use the default integration setting, each identified peak in a chromatogram is given a two-character label that indicates whether the peak starts or ends at a point on the baseline (B) or in a valley (V) above the baseline. The label appears in the **Int Type** column of the **Peaks** tab of the Results and Main windows of Review.

Integration Peak Labels on a Chromatogram:

Peak Start and End Point	Label
Baseline-to-Baseline	BB
Baseline-to-Valley	BV
Exponential-to-Exponential	EE
Exponential-to-Valley	EV
Tangential-to-Tangential	TT
Tangential-to-Valley	TV
Valley-to-Baseline	VB
Valley-to-Exponential	VE
Valley-to-Tangential	VT
Valley-to-Valley	VV

Capitalization of the label indicates that the integration was performed automatically by the software. Lowercase letters indicate that the integration was performed manually.

3.2 References

For further information on the theory of peak detection and integration, see:

- Dyson, Norman, *Chromatographic Integration Methods*, The Royal Society of Chemistry, Thomas Graham House, Cambridge, 1990.
- Massart, D.L., et al.,*Chemometrics: A Textbook*, Elsevier Science Publishers, Amsterdam, 1988.
- Papoulis, Athanasios, *Signal Analysis*, McGraw-Hill, New York, 1977.
- Snyder, L.R. and J.J. Kirkland, *Introduction to Modern Liquid Chromatography*, second ed., Wiley-Interscience, New York, 1979.

4 Peak Identification and Quantitation of Sample Components

This chapter describes how peaks are quantitated.

4.1 Features and capabilities

Empower software identifies and quantifies unknown components using peak matching and quantitation:

- Peak matching – The process of matching unknown peak retention times (RT) against the RT of known standard peaks.
- Quantitation – The process of calculating the amounts of unknown peaks using the integration results of each peak and a calibration curve based on the amounts and integration results of known peaks (standards).

4.2 Peak matching

When performing peak matching, the software chooses the integrated peaks in the chromatogram that most closely match the components in the **Components** table from the processing method.

To accomplish peak matching, the software operates as follows:

1. It uses the time region defined by the RT of the component's listed in the **Components** table, plus or minus the component's RT window, together with the Peak Match type.
2. It matches peaks inside the RT windows of the components by calculating the difference between each unknown peak and component RT defined in the processing method.
3. It uses the differences to choose the unknown peak that most closely matches the component peak.

4.2.1 Matching hierarchy

The software uses a hierarchy of peak match types when matching unknown peaks to components. The software matches each component to the unknown peaks in its RT window. If a

peak matches multiple components, the software determines the most appropriate component for that peak – first by position, then by size, and finally, by RT.

4.2.2 Calculating the match difference

When the peak match type is Closest or Closest Negative, the software calculates the difference as the absolute value of the component's RT minus the unknown peak's RT. For the other match types, the match difference is either 0 (a perfect match) or not matched.

A match is considered perfect if at least one of the following conditions exist:

- A peak is in first, second, third, fourth, fifth, or last position in the RT window that corresponds to its match type.
- The size of a peak, relative to other peaks in the RT window, conforms to its match type:
  - Greatest area or height
  - Least area or height
  - Greatest width (GPCV data only)
- There is a 0.0 difference between the RT of the peak and the component.

4.2.3 Choosing optimal peak match

The next step in the matching process is to determine whether any components match multiple unknown peaks or whether any unknown peaks match multiple components.

These are the three possible outcomes of the initial component matching process:

- Single peak matching a single component
- Multiple peaks matching a single component
- Single peak matching multiple components

4.2.3.1 Single peak with a single component

The peak matching process is straightforward when the RT windows of the components do not overlap and, at most, one unknown peak is found in each window. In such a case, matching type and difference are not necessary.

Tip: Never use the matching types Second, Third, Fourth, or Fifth when there are always fewer than two, three, four, or five peaks, respectively, in the RT window.

4.2.3.2 Multiple peaks with a single component

If there are multiple unknown peaks in the RT window of a component, the software uses the match difference to choose the peak that most closely fits the matching type criteria.

- If two peaks match a single component, the software chooses the peak with the smallest match difference. The remaining peak is then matched to its next closest component.
- If two peaks equally match a single component (the match differences are equal), the software does not match the component to either peak.

4.2.3.3 Single peak with multiple components

If the same peak is matched to two or more components, the software then picks the component with the smallest difference from among the possible matches. If two components have equal differences for a peak, a choice cannot be made and the unknown peak remains unmatched. In the **Peaks** tab of the Main and Results windows of Review, the software lists the components' Peak Types as Missing and a Q04 code is copied into the **Processing Code** field for the unmatched peak, indicating the reason the components are missing.

4.2.4 Shifting RT and RT windows

If the software cannot identify peaks because they are shifting so much that they are outside the RT windows, you can increase the size of the RT windows. If that is not possible, or if it causes peaks to be misidentified, you can use the RT reference peak or the update RT parameter.

4.2.4.1 RT Reference

The **RT Reference** field allows you to temporarily adjust the RT of a component based on where the defined RT Reference peak is found in the chromatogram. The RT of the component temporarily shifts by the same percentage, and in the same direction, as the shift of the RT Reference peak. (The RT Reference peak is determined by comparing the RT for the reference peak listed in the Component table to the actual RT of the reference peak in the chromatogram.) The software uses the adjusted RT to match the unknown peak in the chromatogram and calculates the adjusted RT of a component as follows:

$$RT_{adjusted} = RT_{component\ in\ calibration\ curve} \times \left(RT_{ref\ peak\ found} / RT_{ref\ peak\ in\ calibration\ curve}\right)$$

Use an RT reference peak that is always found in your chromatogram, that is well separated from other peaks, and whose RT shifts with your other components.

4.2.4.2 Update RT

The **Update RT** field adjusts the RT of calibration curves, thus affecting the RT the software uses to match unknown peaks. Adjusting is done to more accurately reflect the actual RT of the peaks in the chromatogram when RT shifting or drifting is problematic.

Recommendation: Ensure that the chromatogram has sufficient resolution between peaks and the retention time windows are large.

Normally during peak matching, the software compares the RT of integrated peaks to the RT of the calibration curves, and also to the RT windows listed for the components in the **Components**

tab of the processing method. When **Update RT** is selected, the software uses the RT where the components were found during the processing of a previous chromatogram. Each time a chromatogram is processed, the software can store a new RT to use for peak matching and processing the subsequent chromatogram (depending on the **Update RT** selection). The updated RT is stored in the calibration curve for the component and is displayed in the **Time** field of the Calibration Curve window. The **Update RT** functionality does not affect the retention times listed in the **Components** tab of the processing method.

You can set the Update RT as follows:

- **Never**: The RT of the calibration curve is not updated.
- **Replace**: The RT of the calibration curve is updated every time a chromatogram is calibrated or quantitated, regardless of sample type. When a chromatogram is calibrated or quantitated, if any peak is identified in the RT window, the software replaces the RT in the calibration curve (not in the Components table) with the newly found RT.
- **Replace Standards**: The RT of the calibration curve is updated only when standards are calibrated. When a chromatogram is calibrated, if a standard peak is identified in the RT window, the software replaces the RT in the calibration curve (not in the Components table) with the newly found standard RT.
- **Average**: The RT of the calibration curve is updated every time a chromatogram is calibrated or quantitated, regardless of sample type. When a chromatogram is calibrated or quantitated, if any peak is identified in the RT window, the software averages the RT in the calibration curve (not in the **Components** table) with the newly found RT.
- **Average Standards**: The RT of the calibration curve is updated only when standards are calibrated. When a chromatogram is calibrated, if a standard peak is identified in the RT window, the software averages the RT in the calibration curve (not in the Components table) with the newly found standard RT.

The software updates the RT using any averaging choice as follows:

$$RT_c = \frac{(Average\ Time\ from\ Calibration\ Curve \times n + New\ Retention\ Time)}{n+1}$$

where:

RT_c = The retention time of the calibration curve

n = The number of times the value was previously averaged

Update RT is a coarse adjustment you should use only when these conditions apply:

- Your chromatography is shifting or fluctuating considerably.
- The overall shift cannot be offset by increasing the RT window of the component peaks or using the RT reference peak.

Tip: Use the **Replace** and **Average** choices only when the unknown samples have no peaks that can be misidentified.

Recommendations:

- Only use Replace and Replace Standards when retention time shifts are consistent (increasing or decreasing) and there are no unknowns that could be misidentified.
- Only use Average and Average Standards when the retention times fluctuate around a mean value and there are no unknowns that could be misidentified.

4.3 Quantitation

You can perform quantitation using these approaches:

- Calibration
- No calibration
- Sample weight and dilution
- Injection volume
- Responses other than peak area and height

See also: For additional information on the processes the software uses during quantitation, see References.

4.3.1 Quantitation by calibration

Empower software performs calibration on a set of processed standards acquired by the chromatographic system. When you run the standards, the software requires that you specify this information:

- Injection volumes in the **Samples** table of the Run Samples window or Sample Set Method Editor.
- Component names and amounts, or concentrations in the **Default Amount** tab of the Processing Method window, the Component Editor of the Run Samples window, or the Alter Sample window.

During processing of chromatograms, the software calculates a response based on the detector signal for each peak. This response can be one of the following:

- Peak area.
- Peak height.
- Another peak value (including a custom peak value).

Once calibration standards are processed, the software generates a calibration curve for each standard component listed in the **Components** table. The calibration curve displays this information:

- Response (**Y-Value** field) versus Amount or Concentration (**X-Value** field) for external standard calibration.
- Response ratio multiplied by the internal standard amount, or concentration, versus Amount or Concentration (**X-Value** field) for internal standard calibration.

Calibration curve shape is based on a fit type you select (as described in Calibration curve fit types). There are three categories of calibration curve fit types:

- Linear: Always results in a linear fit.
- Non-Linear: Allows you to select different fits to a multilevel calibration curve.
- Forced-Through-Zero: Allows you to force the calibration line through zero.

The software calculates and updates calibration curves using individual or averaged points based on an Average By value you specify in the **Components** tab of the Processing Method window.

