

Enhanced CPQ Quote Line Editor

New Configuration guide - Version 1



CONTENTS

1. Introduction.....	3
2. User Interface Customization	5
a. Approach.....	5
b. Custom Column Layout	5
c. Customizing and Pinning.....	6
d. Additional Functionality	6
3. Excel Import Capability	6
4. Multi-User Collaborative Quote Management	7
5. Use Cases	8
6. Pain Points/Features Requested by RCA/RLM/ARM and CPQ customers	9
7. Software Requirement	10
8. Configuration Steps	10
9. Quote Action Field Details	12
10. Understanding the Quote Action Record	15
11. Quote Action Record Fields	15
12. Configuration JSON – Data Section	19
13. Relationship Scoping	20
14. Configuration JSON – Formatting Section	21
15. Field Set Prerequisites	22
16. Custom Metadata: QLE+ Configurations (Global Settings).....	23
17. Key Global Settings Explained	23
18. Security & Visibility	26
19. Best Practices	26

1. INTRODUCTION

QLE+ — A Next-Generation Collaborative Quote Line Editor

To address real-world usability challenges experienced in many RCA/RLM/ARM and CPQ quote editors (including Salesforce RCA/RLM/ARM and CPQ), we developed:

QLE+ — Enhanced Quote Line Editor PLUS and Estimator

QLE+ is the industry's first Multi-User Collaborative Quote Line Editor PLUS — a rich user interface for quoting experiences that can integrate with any Salesforce-based quoting environment *without cost to the customer*.

Think of QLE+ the way you'd think of using **Microsoft Word instead of Notepad** — even though both can edit text, one is significantly richer, more intuitive, and far more productive.

What QLE+ Solves

QLE+ enhances the RCA/RLM/ARM and CPQ user experience by addressing several long-standing pain points in traditional quote line editors. It brings flexibility and power that benefit **any transaction involving line items**, including:

- Quotes
- Orders
- Invoices
- Opportunities
- Subscriptions

Key Capabilities of QLE+

✓ User-Friendly Layout and Customization

- Custom column layouts per user

- Freeze columns
- Column headings tooltips with help text

✓ **Visual Formatting & Reporting**

- Add colors, fonts, bold/italics — just like a spreadsheet
- **Conditional formatting** rules similar to Excel
- Highlight important sales thresholds, exceptions, or alerts

✓ **Advanced Filtering & Grouping**

- Custom filters to sort on any column
- **Dynamic grouping by fields** (example: Product Family)
- Save personal views & filters

✓ **Expanded Data Context**

- Expose any Salesforce object (e.g., Installed Assets) within the editor
- Enable economic analysis directly within the quote
- Preview standard approval summaries without leaving the editor

✓ **Flexible Navigation & Tabs**

- **Unlimited tabs**, each with independent configurations
- Tab content can reference any object, field set, or layout

✓ **Collaborative & Export Features**

- Multi-user collaboration in a unified environment
- Robust export (Excel/PDF) with branding
- Intuitive grid interaction for business analysts, sales ops, and quoting teams

Why QLE+ Matters

QLE+ empowers users to **interact with RCA/RLM/ARM and CPQ data more efficiently** than traditional line-item interfaces by providing:

- **Intuitive customization options**
- **Advanced filtering and grouping**
- **Conditional styling and formatting**
- **Robust export and preview capabilities**

These improvements are especially helpful when standard RCA/RLM/ARM and CPQ UIs become cumbersome due to complexity, large line items, or the need for analytical views.

2. USER INTERFACE CUSTOMIZATION

Any transaction with line items can benefit from this add-on. This includes Orders, Invoices, Opportunities, Subscriptions.

a. Approach

- All data and users remain within Salesforce — no external data storage.
- QLE+ complements RCA/RLM/ARM and CPQ ; it does not replace core RCA/RLM/ARM and CPQ functionality.
- Enhances usability while preserving existing pricing logic and calculations.

b. Custom Column Layout

- Each user can create a personalized column layout.
- Select fields relevant to their role.
- Layout preferences are saved per user for future sessions.

c. Customizing and Pinning

- Columns can be pinned left or right.
- Layouts persist across sessions.
- Users maintain full control over data visibility.

d. Additional Functionality

- Conditional formatting for visual alerts.
- Custom filters for focused analysis.
- Dynamic grouping by product, family, quote type, etc.

3. EXCEL IMPORT CAPABILITY

QLE+ allows direct Excel import without converting to CSV.

- Drag and drop Excel files directly into the quote.
- Automatically split large files into multiple quotes (e.g., 1500 lines → 5 quotes of 300 each).
- No need to rename or rearrange columns.
- Preserves formatting, formulas, macros, tabs, and extra columns.
- All lines can be viewed together in one consolidated screen.

