

From receipt to better decisions

Quality inspections need an owner

Emma: Welcome to Central Perspectives, a Business Central podcast from ITrators. I'm Emma, here with Alex. We're AI hosts, and our scripts and voices are AI-generated. Today, we're looking at what happens between receiving something and being able to make a useful decision about it: goods that need inspecting, purchase emissions that need calculating, and replenishment that needs the right planning rules. There are some genuinely useful improvements in Business Central 29, an earlier feature worth discovering, and a preview worth keeping an eye on. Now, let's get started.

Emma: Quality management first. Business Central can organize inspections of incoming goods and materials, so the checks and their results belong to the business process rather than a separate conversation. The base extension became generally available in April 2026. What's new in version 29 is a set of improvements around doing that work: taking ownership of an inspection, finishing it, and finding the inspection behind a blocked transaction. Alex, what does that give the person receiving the goods?

[Evaluate the quality of goods and materials](#)

[Feature details for Business Central 2026 release wave 2 public preview](#)

[Update 29.0 for Business Central 2026 release wave 2](#)

Alex: A more direct route from a delivery to the decision about whether it meets the required checks. You define what to inspect through templates and rules, then somebody actually performs the inspection and records the result. That's useful because receiving a quantity and accepting its quality are different decisions. Imagine a company receiving components with a dimension that matters when they're assembled. Counting the boxes tells you what arrived. The inspection is where you establish whether those components meet the agreed test. The new improvements make that inspection easier to pick up and complete.

[Work with quality inspections](#)

[Quality management setup and configuration](#)

[Feature details for Business Central 2026 release wave 2 public preview](#)

[Update 29.0 for Business Central 2026 release wave 2](#)

Emma: Taking ownership sounds rather modest next to quality management, but I can see the appeal. A delivery is waiting, two people can inspect it, and neither should have to guess whether the other has started. The new self-assignment prompt seems aimed at exactly that handover. I'd much rather make responsibility visible while someone is opening the work than add another message asking who has it. Does assigning the inspection also decide the result?

[Feature details for Business Central 2026 release wave 2 public preview](#)

[Update 29.0 for Business Central 2026 release wave 2](#)

Alex: No, it identifies who's taking the work, not whether the components pass. That distinction is helpful for your two inspectors. One can take responsibility without the company treating the delivery as accepted. The useful next question is what that person needs to record. For our components, you'd want the relevant test and its expected result to be clear before measuring anything. Otherwise you've tidied the queue while leaving the actual quality decision vague. A well-defined inspection gives the person a job they can finish, rather than a general instruction to have a look.

[Work with quality inspections](#)

[Feature details for Business Central 2026 release wave 2 public preview](#)

[Update 29.0 for Business Central 2026 release wave 2](#)

Emma: Then I'd start with that component's actual acceptance criteria, not a grand project to inspect everything that enters the building. But how does the right inspection appear? If the receiving employee posts the delivery, is that enough to get our dimensional check into somebody's work?

[Work with quality inspections](#)

[Quality management setup and configuration](#)

Alex: Posting permission on its own isn't enough. The quality setup needs an inspection template, an enabled generation rule that applies, and the effective integration permission for that process. Those pieces connect the transaction to the inspection. So your narrow starting point is sensible: configure the intended component and receipt situation, then check that receiving it produces the inspection you expect. If nothing appears, don't immediately conclude that the feature failed. First establish whether that transaction actually met the rule and whether the integration was allowed to run.

[Work with quality inspections](#)

[Quality management setup and configuration](#)

Emma: There could be a general rule for incoming components and a more specific one for this particular component. I'd expect the specific one to win, because that's the one whose test we deliberately designed. Is that a safe expectation, or is the setup deciding something else?

[Work with quality inspections](#)

[Quality management setup and configuration](#)

Alex: The first matching generation rule is selected by sort order. It doesn't infer which rule you consider most important. Your specific test needs to be positioned accordingly. That makes a good configuration check: take the same component and ask which rules match it, then confirm which one is selected first. A broad rule can be perfectly valid and still be the wrong first match. The benefit of noticing this early is that you can keep useful general rules without accidentally losing the more precise inspection you wanted for certain goods.

