CHAPTER 7

Becoming a Cloud Architect

From Technical Implementation to Enterprise Architecture and Strategic Decision-Making

Chapter Introduction — The Next Evolution in Your Career Journey

In Chapter 6, we explored how traditional IT experience transfers into cloud engineering. Now we take that knowledge further. Becoming a Cloud Architect requires more than deploying infrastructure: you must understand how technologies work together, how they support business objectives, and how to evaluate competing solutions.

My journey from Help Desk through Oracle Database Administration, Oracle Applications DBA work, enterprise systems, consulting, and architecture developed that perspective. Cloud introduced new delivery models, but familiar questions remained: What does the business need? What systems depend on each other? What risks can we tolerate? What will it cost? Who will operate it? How will success be validated?

A Cloud Architect must understand technology, but the real responsibility is to make technology serve the business.

Learning outcomes

- Distinguish cloud engineering from architecture
- Translate business objectives into technical requirements
- Perform discovery and readiness assessments
- Design and evaluate enterprise cloud solutions
- Balance security, resilience, performance, cost, and operations
- Prepare architecture decision records and executive recommendations

7.1 — From Cloud Engineer to Cloud Architect

A cloud engineer may deploy virtual machines, configure networks, implement permissions, automate infrastructure, and troubleshoot workloads. An architect must understand those capabilities while deciding which approaches satisfy business requirements and why.

For a database migration, an engineer may configure the chosen target. The architect first evaluates self-managed VMs, compatible managed services, modernization, or retention, using evidence about licensing, dependencies, security, cost, performance, and recovery.

Architecture is not necessarily a linear promotion from Cloud Engineer. Professionals arrive from databases, infrastructure, networking, security, consulting, software engineering, and enterprise architecture. My own career included consulting and enterprise architecture before broader cloud work.

Career connection: Technical experience provides implementation credibility; consulting and enterprise experience expand responsibility from components to business outcomes.

Questions to ask

- How do I deploy this? → Why is this design appropriate?
- Which service is available? → Which service satisfies the requirements?
- How do I troubleshoot a resource? → What happens when a dependency fails?
- How do I automate it? → Who will own, secure, fund, and operate it?

7.2 — Why Cloud Architecture Begins With Enterprise Understanding

Enterprise systems rarely operate alone. An ERP application may depend on databases, application servers, identity systems, networks, integrations, storage, monitoring, backups, and operational support. A healthy database does not guarantee that a business process is functioning.

Oracle DBA work taught me performance, recovery, and availability. Oracle Applications DBA responsibilities expanded my understanding of application tiers and dependencies. Consulting strengthened my ability to connect those technical concerns to business needs.

Cloud is a delivery model, not a business objective. A business may seek better recovery, reduced maintenance, faster delivery, stronger controls, or growth. Sometimes retaining or modernizing an on-premises component is the better near-term decision.

