Enhanced Calculation Engine Changes

This update to the Enhanced Calculation Engine introduces a set of targeted improvements that include defect resolutions, functional enhancements, and new calculation capabilities. These changes are outlined in the list below and are intended to provide more consistent behavior, expanded flexibility, and clearer outcomes across supported functions.

For additional details, including function signatures, input and output parameters, descriptions, and guidance on how to use updated or newly added functions, please refer to the User Documentation.

ID	Description
ARCHER-5763	Previously, SUMIF returned errors when criteria were constructed using concatenated operators and field references (for example, ">="&[YEAR]), even though hardcoded criteria like ">=2012" worked; now, it correctly evaluates concatenated criteria expressions and returns the expected summed values.
ARCHER-7484	Previously, the IF() function could return only a single value from a multi-select list field, limiting formulas that needed to output multiple selections. Now, the IF() function supports returning multiple values when the return expression contains a list, removing the prior single-value restriction.
ARCHER-22633	Previously, calculated fields that depended on other calculated fields in error did not provide enough detail in the UI or logs to identify the underlying source issue, making troubleshooting difficult. Now, the system surfaces clearer diagnostic information, allowing teams to quickly pinpoint which referenced field caused the error.
ARCHER-30027	Previously, IF statements misinterpreted text resembling exponential notation (for example, "4e11" or "5e00") as numeric values, causing calculation errors like “Stack empty”; now, it correctly treats these values as text inside IF expressions and returns the expected text results.
ARCHER-30543	Previously, the GAMMAINV() function could trigger errors and high CPU usage when the computed result was less than 1, preventing records from saving and occasionally causing system timeouts. Now, GAMMAINV() correctly returns values below one, resolving this edge-case defect and improving both mathematical accuracy and system stability.
ARCHER-73131	Archer now supports the SWITCH() function in calculated field formulas. This function evaluates a single expression against a list of specific values and returns the result corresponding to the first matching value. If no match is found, an optional default value can be returned.
ARCHER-121798	The BINOMDIST function previously did not consistently return errors for invalid inputs (for example, when number_s < 0, number_s > trials, or probability_s was outside 0 - 1); the documentation also did not accurately describe this behavior. It has now been updated to correctly error on invalid inputs, and the documentation has been revised accordingly.

Confidentiality Notice: This document is for the sole use of the Archer associates, partners, and clients and may contain confidential or proprietary information. Any unauthorized review, use, disclosure or distribution is prohibited.

ARCHER-121800	Previously, the CHIDIST() function did not always return errors for invalid input values (such as negative x or out-of-range degrees_freedom), causing behavior that differed from Archer's documentation and Excel; now, it correctly validates parameters, and its results and error handling align with the documented behavior and Excel output.
ARCHER-121802	Previously, the CHIINV() function could fail to return errors for out-of-range probability or degrees of freedom values, producing results that did not match documentation; now, it correctly errors on invalid probability and degrees_of_freedom inputs in line with the documented expectations.
ARCHER-121804	Previously, the CONFIDENCE() function did not consistently return errors for invalid alpha, standard deviation, or size values and could produce results that differed from the documentation; now, it correctly validates these parameters, returning errors for invalid inputs in line with the documented expectations.
ARCHER-121806	Previously, the CRITBINOM() function did not always return errors for negative or out-of-range input values, causing behavior that could differ from the Archer documentation and Excel; now, it correctly validates parameters, returning errors for invalid inputs so its behavior aligns with the documented expectations and Excel output.
ARCHER-121812	Previously, the EXPONDIST() function did not reliably return errors for negative or invalid input values, causing inconsistencies with its documented behavior and the corresponding Excel function; now, it correctly validates inputs and its behavior aligns with the Archer documentation and Excel output.
ARCHER-121813	Previously, the FISHER() function did not consistently return errors for inputs outside the valid range (-1, 1), leading to behavior that differed from both the documentation and Excel's FISHER function. Now, the function correctly throws errors for out-of-range values, ensuring full alignment with the documented rules and Excel's output.
ARCHER-121814	Previously, the GAMMADIST() function could fail to return errors for negative or otherwise invalid input values, and its handling of the "cumulative" parameter differed from the documentation; now, the engine correctly errors on invalid parameters and aligns with the documented behavior.
ARCHER-121815	Previously, the LOGINV() function did not consistently return errors for out-of-range probability or negative standard deviation values, causing discrepancies with its documented behavior and Excel's LOGINV function; now, it validates inputs correctly, returning errors for invalid parameters and results that align with the documented behavior and Excel output.
ARCHER-121816	Previously, the NEGBINOMDIST() function did not always return errors for negative or out-of-range input values and could produce results that differed from its documented behavior and from Excel's function; now, it correctly validates inputs, returning errors for invalid parameters and producing results that align with the documented behavior and Excel's output.
ARCHER-121817	Previously, the POISSON() function did not consistently return errors for negative or invalid input values, causing behavior that differed from its documented behavior and

