This is a short preview of features for Release 73. For feature details, please consult the full release notes when they are published on Knowledge Central.

Ticket	**Affected Product Version**	**Description**
DEV-12434	Aria Billing Cloud	When a type 7 contract reaches its final period, Aria issues a prorated final invoice and cancels the plan instance. Previously, if a customer reversed that cancellation, Aria reset the bill anniversary to the next standard bill period instead of aligning it with the end of the prorated period. This ticket updates that behavior so the bill anniversary resets to the end of the prorated invoice period when you remove a type 7 contract cancellation. (DEV-12434)
DEV-12556	Aria Billing Cloud	Aria now resets the usage pooling counter when you void an invoice, keeping it aligned with the usage accumulation reset counter. This keeps usage-related counters consistent after a void. Previously, when you enabled usage accumulation together with usage pooling, voiding an invoice did not reset the usage pooling counter, even when you set the client parameter "Rollback usage records and the account's bill date for voided invoice" (DO_FULL_ROLLBACK_ON_INVOICE_VOID) to true and the usage accumulation reset counter did reset. (DEV-12556)
DEV-12627	Aria Billing Cloud	Aria now prorates invoices generated through the *update_acct_plan_multi_m* API when a contract end date falls within the active billing period, consistent with the *assign_acct_plan_m* API. This prevents your customers from being overbilled mid-contract. Previously, the *update_acct_plan_multi_m* API generated a full invoice instead of a prorated invoice in this scenario. (DEV-12627)
DEV-12680	Aria Billing Cloud	Aria improved the performance of the *update_acct_plan_multi_m* API. Previously, some calls to this API took significantly longer than expected to process. This update reduces processing time for the API call.
DEV-12714	Aria Billing Cloud	Aria now includes a configuration setting that lets you prevent voiding of a payment that has already been partially or fully refunded. This setting enforces the same restriction in the *void_transaction_m* API and updates the user interface to indicate when a payment is not eligible for voiding. Existing clients retain the previous behavior by default. This prevents your customer's account from being credited for both the refund and the voided payment. Previously, Aria allowed voiding of a payment even after it had been partially or fully refunded, which could result in your customer's account being credited twice.
DEV-12718	Aria Billing Cloud	Aria now applies dunning based on an invoice's current billing group and payment method, rather than the billing group and payment method in place when Aria created the invoice. This allows dunning to activate once the invoice

Ticket	**Affected Product Version**	**Description**
		becomes past due. Previously, if your customer changed the payment method or billing group on an existing invoice, that invoice could avoid dunning indefinitely, even after it became past due. (DEV-12718)
DEV-12430	Aria Billing Cloud	This feature allows a rebill invoice to reissue all original invoice lines with correct calculations, including discounts and service credits. It updates the rebill interface to accept decimal amounts for rebill unit values, an amount type it did not previously support. Together, these changes let you generate a rebill invoice that fully replicates the original invoice rather than a subset of its lines.
DEV-12656	Aria Billing Cloud	This feature delivers Phase 8 of Payment Plans: client-level configuration for payment plan terms and auto-cancel. It introduces the payment plan term, which lets you define a reusable payment plan configuration in your Aria client instances instead of setting values individually on each plan. Related Core API changes add the <client*payment_plan_term*id> and <payment*plan_term*no> arguments to the *create_payment_plan_m* API, and add a <recalculate_date> flag to the *update_payment_plan_m* API.
DEV-12573	Aria Billing Cloud	Aria now supports internal electronic invoicing (e-invoicing) in order to realize greater billing efficiency, cost savings, faster payments, and thorough compliance aligned with European e-invoicing standards. Once turned on for your account, Aria automatically emails both an XML file and a .pdf document based on your client configured email template. By default, only invoices that include Aria-managed tax lines are sent, but you can choose to include all invoices regardless of tax lines based on a client parameter setting. This includes the following: • A new EI Email template class for e-invoicing, along with buyer/seller, invoice, payment, and taxation-related replacement strings (including statements). • Dedicated “stubs” within the UI for the e-invoicing template class including sections for payment, tax, and invoice line details. • API support for the following: • e-invoicing email template selection • VAT Registration Number (ID for person or entity registered for the value-added tax in the EU) • Legal Entity email address (non-taxable for the VAT) • Government Invoice ID (Leitweg-ID) for German e-invoicing To enact this feature (along with setting the necessary client parameters), please contact your Aria representative.