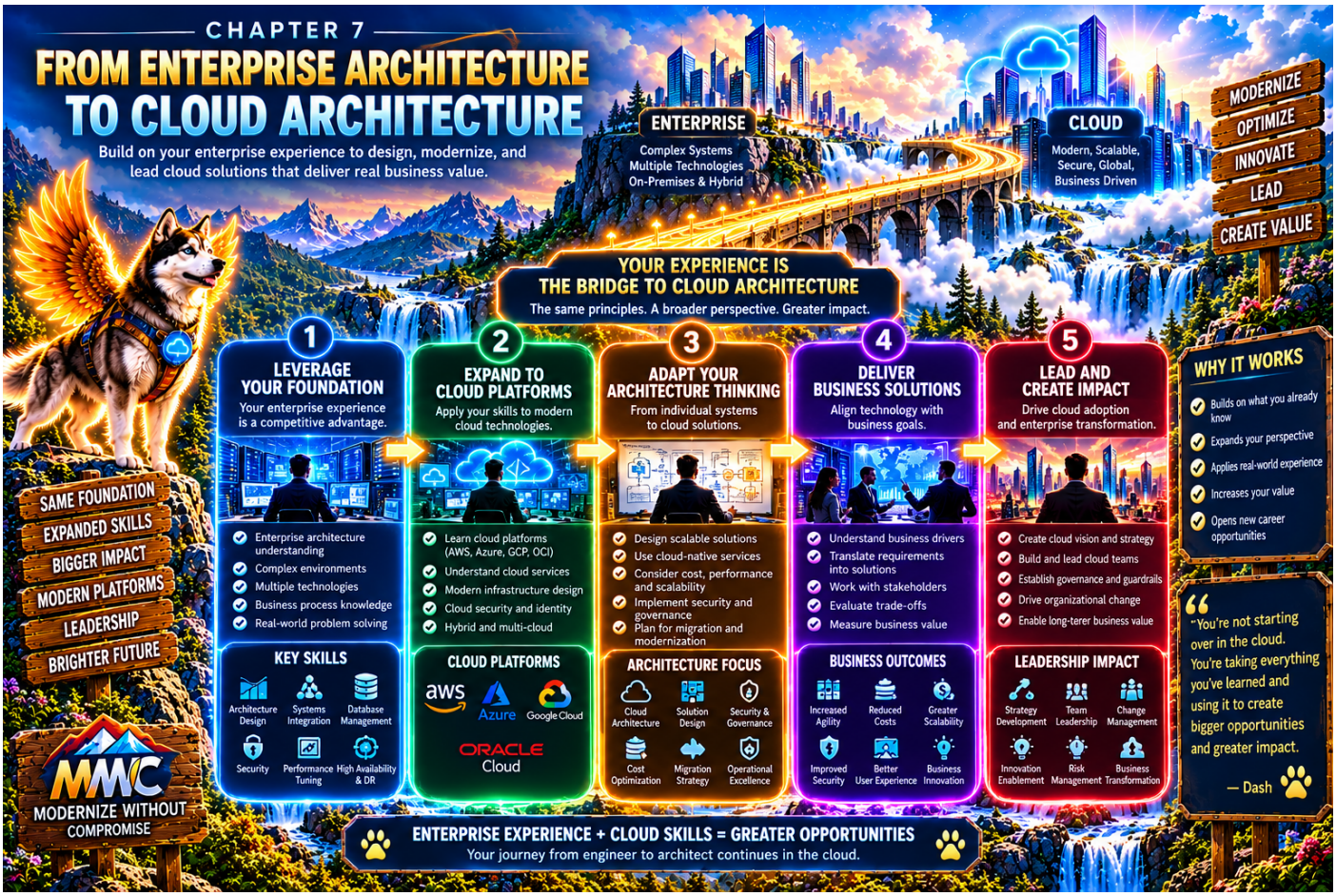
Illustrative scenario: An ERP system has a database, interfaces, and financial reporting. Before recommending cloud VMs, assess dependencies, support, licensing, recovery, and costs. This scenario is instructional, not a specific client claim.

Career connection: Enterprise understanding reveals risks and requirements that product familiarity alone cannot.

Enterprise perspectives

- Business processes and users
- Applications, databases, and data flows
- Infrastructure, integrations, and dependencies
- Security, governance, and compliance
- Operational support and recovery

Infographic 1 — The Bridge From Enterprise to Cloud Architecture



7.3 — Thinking Like an Architect: Business First, Technology Second

An architect translates business goals into functional and nonfunctional requirements. “The application must remain available during sales periods” requires clarification: what availability target, peak demand, failure impact, and recovery time does the business need?

Functional requirements describe what the system does. Nonfunctional requirements describe how it performs and operates, including availability, security, response time, scalability, recovery, maintainability, and cost.

Budget, deadlines, compatibility, support lifecycle, licensing, compliance, and team capabilities are real constraints. Architecture involves choosing among imperfect alternatives and explaining the trade-offs.

Practice: For a small application requiring restricted access, daily processing, backup, and limited cost, compare two designs. Record benefits, assumptions, risks, and validation criteria.