During sample processing, the software performs the following steps in sequence:

- Matches the RT of the integrated peaks found in the unknown chromatogram with the RT of the components in the calibration curve.
- Applies the response of each matched unknown peak to the corresponding component calibration curve.

During quantitation, the software calculates the amount, or concentration, of the unknown sample from the calibration curve. It uses the response of the sample to find the X-value that corresponds to the amount or concentration. The X-values used in the calibration curve are equal to the amounts you entered in sample list or processing method multiplied by sample weight divided by dilution. The amount or concentration for a sample component is divided by the sample weight and multiplied by the dilution. The software then displays the final component amount in the **Peaks** tab of the Main and Results windows of Review.

4.3.2 Quantitation without calibration

If you want to quantitate without performing calibration, the software calculates the relative amount of each unknown peak in the sample as both percent area and percent height. Peak area and height percent are calculated as the percent of each integrated peak relative to the total area or height of all integrated peaks.

4.3.3 Quantitation using sample weight and dilution

Sample weight and dilution values are used to adjust the amounts and concentrations of standard and unknown components. These two values are optional and can be used to compensate for differences due to factors such as these:

- Varying dilutions
- Different initial sample mass or volume

You enter sample weight and dilution on a per-standard or per-sample basis in the **Samples** table of the Run Samples window, Sample Set Method Editor, or Alter Sample window. Typically, sample weights or dilutions are used for either the standard samples or the unknown samples, but not for both.

4.3.3.1 Sample weight

Sample weight is typically used in samples to calculate the ratio of quantity of component injected into the system to total quantity of original sample.

During calibration, the software multiplies the entered amount or concentration of a standard component by the sample weight to calculate amounts and concentrations for the standard sample.

During quantitation, the software divides the amount or concentration (**X-Value** field) determined from the calibration curve by the sample weight to calculate amounts and concentrations for the unknown sample.

For example, if the mass of sample you weighed is 0.5 mg and you want to report the amount determined by the software as the amount of the component as compared to the mass of the total sample, enter a sample weight of 0.5 for your unknown sample. The software quantitates the component amount from the calibration curve and then divides that value by the sample weight to obtain the final ratio of component amount to total sample amount. The amount can then be converted into a percentage by multiplying it by 100, either by using a dilution value of 100 or by creating a custom field and specifying a formula of Amount*100.

When using sample weight, ensure that this value is equivalent to the units for the component amount or concentration you are reporting. For example, if you weigh 1.44 mg of sample and the units of your standard amounts are in µg, use a sample weight of 1440 (µg).

4.3.3.2 Dilution

The **Dilution** field is typically used when you dilute a sample (a standard, an unknown, or a control) prior to injection and want to report the quantity of analyte in the original, undiluted sample. This could occur when an undiluted sample, injected directly onto the column, would fall above the range of the calibration curve. The sample dilution should be entered into the **Samples** table of Run Samples window, the sample set method, or the **Alter Sample** window. The sample should be injected at the usual injection volume.

During calibration, the software divides the entered amounts or concentrations of the standard component or components by the dilution value to calculate amounts and concentrations for the standard component or components.

During quantitation, the software multiplies the amount or concentration (X value) determined from the calibration curve by the dilution value to calculate amounts and concentrations for the unknown sample.

For example, if a 1:10 dilution was performed on a standard sample containing one component at an amount of 100 µg, enter the amount of the standard component into the Component Editor or the **Default Amounts** tab of the Processing Method window as 100 µg (the original, undiluted

quantity) and the dilution as 10. When the software calibrates this standard, it takes the specified amount of 100 µg and divides by the specified dilution of 10. The software reports the resulting amount as 10 µg (the amount injected on the column). This value is also plotted on the calibration curve.

If that same sample were an unknown sample, you would not enter a quantity for the component; but the dilution would still be 10. When the software quantitates the unknown, it reads an amount of 10 µg directly from the calibration curve and multiplies that value by 10 (the dilution value) for a resulting amount of 100 µg (the pre-diluted amount).

When you are working with dilutions of standards, specify the dilutions and the original, undiluted quantities of the standard components. When you are working with dilutions of unknown samples, if you specify the dilutions, the amounts and concentrations reported by the software will be those of the original, undiluted samples. The use of the dilution field eliminates the need to correct for a dilution by adjusting the injection volume.

Tip: You can correct for the dilution of samples by adjusting the injection volume. If, by mistake, you dilute a sample or standard by a factor of 10, you could inject 10 times the usual injection volume instead of specifying a value of 10 in the **Dilution** field. If the sample is a standard, you also need to specify the undiluted amount or amounts for the standard component or components. If the sample is an unknown, do not specify the dilution, but adjust the injection volume 10-fold. The software determines the undiluted quantity of component. This quantity, given the higher injection volume, is 10 times higher than it would be had the normal injection volume been used. For both standard and unknown samples, if a dilution is corrected for by the injection volume, do not adjust the value in the **Dilution** field.

4.3.4 Quantitation using injection volume

The software calculates both amounts and concentrations for standard and unknown samples. You can include the value that is meaningful to you on your reports.

The software determines whether you are entering your standard component quantities in units of amount or concentration by the **Sample Value Type** list in the **Components** tab of the Processing Method window. Selecting amount or concentration causes the software to interpret component quantities as follows:

- Amount: The software interprets the component quantities you enter (in the Component Editor of the Alter Sample window, the Component Editor of the Run Samples window, or the **Default Amount** tab of the Processing Method window) as amounts. The software calculates the corresponding concentration by dividing the specified amount by the injection volume, in µL.
- Concentration: The software interprets the component quantities you enter (in the Component Editor of the Alter Sample window, the Component Editor of the Run Samples window, or the **Default Amount** tab of the Processing Method window) as concentrations. The software calculates the corresponding amount by multiplying the specified concentration by the injection volume, in µL.

Regardless of whether you define your standard component quantities in amounts or concentrations, you can create a calibration curve that uses standard amounts or concentrations. The X-value for the component as entered in the **Components** tab of the Processing Method window determines whether the calibration curve is a plot of Response versus Amount or Response versus Concentration.

If the calibration curve is a plot of Response versus Amount (which occurs when the **X-Value** field is set to **Amount**), the software quantitates unknown samples by using a component's response to determine its amount directly from the calibration curve. The software then determines the component's corresponding concentration value by dividing the calculated amount by the injection volume, in µL.

Tip: If the **X-Value** field is set to **Amount**, the sample injection volume affects the calculated concentrations for unknown samples, but not the calculated amounts.
Likewise, if the calibration curve is a plot of Response versus Concentration (which occurs when the **X-Value** field is set to Concentration), the software quantitates unknown samples by using a component's response to determine its concentration directly from the calibration curve. The software then determines the component's corresponding amount value by multiplying the calculated concentration by the injection volume, in µL.

Tip: If the **X-Value** field is set to Concentration, the sample injection volume affects the calculated amounts for unknown samples, but not the calculated concentrations.
Use caution when specifying a component's unit label (µg, µg/µL, and so forth), because the software reports the label exactly as you specify it. In cases where the component's quantity is affected by the injection volume, ensure that the unit label is appropriate. The software always uses microliters for injection volume units.

4.3.5 Quantitation using responses other than peak area and height

The software allows you to use the responses of area, height, % Area, and % Height for response. Also, any peak type custom field using a data type of real can be used for response, except for custom fields using these parameters:

- Time fields
- Baseline fields
- Response
- Amount
- Concentration
- % Amount

Make the appropriate selection in the **Y-Value** field in the **Components** tab of the Processing Method window.

4.3.6 External and internal standard quantitation

The component amount calculations use one of these quantitation methods:

- External standard
- Internal standard with separate standard and unknown samples
- Internal standard without separate standard and unknown samples (typically used with gas chromatography)

See also: For assistance reproducing amounts or concentrations calculated by the software, see Calibration curve fit types.

4.3.7 External standard quantitation

The external standard method of quantitation determines component amounts or concentrations by applying the detector response of a component peak to a calibration curve. The calibration curve is generated from a separately acquired and processed set of standards.

Tip: The standard set must contain at least one standard (referred to as single-level calibration).

The following criteria are also required:

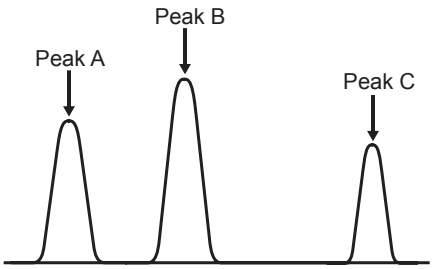
- You must define standard samples as Standards by using the Inject Standard function during sample loading in Run Samples or (after the sample is acquired) by defining a Sample Type of Standard in Alter Sample.
- You must define unknown samples as Unknowns by using the Inject Unknown function during sample loading in Run Samples or (after the sample is acquired) by defining a Sample Type of Unknown in Alter Sample.
- You must define component names and amounts or concentrations of each standard component in the **Default Amount** tab of the Processing Method window, the Component Editor of the Run Samples window, or the Alter Sample window.
- The response is the Y-value of the calibration curve. You choose the parameter to use as the *y* axis by selecting the **Y-Value** field in the **Components** tab of the Processing Method window.
- The X value of the calibration curve is amount, concentration, or a custom field. You choose the *x* axis in the **X-Value** field in the **Components** tab of the Processing Method window. You can use any peak type custom field using a data type of real for the X-value, except for custom fields using:
  - Time fields
  - Baseline fields
  - Response
  - % Amount

- External standard quantitation generates each calibration curve by plotting the detector response of a standard component versus the amount or concentration of the standard component.
- You define the fit type used for the calibration curve in the **Components** table of the Processing Method window.

To perform single-level external standard quantitation, the software operates as follows:

1. It identifies one or more component peaks in the standard injection using peak matching.
2. It determines the response and amount or concentration for each standard peak, then plots the two values as a calibration point on the calibration curve for the component with the same name.
   Given a chromatogram such as that in the following figure, the values used to determine the calibration points are the concentration and response values from the table that follows the figure.

