**BCS Syllabus and Exam Overview**

**(Each Section on a Page)**

**Introduction**

The primary aims of the BCS examinations and associated training are to give enterprise and solution architects a broad framework that covers the range of architecture work that precedes and steers system development, and to focus attention on areas where the architect is responsible for effective design and risk management.

A secondary aim is to provide architects with generally applicable knowledge and training. General here means independent of any specific architecture framework (Gartner, TOGAF, etc). This enables Training Providers to teach general knowledge and skills, rather than framework-specific terms, concepts, structures and processes.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Section** | **Intermediate Questions** | **Practitioner Questions** |
| 1 | Architecture & architects | 4 | 0 |
| 2 | Architecture precursors | 6 | 5 |
| 3 | Architecture frameworks | 5 | 1 |
| 4 | Business architecture | 5 | 5 |
| 5 | Data architecture | 3 | 4 |
| 6 | SW / Component architecture | 5 | 3 |
| 7 | Applications architecture | 4 | 6 |
| 8 | Design for non-functionals | 3 | 6 |
| 9 | Infrastructure architecture | 5 | 4 |
| 10 | Migration planning | 0 | 2 |
| 11 | Architecture management | 0 | 4 |
|  | **Total** | **40** | **40** |
|  | **Pass mark** | **26** | **26** |

1. Architecture and Architects
   1. **Granularity** – Enterprise, Solution, Software
   2. **Primary Domains** – Business, Data, Applications, Infrastructure
   3. **Other Domains** – Information, Information systems, Software, Security, Hierarchical, Platform
   4. **Roles** – Enterprise architect, solution architect
   5. **Knowledge** – business and technical goals, business and technical environment, technical knowledge, methodology knowledge, requirements, analysis, innovation, leadership, stakeholder management, communication, political, soft, project management, commercial, risk
   6. **Enterprise** – offers services, via divisions which offer services to each other
   7. **Application** – offers services, via components, provide services to each other

## Foundation Learning Objectives Recognise the ways base terms like architecture, system, structure, behaviour, interface, service and function are used in the BCS reference model (and so, examination question wording).

**Intermediate Learning Objectives (4 questions)  
Architecture granularity**

Distinguish the following three levels of granularity in architecture definition: Enterprise, Solution(s) and Software.

**Architecture domains**

Recognise the need for separation of concerns. Recognise that architecture descriptions are sliced into high-level views to address the separate concerns of different stakeholders. Distinguish the following four broad perspectives: Business, Information/Data, Applications and Technology/Infrastructure.   
**Hierarchical or layered architecture**Recognise division into layers as a fundamental and widely-used technique of architecture and design. Recognise how the architecture domains (above) may be regarded as layers.   
**Architect roles, goals and skills**List a variety of architect roles. Distinguish the goals of enterprise architects and solution architects. List some knowledge and skills required for those roles.   
  
**Practitioner Learning Objectives (0 questions)**Discuss the practical use of the concepts above and apply them a case study.   
Draw analogies between human activity systems and software systems. Recognise the limits to these analogies.

1. Architecture **Precursors** (requirements and context)
   1. ***Stakeholders*** *– sponsors, concerns*
   2. ***Architect stakeholders*** *– owners, managers, buyers, suppliers, designers, users, operators and maintainers*
   3. **Drivers** – Internal, External, Competitive
   4. **Aims** – Goal, Objective, Requirement
   5. **Balanced Scorecard**
   6. **SMART** – Specific, Measurable, Actionable, Realistic, Time Bound
   7. **Directives** – Principle, Policy, Business Rule
   8. Business Mission
   9. **Business Vision** – Aspirational target state to achieve goals, may be a guide or specific desired outcome.
   10. **Mission Statement** – Mission, Vision, Main Goals, and Values
   11. **Target Solution Hierarchy** – Vision, Outline, To Be Built
   12. **Plans** – Strategy, Programme, Project
   13. **Standards** – Bodies, Enterprise Standards Information Base (SIB), Profile, Profiling
   14. **Scope of Architecture Work** – Breadth, Focus, Depth, Constraints
   15. **Context Diagram** – Systems Black Box, Entity – Actor, Role
   16. **Requirements** - Requirements Statement, Functional, Audit, Non-functional, Performance, Availability, Recoverability, Reliability, Integrity, Scalability, Security, Serviceability, Usability, Maintainability, Portability, Interoperability, Integratability, Extensibiltiy, Service Level Agreement – written agreement between service provider and customer, Service Level Requirement – Criteria for level of service to meet business objectives, Regulatory (Accountability and Procurement, Data Protection, Disability and Accessibility, Shareholder Protection and Audit, Intellectual Property, Health and Safety
   17. **Business Case** – (see migration planning), Business scenario – a process or story about how the system should be structured and operate.