Career connection: Database troubleshooting taught evidence-based investigation; consulting taught requirements discovery.

Decision areas

• Reliability: required availability and recovery targets

- Security: permissions and control validation
- Performance: latency, utilization, and demand evidence
- Cost: migration, licensing, support, and run costs
- Operations: incident, patching, monitoring, and recovery ownership
- Migration: dependencies, testing, and rollback

7.4 — Cloud Assessment, Discovery & Readiness

Discovery precedes design. Inventories and diagrams can be outdated, incomplete, or inaccurate. Validate applications, dependencies, ownership, usage, and operating requirements with the appropriate teams.

A server allocated 64 GB of memory does not necessarily require a 64-GB cloud instance. Review historical utilization and performance. Moving an application while retaining a database on-premises may introduce latency, security, and support implications.

Useful deliverables include a validated inventory, dependency map, business criticality, readiness findings, risk register, recovery requirements, and candidate migration approaches.

Career connection: Enterprise applications, databases, consulting, and cloud assessments reinforced assessment before solution selection.

Seven discovery steps

- 1. Business objectives and success measures
- 2. Applications, servers, databases, environments, and owners
- 3. Integrations, identity, networks, and data flows
- 4. Utilization, incidents, monitoring, backup, and recovery
- 5. Compatibility, licensing, support, security, and skills
- 6. Risks, assumptions, and unanswered questions
- 7. Validated target architecture requirements

7.5 — Designing Enterprise Cloud Architecture

A target architecture is an organized design showing how applications, data, infrastructure, security, integration, and operations work together to satisfy requirements. It is more than a list of cloud products.

A landing zone can establish shared foundations such as account structures, network connectivity, identity, logging, governance, and cost visibility. Appropriate patterns vary by provider and organization.

A conceptual application design may use controlled entry, separated network tiers, application compute, a compatible database, identity and secrets management, logging, backup, resilience, and cost controls.

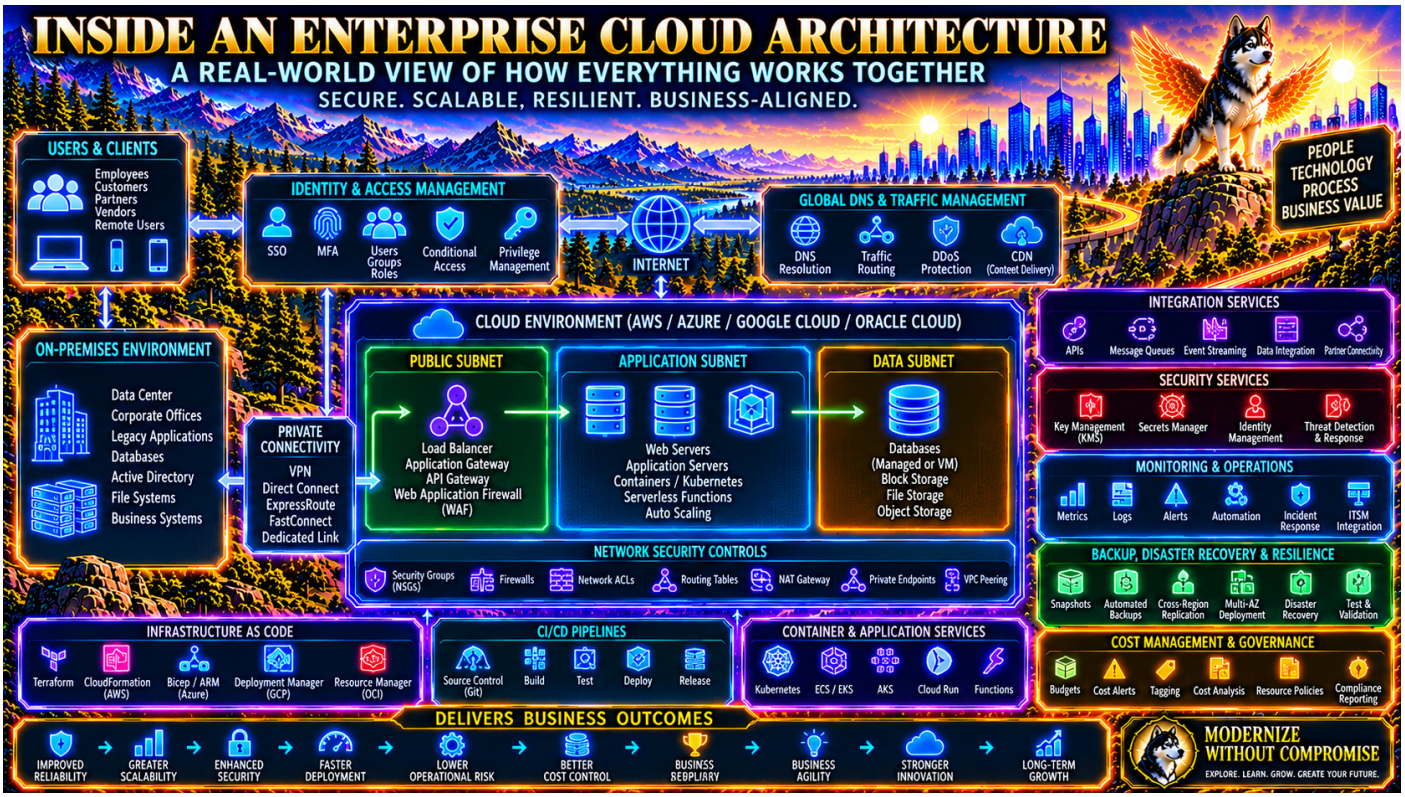
Production suitability requires further validation.