   Figure 4–1: External standard chromatogram:



   Tip: In the following table, the X-values are set to concentration.

   Table 4–1: Standard peak values, external standard calibration:

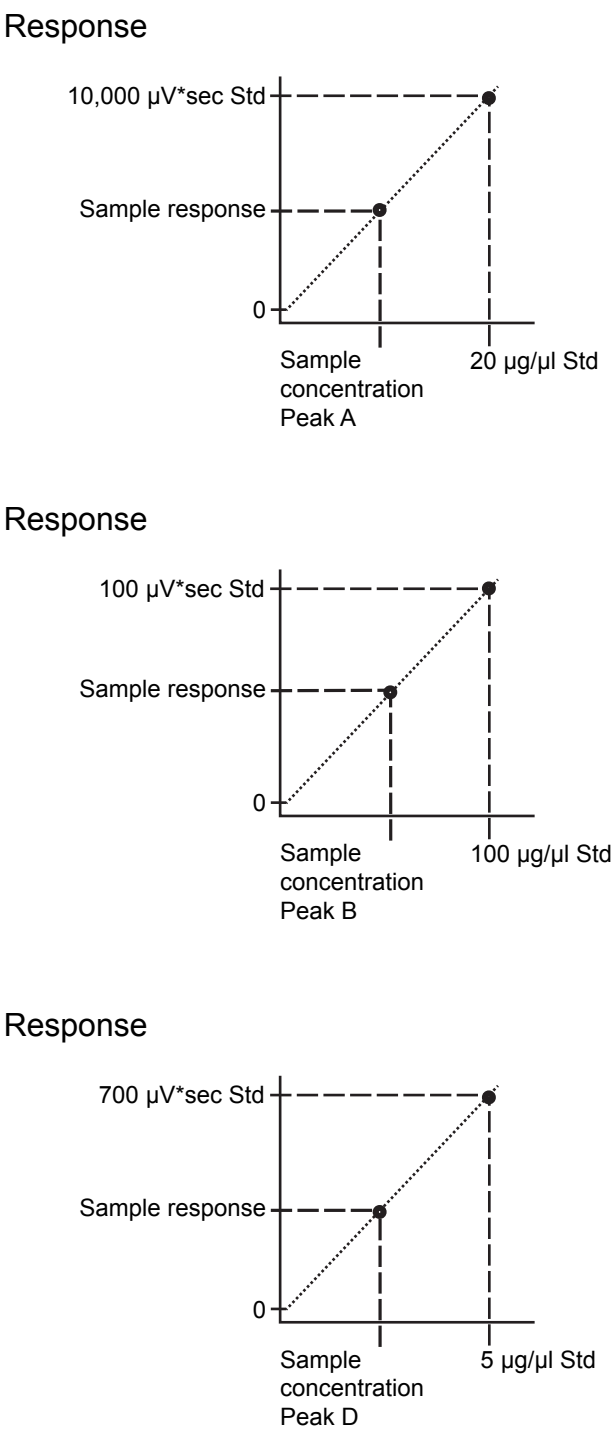
Component name	Quantitation basis (user-specified Y-value)	Amount in standard (user-specified X-value)	Component area	Component height	Response
A	Area	20 µg/µL	10000	900	10000
B	Height	100 µg/µL	12000	1100	1100
C	Height	5 µg/µL	8000	700	700

3. Calculates a calibration curve (response versus concentration) for each component listed in the **Name** column of the **Components** table.
4. Quantitates unknown samples comparing them against the generated calibration curve, completing the following steps in sequence:

- Identifies each unknown peak by matching its retention time with a component from the **Components** table.
- Calculates amount and concentration for each unknown peak from the component calibration curve using sample peak response and injection volume.
- Adjusts the amount and concentration by the sample weight and dilution fields as entered during sample loading. The final calculated amount and concentration appear in the **Peaks** tab of the Main and Results windows of Review.

The following figure illustrates the quantitation for peak components A, B, and C. Each calibration curve uses the single-level fit type (linear through zero).

Figure 4–2: External standard component calibration curves (single-level, concentration):



Tip: If you are using a multilevel calibration curve, the software performs an equivalent process.

4.3.8 Internal standard quantitation with separate standard and unknown samples

This technique uses an internal standard added to both the standard and unknown samples as a recovery standard. It is commonly used to correct for losses during sample preparation.

This technique determines component amounts and concentrations by applying a response to a calibration curve generated first by calculating the response for the set of standards containing the internal standard. The response is calculated from the responses of the component peak and its internal standard peak. You select the type of response by using the **Y-Value** field in the **Components** table of the Processing Method window.

The classic internal standard quantitation method plots the response ratio of the standard component to the internal standard versus the amount or concentration ratio of the standard component to the internal standard to generate a calibration curve. The software uses an equivalent calibration curve produced by plotting the response ratio times the internal standard X-value versus the component X-value, where the **X-Value** field is set to amount or concentration in the **Components** table of the Processing Method window.

Figure 4–3: Classic response versus amount (or concentration) plot:

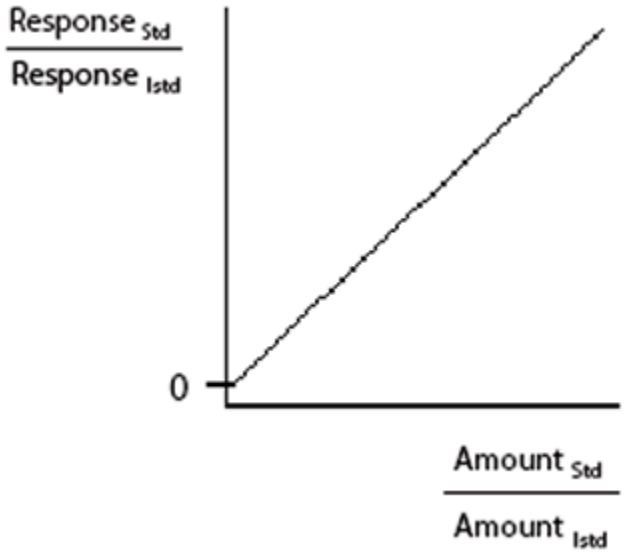
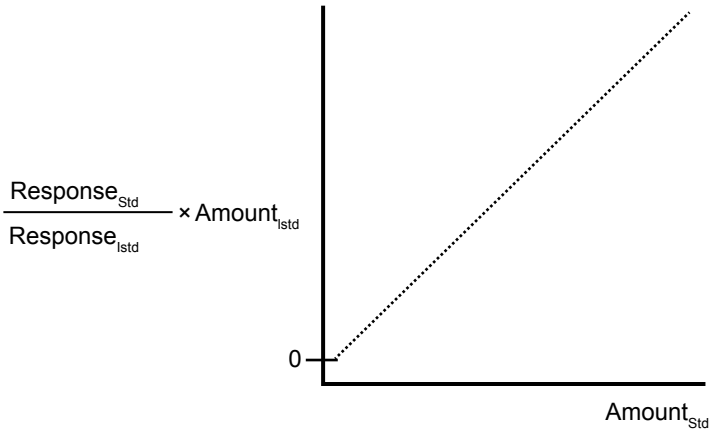


Figure 4–4: Multiplying response by internal standard amount (or concentration):



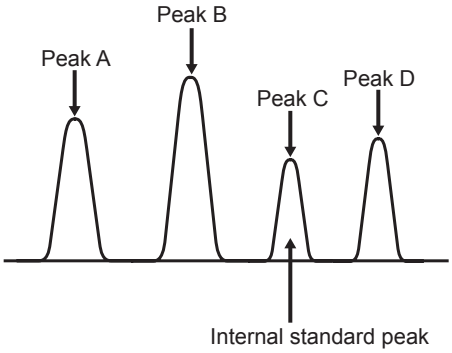
The following criteria are also required:

- You must define standard samples as standards, by using the Inject Standards function during sample loading in Run Samples window or (after the sample is acquired) by defining a Sample Type of Standard in Alter Sample.
- You must define unknown samples as unknowns, by using the Inject Unknowns function during sample loading in Run Samples window or (after the sample is acquired) by defining a Sample Type of Unknown in Alter Sample.
- You must enter component names and amounts or concentrations of each standard component in the **Default Amount** tab of the Processing Method window, the Component Editor of the Run Samples window, or the Alter Sample window.
- The X-value of the calibration curve is amount or concentration. You choose amount or concentration as the *x* axis in the **X-Value** field in the **Components** tab of the Processing Method window.
- You define the fit type used for the calibration curve in the **Components** table of the Processing Method window.

To perform internal standard quantitation (with separate standard and unknown samples), the software operates as follows:

1. It identifies the component peaks in the chromatogram using peak matching.
2. It determines the responses and amounts, or concentrations, for standard peaks and internal standards. The software calculates a response for each standard peak and multiplies that value by the amount, or concentration, of the internal standard component divided by the response of the internal standard. The resulting response value is plotted against the amount, or concentration, value of the standard peak on the calibration curve for the component with the same name.
   Given a standard chromatogram like the one in the following figure, the values used to determine the calibration points are the amount and response values from the table that follows.

Figure 4–5: Internal standard chromatogram:



Tip: In the following table, the X-values are set to amount.