[Work with quality inspections](#)

[Quality management setup and configuration](#)

Emma: That's a useful reason to try a specific delivery, rather than admire an impressive list of rules. Let's give our inspector the correct tests now. Another version 29 improvement is that finishing the inspection calculates passed and failed quantities automatically. That sounds like less arithmetic after the actual work. What quantity is it calculating against if we inspected a sample rather than every component?

[Feature details for Business Central 2026 release wave 2 public preview](#)

[Update 29.0 for Business Central 2026 release wave 2](#)

Alex: The documented basis is the sample size when that is positive; otherwise it uses the base quantity. That matters when you read the result. Suppose the delivery is a hundred components and the inspection concerns a sample of ten. You shouldn't read a quantity on the inspection as though a person individually measured all hundred. Automatic calculation removes a manual step, but the inspection still means what its sampling and test design mean. The company needs to decide what that evidence permits it to do with the delivery.

[Work with quality inspections](#)

Emma: And the person waiting to use the components may not be the inspector. That's where I like the other improvement: a blocked transaction can lead directly to the inspection responsible for the block. A message that says no is useful; a message that helps you find the unresolved work is considerably more useful. It gives the next person somewhere sensible to go without turning them into a detective.

[Feature details for Business Central 2026 release wave 2 public preview](#)

[Update 29.0 for Business Central 2026 release wave 2](#)

Alex: And it preserves the purpose of the block. You want the person to resolve the inspection, not search for a way around a mysterious error. Combined with visible responsibility, that navigation can shorten the conversation to: this is the inspection, and this is who is handling it. Assisted destination-bin selection is another version 29 improvement in the same practical direction: helping the inspector finish the handling work, not just collect a result.

[Feature details for Business Central 2026 release wave 2 public preview](#)

[Update 29.0 for Business Central 2026 release wave 2](#)

Emma: What if the delivery doesn't pass first time, but the supplier sorts out the problem? I'd want the second inspection connected to the first, so we can follow the story. I would not want yesterday's measurements quietly becoming today's evidence.

[Work with quality inspections](#)

Alex: Reinspection rebuilds the test lines from the current template without copying the entered test values. So the new inspection needs new results. There is another consequence in that word current, though. If the template has changed since the first inspection, you're not necessarily repeating exactly the same set of tests.

[Work with quality inspections](#)

Emma: That could be an improvement. Perhaps the first failure showed that our checks were too vague, and we made them more precise. But then a pass after reinspection isn't simply the same exercise with a happier answer. We'd need to understand what changed before comparing the two results.

[Work with quality inspections](#)

Alex: Exactly that comparison: what was tested, against which criteria, and what result was recorded each time. That gives the company something meaningful to review rather than two status labels. For a first use, I'd choose one receiving process with a clear quality question and follow it through both a successful inspection and a failed one that needs reinspection. The extension has to be installed and configured, and inspectors need the appropriate permissions. Manufacturing-related quality scenarios also bring Premium licensing into the picture. Those are setup boundaries, not reasons to overlook a useful incoming-goods workflow.

[Work with quality inspections](#)

[Quality management setup and configuration](#)

Emma: My takeaway is that these are improvements to the work around the measurement, and that work matters. Somebody takes it, the result has a quantity behind it, and a colleague can find out why they're waiting. It doesn't make a passed inspection a certificate of legal compliance. It does make a defined quality process easier to follow. And we should keep the dates straight: an April foundation, with useful improvements in version 29.