Career connection: My database, infrastructure, enterprise application, and architecture experience helped me reason about complete solutions.

Major design domains

- Account or subscription organization and ownership
- Virtual networking, DNS, routing, and hybrid connectivity
- Identity, authentication, and least privilege
- Compute, containers, or managed application platforms
- Database, storage, and integration choices
- Monitoring, incident response, and operational ownership
- Security, recovery, governance, and cost controls

Infographic 2 — Inside an Enterprise Cloud Architecture



7.6 — Security, Governance & Compliance by Design

Security is not a feature added after deployment. Architecture affects access, data protection, auditability, and risk. Shared responsibility differs by cloud service model.

Cloud provider compliance certifications do not automatically make every customer workload compliant. Customer configuration, permissions, data handling, and processes must satisfy applicable obligations.

An internal financial application requires decisions about authorized access, authentication, audit logging, encryption, retention, and compliance before selecting the final design.

Career connection: Database administration and enterprise applications introduced the importance of protecting business-critical data.

Security and governance review

- Least privilege, MFA, and separation of duties
- Restricted network and administrative access
- Data classification, encryption, keys, and retention
- Logging, monitoring, and incident response
- Ownership, naming, tagging, and approved regions

• Security, legal, risk, and compliance stakeholder review

7.7 — High Availability, Disaster Recovery & Resilience

High availability aims to reduce interruptions from certain failures. Disaster recovery addresses restoration following significant disruption. Architects must evaluate the complete business service, not only a database or server.

Recovery Point Objective (RPO) describes acceptable potential data loss. Recovery Time Objective (RTO) describes target restoration time. Both come from business needs and must be validated against the design.

Multiple availability zones can support resilience but do not guarantee application availability. A recovered ERP database does not restore the business process if an essential integration remains unavailable.

Career connection: Oracle DBA recovery and production responsibilities supported broader resilience thinking.

Resilience checklist

• Failure domains, application redundancy, and dependencies

• Backup retention, restore procedures, and data consistency

• Network and identity recovery dependencies

• Monitoring, incident response, and support ownership

• Controlled recovery testing and business process validation

7.8 — Performance, Scalability, Cost & FinOps

Architecture balances performance, reliability, security, operations, and cost. Larger resources may increase spending; managed services may reduce some maintenance but introduce different compatibility and pricing considerations.

FinOps is a collaborative approach to cloud financial accountability. Architects contribute through design decisions, tagging, cost visibility, sizing, and transparent trade-offs.