Confidentiality Notice: This document is for the sole use of the Archer associates, partners, and clients and may contain confidential or proprietary information. Any unauthorized review, use, disclosure or distribution is prohibited.

	Excel's POISSON function; now, it validates inputs correctly, and its return values align with the documented behavior and Excel output.
ARCHER-121818	Previously, the WEIBULL() function did not consistently return errors for invalid parameter inputs (such as negative x or non-positive alpha/beta) and could produce results that differed from Excel; now, it errors for invalid parameters as expected and returns correct results for valid inputs, matching Excel's behavior and the documented requirement.
ARCHER-121849	Previously, the NORMDIST() function could return values that differed from its documented behavior and from Excel's NORMDIST () function; now, its return values align with Excel's output.
ARCHER-121963	Previously, the RANK() function could return incorrect results for certain inputs, differing from both Archer's documented behavior and Excel's RANK function; now, it correctly returns results that align with the documentation and match the corresponding Excel function's output.
ARCHER-154656	Text functions such as SUBSTRING and LEN may include quotation marks and HTML tags in their calculations for single-line and multi-line text fields, leading to inconsistent results.
ARCHER-170470	Date/time extraction functions used in calculated fields (e.g., YEAR, MONTH, DAY, HOUR, MINUTE, MONTHNAME, QUARTER, WEEKDAY, WEEKNUMBER) previously returned errors when an empty date was passed. These functions now return empty when the input date is empty, preventing calculation errors.
ARCHER-170688	Fixed an issue where IF statements failed to process value list entries containing quotation marks returned from the VALUEOF() function (for example, ab"c), causing calculation errors.
ARCHER-171074	Previously, dependent field errors were not consistently honored in the defined calculation order, leading to incorrect or missing error propagation. Calculations now always respect the configured sequence on initial runs and recalculations, ensuring dependent errors propagate correctly.
ARCHER-171184	Fixed an issue where the {=YEAR(DATEVALUE(GETTOKEN(LastRunTime)))} formula token in DB Query Transporter data feeds failed or caused faulted runs unless LastRunTime was capitalized and/or quoted. The expression now works as documented when enclosed in quotes.
ARCHER-171574	Formulas using the VALUEOF function may not be calculated if the dependency graph fails to recognize referenced fields, resulting in empty values. For example, VALUEOF([vl], "no") would fail because [vl] wasn't used outside the function. In contrast, IF(CONTAINS(any, [vl], VALUEOF([vl], "no")), "x", "y") would work because [vl] appeared outside of VALUEOF() in the CONTAINS() clause.
ARCHER-171627	Earlier, formulas without spaces between nested IF statements could cause single-value exceptions due to improper parsing of parameters. Formulas are now parsed correctly without requiring whitespace between comma-separated parameters, while any spaces that are present are ignored during execution.

Confidentiality Notice: This document is for the sole use of the Archer associates, partners, and clients and may contain confidential or proprietary information. Any unauthorized review, use, disclosure or distribution is prohibited.