Ticket	Affected Product Version	Description
DEV-12704	Aria Billing Cloud	Aria has enhanced its integration with the OneSource tax service to facilitate taxation for clients doing business primarily in Canada, Germany and India. Now, Aria sends OneSource customer names and ID numbers, delivery terms, product codes, and registration numbers for both buyers and sellers. This ensures tax calculation to the maximum extent possible, including honoring of tax exemptions and streamlined association of invoices with tax calculations.
DEV-12640	Aria Billing Cloud	The *get_dunning_processes_m* Admintools API now supports an optional <include_dunning_steps> input field. When set to true, this field returns full configuration for each dunning process step—step duration, notification template, email destination, fee type and amount, collection indicator, and step description—along with whether or not the process applies to all subscriptions, and the final suspension step's template. This update also corrects the <total_step_count> field description to correctly exclude the final suspension step.
DEV-12753	Aria Billing Cloud, CCP	An upgrade for Billie and Billie Connect (1.73.0) has updated the underlying language model to use Claude Sonnet 4.6 and add a retry mechanism for empty model responses. Dunning-related utterances now return dunning step and suspension details, viewable via the **Open Table** option which is grouped by dunning group and suspension date (the likely suspension date is calculated dynamically from the account's dunning configuration).
DEV-12617	Aria Billing Cloud	The *get_surcharge_details_m* Admin Tool API now returns full surcharge rate schedule details in a new <surcharge*rate_schedule*details> array—including schedule name, currency, default indicator, and ordered rate tiers with sequence number, unit range, rate per unit, and tier description—instead of only basic tier data. Existing locale and translation inputs are honored for schedule and tier names. This is an additive, backward-compatible change; the existing <surcharge_tier> field is unchanged.
DEV-12686	Aria Billing Cloud, CCP	Multiple REST API fixes and improvements have been completed including: • Restored case-insensitive sort ordering on several paged REST endpoints • A new distinct query parameter on multiple account transaction endpoints • A 1000-character limit on the E-Mandate Signature field • E-mandate signature and mandate date support added to Billing Group Update and Subscription Create • A fix allowing custom rates on usage-only plans • A new CCP "Queued Service Credits" tab for future-

Ticket	Affected Product Version	Description
		dated recurring credits • Sanitization of CCP content-preview HTML to close an XSS vulnerability • Search-overlay, notification-template, client-list, grid-refresh, and client-switch-popup fixes in CCP • Several REST API/Elasticsearch-sync corrections. No API contract changes accompany these corrections; customer-facing changes are individually backward-compatible.
DEV-12689	CCP	In the Customer Care Portal (CCP), you can now edit recipients before resending a statement or communication. You can add multiple **To** recipients plus **Cc** and **Bcc** recipients, including ad-hoc email addresses that are not already saved as account contacts. Aria validates addresses inline and flags invalid entries before the communication can be resent, and enforces recipient length limits (900 characters for To; 200 characters each for Cc and Bcc).
DEV-12740	Aria Billing Cloud, CCP	The php54 (aria-php54-frontend) container build and Helm chart have been relocated from the dev-devops repo into a new php54/ directory within aria-billing, mirroring the existing uiu/ layout so deployment artifacts live alongside the application code. The build pipeline was rewired to source from this new location; pipeline configuration and per-namespace Helm values remain in dev-devops. This is an internal build/deployment change only, with no application code, API, or behavioral impact.