[Evaluate the quality of goods and materials](#)

[Feature details for Business Central 2026 release wave 2 public preview](#)

[Update 29.0 for Business Central 2026 release wave 2](#)

What a purchase emissions formula calculates

Alex: There's another receipt-related improvement where the numbers deserve a closer look: emissions calculations on purchases. A formula can turn configured inputs and factors into emissions on a purchase document, rather than asking someone to enter the resulting emissions manually each time. That's a useful way to make a repeated calculation consistent. The question I'd bring to it is what the total represents. Are we calculating for this whole delivery, for a quantity of material, or using some other defined input? That meaning needs to survive ordinary changes to the order.

[Update 29.0 for Business Central 2026 release wave 2](#)

[Record sustainability entries](#)

Emma: And to put a date on that, purchase-document formulas are listed as generally available in Business Central 29. Sustainability journals and purchase emissions themselves aren't new with this release. The formulas apply to purchase orders, invoices, credit memos and returns. I like the prospect of keeping the calculation near the purchase, because an order is where somebody will notice that the delivery has changed. Let's say we order a hundred units, then increase it to two hundred. I would expect the emissions to increase too.

[Update 29.0 for Business Central 2026 release wave 2](#)

[Record sustainability entries](#)

Alex: That's the expectation to check. Changing the purchase quantity changes the per-unit emissions amount, not the formula's total. If your formula total was a hundred kilograms of emissions, changing the order from a hundred units to two hundred doesn't, by itself, turn that total into two hundred kilograms. It spreads the hundred across a different purchase quantity.

[Record sustainability entries](#)

Emma: So the per-unit figure gets smaller in that example. That might look like an improvement if you only looked at that one number, although nobody has changed how anything was produced or transported. We changed the order. I'd want the buyer to recognize that before using the figure to compare two purchases.

[Record sustainability entries](#)

Alex: Which brings us back to what the formula describes. If the original inputs describe a delivery whose emissions remain the same despite the extra units, that unchanged total could be appropriate. If doubling the units also doubles the material or activity that the formula is meant to capture, those inputs need attention. The software's calculation behavior is definite; which inputs represent the real purchase is the business question. Your comparison needs both a meaningful total and a meaningful quantity, not simply a reassuringly low amount per unit.

[Record sustainability entries](#)

Emma: That's still useful automation. We're asking the buyer to notice a changed assumption, not to redo every calculation with a calculator. But somebody has to make the formula understandable enough that the next person can tell when an input has become stale. What has to be in place before the purchase line can use it?

[Record sustainability entries](#)

Alex: Purchase emissions and formula use have to be enabled, with the relevant sustainability account, categories and factors configured. The factors are particularly important: the system needs a basis for turning an activity or input into an emissions amount. A consistent formula is valuable because the same defined calculation can be repeated, and its inputs can be inspected. But it doesn't independently measure what happened at the supplier. If the factor is unsuitable for this purchase, applying it very consistently won't make the result more representative.

[Record sustainability entries](#)

Emma: There's a small detail in the documentation that I'd put beside the formula, not bury in a footnote: changing its unit doesn't convert or recalculate it. That's easy to miss when a screen lets you choose a different unit. You might reasonably think you've asked for the same calculation expressed differently, rather than changed a label that now needs matching inputs.

[Record sustainability entries](#)

Alex: Yes. Think of a quantity entered in kilograms and a factor defined per tonne. Those quantities need to agree before the multiplication means anything. Selecting a different unit isn't a substitute for checking that agreement. I'd make one worked example part of the setup explanation: these are the inputs, these are their units, this is the expected total. Then the person maintaining the formula has something concrete to compare against after a change.

[Record sustainability entries](#)

Emma: Let's take our hundred units again, keep the inputs unchanged, and say the purchase carries a hundred kilograms of emissions in total. Only forty units are posted now; the rest will follow. I'd want the emissions attached to that partial posting to represent forty units, not the whole order. Does the calculation handle that split?

[Record sustainability entries](#)

Alex: Partial posting apportions the emissions. With those simple numbers and unchanged inputs, forty out of a hundred units carries forty out of the hundred kilograms. That's a good reason to follow an example beyond entering the formula. You want to see the total, the per-unit amount and the posted portion tell a consistent story. Then, if the order quantity changes between deliveries, you have a clear reason to inspect the calculation again instead of assuming every later posting follows the original example unchanged.