Compare self-managed and compatible managed services using full costs over a defined period, including migration, support, licensing, operations, and backup.

Career connection: Database performance tuning developed evidence-based sizing; consulting added financial and delivery constraints.

Evaluate with evidence

• Response time, throughput, transactions, and peak demand

• CPU, memory, storage, database, and network utilization

• Vertical and horizontal scaling feasibility

• Compute, transfer, licensing, monitoring, backup, and support costs

• Operations ownership and lifecycle costs

7.9 — Cloud Migration & Modernization Decisions

Migration strategy follows validated discovery, business objectives, dependencies, and constraints. Options include rehost, replatform, refactor, replace, retain, and retire.

Migration waves should reflect application and data dependencies, business schedules, and support readiness. Cutover needs technical and business validation, rollback planning, monitoring, recovery readiness, and accountable approvals.

A legacy reporting workload may merit rehosting, replatforming, or temporary retention depending on integration, licensing, support, risk, and cost evidence.

Career connection: Enterprise migration and consulting work reinforced the importance of dependency discovery and business acceptance.

Migration choices

• Rehost: move with limited changes

• Replatform: targeted platform changes

• Refactor: substantial application redesign

• Replace: different product or SaaS

• Retain: keep in current environment

• Retire: decommission after dependency and data checks

7.10 — Architecture Documentation & Technical Decision-Making

A diagram shows design structure; a defensible recommendation also explains why it was selected. Documentation supports review, implementation, operations, and future changes.

Diagrams should distinguish current and target state, confirmed findings and assumptions, network boundaries, dependencies, and data flows. Avoid adding technology simply to make diagrams look sophisticated.

An Architecture Decision Record (ADR) captures decision title, context, requirements, alternatives, selected design, rationale, consequences, risks, assumptions, and validation.

Career connection: Consulting and enterprise architecture reinforced explaining and defending design decisions.

Core architecture deliverables

• Current-state and target-state diagrams

• Functional, nonfunctional, and operational requirements

• Architecture Decision Records

- Risk and assumption registers
- Implementation and validation plans

7.11 — Communicating With Executives, Clients & Technical Teams

Engineers need technical assumptions, controls, and testing. Business owners need service impacts, risks, and acceptance criteria. Executives need the problem, options, recommendation, benefits, costs, major risks, and decision required.