**Foundation Learning Objectives**

**Stakeholders**

* Identify stakeholders and their concerns.
* Apply stakeholder management techniques.

**Elaboration of inputs to become deliverables**

* Recognise that architecture involves hierarchical decomposition inputs into outputs, so the outputs at one level are inputs to the level below.

**Intermediate Learning Objectives (6 questions)**

**Drivers aims and directives**

* Distinguish drivers, aims and directives. Recognise the hierarchical decomposition of both aims and directives. Define SMART aims.

**Solution descriptions and plans**

* Distinguish business from IT. Recognise the hierarchical decomposition of solutions and the plans to deliver them.

**Standards**

* Identify several standard bodies. Recognise the need for the enterprise to have a Standards Information Base, and to define the standards profile of solution components.

**Scope of architecture work**

* Recognise ways to define the scope of architecture work, and the scope of an enterprise or system. Draw a context diagram showing interfaces to external systems). Distinguish the concepts of external entity, actor, role.

**Requirements**

* Recognise the difference between functional and non-functional requirements. List ten kinds of non-functional requirement.
* Distinguish requirements, Service Level Agreements and Service Level Requirements.

**Regulatory requirements**

* Recognise several kinds of regulatory requirement relevant to architecture definition.

**Business case**

* Recognise the contents of a business case; notably RoI, cost-benefit analysis, solution options and risk analysis.
* Recognise in this connection the nature and purposes of gap analysis (between options), trade-off analysis, and business scenarios.

**Practitioner Learning Objectives (5 questions)**

* Discuss the practical use of the concepts above and apply them a case study.

1. **Architecture Frameworks**
   1. *Architecture Framework*
   2. **Architecture Process Frameworks** – Architecture states (baseline – transition – target), TOGAF, ADM, Avancier
   3. **Architecture Descriptions** – Deliverable, Artefact, Entity, Mapping, ISO/IEC42010 Standard, View, Viewpoint
   4. **Architecture Models and Languages** – Idealisation Hierarchy – Conceptual, Logical, Physical, MDA, System Modelling Techniques (Structured and UML), Modelling Language – IDEF, UML, Archimate
   5. **Architecture Description** – Repository, Zachman framework, Enterprise Continuum (Architecture, Solutions), Reference Model

**Foundation Learning Objectives**None.   
 **Intermediate Learning Objectives (5 questions)  
Architecture process frameworks**Recognise the phases of an architecture process framework that helps architects to describe a baseline architecture, a target architecture and the transformation between them.   
Recognise the ten phases of the method for enterprise architecture development (ADM) in the Open Group Architecture Framework (TOGAF) Recognise that a process for solution architecture involves similar activities but with different goals and a different level of detail.   
**Architecture descriptions**Distinguish the key concepts of architecture description (system, view, view point, concern, and stakeholder) and the relationships between them, as described in the standard popularly known as ANSI 1471. Identify building blocks used in architecture descriptions.   
**Architecture models**Recognise that models are abstractions composed of instances of artefact types and mappings between them.   
Distinguish three kinds of abstraction: by composition, generalisation and idealisation.   
Distinguish three levels in an idealisation hierarchy: conceptual, logical and physical. Recognise the possible transformations involved in Model-Driven Architecture (MDA).   
Recognise the relationship (line) symbols shared by the Unified Modelling Language (UML) and ArchiMate.   
**Architecture description frameworks**Identify the rows and columns of the architecture description framework known as the Zachman framework. Recognise the meta model of an architecture repository. Recognise the two dimensions of the Enterprise Continuum in TOGAF.

**Practitioner Learning Objectives (1 question)**Discuss the practical use of the concepts above and apply them a case study.