This significantly improves efficiency and eliminates manual preparation.

4. MULTI-USER COLLABORATIVE QUOTE MANAGEMENT

QLE+ enables collaborative editing across multiple quotes in a single aggregated view.

Key features:

- Combine multiple quotes into one editable interface
- Overcome Salesforce CPQ quote line limits
- Individualized user perspectives within shared environment
- Real-time collaboration and updates
- Customized tabs for service, manufacturing, procurement costs
- Integration with CPQ for unified estimation
- Usage-based billing visibility
- Child object exposure (BOM, routings, service costs)

This enhances productivity and cross-functional collaboration.

5. USE CASES

- a. Any transaction with line items can benefit from this add-on. This includes Orders, Invoices, Opportunities, Subscriptions.
- b. Inclusion of AI type capability where in a quote is proposed automatically based on historical quotes for the customer's industry, region, win/loss history and other configurable parameters
- c. Ability to see all associated transactions on one screen powered by user configurable tabs

☐ Lead

o Opportunity

☐ Quote

☐ Order

o Contracts

☐ Renewal Opportunity

☐ Renewal Quote

6. PAIN POINTS/FEATURES REQUESTED BY RCA/RLM/ARM AND CPQ CUSTOMERS

a. User Interface

- Ability to set custom column layout by User, Freeze columns, Column Heading Help Text Tool Tips.
- Add colors, fonts, formatting, bold/italics.
- Have custom filters to be able to sort on any columns.
- Grouping in the quote line editor by any field such as Product Family on the fly.
- Conditional formatting: Based on rules, just like in Excel, be able to show different colors, fonts, etc.
- Preview approvals from within the QLE for Standard approvals (as you know, there is no preview approval capability for standard approvals).
- Ability to expose any object such as installed assets within the quote line editor.
- Economic analysis of the quote with a configurable field set.
- Ability for the user to save his/her own selection of columns and the order in which they would like to see them.
- Unlimited Tabs each with its field set, colors, columns, or referring to any object in Salesforce.

b. Excel Upload (Not CSV, but existing Excel workbooks)

This is a separate module

- Upload quote line items by dragging an existing Excel (not CSV but Excel) sheet onto the quote without having to convert to CSV. Works with existing excel sheets.
- Automatically create multiple quotes from a single Excel file based on a configurable parameter, such as creating one for every 300 lines. So an Excel file with 1500 lines will create 5 quotes. While QLE+ allows you to view the 1500 lines all together in a single screen and work on it as if it were a single quote.

c. Multi-User Collaborative Quote Management with Service Cost Estimation

- Combining multiple quotes into a single view and being able to make edits across the quotes.
- Overcoming the Salesforce CPQ quote line limit while also giving multiple users their view of an
- overall aggregated quote.
- Ability to maintain separation of Service Items, Manufacturing Costs, Procurement cost with
- customized tabs for each.
- Ability to leverage CPQ to estimate projects within the context of the Sales Quote for a unified
- quoting and estimation experience.
- Can surface Consumption Schedules for Usage Based Billing.

7. SOFTWARE REQUIREMENT

The recommended Salesforce CPQ version should be v238 and up to avail all the features in the application.

After Successful installation of the package in your environment, following steps should be performed:

8. CONFIGURATION STEPS

d. Step 1: Authorize new calculation services

Go to Setup>Installed Packages> Salesforce CPQ> Configure> Pricing and Calculation (Tab)

Uncheck the “Use Integration User for Calculations” checkbox, wait for a few seconds.

The screenshot shows the 'Settings Editor' for 'Salesforce CPQ'. The 'Pricing and Calculation' tab is selected. The 'Use Integration User for Calculations' checkbox is highlighted with a red box. Other settings include 'Currency Symbol', 'Unit Price Scale', 'Enable Quick Calculate', 'Allow Non-Consecutive Custom Segments', 'Enable Pricing Guidance', 'Quote Line Edits for Usage Based Pricing', 'Use Legacy Calculator', 'Use Inactive Prices', 'Calculate Immediately', 'Disable Background Calculation Refresh', 'Enable Usage Based Pricing', and 'Hide uncalculated quote warning'.

After Unchecking the “Use Integration User for Calculations” checkbox, to authorize a new calculation service, click on the Authorize new calculation service link (at the bottom).

Note: This step is only applicable on Authorization of new calculation services for the first time.