[Record sustainability entries](#)

Emma: And if a supplier already gives us an emissions amount for a particular purchase, we shouldn't automatically invent a formula just because there is now somewhere to put one. The documentation treats manual emissions and formula inputs as alternative paths. That seems a sensible choice to make deliberately: are we recording an amount we already have, or calculating one from defined inputs?

[Record sustainability entries](#)

Alex: I'd make that choice based on the information available and how repeatable the purchase is. A formula earns its place when the inputs and factors are understood and the same calculation will be useful again. A supplied amount still needs a clear basis, but it doesn't become better merely by being reconstructed through a formula. Also, this purchase-document formula capability doesn't support Water or Waste calculations. We're talking about the supported emissions calculation here, not a general calculator for every sustainability measure the company might track.

[Record sustainability entries](#)

Emma: I started by expecting the total to double when the order doubled. Now I'd first ask what the formula's inputs actually describe. That's a better conversation for the buyer and whoever owns the sustainability data: make the calculation visible, make its basis understandable, and notice when the purchase no longer matches it. The benefit is less repeated arithmetic with a clearer trail behind the number. It isn't independent certification of the number. It can be useful without pretending to provide that.

[Record sustainability entries](#)

Planning when the location has no SKU

Emma: Here's an earlier feature that deserves a look if you plan inventory across locations. Since the April 2026 release, you can choose how planning should behave at a location when an item has no stockkeeping unit there. That's not October news, but it answers a very ordinary question: should the system use the item's general settings, take a minimal approach, or leave that location out of planning? Before we choose, Alex, explain what a stockkeeping unit adds in Business Central.

[About planning functionality](#)

[Choose planning behavior for items without SKUs at a location](#)

Alex: It lets the planning parameters be specific to the item at a location, rather than relying only on the general item card. People often shorten stockkeeping unit to SKU. If an existing SKU applies, planning uses its parameters. The feature we're discussing is the fallback when that record isn't there. That's useful because a missing record doesn't always mean the same thing. At one location, the general item settings might be exactly what you want. At another, you might deliberately want no planning suggestions at all.

[About planning functionality](#)

Emma: I'd be tempted to use the item card as the fallback. If the company already decided how it wants to replenish that item, using those settings seems more useful than making somebody maintain another record just to repeat them. Suppose we add a second ordinary stock location and use the same buying pattern. That feels like a good place for the Item Card option.

[About planning functionality](#)

Alex: It can be. Item Card creates a temporary SKU for the planning calculation using the item parameters, and the fallback is logged. You're not obliged to create a permanent duplicate simply to get that behavior. But before applying it to every location, what if your second location holds goods awaiting inspection rather than stock you normally replenish? Would the same buying pattern still describe what should happen there?

[About planning functionality](#)

Emma: No, then the location's purpose is different. I wouldn't want a general replenishment setting to make that holding location look like another place we should keep topping up. My shortcut only works if the item card really expresses the policy for that location. That's where Don't Plan becomes a deliberate choice rather than an admission that nobody has finished the setup.

[About planning functionality](#)

Alex: Yes, for the items without SKUs covered by that fallback. Don't Plan skips planning in that situation. It doesn't mean the goods vanish, and it isn't a statement that every item at the location must be excluded regardless of its setup. Existing SKUs still supply their own parameters. So you can distinguish an intentional fallback from a particular item that needs location-specific planning. That is more expressive than treating every missing SKU as the same kind of omission.

[About planning functionality](#)

Emma: We have two understandable choices now. Use the general item settings where they're appropriate, or deliberately skip that fallback planning. The third option is called Minimal. That name sounds reassuringly light, but it doesn't tell me what will happen. Is it a smaller version of Item Card?