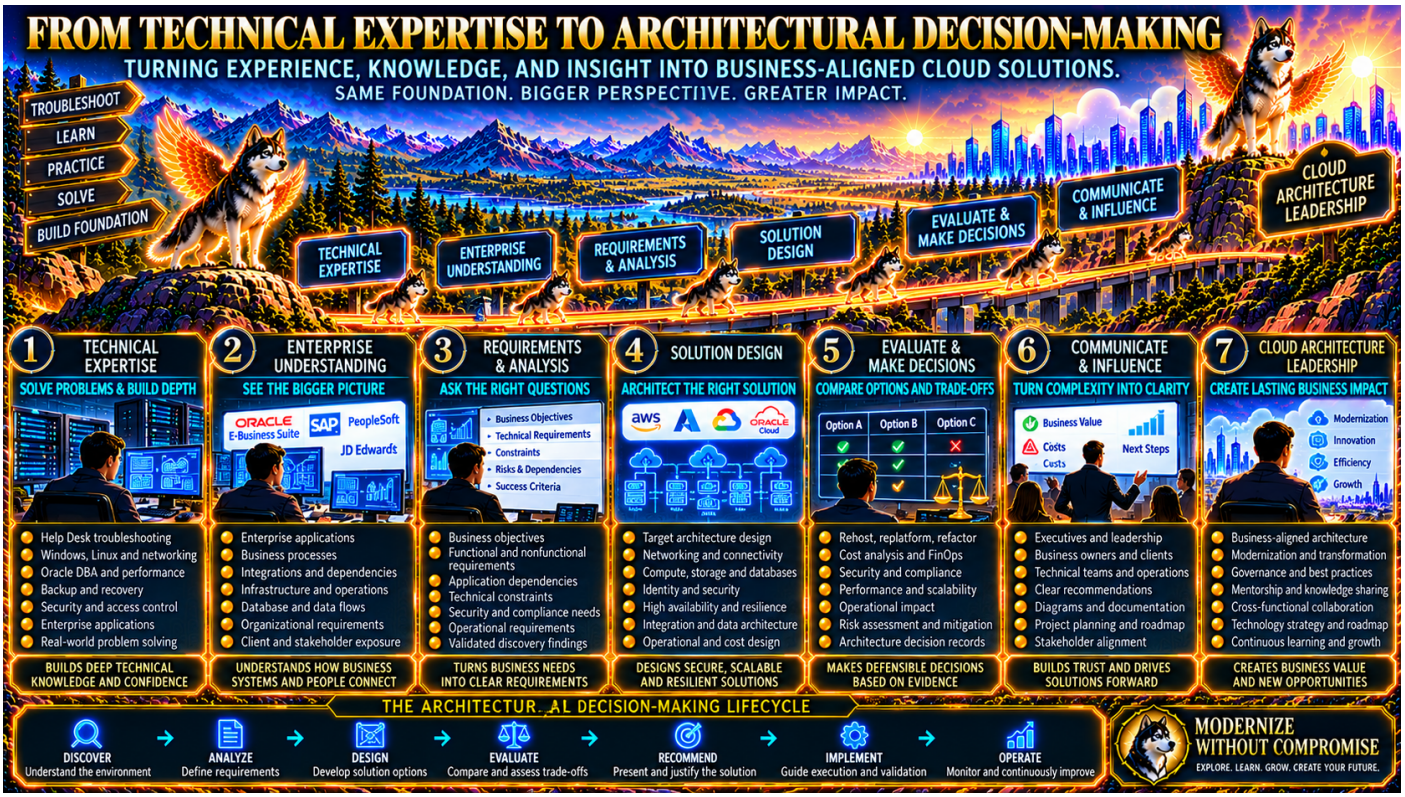
Technical knowledge gives credibility; clear communication enables action. Adapt detail to the audience without hiding important uncertainty.

Career connection: Help Desk developed user communication; database administration built technical confidence; consulting expanded client and executive engagement.

Executive recommendation structure

- 1. Business objective and current challenge
- 2. Evidence and options considered
- 3. Preferred architecture and why
- 4. Expected benefits and trade-offs
- 5. Cost assumptions, risks, and dependencies
- 6. Decisions required and next steps

Infographic 3 — From Technical Expertise to Architectural Decision-Making



7.12 — Practical Toolkit: Cloud Architect Worksheets

Use a fictional scenario or an environment you are authorized to assess. Do not publish confidential client information.

WORKSHEET A — Cloud Architecture Requirements & Discovery Assessment

Purpose: Translate business objectives and current-state evidence into validated requirements.

Business objective, problem, desired outcome, owner, users, and success criteria

_____

_____

_____

_____

_____

Discovery area	Known findings	Evidence	Unknowns
Applications			
Databases			
Compute			
Storage			
Networking			
Identity			
Integrations			
Security			
Monitoring			
Backup/recovery			
Licensing			
Operations			

Requirement	Importance	Evidence	Priority
Availability			
Recovery time			
Recovery point			
Performance			

Requirement	Importance	Evidence	Priority
Scalability			
Security			
Compliance			
Data protection			
Operations			
Cost			

Stakeholder	Responsibility	Information needed
Business owner		
Application owner		
Database team		
Infrastructure team		
Network team		
Security team		
Operations team		
Finance/procurement		

Five confirmed requirements and three unknowns

_____

_____

_____

_____

Three architecture risks, mitigations, and owners

_____

_____

_____

_____

Career reflection — existing skills used and skills to develop

_____

_____

_____

Portfolio artifact: Cloud Architecture Requirements & Discovery Assessment.