Table 4–2: Standard peak values, internal standard calibration with separate standard and unknown samples:

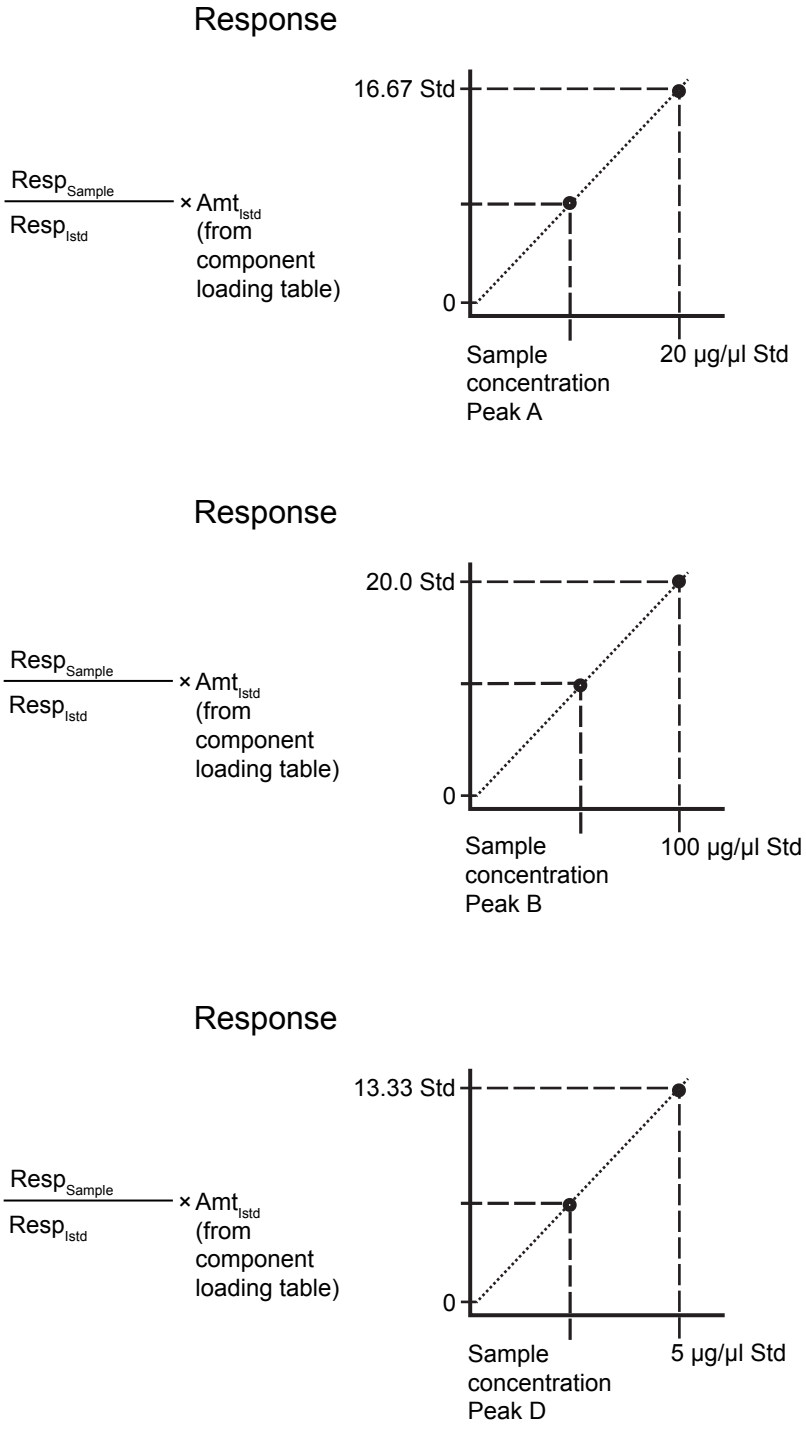
Component name	Quantitation basis (user-specified Y-value)	Concentration in standard (user-specified X-value)	Component area	Response
A	Area	20 µg	10000	$\frac{Area\,A}{Area\,C} = \frac{10000}{6000} \times 10 = 16.67$
B	Area	100 µg	12000	$\frac{Area\,B}{Area\,C} = \frac{12000}{6000} \times 10 = 20$
C (Int Std)	Area	10 µg	6000	N/A
D	Area	5 µg	8000	$\frac{Area\,D}{Area\,C} = \frac{8000}{6000} \times 10 = 13.33$

3. Calculates a calibration curve (peak response/internal standard response times internal standard amount versus Amount or Concentration) for each component listed in the **Name** field of the **Components** table of the Processing Method window.
4. Quantitates unknown samples, comparing them against the generated calibration curve, by performing the following steps in sequence:
   - Identifies each unknown peak by matching its retention time with a component from the **Components** table.
   - Calculates a response for each matched peak by dividing the peak's response by its internal standard response, and then multiplies this ratio by the amount or concentration of the internal standard.

- Calculates amount and concentration for each sample peak from the component calibration curve using the sample peak response and the injection volume.
- Adjusts amount and concentration by the **Sample Weight** and **Dilution** fields as specified in the **Samples** tab of Run Samples or in Alter Sample. The final calculated amount and concentration appear in the **Peaks** tab of the Main and Results windows of Review.

The following figure illustrates quantitation for peak components A, B, and D (internal standard C is not shown). Note that each calibration curve uses the single-level fit type.

Figure 4–6: Internal standard component calibration curves (single-level, amount):



Tip: If you are using a multilevel calibration curve, the software performs an equivalent process.

4.3.9 Internal standard quantitation without separate standard and unknown samples (RF internal standard)

This technique is often used in gas chromatography. All samples are spiked with one or more standard compounds that elute at a different time than the unknown component or components you want to quantitate. When each chromatogram is processed, it is treated as an unknown sample type. The software calculates a response factor (RF) for one or more standard components and then quantitates the unknown components using the RF of one or more standard components rather than using coefficients of a calibration curve.

The following criteria are also required:

- You must define all samples as an RF internal standard, either by using the Inject RF Internal Standards function during sample loading in Run Samples window or (after the sample is acquired) by defining a Sample Type of RF Internal Standard in Alter Sample.
- The X-value of the calibration curve is either amount or concentration (defined in the **X-Value** field in the **Components** tab of the processing method).
- You must specify the names of the standard components in the **Components** tab of the processing method.
- You must specify amounts or concentrations of each standard component in the **Default Amount** tab of the processing method in the Component Editor of the Run Samples window, or the Alter Sample window.
- You can enter component names of the unknown components in the **Components** tab of the processing method (optional).
- Unknown components whose names are defined in the **Components** tab of the processing method are typically quantitated using the RF of a standard component defined by using a curve reference peak. However, they can alternatively be quantitated using a default peak.
- Unknown components whose names are not defined in the **Components** tab of the processing method are quantitated using a default peak.

Tip: Use the **Default Pk** field to define which standard component's response factor to use during quantitation of unnamed components. To use a standard peak as a default peak, in the **Components** tab of the processing method, on the row containing the default peak, check the **Default Pk** field. Define the region of the chromatogram in which you want to use this default peak in the **Default Pk Start** and **Default Pk End** fields. These fields allow you to use a different default peak for different regions of your chromatogram, if necessary. During quantitation, any unknown peak that is detected within the default peak start and end range uses the RF of that default peak in determining its amount or concentration.

Tip: Use the **Curve Reference** field to define which standard component's RF to use during quantitation of named components. To use a Curve Reference, in the **Components** tab of the processing method, on the row for the named unknown component, specify the name of the appropriate standard component in the **Curve Reference** field.

- The RF is calculated using the Y-value and the X-value (amount or concentration) defined in the **Components** tab of the Processing Method window.

To perform internal standard quantitation (without separate standards and unknown samples), the software operates as follows:

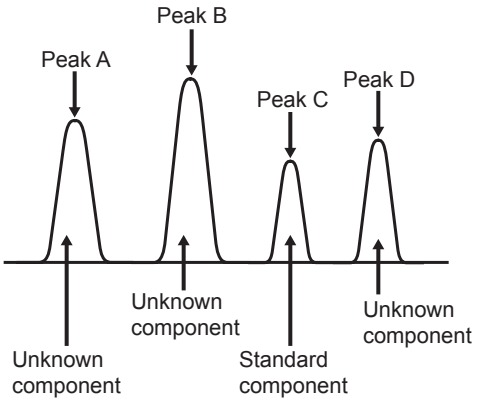
1. It identifies one or more component peaks in the injection using peak matching.
2. It determines the RF for each standard component using the following formula:

   $$RF = \frac{Y\ Value}{X\ Value}$$

   where:
   RF = Response factor
   Y-Value = Response of the standard component calculated by the software
   X-Value = Component amount or concentration of the standard component
   Given an RF internal standard chromatogram such as the one in the next figure, the values used to determine the RF are the amount and response values from the table that follows.

   Tip: When this type of internal standard method is used, no calibration curves are generated.

   Figure 4–7: RF Internal Standard Chromatogram:

   Tip: In the following table, the X-values are set to amount.

   Table 4–3: Standard peak values, RF internal standard calibration without separate standard and unknown samples:

Component name	Quantitation basis	Amount or concentration in standard (user-specified X-value)	Component area (response or Y-value)	Response factor
C (standard component)	Area	10 µg	6000	$\frac{6000}{10} = 600$

3. Uses the response of each unknown component and the appropriate RF to determine unknown component amounts (or concentrations) as follows:

   $$X\ Value = \frac{Y\ Value}{RF}$$

where:
RF = Response factor value calculated for the standard peak
Y-Value = Response of the unknown component calculated by the software
X-Value = Component amount or concentration
The values used to determine the amounts of the unknown components in the following table are the RFs determined in the previous table and the unknown component values in the following table.

Tip: In the following table, the software-determined X-values are amounts.

Table 4–4: Unknown component values, RF internal standard calibration without separate standard and unknown samples:

Component name	Quantitation basis	Component area (response or Y-value)	Amount or concentration in unknown component (software-determined X-value)
A (unknown component)	Area	10000	$\frac{10000}{600} = 16.667 \mu g$
B (unknown component)	Area	12000	$\frac{12000}{600} = 20 \mu g$
D (unknown component)	Area	8000	$\frac{8000}{600} = 13.333 \mu g$

4.4 Calibration curve fit types

A variety of calibration curve fits are available for external and internal standard calibration with multiple levels of standards. The calibration curve fit types are divided into three groups of increasing complexity:

- Single-level calibration (linear through zero and response factor)
- Multilevel calibration matrix operations:
  - Multilevel calibration (linear, inverse linear, log-log linear, quadratic, cubic, fourth-order, fifth-order, and power curve)
  - Multilevel forced through zero (linear, quadratic, cubic, fourth-order, fifth-order, and response factor)
- Multilevel calibration (point-to-point and cubic spline)

Tip: You can apply weighting only to the linear, quadratic, cubic, fourth-order, and fifth-order fit types.