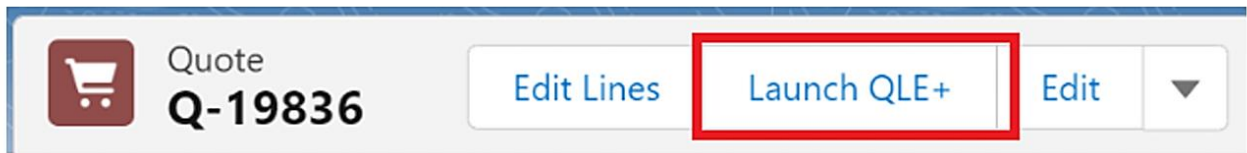
e. Step 2: Assign "Enhanced Quote Line Editor Access" permission set to the users.

In order to provide access to the Enhanced Quote Line Editor, "Enhanced Quote Line Editor Access"

permission set needs to be assigned to the user.

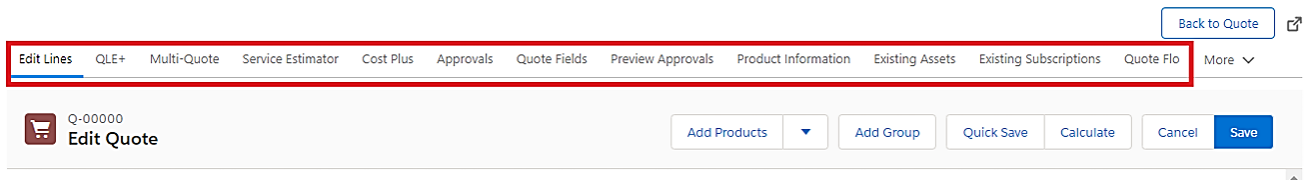
f. Step 3: Add “Launch QLE+” Action on the Page Layout/ Lightning Page Layout of Quote object.

To experience Enhanced Quote Line Editor, Launch QLE+ Action needs to be added on the assigned page layout/ lightning page layout of the Quote object.



g. Step 4: Create Quote Actions records.

In Enhanced Quote Line Editor screens, there are different configurable tabs.



- These tabs and their configuration are controlled by the Quote Action object.
- Each Quote Action record represents a Tab in Enhanced Quote Line Editor screen. To add a new Tab clone an already existed Quote Action record as per the required view.
- Accessibility of the tab is based on accessibility of the record, it can be controlled using sharing rules.

Quote Actions							
All							
12 items • Sorted by Display Order • Filtered by All quote actions • Updated 8 minutes ago							
☐	Quote Action N...	Display Order	Active	Label	Name	Type	Component Name
☐	QA-0000	10	☒	Edit Lines	Edit Lines	VF Page	SBQQ_sb
☐	QA-0001	20	☒	QLE+	Quote Lines	VF Page	CPQ2_QLEGridContainerPage
☐	QA-0002	25	☒	Multi-Quote	Quote Lines	VF Page	CPQ2_QLEGridContainerPage
☐	QA-0003	26	☒	Service Estimator	Quote Lines	VF Page	CPQ2_QLEGridContainerPage
☐	QA-0004	27	☒	Cost Plus	Quote Lines	VF Page	CPQ2_QLEGridContainerPage
☐	QA-0005	29	☒	Approvals	Quote Lines	VF Page	CPQ2_QLEGridContainerPage
☐	QA-0006	30	☒	Quote Fields	Economic Analysis	Aura/LWC	
☐	QA-0007	40	☒	Preview Approvals	Preview Approvals	Aura/LWC	
☐	QA-0008	50	☒	Product Information	Product Information	VF Page	CPQ2_QLEGridContainerPage
☐	QA-0009	60	☒	Existing Assets	Existing Assets	VF Page	CPQ2_QLEGridContainerPage
☐	QA-0010	70	☒	Existing Subscriptions	Existing Subscriptions	VF Page	CPQ2_QLEGridContainerPage
☐	QA-0011	80	☒	Quote Flo	Quote Flo	Aura/LWC	CPQ2:revenueFloMain

9. QUOTE ACTION FIELD DETAILS

1. **Label:** Label field determines the name of the Tab in Enhanced Quote Line Editor.
2. **Display Order:** Display Order field determines the display order in which the tab will be displayed at the top bar of Enhanced Quote Line Editor.

3. **Active:** Active field determines the availability of the tab.
4. **Type:** Type field is a configuration field used in the managed package to determine the view of the screen.
5. **Component Name:** Component Name is a configuration field used in the managed package to determine the view of the screen.
6. **Configuration:** Configuration field is an editable text field which stores a JSON to determine the configuration of the screen.