[About planning functionality](#)

Alex: Not quite. Minimal preserves the earlier planning behavior. In the ordinary fallback case it uses Lot-for-Lot, includes inventory, and leaves the other planning parameters empty. In plain terms, that's different from copying your chosen item-card settings wholesale. There are exceptions where item parameters still apply, involving Components at Location and the Order reordering policy. So compare the two options against a known demand situation in a test setup. Keep demand and supply unchanged, and explain the different proposals. You'll learn whether the general item settings actually improve planning for that location, rather than choosing by the option's name.

[About planning functionality](#)

Emma: I'd include an item that already has a SKU in that comparison too. Otherwise we could see a difference for one item and tell everybody that the location now uses the item card. Your point is that the new choice answers what happens when the SKU is missing. It doesn't replace the planning decisions we've already made in an existing SKU.

[About planning functionality](#)

Alex: That existing SKU is a useful control case. It helps show whether you're examining the fallback at all. For the missing-SKU case, the log from Item Card can help explain why item parameters were used. Then the planner has a traceable answer to a proposal that otherwise might look surprising. The value isn't just fewer records to maintain; it's making the source of the planning policy explicit. When somebody asks why the system suggested that quantity here, you have a better starting point than guessing which settings it found.

[About planning functionality](#)

Emma: You called the SKU temporary. If a company blocks creation of permanent stockkeeping units, does that stop this fallback too? I'd hate to recommend a maintenance shortcut and then discover it depends on creating the very records the company has chosen not to create.

[About planning functionality](#)

Alex: Blocking creation of persisted SKUs doesn't block the temporary ones used for this planning behavior. The temporary record is part of how the calculation works; it isn't the same as saving a new permanent SKU for ongoing maintenance. So that control doesn't automatically prevent Item Card fallback. It does make the distinction worth explaining to the people responsible for master data, because they may reasonably hear creates a SKU and assume their record-creation policy has been ignored.

[About planning functionality](#)

Emma: There is one manufacturing detail I'd keep in the conversation rather than leave entirely in the notes. Don't Plan overrides Components at Location in this missing-SKU situation. So someone can't rely on that other setting to quietly bring the item back into planning. If they've chosen to skip it, that choice has a real consequence.

[About planning functionality](#)

Alex: And that's why I would review the purpose of each affected location before changing the fallback. For an ordinary stocking location, using the item card may remove unnecessary duplication. For a location deliberately outside replenishment, Don't Plan can express that intent. Where location-specific parameters matter, an actual SKU still has a job. None of those is the universally advanced option. The good outcome is that the planner can explain the proposal, or its absence, in terms of the company's intended policy. A location with no SKU should no longer be an unexplained gap in that conversation.

[About planning functionality](#)

Emma: That makes this a useful discovery rather than a recommendation to rebuild everyone's planning setup. My second ordinary stock location could be a good candidate for Item Card; the inspection holding location needs a different conversation. I'd take the feature to the planner who currently has to explain inconsistent-looking suggestions, not introduce it as another master-data exercise. It's been available since April, and the practical question is simply whether this fallback now lets the setup say what the business actually means.

[About planning functionality](#)

[Choose planning behavior for items without SKUs at a location](#)

EDI exchange enters public preview

Emma: Two shorter items before we finish. First, electronic document exchange. Business Central 29 lists EDI document exchange as a public preview, extending the existing E-Documents framework beyond the familiar invoice conversation into order exchange and sending remittance advices. That could matter to companies still passing orders between systems by hand. It's a preview, not generally available functionality, and we don't have a verified general-availability date to give you. Alex, what's the useful opportunity to investigate here?

[Feature details for Business Central 2026 release wave 2 public preview](#)

[Update 29.0 for Business Central 2026 release wave 2](#)

Alex: Whether a specific exchange with a trading partner can become structured rather than re-entered. EDI is electronic data interchange: the document's information travels in a form the receiving system can work with. That can remove a lot of typing, but both sides still need to agree on the document and its meaning. The documented direction uses Peppol BIS3 and allows extension for local or customer formats. That is a useful starting point, not a promise that every existing EDI connection will work without adaptation.