When you refer to the calibration curve fit types in this section, keep this information in mind:

- The software uses matrix operations to perform multilevel calibration.
- For background material on the processes used by the software to create calibration curves, (see References).
- The equations shown in the following examples are not adjusted for sample weight and dilution (see Quantitation).

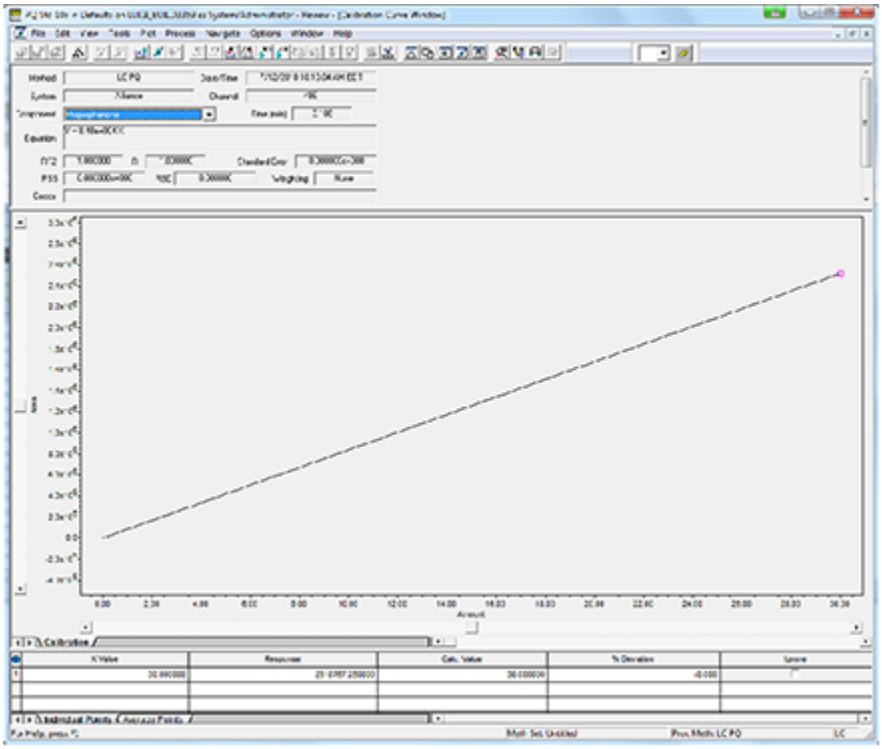
4.4.1 Single-level calibration curve

For single-level calibration, the curve fit is linear; the intercept is zero. The software supports the following single-level calibration curve fits:

- Linear through zero
- Response factor

The following figure illustrates a single-level calibration curve.

Figure 4–8: Single-level calibration curve:



4.4.1.1 Linear through zero

The linear-through-zero calibration curve is represented by the equation:

$y = Bx$

where:

y = Response of the standard component calculated by the software

B = Slope of the calibration curve

x = Component amount or concentration

The component amount or concentration for a quantitated sample can be determined by the equation:

$$x = \frac{y}{B}$$

where:

x = Component amount or concentration

y = Response of the sample peak calculated by the software

B = Slope of the calibration curve

4.4.1.2 Response factor

The response factor (RF) fit type eliminates the need to create an RF custom field. When using an RF fit type, you should specify the appropriate X-Value and Y-Value in the Components tab of the processing method as when using a linear-through-zero fit.

The software plots the standard component's response versus its amount (or concentration) on the calibration curve. The RF is the slope of the curve. If multiple data points are plotted on the calibration curve, the RF for each point is determined, and then the average RF is used as the slope of the curve.

The RF is represented by the equation:

$$RF = \frac{Y\ Value}{X\ Value}$$

where:

RF = Response factor (slope of the calibration curve)

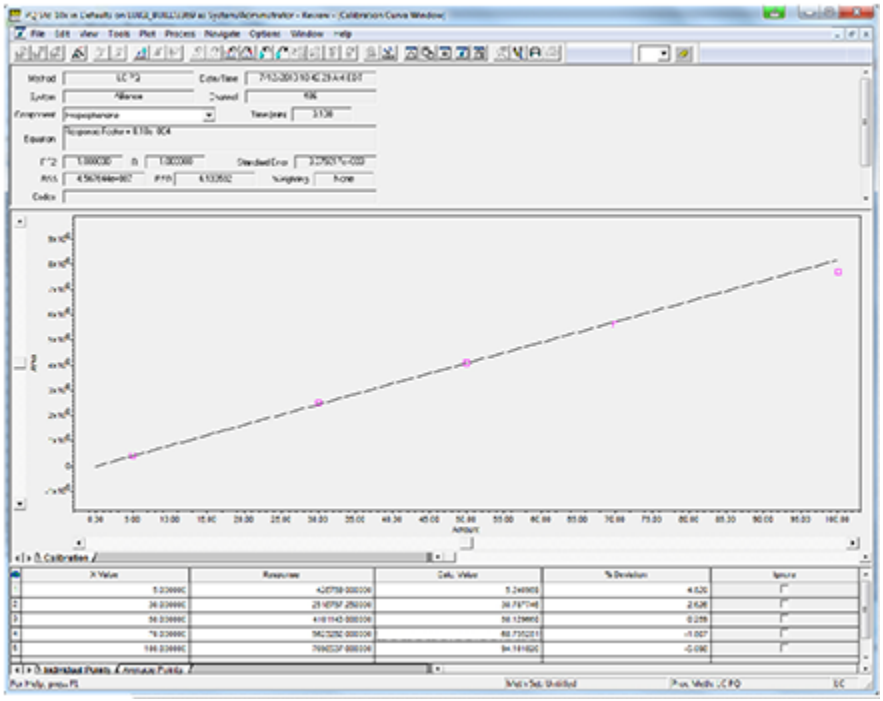
y = Response of the standard component calculated by the software

x = Component amount or concentration of the standard component

A linear-through-zero fit to the average RF point results in the equation of the curve.

The following figure illustrates an RF calibration curve.

Figure 4–9: Response factor calibration curve:



4.4.2 Multilevel calibration matrix operations

For the multilevel polynomial curve fits (linear, inverse linear, log-log linear, quadratic, cubic, fourth-order, fifth-order, and through zero fits), the software uses matrix operations to obtain the required coefficients.

For all polynomial fits, the software uses an unweighted or weighted least-squares fitting technique to a set of x-y points or x, y, and weight points. This technique is the numerical routine called LU (lower and upper triangular matrix) Decomposition for weighted and unweighted fits.

- When weighting is disabled in the **Components** table, the software uses an unweighted least-squares fitting technique to a set of x-y points.
- When weighting is enabled in the **Components** table, the software uses a weighted least-squares fitting technique to a set of x-y weight points.

4.4.2.1 Matrix operations

The objective is to use the least-squares fit to calculate the coefficients for the curve:

$[Y] = [A] \times [C]$

where:

$[Y]$ = Response vector

$[A]$ = Design matrix

[*C*] = Coefficient vector

To accomplish this, the least-squares fit solves the following linear normal equations:

$([A]^T \times [W] \times [A]) \times [C] = [A]^T \times [W] \times [Y]$

where [*W*] is a diagonal weight matrix, which becomes unity (all diagonal elements equal to 1) for an unweighted fit. The solution is:

$[C] = ([A]^T \times [W] \times [A])^{-1} \times ([A]^T \times [W] \times [Y])$

The matrix inversion is done by using LU Decomposition. The least-squares fit is described in Section 15.4 General Linear Least Squares in *Numerical Recipes* in C William H. Press, et al., (2nd Edition).

4.4.2.2 Matrix operations example

The software uses the following matrices to calculate the coefficients based on the number of standards run. These operations illustrate an unweighted multilevel fifth-order fit.

As an example, assume that seven standards are run, one at each level. The software attempts to apply a fifth-order fit to the calibration points.

To find the coefficients of the calibration equation:

1. The seven standards produce the following amount, response sets (x, y plotted points on the calibration curve):
   $(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4), (x_5, y_5), (x_6, y_6), (x_7, y_7)$
2. The following seven equations, using the data in step 1, contain the six unknown coefficients (c_5 through c_0) and the seven sets of points:
   $y_1 = c_5(x_1)^5 + c_4(x_1)^4 + c_3(x_1)^3 + c_2(x_1)^2 + c_1(x_1)^1 + c_0(x_1)^0$
   $y_2 = c_5(x_2)^5 + c_4(x_2)^4 + c_3(x_2)^3 + c_2(x_2)^2 + c_1(x_2)^1 + c_0(x_2)^0$
   $y_3 = c_5(x_3)^5 + c_4(x_3)^4 + c_3(x_3)^3 + c_2(x_3)^2 + c_1(x_3)^1 + c_0(x_3)^0$
   $y_4 = c_5(x_4)^5 + c_4(x_4)^4 + c_3(x_4)^3 + c_2(x_4)^2 + c_1(x_4)^1 + c_0(x_4)^0$
   $y_5 = c_5(x_5)^5 + c_4(x_5)^4 + c_3(x_5)^3 + c_2(x_5)^2 + c_1(x_5)^1 + c_0(x_5)^0$
   $y_6 = c_5(x_6)^5 + c_4(x_6)^4 + c_3(x_6)^3 + c_2(x_6)^2 + c_1(x_6)^1 + c_0(x_6)^0$
   $y_7 = c_5(x_7)^5 + c_4(x_7)^4 + c_3(x_7)^3 + c_2(x_7)^2 + c_1(x_7)^1 + c_0(x_7)^0$
3. The previous equations can now be written using matrix notation as:

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{bmatrix} = \begin{bmatrix} x_1^5 & x_1^4 & x_1^3 & x_1^2 & x_1^1 & x_1^0 \\ x_2^5 & x_2^4 & x_2^3 & x_2^2 & x_2^1 & x_2^0 \\ x_3^5 & x_3^4 & x_3^3 & x_3^2 & x_3^1 & x_3^0 \\ x_4^5 & x_4^4 & x_4^3 & x_4^2 & x_4^1 & x_4^0 \\ x_5^5 & x_5^4 & x_5^3 & x_5^2 & x_5^1 & x_5^0 \\ x_6^5 & x_6^4 & x_6^3 & x_6^2 & x_6^1 & x_6^0 \\ x_7^5 & x_7^4 & x_7^3 & x_7^2 & x_7^1 & x_7^0 \end{bmatrix} \bullet \begin{bmatrix} c_5 \\ c_4 \\ c_3 \\ c_2 \\ c_1 \\ c_0 \end{bmatrix}$$

or

$[Y] = [A] \times [C]$

where:

$[Y]$ = Response vector

$[A]$ = Design matrix

$[C]$ = Vector of coefficient to be computed

Design Matrix *A* is constructed with *n*+1 columns and *i* rows (where *n* is the order of the polynomial and *i* is the number of levels). The construction of Design Matrix *A* for the fifth-order fit type is previously illustrated.