This JSON contains the following keys:

Data: This section contains information to fetch and publish the data.

objectName	Contains the API name of the Object to query the data
treeview	It is of type boolean.if enabled then it will show data in tree view (eg. for bundle product)
fieldSetName	Contains the API name of the fieldset to publish fields on Enhanced Quote Line Editor
aggregateField	Contains API name of the QuoteLine field for aggregation after grouping.
parentObjectField	Contains API name of the Lookup field to create Parent-Child Quote relationship.
parentObjectId	Contains API name of the Lookup to query the data.
quoteld	Contains API name of ID field of the Quote

Formatting:

This section is used to achieve conditional formatting in Enhanced Quote Line Editor Screens.

Active	It is used to activate/deactivate the formatting.	
appliesTo	Contains the API name of the field(column) on which conditional formatting will be applied.	
fontColor	Contains the color code of the required font color.	
backgroundColor	Contains the color code of the required background color.	
Criteria	It is a List of objects to determine the criteria to apply formatting.	
	fieldName	Contains API name of the field to calculate the criteria.
	Operator	Contains the operator (english expression) to calculate the criteria Eg. less than or equals. All the operators are mentioned in Table below.
	inputValue	Contains the value to calculate the criteria.

Operators:

Symbol	Operator Input
==	Equals
!=	not equals
>	greater than
>=	greater than or equals
<	less than
<=	less than or equals

10. UNDERSTANDING THE QUOTE ACTION RECORD

The **Quote Action Record** is the primary configuration object that defines how each tab behaves within QLE+.

Each Quote Action record represents **one tab** in the QLE+ interface.

Through this record, administrators can define:

- What data should be displayed
- How that data is related to the current Quote
- Whether users can edit the grid
- How data should be visually formatted
- Whether the tab should be visible to users

QLE+ reads these records dynamically when a Quote is opened and builds the UI accordingly. There are no hardcoded tabs — everything is metadata-driven.

11. QUOTE ACTION RECORD FIELDS

a. Label

The **Label** field determines the name displayed in the QLE+ top navigation bar.

This value appears exactly as entered and should be:

- Business-friendly
- Clear and intuitive
- Aligned with business terminology

Examples:

- Quote Lines
- Extended Quote Lines
- Subscriptions
- Approval Summary

Since this is user-facing, avoid technical API names.

b. Display Order

The **Display Order** field controls the position of the tab in the QLE+ navigation bar.

Tabs are rendered in ascending order.

Example:

Display Order Tab

10	Quote Lines
20	Extended Quote Lines
30	Assets

Best Practice:

Use increments of 10 (10, 20, 30) to allow flexibility for future insertions.

If two tabs share the same Display Order, rendering order is not guaranteed.

c Active

The **Active** checkbox determines whether the tab is available.

- If Active = True → Tab is eligible to be displayed.
- If Active = False → Tab is completely hidden from QLE+.

This is useful for:

- Temporary deactivation
- Controlled rollouts
- Sandbox testing

Note: Salesforce security still applies even if Active = True.

d. Type

The **Type** field defines how the tab content is rendered.

Supported values:

- VF Page
- Aura/LWC

Behavior:

If VF Page is selected:

- Component Name must contain a Visualforce page.

If Aura/LWC is selected:

- Component Name must contain Aura or LWC component name.
 - LWC must implement @api recordId to receive the Quote Id.
-

Standard Grid Rendering (Excel-like Experience)

To render the QLE+ grid:

- Type = VF Page
- Component Name = CPQ2__QLEGridContainerPage

This is the standard configuration used for grid-based tabs.

e. Component Name

The **Component Name** field defines the UI component loaded when the tab is clicked.

Depending on Type, this can represent:

- A Visualforce page
- An Aura component
- A Lightning Web Component

The managed package dynamically loads this component at runtime.

f. Export Logo Resource Name

This field supports Excel export functionality.

It stores:

- The name of a Static Resource
- The static resource must contain a company logo image

Important:

- Only the resource name is stored
- Do not include file paths
- The image is embedded into exported Excel files

This ensures branding consistency.

g. Filter JSON

The **Filter JSON** field stores the default filter state of the tab.

Key characteristics:

- Stored in JSON format
- Generated via “Save Filter” button
- Applies globally to all users
- Not user-specific
- System-managed

Manual editing is not recommended. Invalid JSON can break grid rendering.

h. Configuration (Most Critical Field)

The **Configuration** field contains structured JSON that defines:

- Data source
- Relationships
- Grid behavior
- Editability
- Filtering capability
- Conditional formatting

The JSON is divided into two primary sections:

- Data
- Formatting

The configuration is parsed at runtime. No deployment is required for changes.

12. CONFIGURATION JSON – DATA SECTION

The Data section controls how records are fetched and displayed.