WORKSHEET B — Enterprise Cloud Solution Design & Decision Matrix

Purpose: Compare credible designs and document the reasons behind a recommendation.

Workload, objective, current architecture, cloud platform, and constraints

_____

_____

_____

_____

_____

Component	Proposed approach	Rationale	Risk / assumption
Account organization			
Networking			
Identity			
Compute			
Database			
Storage			
Integrations			
Security			
Monitoring			
Backup/recovery			
Cost controls			

Describe Option A and Option B

_____

_____

_____

_____

Criterion	Option A	Option B
Business fit		

Criterion	Option A	Option B
Compatibility		
Security		
Availability		
Recovery		
Performance		
Scalability		
Operations		
Migration effort		
Cost		
Skills required		
Vendor dependencies		

Architecture Decision Record: choice, evidence, alternatives, trade-offs

_____

_____

_____

_____

_____

Risks, assumptions, decision owner, and status

_____

_____

_____

_____

Validation area	Acceptance criteria	Owner
Functionality		
Performance		
Security		

Validation area	Acceptance criteria	Owner
Availability		
Recovery		
Monitoring		
Cost		
Business acceptance		

Career reflection — technical skills supporting these decisions

_____

_____

_____

Portfolio artifact: Enterprise Cloud Solution Design & Architecture Decision Record.

WORKSHEET C — Cloud Architect Executive Recommendation & Presentation

Purpose: Turn technical findings into a clear business recommendation.

Business objective, current challenge, recommended solution, expected value

_____

_____

_____

_____

_____

Option	Benefits	Risks	Cost considerations
Option A			
Option B			
Option C, if applicable			

Preferred option, evidence, and business/technical assumptions

_____

_____

_____

_____

_____

Risk	Business impact	Mitigation	Owner

Implementation cost, recurring cost, support, ownership, skills

_____

_____

_____

_____

_____

Seven-slide executive presentation outline

• Slide 1 — Business problem

• Slide 2 — Current environment and evidence

• Slide 3 — Architecture options

• Slide 4 — Recommended design

• Slide 5 — Benefits and trade-offs

• Slide 6 — Costs, risks, and dependencies

• Slide 7 — Decision required and next steps

Explain the recommendation to an engineer versus an executive

_____

_____

_____

_____

Anticipated leadership questions and missing evidence

_____

_____

_____

_____

Portfolio artifact: Cloud Architecture Executive Recommendation & Presentation.

7.13 — Chapter Takeaways & Preparing for Career Growth

Becoming a Cloud Architect expands responsibility. It requires technical credibility, enterprise understanding, discovery, solution evaluation, risk awareness, documentation, and communication.

• Architecture begins with business understanding

• Discovery and validation precede design

• Security, resilience, performance, operations, and cost require trade-offs

• Technical implementation knowledge remains important

• Documentation makes decisions defensible

- Business validation matters beyond technical success
- Communication is part of architecture leadership

Career stage	Core capability	Architecture contribution
Help Desk	Troubleshooting and communication	User and service understanding
Oracle DBA	Performance, data, recovery	Database and resilience design
Oracle Applications DBA	Enterprise application support	Business dependencies
Enterprise consulting	Client engagement and delivery	Requirements and stakeholders
Cloud engineering	Implementation and operations	Technical feasibility
Enterprise/cloud architecture	Design and decisions	Business-aligned solutions

Your 30-day architecture challenge

- Week 1 — Discover requirements, dependencies, and constraints
- Week 2 — Design two architecture options
- Week 3 — Evaluate security, reliability, performance, operations, and cost
- Week 4 — Produce an ADR and executive presentation

Preparing for Chapter 8 — Networking, Mentorship & Opportunity

Career advancement also depends on relationships, visibility, mentorship, and opportunity. Connections with Oracle DBAs, training requests, and consulting exposure helped expand my career possibilities. Chapter 8 explores those human dimensions.

Final lesson: Becoming a Cloud Architect means connecting technical expertise, enterprise understanding, business requirements, and judgment into solutions that can be explained, implemented, operated, and validated.