4. The software then uses LU Decomposition to solve the normal equations of the least-squares fit:

   $([A]^T \times [A]) \times [C] = [A]^T \times [Y]$

4.4.3 Multilevel calibration curves

The software supports the following multilevel calibration curve fits:

- Point-to-point
- Cubic spline
- Linear
- Inverse linear
- Log-log linear
- Quadratic
- Cubic
- Fourth-order
- Fifth-order
- Power curve

The equations used to calculate the goodness-of-fit statistics are described in Statistics, with the following results:

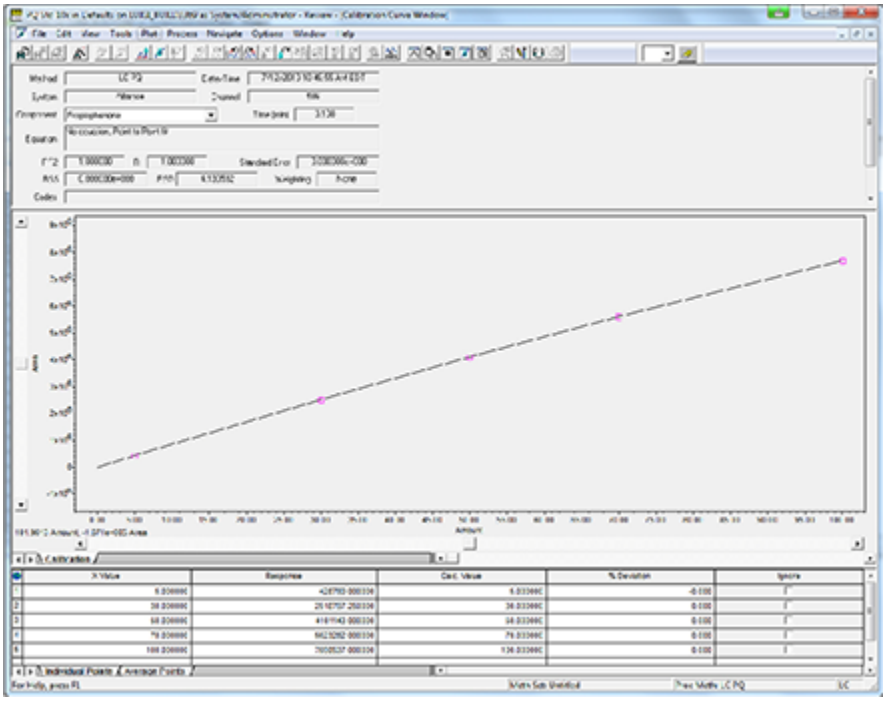
- For all fit types, the software reports only positive X-value amounts or concentrations.
- For linear fit types, the software reports X-values within the range of the calibration curve (from 0 to the highest X-value) and X-values outside the range of the calibration curve. For values greater than the highest X-value, the software extrapolates values above the highest X-value of the standard data point. For values less than the lowest X-value, the software extrapolates values below the lowest X-value of the standard data point.
- For all nonlinear fit types, the software reports X-values from 0 to the highest X-value of the standard points.

4.4.3.1 Point-to-point fit

To calculate a point-to-point calibration curve, the software performs a linear fit between the different levels. The first and last segments of the curve are extrapolated linearly so that they can be used to calculate X-values that fall outside the range of the lowest to highest X-value.

Because the point-to-point calibration curve is fit through every point, the correlation coefficient equals 1, and the standard error equals 0. No curve coefficients are calculated or stored for this fit type. The following figure illustrates a point-to-point calibration curve.

Figure 4–10: Point-to-point calibration curve:



The software calculates each point-to-point segment of the calibration curve using the following equation:

$y = A_i + B_i x$

where:

y = Response of the standard peak calculated by the software

A_i = y-intercept of the *i*th curve segment

B_i = Slope of the *i*th curve segment

x = Component amount or concentration

4.4.3.2 Determining component amount and/or concentration

Component amount and/or concentration for a quantitated sample peak can be determined by the equation:

$$x = \frac{y - A_i}{B_i}$$

where:

x = Component amount or concentration

y = Response of the sample peak, calculated by the software

A_i = y-intercept of the $_i$th curve segment

B_i = Slope of the $_i$th segment

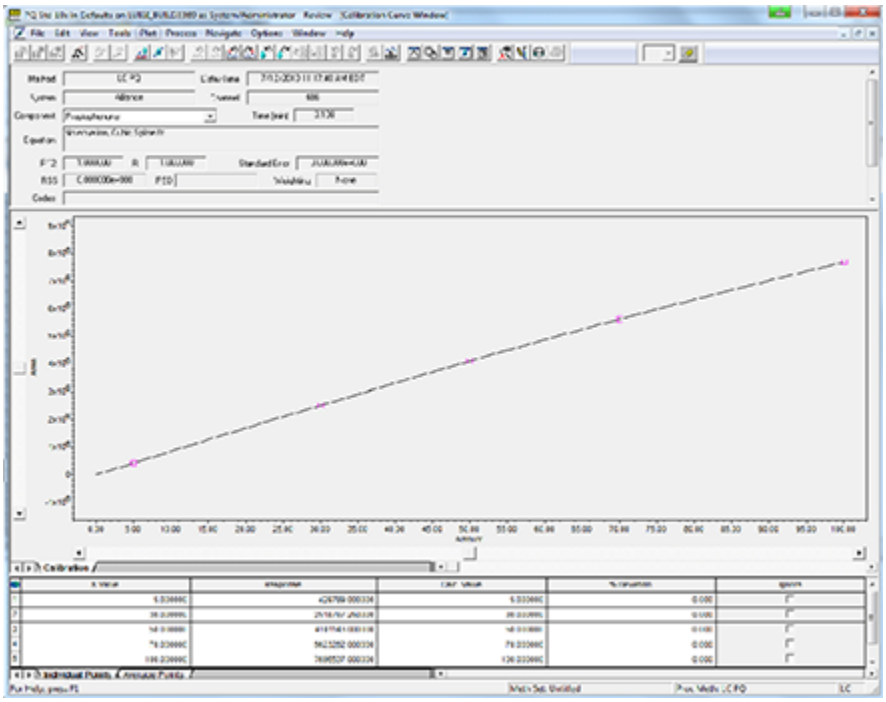
4.4.3.3 Cubic spline fit

To generate a cubic spline calibration curve, the software performs a cubic polynomial fit between every two successive levels, matching the slope and second derivative at every point boundary. The cubic spline fit adjusts the shape of the calibration curve on a point-by-point basis.

Because the cubic spline calibration curve is fit through every point, the correlation coefficient = 1 and the standard error = 0. No curve coefficients are calculated or stored when the cubic spline fit type is used.

The following figure illustrates a cubic spline calibration curve.

Figure 4–11: Cubic spline calibration curve:



Each cubic spline segment of the calibration curve is represented by the equation:

$y = A_i + B_i + C_i x^2 + D_i x^3$

where A_i, B_i, C_i, and D_i are the polynomial coefficients of the segment.

Tip: The software uses an iterative method to solve for x when given y.

4.4.3.4 Linear fits

Linear fits include linear, inverse linear, log-log linear, and power curve.

4.4.3.4.1 Linear fit

To calculate a linear calibration curve, the software calculates the line that best fits the amounts or concentrations and responses of the calibration points. The Y-value of each point is the response of the standard peak and the X-value of each point is the amount or concentration of the standard peak. The following figure illustrates a linear least-squares fit calibration curve.

4.4.3.4.2 Inverse linear fit

To calculate an inverse linear calibration curve, the software performs a linear fit to the X- and Y-values of the calibration points. The Y-value of the point is the response of the standard peak and the X-value is the 1/X value (amount or concentration) of the standard peak.

4.4.3.4.3 Log-log linear fit

To calculate a log-log linear calibration curve, the software performs a linear fit to the X- and Y-values of the calibration points. The Y-value of each point is the log of the response of the standard peak and the X-value is the log of the X-value (either amount or concentration) of the standard peak.

Tip: Inverse linear and log-log linear use the linear fit equation.

4.4.3.4.4 Power curve fit

To calculate a power curve, the software performs a linear fit of log y versus log x, where y is the response and x is the X-value. The plot and equation of the curve are reported in terms of the X-value and response.