It defines:

- Which object is queried
- How it relates to the Quote
- Which fields appear as columns
- Whether records are editable

Data Attributes

Key	Description
objectName	Target object displayed in grid
Treeview	Enables hierarchical display (bundles)
fieldSetName	Controls visible columns
aggregateField	Used for grouping and totals
parentObjectField	Relationship path used for scoping
parentObjectId	Runtime filter field
quoteId	Field used to identify current Quote
Editable	Enables inline editing
enableSaveFilter	Enables Save Filter button

13. RELATIONSHIP SCOPING

These two attributes define how non-Quote Line objects are filtered in the context of the current Quote.

They allow QLE+ to:

- Traverse object relationships

- Resolve parent context dynamically
- Display indirectly related records

Example 1: Extended Quote Lines

"parentObjectField": "Quote_Line__r.SBQQ__Quote__r.Id",
"parentObjectId": "Id"

Flow:

Extended Quote Line → Quote Line → Quote

All related records are scoped to the opened Quote.

Example 2: Assets via Account

"parentObjectField": "AccountId",
"parentObjectId": "SBQQ__Account__c"

Flow:

Quote → Account

Asset → Account

All Asset records associated with the Account of the opened Quote are displayed.

14. CONFIGURATION JSON – FORMATTING SECTION

The Formatting section controls visual styling of grid data.

It does not affect database records.

Formatting Attributes

Key	Description
Active	Enables the rule

appliesTo	Column API or ROW
fontColor	Font color applied
backgroundColor	Background color applied
Criteria	Conditions triggering formatting

Criteria Logic

- Multiple criteria use AND logic
- All criteria must evaluate true
- Evaluation occurs per row

Row-Level Formatting

To highlight entire records:

"appliesTo": "ROW"

Common use cases:

- Pricing exceptions
- Approval thresholds
- Validation warnings

15. FIELD SET PREREQUISITES

These configurations are required outside of JSON.

Read-Only Columns

Mark a field as Required in the field set.

QLE+ interprets required fields as read-only.

Quick Save Dependency

The Quote lookup field must exist in the field set.

- Can be hidden
- Required for save context resolution

Without it, save operations may fail.

16. CUSTOM METADATA: QLE+ CONFIGURATIONS (GLOBAL SETTINGS)

The QLE+ Configurations custom metadata controls cross-tab behavior.

Unlike Quote Action records (tab-specific), these settings apply globally.

They allow administrators to:

- Control UI behavior
- Tune performance
- Customize labels
- Optimize calculation timing

All settings are applied dynamically at runtime.

17. KEY GLOBAL SETTINGS EXPLAINED

a. Enable Undo For Column

Displays Undo Reset button.

Resets:

- Pinned columns
- Custom column ordering
- User layout rearrangements

Does not impact data.

b. Undo Reset Button Label

Customizes the button label.

Examples:

- Reset Columns
- Restore Default View
- Undo Layout Changes

c. Grid Height Adjustment

Controls vertical height of the grid.

Useful for:

- Large monitors
- Laptops
- Experience Cloud layouts

d. Custom Scale Value

Defines number of digits displayed after decimal.

Example:

- 2 → 100.25
- 4 → 100.2537

Display-only setting.

e. Quick Save Button Label

Changes Quick Save button text globally. Does not affect save logic.

f. Refresh Component on Select

When enabled:

- Switching tabs refreshes data

When disabled:

- Data is cached until manual refresh

Useful in dynamic pricing environments.

g. Override ReadOnly Style

Enables custom styling of read-only column headers.

Works with `ReadOnlyHeaderFontColor`.

h. ReadOnlyHeaderFontColor

Accepts:

- Hex codes (`#A9A9A9`)
- Named colors

Applied only if `Override ReadOnly Style` is enabled.

i. Save Timeout MilliSeconds

Defines delay before grid refresh after async CPQ calculations.

Important because:

- Price rules and QCPs run asynchronously
- Early refresh may show stale data

Tuning guidance:

- Lower value for smaller orgs
- Higher value for complex pricing logic
- Depends on price rule volume and QCP complexity

18. SECURITY & VISIBILITY

QLE+ fully respects Salesforce security.

Tabs are visible only if:

- User has object access
- User has record access
- Quote Action record is Active

Controlled via:

- Profiles
- Permission Sets
- Sharing Rules
- Manual Sharing

This enables secure, persona-based UI design.

19. BEST PRACTICES

- Avoid manual JSON edits
- Prefer metadata over custom code
- Use sharing for visibility control
- Keep formatting rules minimal for performance
- Leave numeric gaps in Display Order
- Tune Save Timeout carefully in heavy pricing environments