SAP
C_S4CPB_2608
SAP Certified - Implementation Consultant - SAP S/4HANA Cloud Public Edition

Latest Version: 4.0

1. Unified Scenario Simulation
2. Micro Skill Drill Exam

Topic: 1
Unified Scenario Simulation
Ravensmoor Travelware Rolls Out a Configuration Change
Business Context
Ravensmoor Travelware designs and assembles travel luggage and backpacks in Amman, Jordan, supplying regional retailers and a growing online storefront. Orders arrive through both channels and flow through a single order-to-cash process: a sales document is created, stock is confirmed from the assembly warehouse, and the goods are shipped and billed. A seasonal surge each spring, when travel demand rises, pushes far more orders through that process in a few weeks than the company handles in an ordinary month.
System Landscape
The company has run SAP S/4HANA Cloud Public Edition for several months, using the standard separated landscape of a Development system, a Test system, and a Production system. Business configuration is meant to originate centrally and move through those systems in sequence, and the cloud service receives scheduled upgrades on a continuous-delivery cadence. A small internal team, supported by an implementation partner, owns configuration and keeps the landscape aligned with the standard processes the company adopted at go-live.
Change Context
Ahead of the spring surge, the sales team asked for a small adjustment to how one sales document behaves, so that a particular order category is handled the way the retail channel now expects. A consultant who joined the team only recently picked up the request. Wanting to show progress quickly, the consultant applied the adjustment in one of the downstream systems and demonstrated the new behavior to the sales team in a short working session.
Process Flow Context
In that session the adjusted document behaved exactly as the sales team wanted, and the request was reported as done. When operations later processed live orders in the Production system, though, those orders continued to follow the original behavior. The category the sales team cared about was still handled the old way for real customers, even though the demonstration a few days earlier had shown the change working. To keep those orders moving, operations began adjusting the affected records by hand after they were billed, quietly absorbing the extra effort while volumes were still light.
Operational Context
As the team looked more closely, the picture varied from system to system. The new behavior was visible where the consultant had entered it, absent in Production, and after a routine movement of other configuration through the landscape the two systems no longer matched at all. Some team members saw the adjusted behavior in their environment while others did not, and it became harder to say which system reflected the setting the business had actually approved.
Governance Context
The platform lead has asked that every configuration change begin in the central configuration environment and travel through the landscape from there, rather than being typed into whichever system is closest to hand. The sales team, with the surge approaching, would prefer the change simply be entered straight into Production now so the live orders are handled correctly this season. A suggestion has been floated to re-key the same setting by hand in each system to make them agree.

Readiness Context

With only weeks before order volumes climb, the team needs to settle where this change is authored, how it is checked before it reaches live customers, and how the three systems are kept identical once it is in place. The same document setting will be touched again at future upgrade cycles, so however the team resolves it now will shape how routine changes are handled well beyond this one request.

Your assignment

You are a member of the configuration team at Ravensmoor Travelware, working in a live SAP S/4HANA Cloud Public Edition landscape made up of a Development, a Test, and a Production system. A requested adjustment to how one sales document behaves has been applied in a downstream system but does not appear consistently across the landscape, and the spring order surge is approaching. Treat each instruction as a practical configuration and validation task. Do not assume any fixed technical identifier, screen label, transaction code, or parameter default beyond what the environment confirms; reason from where a change is authored, how it is validated, and how it is kept consistent

1. CHALLENGE 1 — Authoring the Configuration Change in the Correct System

• Establish that a business configuration change in this landscape is authored in the central configuration environment and then moved through Development, Test, and Production in sequence.

• Distinguish a change that can travel through the landscape from a value typed directly into a downstream system, which creates nothing to move onward.

• Weigh a proposal to edit the Production system directly for speed against the governed authoring path the landscape depends on.

Checkpoints

• The setting is created centrally and can then be observed in the target system after it is moved, rather than existing only where it was first typed.

• The chosen authoring location is justified by whether the change reaches Production consistently, not by which system was quickest to open.

2. CHALLENGE 2 — Validating the Change in the Test System First

• Plan a structured check of the configured behavior in the Test system, exercising the affected process before the change is transported to Production.

• Separate a genuine validation in the Test system from proving the behavior on live customer orders in Production during the surge.

• Confirm that the change is signed off against the configured process, not accepted on the strength of a single earlier demonstration.

Checkpoints

• The configured behavior is validated in the Test system and approved before any transport to Production.

• Live customer orders are not used as the proving ground for an untested configuration change.

3. CHALLENGE 3 — Keeping the Landscape Consistent Through Transport

• Trace the behavior that appears in one system but not another back to how the change was applied, before adjusting any downstream setting.

• Move the single governed change through the landscape so that Development, Test, and Production each receive an identical copy from one source.

• Resolve the request for a quick local correction against the need for one governed transport path across the landscape.

Checkpoints

• The same transported setting is confirmed present and identical across all three systems.