The power curve fit generates a calibration curve represented by the equation:

$y = Ax^B$

where:

y = Response of the standard peak calculated by the software

A = Multiplier in the calibration curve formula

B = Exponent in the calibration curve formula

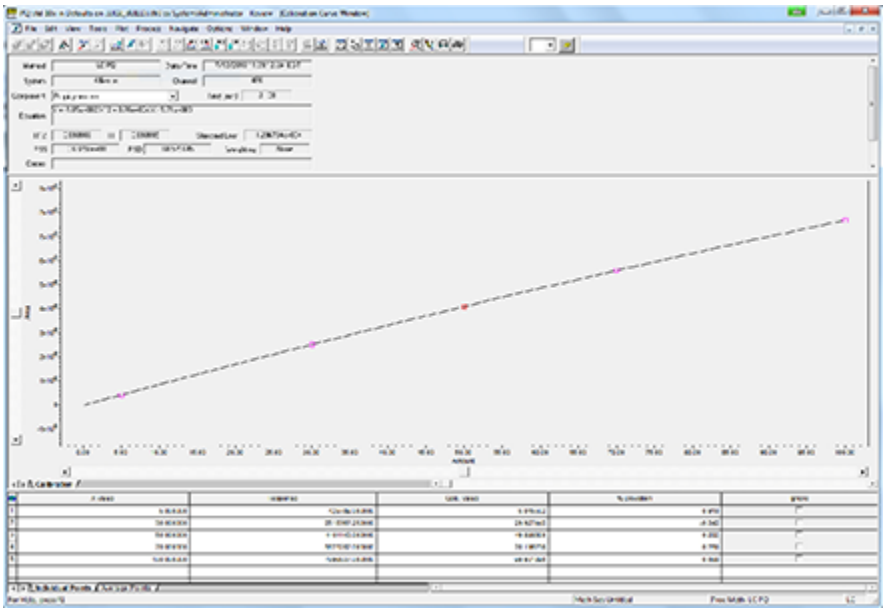
x = Component amount or concentration

4.4.3.5 Quadratic fit

To calculate a quadratic calibration curve, the software performs a least-squares fit of a quadratic polynomial to the calibration points. The fit cannot be performed with fewer than three calibration points, and a minimum of five points is strongly recommended.

The following figure illustrates a quadratic fit calibration curve.

Figure 4–12: Quadratic fit calibration curve:



The quadratic fit generates a calibration curve that is represented by the equation:

$y = A + Bx + Cx^2$

where:

y = Response of the standard peak calculated by the software

x = Component amount or concentration

A, B, and C = Polynomial coefficients of the curve

Determining component amount or concentration

Component amount or concentration for a quantitated sample peak can be determined by solving for x:

$$x = \frac{-B \pm \sqrt{B^2 - 4C(A-y)}}{2C}$$

where:

y = Response of the sample peak calculated by the software

x = Component amount or concentration

A, B, and C = Polynomial coefficients of the curve

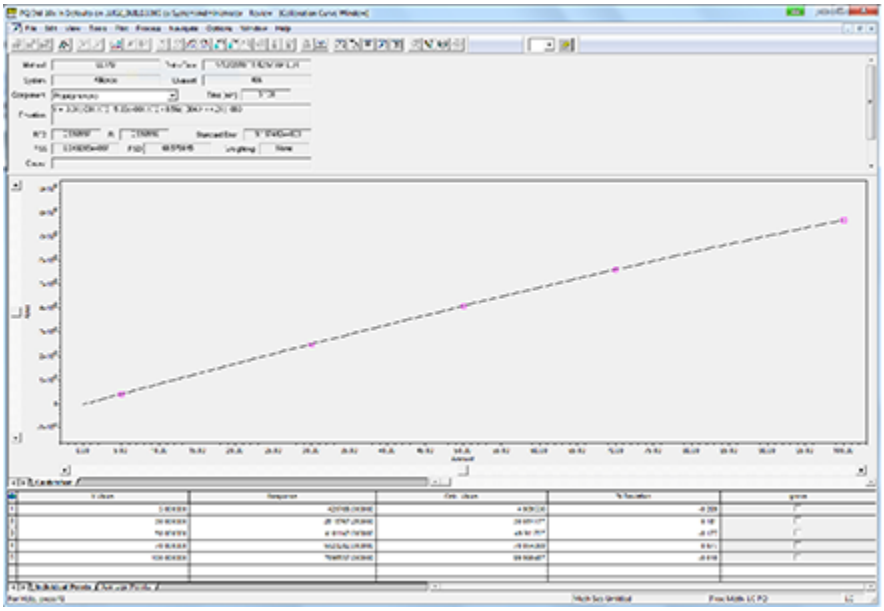
The software reports only positive values of x that are within the range of the calibration curve.

4.4.3.6 Cubic fit

To calculate a cubic fit calibration curve, the software performs a least-squares fit of a cubic polynomial to the calibration points. The fit cannot be performed with fewer than four calibration points, and a minimum of six points is strongly recommended.

The following figure illustrates a cubic fit calibration curve.

Figure 4–13: Cubic fit calibration curve:



The cubic fit generates a calibration curve that is represented by the equation:

$y = A + Bx + Cx^2 + Dx^3$

where:

y = Response of the standard peak calculated by the software

x = Component amount or concentration

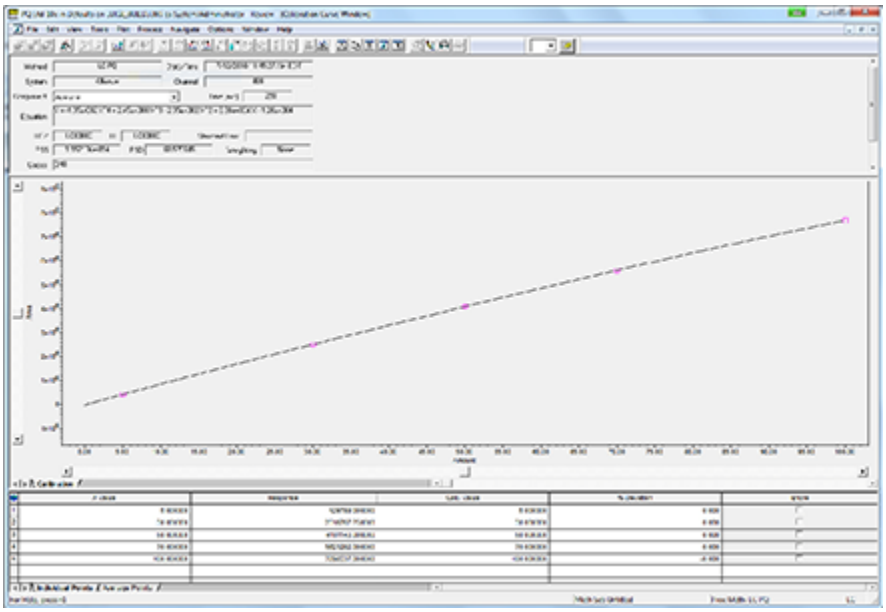
A, B, C, and D = Polynomial coefficients of the curve

4.4.3.7 Fourth-order fit

To calculate a fourth-order fit calibration curve, the software performs a least-squares fit of a fourth-order polynomial to the calibration points. The fit cannot be performed with fewer than five calibration points, and a minimum of seven points is strongly recommended.

The following figure illustrates a fourth-order fit calibration curve.

Figure 4–14: Fourth-order fit calibration curve:



The fourth-order fit generates a calibration curve that is represented by the equation:

$y = A + Bx + Cx^2 + Dx^3 + Ex^4$

where:

y = Response of the standard peak calculated by the software

x = Component amount or concentration

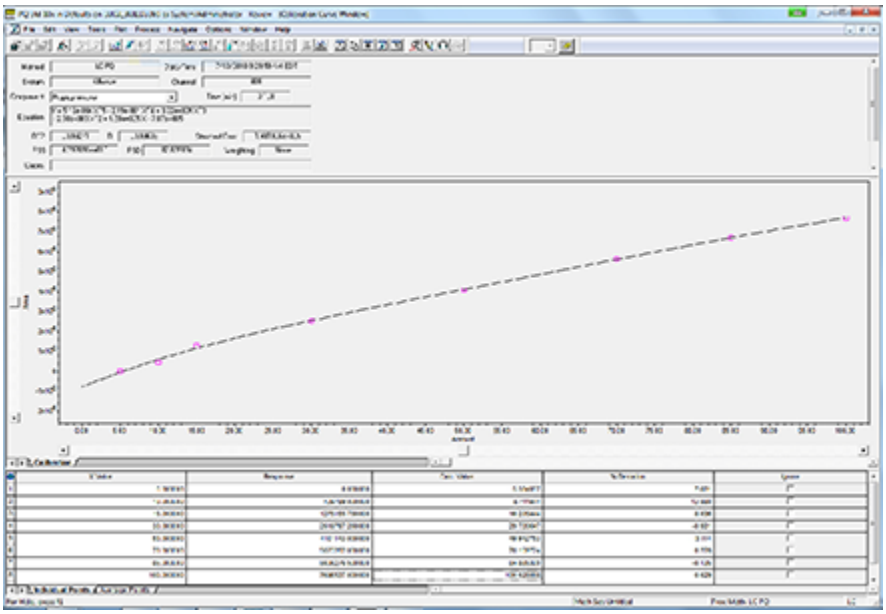
A, B, C, D, and E = Polynomial coefficients of the curve

4.4.3.8 Fifth-order fit

To calculate a fifth-order fit calibration curve, the software performs a least-squares fit of a fifth-order polynomial to the calibration points. The fit cannot be performed with fewer than six calibration points, and a minimum of eight points is strongly recommended.

The following figure illustrates a fifth-order fit calibration curve.

Figure 4–15: Fifth-order fit calibration curve:



The fifth-order fit generates a calibration curve that is represented by the equation:

$y = A + Bx + Cx^2 + Dx^3 + Ex^4 + Fx^5$

where:

y = Response of the standard peak calculated by the software

x = Component amount or concentration

A, B, C, D, E, and F = Polynomial coefficients of the curve

4.4.4 Multilevel forced-through-zero calibration curves

The software supports forced-through-zero for the following multilevel curve fits:

- Linear
- Quadratic
- Cubic
- Fourth-order
- Fifth-order
- Response factor

Each forced-through-zero fit is similar to the corresponding nonforced-through-zero fit, except that the curve is mathematically constrained to pass through zero. Forcing the calibration curve through zero results in different coefficients than those for nonforced calibration curves. For forced-through-zero fits, the zeroth order coefficient (C_0) is set to 0 and the software computes the remaining coefficients.

