SAP-C02 Cheat Sheet

High-yield AWS Solutions Architect Professional review for the last pass before practice tests.

BEST FOR	Last review before mini test or app session
FOCUS	Multi-account, hybrid, migration, governance, and optimization
USE WITH	Quick Summary + Service Map

1. Domain weights and architect mindset

26 / 29 / 25 / 20 — Organizational complexity, New solutions, Continuous improvement, and Migration are the scored domains.
Design for many accounts — Expect Organizations, SCPs, Control Tower, landing zones, and delegated administration.
Architect with trade-offs — Choose patterns by resilience, cost, performance, operations, and security together.
Think at enterprise scale — Cross-account networking, data sharing, identity boundaries, and central logging appear often.
Justify the design — The best answer usually explains why one pattern scales better or lowers long-term risk.

2. Identity, governance, and multi-account

Organizations + OUs — Use OUs to segment environments and apply broad governance with SCP guardrails.
IAM Identity Center — Prefer centralized workforce access across accounts instead of duplicating IAM users.
SCP vs IAM policy — SCP sets the maximum permission boundary; IAM policies grant what a principal can actually do.
KMS strategy — Plan key ownership, grants, cross-account use, and rotation requirements early.
CloudTrail + Config + Security Hub — Centralize auditing and compliance signals into shared security accounts.

3. Networking, hybrid, and edge design

Transit Gateway — Standard answer for large hub-and-spoke connectivity across many VPCs and accounts.
Direct Connect + VPN — DX for predictable throughput; VPN for faster setup, backup, or encryption overlay.
PrivateLink vs VPC peering — PrivateLink exposes a service privately; peering gives broader network reach.
Route 53 patterns — Use latency, weighted, failover, geolocation, or geoproximity based on the routing goal.
CloudFront + Global Accelerator — CloudFront is content/app acceleration; Global Accelerator improves anycast entry to regional apps.

4. Migration, modernization, and continuous improvement

MAP thinking — Pick rehost, replatform, refactor, repurchase, retire, or retain based on business constraints.
DMS vs SCT — DMS moves data; SCT helps convert schema when engines differ.
Observability first — Use CloudWatch, X-Ray, logs, traces, and metrics to identify where cost or latency is really coming from.
Cost optimization — Match pricing model to steady vs variable workloads with Savings Plans, RIs, Spot, and rightsizing.
Well-Architected habit — Review architectures continuously, not only before launch.