1. **Business Architecture**
   1. *Goal, Objective, Enterprise, Business, Organisation, Organisation Unit, Management Structure*
   2. **Structure and Behaviour** – Business function catalogue or portfolio, Business function, Core business functions, Core competency, Support business function, Business capability, Capability based planning, Business domain, Business process, Value stream, Value chain, Business service, Business service oriented architecture, Business data model, Business semantics, Business model, Process map, Business process decomposition, OPOPOT, Process automation hierarchy, Information system service, Workflow
   3. **Business Security** – Design for human and organisation security (non-IT)

**Foundation Learning Objectives**Distinguish the physical and logical structures of a business (organisation units and business functions or capabilities). Recognise several synonyms in this area.   
Distinguish business services and business processes. Recognise synonyms in this area, and the use of the terms value stream and value chain.   
Recognise mappings of the above to place and time.   
 **Intermediate Learning Objectives (5 questions)  
Business architecture structure and behaviour.**Recognise ways to model a business system, including business process structures, business function (or capability) structures, business data models and business rules.   
**Business process decomposition and automation.**Recognise three levels of business process granularity: workflow, use case and automated service.   
Recognise the benefits of implementing a process as an ACID transaction and the need for compensating transactions where this is not possible.   
Distinguish business services from data services.   
**Design for business security.**Identify several features in design for human and organisational security.   
  
**Practitioner Learning Objectives (5 questions)** Discuss the practical use of the concepts above and apply them a case study.

1. **Data Architecture**
   1. *Entity, Event, Data, Information, Structured Data, Data Item, Data Structure, Data Entity, Data Event, Data Lifecycle, Type, Data Type (primitive), Data Type (user defined), Constraint (rule), Derivation rule, Meta Data, Data Dictionary, File, Data Sources*
   2. **Unstructured Data Management** – Content, Document, Knowledge
   3. **Data Architecture** – Data in Storage, Data Store, Data Model, State, Database, Cache, Data In Motion, Data Flow, Regular Expression, Data Format, Data Format Standard, Canonical data Model
   4. **Data Qualities and Integration** – Data Quality, Data Integrity, Data Flow (or message) Integrity, Data Dissemination View, Data Warehouse, Master Data Management
   5. **Design For Data Security** – Data Security, Security Protection, Security Feature, Security Policy, Information Domain, Identify, Encryption, Checksum, Digital Signature

**Foundation Learning Objectives**Recognises how the data stored in and transmitted between information systems is a model of entities and events in the external environment, or real world.   
Recognise how data is described using meta data, data structures, data types and data dictionaries.   
  
**Intermediate Learning Objectives (3 questions)  
Knowledge and/or content management**Distinguish knowledge and/or content management from data management.   
**Data architecture structure**Understand how to describe data in storage using data models. Recognise the issues in storing state outside of a database in a cache.   
Recognise the functions of database management system and concept of a federated transaction across a distributed database.   
Understand how to describe data in motion, data flow definitions, regular expression, message formats and canonical data models.   
**Data qualities and integration**Recognise how the three primary data qualities (CIA) may be measured at three or more levels.   
Distinguish data store integrity from data flow (or message) integrity.   
Recognise the dimensions of a data dissemination view.   
Recognise the way denormalization is used to optimise output from a data warehouse.  
Recognise the concept of master data management, and ways to implement it.   
D**esign for data security**Recognise key concepts in data security: security protection, security feature, security policy, information domain, identity, encryption, checksum and digital signature.   
  
**Practitioner Learning Objectives (4 questions)**Discuss the practical use of the concepts above and apply them a case study.

1. **Software Architecture**
   1. ***Foundation*** *– Modular Design, Client, Server, Design Time Structure and Runtime Behaviour, Encapsulation, Cluster or Affinity Analysis, Stateless, Façade, Transactional, Delegation, Dependency, Cyclic Dependency, Hierarchical (non-cyclic) Dependency, Service Quality, Service Oriented Design (SW), Service Oriented Design Challenges*
   2. **Component Interfaces** – IDL, Realisation, Synchronicity, Synchronous, Asynchronous, Loosely-coupled, Tightly-coupled
   3. **Component Structures and Patterns** – Design Pattern, Hierarchical and Peer to Peer Structures, Fork and Chain Structures, Model View Controller, Controller, OO Design Pattern, Singleton, Proxy, Adapter, Observer
   4. **Component Interoperation Styles** – Distributed Objects (DO), Service Oriented Architecture (SOA), Representational State Transfer (REST), Event Driven Architecture (EDA)
   5. **Component Communication Styles** – Point to Point, Introduction Agent (Direct Broker), Mediator, Message Broker (Indirect Broker), Message Router, Message Bus (Schema Based), Passive Mediator, Publish and Subscribe (Topic, List or Content Based) – Broadcast (sends topic everywhere).

**Foundation Learning Objectives**Recognise system modelling techniques including at least context diagrams, data flow diagrams, use case diagrams, process flow charts (activity diagrams), interaction diagrams (sequence diagrams) and state charts.   
Recognise the core concepts of modular design including: encapsulation, façade, aggregation by cluster or affinity analysis, stateful and stateless components.   
Recognise the core elements of a service contract and principles of service-oriented design.   
Recognise the concepts of transactional processing.   
Recognise the concept of delegation from clients to servers. Distinguish cyclic dependency from hierarchical (non-cyclic) dependency. Recognise several service qualities and service-oriented design challenges.