[Feature details for Business Central 2026 release wave 2 public preview](#)

[Connect E-Documents to external access points](#)

Emma: I'd judge that by what happens at the other end. Sending an order successfully is less interesting than the supplier receiving the right item, quantity and reference without somebody repairing it. A document can travel beautifully and still leave a person with the same job. So the sensible preview exercise seems to be one real type of exchange with one willing trading partner, rather than declaring that the company now does EDI.

[Feature details for Business Central 2026 release wave 2 public preview](#)

Alex: And agree in advance what receiving it correctly means. For that pilot, the E-Document core setup comes before configuring the external connector. An access-point service can also involve its own contract, credentials and charges, so this isn't necessarily a feature you enable with no other arrangement. Your one-partner approach makes those dependencies concrete. You can investigate the intended document flow and the work it removes, rather than connecting a service first and searching for a useful transaction afterward.

[Connect E-Documents to external access points](#)

Emma: That earns a place on the watch list, especially if order re-entry is a recurring job today. The interesting direction is a broader document framework, not just another way to send an invoice. We'll keep the preview label attached and avoid guessing a launch date. A company can investigate its trading-partner fit now without pretending that preview support proves an entire production integration.

[Feature details for Business Central 2026 release wave 2 public preview](#)

[Update 29.0 for Business Central 2026 release wave 2](#)

Read the actual AL test result

Emma: Finally, something for the people building and maintaining extensions. Business Central 29 brings command-line test execution through the AL tool: the command is `al runtests`. It can execute test codeunits on a configured server without needing an MCP connection. That sounds particularly useful for automated build and validation jobs. What's the important distinction when a pipeline reports that the command ran successfully?

[Update 29.0 for Business Central 2026 release wave 2](#)

[Develop AL extensions with ALTool](#)

Alex: Successful invocation and successful tests are different things. The tool can reach the target and execute a request while the actual tests report failures. You need to inspect the test success information and the individual method results, not just whether you received a response. The command's exit code is also intended for automation: zero means no failing tests, otherwise one. That makes it possible for a build job to react to the test outcome, rather than treating the act of starting a test run as the achievement.

[Develop AL extensions with ALTool](#)

Emma: No failing tests could still be less reassuring than it sounds if the tests we meant to run never ran. I'd want the result to say how many were executed and how many were skipped. Otherwise a green result can answer a narrower question than the team thinks it asked.

[Develop AL extensions with ALTool](#)

Alex: Those counts belong beside the result. The target environment and company must be configured, authentication must work, and the intended test code units and their dependencies must already be installed. The command doesn't manufacture that test environment for you. A useful first check is to run a known small set and verify that the named methods appear in the output. Then deliberately include a known failing test in a controlled check and make sure the pipeline reacts correctly. You're establishing that the connection from execution to the build decision actually works.

[Develop AL extensions with ALTool](#)

Emma: That's a practical improvement without reopening the whole debate about how many tests a project should have. The news is a more direct execution route. A team with useful tests can make their results part of the automated job, provided the job checks what actually happened. And not needing MCP here means we shouldn't introduce an extra integration layer just to run this command.

[Develop AL extensions with ALTool](#)

Alex: Use the direct route when it meets the need. The tool also supports test groups sharing a connection, which can be relevant when organizing a run. The command options belong in the technical notes; the main point is to retain the actual results with the job that used them. We've described the documented capability here, not executed tests in a listener's environment. That first controlled run is where the team can establish its own evidence.

[Develop AL extensions with ALTool](#)

Emma: That's it for this Central Perspectives. The useful thread today was making a decision easier to explain: who is handling an inspection, what an emissions number represents, and which planning policy produced a suggestion. Some improvements save typing; others save the conversation needed to understand the result. Both can matter. You'll find the direct sources, release qualifications and technical detail in the episode notes. Thanks for listening to Central Perspectives from ITrators. Until next time.