ARCHER-171850	Previously, SLOPE() returned an error when only two data points were provided; it now computes results correctly with as few as two. Previously, ROUND(Function(...)) returned 0 when the wrapped function produced an error; it now surfaces those errors instead of masking them.
ARCHER-171866	Previously, field-level and record-level recalculations could produce inconsistent results when dependent calculated fields were in error. Now, recalculation consistently treats dependent calculated fields with errors as failures and correctly displays the dependent field error in all recalculation modes.
ARCHER-171915	Previously, the COUNT function did not work correctly when a referenced field had more than one record selected, returning incorrect values such as 0 or “Not Calculated.” Now, COUNT correctly returns the number of selected referenced records, even when multiple records are selected.
ARCHER-171933	Previously, several statistical and financial functions (such as CONFIDENCE, KURT, PROB, PEARSON, LOGNORMDIST, PERCENTRANK, VDB, and NPV) returned results that did not match the corresponding Excel functions. Now, these functions align their outputs with those of Excel functions, ensuring consistent, expected calculation results.
ARCHER-172018	Previously, the ATAN2 function was documented as supported but was not recognized or executed, resulting in formulas that used it failing validation. Now, the new engine fully supports and evaluates ATAN2 correctly in line with the documented behavior.
ARCHER-172091	Previously, the IF() function required three arguments but did not always error when parentheses were misused, allowing expressions like IF(1=1, (1+1, 2+2)) to be interpreted incorrectly instead of failing due to a missing comma. Now, the function properly validates argument structure and returns an error when only one result is parsed, while correctly evaluating well-formed expressions such as `IF(1=1+1, 2+2)
ARCHER-172395	Previously, IF statements using LEFT or RIGHT would return an error when the matched substring contained leading or trailing spaces (for example, a number followed by a space). Now, these functions correctly match substrings even when whitespace is present, allowing the IF expression to evaluate without errors.
ARCHER-172466	Previously, COUNTBLANK could error when counting referenced text fields containing special characters; now, it correctly handles those fields and no longer throws errors.
ARCHER-172470	Previously, GETUSERS and GETGROUPS would throw errors when used with REF and if the referenced field had no selected record (null); now, they correctly handle null references and return empty results instead of errors.
ARCHER-172472	Previously, COMBINESELECTIONS did not work with numeric fields referenced via REF because the engine prefixed numeric values (e.g., n123) and failed to match them to value list entries; now, it correctly handles numeric fields via REF and properly maps them to the corresponding value list entries.
ARCHER-172473	Previously, dependent field errors from REF applications were attributed only to the dependent field, making it difficult to identify the root cause; now, in the new engine, these errors provide a clear message indicating where the error originated, improving clarity for users.

Confidentiality Notice: This document is for the sole use of the Archer associates, partners, and clients and may contain confidential or proprietary information. Any unauthorized review, use, disclosure or distribution is prohibited.

ARCHER-172534	Previously, calculated value list fields could mis-evaluate selections when value names differed only by a “+” suffix (e.g., “90” vs. “90+”), causing comparisons such as IF([Values List Field] = VALUEOF([Values List Field], “90+”), "Yes", "No") to return "Yes" even when “90” was selected. Now, the calculation engine correctly distinguishes between these values, ensuring formulas return “Yes” only when “90+” is actually selected and “No” when “90” or no value is selected.
ARCHER-172539	The STRIPHTML() function, previously available only in data feed calculations, is now available for calculated application-level fields.
ARCHER-172541	Date calculation functions now support business day logic and holiday exclusions. Users can specify which weekdays and holidays to exclude from calculations.
ARCHER-172599	Previously, user attribute functions such as USERFIRSTNAME, USERMIDDLENAME, and USERLASTNAME accepted only 'user_type' inputs and could not process Users/Groups List or Record Permissions fields. Now, all user-attribute functions support fetching user attributes from the Users/Groups List and Record Permissions field types; however, if the input contains a group or multiple entries, the function correctly returns an error.
ARCHER-172600	Previously, the GETUSERS() function returned only a single User ID even when multiple users were selected, limiting its usefulness compared to GETGROUPS(). Now, GETUSERS() returns an array of User IDs, correctly outputting multiple selections when more than one user is selected.
ARCHER-172609	Previously, the >= and <= operators produced incorrect results when comparing strings. For example, expressions like IF(“10a” <= “20a”, “10”, “20”) or “A” <= “B” could return wrong results. Now, string comparisons using >= and <= behave correctly and consistently, ensuring predictable lexicographical comparison.
ARCHER-172664	The new USERSTATUS() function retrieves a user’s account status (Active, Inactive, or Locked) for the User/Group and Record Permissions fields. If the input contains a group or multiple entries, the function returns an error.
ARCHER-172687	Previously, ROUND did not consistently follow a single rounding rule and often behaved like “round half up” with floating-point inconsistencies; now, ROUND uses a clear and consistent rule - banker’s rounding (round half to even).
ARCHER-172688	Previously, value list matching treated trailing spaces as distinct in some parts of the platform, causing inconsistencies in data import and value list comparison; now, it ignores trailing spaces while consistently preserving leading spaces as valid characters across the platform.
ARCHER-172689	The calculation system now supports a new LOOKUPVALUEBYNUMBER function, which enables value lookup based on a specified numeric weight from a list of values.
ARCHER-172729	Support for the new IFS() function has been added to evaluate multiple conditions and return the value of the first TRUE condition, simplifying complex logic by replacing nested IF statements and making formulas more straightforward to read, write, and maintain.