**Intermediate Learning Objectives (5 questions)  
Component structures and patterns**Recognise the essential ideas in component structures and patterns: client versus server, loosely coupled versus tightly-coupled.   
Recognise basic design patterns: hierarchical and peer-to-peer structures, fork and chain structures. model-view controller (MVC).   
Recognise a few common OO design patterns.   
**Component interfaces**Recognise the concepts of an Application Programming Interface (API) and Interface Description Language (IDL).   
Recognise how an interface is realised by a component.   
Distinguish asynchronous from synchronous communication, from both client and server perspectives.   
**Component interoperation styles**Recognise how the following component interoperation styles are successively more loosely coupled: Distributed Objects style (DO): Service-Oriented Architecture style (SOA):   
Representational State Transfer style (REST): Event-Driven Architecture style (EDA).  
**Component communication styles**Recognise how the following component communication styles increasingly decouple client/sender from server/receiver: point-to-point communication, introduction agent (direct broker) and mediator (indirect broker).   
Recognise at least one middleware technology used for each style.   
**Publish and subscribe distribution (3 questions)**Distinguish different kinds of publish and subscribe distribution.   
  
**Practitioner Learning Objectives (3 questions)**Discuss the practical use of the concepts above and apply them a case study.

1. **Applications Architecture**
   1. ***Foundation*** *– Information System, User, Application, Business Application, Generic Application, Platform Application, ERP, CRM,*
   2. **Application Catalogue or Portfolio** – Application Portfolio Management, Application Architecture Structure
   3. **Application Architecture Management** – Architecture behaviour model, Information System Service, Use Case, Automated Service (SOA sense), Business Service (SOA sense), Data Service (SOA sense), Transaction, ACID (Atomic, Consistent, Isolated, Durable, Compensation Transaction
   4. **Applications Integration** – Batch Process, Extract Transform and Load, Application Consolidation, Point to Point Integration, Hub and Spoke Integration, Boundaryless Information Flow, Integrated Information Infrastructure Reference Model
   5. **Design For Applications Security** – Identification, Authentication, 3 Factor Authentication (Are, Have, Know), Authorisation, Access

**Foundation Learning Objectives**Distinguish three kinds of application: user application, infrastructure application, platform application.   
Recognise the main purposes of Enterprise Resource Planning (ERP) and Customer Relationship Management (CRM) systems.  
  
**Intermediate Learning Objectives (4 questions)  
Applications architecture structure and behaviour.**Recognise the goals and concepts of application portfolio management.   
Distinguish structural and behavioural models of applications architecture.   
Recognise a variety of approaches to application integration: screen scrapers, ETL, application consolidation.   
Distinguish point-to-point from hub and spoke application integration.   
Recognise the TOGAF concepts of Boundaryless Information Flow and Integrated Information Infrastructure Reference Model (III-RM).   
**Design for applications security**Distinguish the steps in applications security: identification, authentication, authorisation and access.   
Recognise the concept of three-factor authentication.   
Recognise different security controls needed in different layers of an enterprise application.   
**Application platform**Recognise the purposes of application platform technologies: notably component distribution middleware, database and transaction middleware (remote database access, transaction processing and distributed transaction manager).  
  
**Practitioner Learning Objectives (6 questions)**Discuss the practical use of the concepts above and apply them a case study.

1. **Design For Non Functional Performance**
   1. Database Optimisation, Normalisation, Index, Access Path Analysis, Caching, Scale Up, Scale Out (Clustering), Design for Resilience (Availability and Reliability), Fail Over, Defensive Design, Design for recoverability, Back Up,
   2. Back Up Site, Design For Integrity, Design For Serviceability, Design For Security, ISO/IEC 17799 – Code of Practice for IT Security, ISO/IEC 24762:2008 Techniques for IT Security – Guidelines For Information, ISO/IEC 27001 - Techniques for IT Security – Guidelines For Requirements

**Foundation Learning Objectives**None.   
  
**Intermediate Learning Objectives (3 questions)**Recognise four designs for performance techniques: Database optimization (normalisation, denormalisation, index, and access path analysis), cache, scale up, scale out (aka clustering).   
Recognise two design for resilience techniques: Fail over, Defensive design.   
Recognise two design for recoverability techniques: Back up, Backup site.   
Recognise two design for integrity techniques.   
Recognise two design for serviceability techniques.   
Recognise design for security techniques mentioned in other sections of the syllabus and identify relevant standards (ISO/IEC 27001, ISO/IEC 17799).