$y_1 = c_5(x_1)^5 + c_4(x_1)^4 + c_3(x_1)^3 + c_2(x_1)^2 + c_1(x_1)^1 + 0(x_1)^0$

$y_2 = c_5(x_2)^5 + c_4(x_2)^4 + c_3(x_2)^3 + c_2(x_2)^2 + c_1(x_2)^1 + 0(x_2)^0$

$y_3 = c_5(x_3)^5 + c_4(x_3)^4 + c_3(x_3)^3 + c_2(x_3)^2 + c_1(x_3)^1 + 0(x_3)^0$

$y_4 = c_5(x_4)^5 + c_4(x_4)^4 + c_3(x_4)^3 + c_2(x_4)^2 + c_1(x_4)^1 + 0(x_4)^0$

$y_5 = c_5(x_5)^5 + c_4(x_5)^4 + c_3(x_5)^3 + c_2(x_5)^2 + c_1(x_5)^1 + 0(x_5)^0$

$y_6 = c_5(x_6)^5 + c_4(x_6)^4 + c_3(x_6)^3 + c_2(x_6)^2 + c_1(x_6)^1 + 0(x_6)^0$

$y_7 = c_5(x_7)^5 + c_4(x_7)^4 + c_3(x_7)^3 + c_2(x_7)^2 + c_1(x_7)^1 + 0(x_7)^0$

The following table shows the differences between standard equations and forced-through-zero equations.

Table 4–5: Standard and forced-through-zero equation format comparison:

Fit type	Standard equation	Forced-through-zero equation
Linear	$y = A + Bx$	$y = Bx$
Quadratic	$y = A + Bx + Cx^2$	$y = B + Cx^2$
Cubic	$y = A + Bx + Cx^2 + Dx^3$	$y = Bx + Cx^2 + Dx^3$

For information on forced-through-zero fits, see the relevant nonforced-through-zero section for that type of fit. The software performs an equivalent computation, but coefficient A is always zero.

4.4.5 Weighting

Weighting is applied while fitting curves to multilevel points to achieve the following ends:

- Ensure that the points that have the most certainty (least error) contribute most strongly to the determination of the coefficients.
- Adjust for the differences in precision of the Y-value (response or response ratio) with respect to the X-value (amount or concentration).

To fit a curve to the calibration data, the software performs a least-squares fit to select the coefficients that minimize the sum of the differences between the individual points in the curve.

- Without weighting, all points contribute equally to that sum.
- With weighting, the contributions are adjusted to reflect the variance at each calibration level.

The equation that is minimized is:

$$\sum_{i=1} \frac{\left(\hat{y}_i - y_i\right)^2 w_i}{Degrees\,Of\,Freedom}$$

where:

y_i = Observed data point

$\hat{y}_i$ = Calculated data point

w_i = Weighting factor for each data point

Degrees Of Freedom = The number of points minus the number of coefficients calculated

Unweighted data assumes equal precision at all levels ($w_i = 1$).

To select the weighting type, plot the standard deviation for each level versus the X-value. Then select the weighting type based on the observed variation of the standard deviation by level.

Weighting can be applied to the standard fit types:

- Linear
- Quadratic
- Cubic
- Fourth-order
- Fifth-order

The types of weighting and the results of their application are described in the following table.

Table 4–6: Weighting application results:

Weighting type	Weighting equation
x	$w_i = x_i$: This results in a fit to the points at the high end of the curve (amounts or concentrations).
1/x and $1/x^2$	$w_i = 1/x_i$ or $w_i = 1/x_i^2$: This results in a fit to the points at the low end of the curve (amounts or concentrations). The weight for the point and the coefficients cannot be calculated if x = 0. If there is a point where x = 0, the coefficients of the curve are not calculated and processing code Q28 is copied into the curves code field.
1/y and $1/y^2$	$w_i = y_i$: This results in a fit to the points at the low end of the curve (response). The weight for the point and the coefficients cannot be calculated if y = 0. If there is a point where y = 0, the coefficients of the curve are not calculated and processing code Q30 is copied into the curves code field.
x^2	$w_i = x_i^2$: This results in a fit to the points at the high end of the curve (amounts or concentrations).

Table 4–6: Weighting application results: (continued)

Weighting type	Weighting equation
log x	w_i = log x_i: Produces a fit that weights the points on the calibration curve by a factor of log base 10, resulting in a logarithmic fit to the points on the high end of the calibration curve. If $x_i < 0$, the weight for the point and the coefficients of the calibration curve are not calculated, and processing code Q29 is copied into the curves code field.
ln x	w_i = ln x_i: Produces a fit that weights the points on the calibration curve by a factor of the natural log of X, resulting in a logarithmic fit to the points on the high end of the calibration curve. If $x_i < 0$, the weight for the point and the coefficients of the calibration curve are not calculated, and processing code Q29 is copied into the curves code field.

where:

w_i = Weighting factor for each data point

x_i = X value of the point

y_i = Y value of the point

Tip: If the software cannot calculate the weighted points, the coefficients for the curve are not calculated and a processing code is generated, indicating the reason it is not calculated.

4.4.6 Statistics

Statistics indicate goodness of fit. The software calculates the following statistical criteria:

- Coefficient of determination
- Correlation coefficient
- Residual sum of squares
- Standard error of estimate of y on x (no report)[1]
- Standard variance (no report)[1]
- Standard error of calibration
- Percent Relative Standard Deviation
- Calculated value and percent deviation of the calibration points

[1] The software calculates these two criteria that are not reported as intermediate values.

4.4.6.1 Coefficient of determination

Coefficient of determination (R^2) is a rough indicator of the goodness of fit and is calculated by:

$$R^2 = 1 - \frac{(S_y)^2}{\sigma_y^2}$$

where:

R^2 = Coefficient of determination

R = Correlation coefficient

S_y = Standard error of estimate of y on x

σ^2_y = Standard variance

4.4.6.2 Correlation coefficient

The correlation coefficient (R) is an indicator of goodness of fit. It is the square root of the coefficient of determination.

4.4.6.3 Standard error of estimate of Y on X

The standard error of estimate of y on x (S_y) determines R^2 (the coefficient of determination) and R (the correlation coefficient) and is calculated by:

$$S_y = \sqrt{\frac{1}{n}\sum_{i=1}^{n} W_i\left(\hat{y_i} - y_i\right)^2}$$

where:

n = Number of points

The software calculates these two criteria that are not reported as intermediate values.

w_i = Weighting factor (set to 1 for uniform weighting)

$\hat{y_i}$ = Weighting factor (set to 1 for uniform weighting)

y_i = Response of a calibration point

4.4.6.4 Standard variance

The standard variance ($\sigma^2 y$) is used to calculate the coefficient of determination and correlation coefficient. It is computed as follows:

$$\sigma^2 y = \frac{1}{n}\sum_{i=1}^{n} w_i\left(y_i - \bar{y}\right)^2$$

where:

w_i = Weighting factor (set to 1 for uniform weighting)

y_i = Response of a calibration point

$\bar{y}$ = Weighted mean given by the equation:

$$\bar{y} = \frac{\sum_{i=1}^{n} w_i y_i}{\sum_{i=1}^{n} w_i}$$

4.4.6.5 Residual sum of squares

Residual sum of squares (RSS) is an indicator of goodness of fit and precision of data. It is used to calculate the standard error of estimate and the standard error of calibration. It is calculated by:

$$RSS = \sum_{i=1}^{n} w_i \left(\hat{y}_i - y_i \right)^2$$

where:

RSS = Residual sum of squares

n = Number of points

w_i = Weighting factor (set to 1 for uniform weighting)

$\hat{y}_i$ = Responses as predicted using the calibration curve

y_i = Response of a calibration point

4.4.6.6 Standard error of calibration

The standard error of calibration (E) is the square root of the sum that is minimized when fitting coefficients to the curve and is calculated by:

$$E = \sqrt{\frac{1}{d}\left(\sum_{i=1}^{n} w_i \left(\hat{y}_i - y_i \right)^2 \right)} = \sqrt{\frac{1}{d} RSS}$$

where:

d = Degrees of Freedom = Number of points minus the number of coefficients calculated

w_i = Weighting factor (set to 1 for uniform weighting)

$\hat{y}_i$ = Responses as predicted using the calibration curve

y_i = Response of a calibration point

RSS = Residual sum of squares

4.4.6.7 Calculated value and percent deviation of calibration points

The calculated value and percent deviation of calibration points can be used to assess how well the points fit the curve by visual inspection or by plotting against the X value.

Percent deviation is calculated by:

$$\%\ Deviation = 100\left(\frac{\hat{x}_i - x_i}{\hat{x}_i}\right)$$

where:

$\hat{x}_i$ = X value as predicted using the calibration curve (the calculated value)

x_i = X value of the calibration point

Plots of percent deviation versus amount should display random scatter if the fit type is correct. Plots of calculated value versus amount or concentration should be linear.

4.4.6.8 Percent relative standard deviation

The Percent RSD is an indication of goodness of fit and precision of the data.

The software calculates Percent RSD using the following equation:

$$\%\ RSD = \frac{\left(\sum_{i=1}^{n}\left[w_i \bullet y_i - YWM\right]^2 \div (n-1)\right)^{\frac{1}{2}}}{YWM} \bullet 100$$

where:

› w_i = Weighting factor (set to 1 for uniform weighting)

› Y_i = Response of a calibration point

› *YWM* = Weighted mean response of all calibration points, which is expressed as:

$$YWM = \frac{\sum_{i=1}^{n}(w_i \bullet y_i)}{n}$$

› *n* = The number of points

4.5 References

For further information on the theory of quantitation, see:

- Press, William H., Teukolsky, Saul A., Vetterling, William T., and Flannery, Brian P., *Numerical Recipes*, Cambridge University Press, Cambridge, UK, 2007.
- Massart, D.L., et al., *Chemometrics: A Textbook*, Elsevier Science Publishers, Amsterdam, 1988.
- Papoulis, Athanasios, *Signal Analysis*, McGraw-Hill, New York, 1977.
- Snyder, Lloyd R., Kirkland, Joseph J., and Dolan, John W., *Introduction to Modern Liquid Chromatography*, third ed., Wiley-Interscience, Hoboken, New Jersey, 2010.
- Strang, Gilbert,*Linear Algebra and Its Applications*, Harcourt Brace Jovanovich, Inc., New York, 1988 .