Confidentiality Notice: This document is for the sole use of the Archer associates, partners, and clients and may contain confidential or proprietary information. Any unauthorized review, use, disclosure or distribution is prohibited.

ARCHER-172831	Previously, MIN() and MAX() could return 0 when all inputs were empty (For example, MIN(””, ““) or MIN(SELECTEDVALUENUMBER([vl])) when all selected values had empty numeric values); now, these functions consistently return an empty result when no values are associated with the input parameters.
ARCHER-173392	Previously, mapping a multi-select Values List field to a Text or Numeric field resulted in an error. Now, all selected values are concatenated into a comma-separated string when mapped to a Text field, and when mapped to a Numeric field, the first selected value is converted to a number and assigned where possible.
ARCHER-173405	Previously, COMBINESELECTIONS() could not be used in IF statements within Values List calculations, forcing customers to use workarounds to conditionally combine selections from related records. Now, COMBINESELECTIONS() can be used inside IF statements within Values List calculations, allowing customers to directly combine and conditionally filter selections based on related records without relying on workarounds.
ARCHER-173625	Error messages for cascaded calculation failures are now more informative, providing details about all related fields in the error chain to improve traceability and debugging.
ARCHER-174859	Previously, the COMBINESELECTION function returned only the first selected value, rather than all selected Value List options. Now, the COMBINESELECTIONS function correctly aggregates all referenced selections and outputs them as a comma-separated list within the text field where the formula is applied.
ARCHER-175182	Previously, the SUBSTRING function could throw an error when its returned value began with a space - for example, in nested formulas like IF(…, SUBSTRING(“Finding name 3”, 13)), where the expected result is “ 3”; now, SUBSTRING no longer errors in these cases and correctly returns the expected value.
ARCHER-175186	Previously, when an internal error occurred, the system could rethrow a previous field’s error message instead of generating a new, relevant one - for example, displaying a calculation error from Field1 as the error for Field2, making it difficult to identify the true source of the problem; now, each field reports its own correct error, ensuring accurate and clear information presentation.
ARCHER-175410	Previously, nested IF expressions comparing a Values List option that included a trailing space could return an incorrect result; now, the system correctly evaluates these conditions and returns the intended value.
ARCHER-175597	Previously, the calc engine exposes private field values to users who don’t have access, for example, showing “Private Content” in a public field after recalculation and return seemingly random results when text resembled grid references (for example, A10 or A11); now the calculation engine prevents exposure of private data and correctly returns the intended field values for these inputs.
ARCHER-175819	The calculation engine has been extended to support the logical functions TRUE() and FALSE() in standard calculated fields in App builder functions, previously limited to data feed token calculations.

Confidentiality Notice: This document is for the sole use of the Archer associates, partners, and clients and may contain confidential or proprietary information. Any unauthorized review, use, disclosure or distribution is prohibited.

ARCHER-175820	The calculation engine has been extended to support the text functions CHAR(), EXACT(), ISNONTEXT(), ISTEXT(), MID(), SUBSTITUTE(), and TEXT() in App builder functions, which were previously limited to data feed token calculations.
ARCHER-175821	The calculation engine now supports the date functions DATE(), DATEVALUE(), DAYS360(), SECOND(), TIME(), and TIMEVALUE() in App Builder functions. Previously, they were limited to calculations involving only data feed tokens.
ARCHER-175822	The calculation engine has been extended to support the financial function Dollar(), which was previously limited to data feed token calculations only.
ARCHER-175823	The calculation engine has been extended to support the system functions ISBLANK(), previously limited to data feed token calculations, and ISERROR().

Confidentiality Notice: This document is for the sole use of the Archer associates, partners, and clients and may contain confidential or proprietary information. Any unauthorized review, use, disclosure or distribution is prohibited.