• Configuration is not left divergent through separate manual entry in individual systems.

Question: 1

CHALLENGE 1 — Authoring the Configuration Change in the Correct System

The requested adjustment to the sales document has to take effect across the live landscape before the spring surge begins.
Where should this configuration change be created so that it reaches every system?

A. Enter the adjustment directly in the Production system, so the live behavior updates at once for the peak-season orders that need it.
B. Enter the adjustment directly in the Test system, since that is where the earlier demonstration already showed the new behavior working.
C. Author the adjustment in the central configuration environment and move it through the landscape, so each system receives the same governed change.
D. Enter the adjustment by hand in each of the systems in turn, so that Development, Test, and Production all separately reflect the new behavior.

Answer: C

Explanation:
In SAP S/4HANA Cloud Public Edition business configuration is authored centrally and then transported through Development, Test, and Production, so a single governed change reaches every system consistently. Authoring it in the central environment is what makes the change travel rather than stay stuck in one place.

Question: 2

CHALLENGE 1 — Authoring the Configuration Change in the Correct System

The consultant entered the change in a downstream system and it behaved correctly in a demonstration there, yet the Production system still follows the original behavior.
What is the most likely reason Production did not pick up the change?

A. The change was never authored in the central configuration environment, so nothing was created that could be transported into the Production system.
B. The Production system runs on an older release than Test, so it is unable to display a configuration setting that a newer release introduced.
C. The change depends on an overnight background job, so the Production system will reflect the new behavior automatically after the next scheduled cycle.
D. The user working in Production lacks the business role for the new behavior, so the setting is present but stays hidden from that particular user.

Answer: A

Explanation:

Because the setting was typed directly into a downstream system rather than authored centrally, there was no governed change to transport onward, so Production never received it. The behavior existed only where it was entered, which is why the live system kept the original handling.

Question: 3

CHALLENGE 1 — Authoring the Configuration Change in the Correct System

The sales team wants the change live quickly and proposes editing the Production system directly, but the same document setting will be revisited at each upgrade cycle.
How should the team weigh this proposal?

A. Edit the Production system directly just this once, because a single quick change carries very little risk and comfortably meets the tight peak-season deadline.
B. Edit the Production system directly but keep a written note, so the team can re-apply the change by hand if a later upgrade removes it.
C. Edit the Test system now for another demonstration and the Production system later, so each system is handled at the moment it is actually needed.
D. Author the change in the central configuration environment and transport it, so the change survives upgrades and stays consistent across the landscape.

Answer: D

Explanation:
Authoring centrally and transporting keeps the setting governed and consistent, and because it lives on the standard path it is carried through upgrade cycles instead of being at risk each time. This meets the deadline through the supported route rather than trading durability for speed.

Question: 4

CHALLENGE 2 — Validating the Change in the Test System First

The change now needs to be confirmed before it reaches customers, and the landscape provides a dedicated Test system for that purpose.
Which validation step fits this landscape before the change moves to Production?

A. Confirm the behavior in the Production system with a handful of live orders, since real transactions are the most convincing proof that it works.
B. Test the configured behavior in the Test system and obtain sign-off there before the change is transported onward to the Production system.
C. Confirm the behavior in the Development system alone, since that is where the change is authored and first becomes visible to the team.
D. Skip a formal test because the earlier demonstration already showed the new behavior working once in a downstream system.

Answer: B

Explanation:
The Test system exists to validate configured behavior against the affected process before it reaches live customers, so testing there and signing off is the fit. It confirms the change works as intended while any defect stays away from real orders.

Question: 5

CHALLENGE 2 — Validating the Change in the Test System First

Under deadline pressure, the team considers proving the new behavior by watching live customer orders in Production during the surge.
Why is validating on live Production orders the wrong path here?

A. Live customer orders complete faster than prepared test data, so the validation would finish before the team is able to review the results properly.
B. The Production system is unable to record test outcomes, so any defect found while checking there would stay invisible to the team reviewing the change.
C. Checking against live orders exposes real customers to any defect in the change, whereas the Test system lets the same behavior be validated safely first.
D. The Production system blocks all configuration testing entirely by design, so the attempt itself would be rejected automatically before any customer order was processed.

Answer: C

Explanation:
Validating on live orders means real customers absorb any defect before it is caught, while the Test system is built to exercise the same behavior without that exposure. Confirming in Test first keeps the surge weeks safe and only then moves a proven change to Production.

Question: 6

CHALLENGE 2 — Validating the Change in the Test System First

Before the change is transported to Production, the team wants a clear signal that it is genuinely ready.
Which outcome confirms the change is ready to move to Production?

A. The configured behavior is validated in the Test system and signed off, showing the process works as intended before the change is transported onward.
B. The change is already visible in the Development system, which shows it was authored and therefore needs no further checking at all before it reaches Production.
C. A single order in the Production system completes without an error message, which shows the intended new behavior must already be active there.
D. The business approves the plan in a review meeting, which confirms the change is ready to go regardless of any testing in the systems.

Answer: A

Explanation:
Readiness is demonstrated when the configured behavior is exercised and signed off in the Test system, because that proves the process works before it reaches live customers. That validated sign-off is the proper trigger for transporting the change to Production.

Question: 7

CHALLENGE 3 — Keeping the Landscape Consistent Through Transport

Team members report that the adjusted behavior appears in one system but is missing in another across the landscape.
What is the underlying cause of the systems disagreeing?

A. The systems run on different releases, so the newer behavior only appears in whichever system happens to have the latest release installed.
B. The Fiori launchpad tiles differ between the individual systems, so the same configured behavior is really present but is simply displayed under a different app there.
C. The Production system was refreshed from a copy recently, so the change was in fact present everywhere but was removed only during that refresh.
D. The change was re-keyed separately in one system instead of being transported, so the systems now hold divergent configuration from different sources.

Answer: D

Explanation:
Because the setting was applied separately in one system rather than transported from a single source, the systems ended up holding different configuration, which is the second-order cause behind the visible symptom. The behavior appears where it was typed and is absent where the governed change never arrived.

Question: 8

CHALLENGE 3 — Keeping the Landscape Consistent Through Transport

The team wants Development, Test, and Production to hold the same configuration again after the change drifted apart.
How should the team restore consistency across the three systems?

A. Re-enter the change by hand in whichever system is currently missing it, so that each system quickly comes to match the others again.
B. Transport the single governed change through the landscape, so that every system receives an identical copy of it from one authored source.
C. Transport the current Production configuration backward into Test and Development, so the systems align on whatever Production currently happens to hold.

D. Leave each system as it was configured locally, since users can adapt to whichever behavior their own system shows them day to day.

Answer: B

Explanation:
Moving one governed change through the landscape gives every system an identical copy from a single authored source, which is what keeps the three systems consistent. It replaces divergent manual entries with the supported transport path.

Question: 9

CHALLENGE 3 — Keeping the Landscape Consistent Through Transport

The sales team asks for a quick local correction in Production right now, while the platform team insists on a single governed transport path across the landscape.
How should this be resolved?

A. Route the correction through the governed transport path, so that a single authored change reaches every system consistently and predictably.
B. Apply the quick local correction in Production now and transport a matching change later, so that both the sales and platform goals are partly met.
C. Apply the quick local correction and treat Production as the reference, so that the other systems can be aligned back to it at a later point.
D. Let each team proceed in its own way, applying the local correction now while the governed transport path is prepared separately in parallel.

Answer: A

Explanation:
Routing the correction through the governed transport path delivers one authored change to every system consistently, which serves the shared goal of a trustworthy configuration before the surge. It resolves the tension by meeting the real need — a correct, consistent setting — through the supported route.
Topic: 2
Micro Skill Drill Exam

Question: 10

Nordfalt Interiors, a mid-sized furniture retailer headquartered in Helsinki, is running a new-implementation project for SAP S/4HANA Cloud Public Edition through GROW with SAP. A consultant is facilitating a fit-to-standard workshop for the order-to-cash process in the Starter system, working from the responses that the finance and sales teams provided in the Business Driven Configuration Questionnaire. Most of the sales flow maps cleanly to the standard best-practice process shown in the workshop. However, the sales manager insists on one step carried over from the company's legacy system: a manual discount-approval screen that the standard order type handles through a different,

automated mechanism. The workshop is time-boxed, and the project follows a clean-core approach in which configuration is performed in SAP Central Business Configuration and unnecessary core modifications are avoided. The consultant must decide how to handle this single mismatch so that the workshop stays on track and the scope record stays accurate.
How should the consultant handle the legacy discount-approval step during the fit-to-standard workshop?

A. Record the delta as a genuine gap in scope and adopt the standard best-practice process for the other steps, noting it in the workshop log.
B. Replicate the customer's full legacy order process across the entire configuration scope so that no business user has to change how they work today.
C. Rebuild the standard flow as a custom core modification inside the Starter system so the workshop demo screens match the legacy system exactly.
D. Leave the gap undocumented and defer every configuration decision to the Realize phase, when work begins directly in the target production system.

Answer: A

Explanation:
Recording the mismatch as a genuine gap in scope while adopting the standard best-practice process for the remaining steps keeps the project clean-core and produces an accurate scope record. Fit-to-standard is designed to validate the standard process first and capture only true deltas, so this decision belongs in the workshop log where it can be evaluated later. This action satisfies both the workshop purpose and the clean-core approach.

Thank You for Trying Our Product

For More Information – Visit link below:

https://www.examsboost.com/

15 USD Discount Coupon Code:

G74JA8UF

FEATURES

- ✓ 90 Days Free Updates
- ✓ Money Back Pass Guarantee
- ✓ Instant Download or Email Attachment
- ✓ 24/7 Live Chat Support
- ✓ PDF file could be used at any Platform
- ✓ 50,000 Happy Customer