**Practitioner Learning Objectives (6 questions)**Discuss the practical use of the concepts above and refer to them during a case study.

1. **Infrastructure Architecture**
   1. ***Basic Infrastructure Components*** *– Node, Computer, Processor (CPU), Operating System*
   2. ***Network Scopes*** *– Network, PAN/LAN/MAN/WAN, VPN, Cloud Computing, Topology, Topology Shape (PtP, Bus, Hub, Ring)*
   3. ***Network Layers*** *– 5 – Application/Component Connection, 4 – Transport, 3- Network Level, 2 – Data Transport, 1 – Physical Level*
   4. ***Network Protocols*** *– Protocol, Protocol Stack – OSI 7 Layer (7-Applicatiion 6-Presentation 5- Session 4-Transport 3-Network 2-DataLink 1-Physical, TCP/IP 5 layer (5-Aplication, 4- Transport 3-Network/Internet 2-Data Link 1-Physical*
   5. ***The Internet*** *– IP, Subnet, Convergence, VOIP*
   6. **Infrastructure Services and Compo**nents – Infrastructure Service, Platform Service, Technology Catalogue or Portfolio
   7. **Enterprise Technology Rationalisation** - Enterprise technology Classification Technical Reference Model (TRM – TOGAF), Virtual Machine, Service Consolidation
   8. **Solution Technology Definition** – (Clarify precursors, Establish baseline and constraints, Define client end devices, Define data servers, Define intermediate layers, Map software layers to platform and hardware tiers, Define the network, Handle Non-functional, Define the environment strategy)
   9. **Connection Applications To Networks** – Network Address, MAC Address, IP Address, Port, Service Type, Port, Socket, Active Process
   10. **Design For Infrastructure Security** – Client Security, Server Security, Firewall, DMZ, HTTPS, Web Site Security, Public Key Certificate, Certificate Authority

**Foundation Learning Objectives   
Computers**Recognise the basic units of computer network: computer, processor, operating system, peripheral, router, bridge.   
**Connecting computers to networks**Distinguish physical and logical network addresses: MAC address, IP address.   
Recognise the use by a process of a socket of a port to send and receive data of a service type.   
**Topologies**Distinguish four topology shapes: hub and spoke, point to point, bus and ring. Recognise these terms have different meanings in different areas of computing.   
Recognise different shapes may apply at different topology levels.   
**Networks and protocols**Distinguish four levels of network geography: PAN, LAN, MAN, and WAN.   
Recognise the convergence of telecommunications media, and Voice Over IP.   
Distinguish the seven layers of the OSI model of communication protocols.   
Distinguish the TCP/IP 5 layer stack from the OSI 7 layer model. Recognise the protocols used in the web services stack. Recognise the division of an IP address by a network administrator to identify a subnet.   
  
**Intermediate Learning Objectives (5 questions)  
Infrastructure architecture structure and behaviour**Recognise why TOGAF recommends listing platform services under a logical hierarchical structure, known as a Technical Reference Model.   
Recognise the building blocks commonly shown in a hardware configuration diagram, and the process of infrastructure architecture design.   
Recognise the concepts of virtualisation and server consolidation.   
**Design for infrastructure security**Recognise techniques for infrastructure security used to protect client devices, web sites and services, including https, firewalls and a De-Militarised Zone (DMZ).   
  
**Practitioner Learning Objectives (4 questions)**Discuss the practical use of the concepts above and apply them a case study.

1. **Migration Planning**
   1. *Foundation and Intermediate – Business Case, ROI, Cost Benefit Analysis, Solution Options, Risk Analysis, Gap Analysis, Trade Off Analysis, Business Scenario*
   2. Practitioner – Gap Analysis, Migration Path/Plan, Roadmap, Critical Path Analysis, PERT, RAID Catalogue, Risk, Assumption, Issue, Dependency (risk sense) Management Methodology, Programme, MSP, Project, Prince 2

**Foundation and Intermediate Learning Objectives**None.   
  
**Practitioner Learning Objectives (2 questions)**Recognise the use of analysis techniques such as gap analysis, and critical path analysis in defining a migration path. Recognise the formation of a roadmap with time, costs and resources, from a migration path.   
Recognise the need to maintain a RAID catalogue and work alongside managers using methods for programme and project management.

1. **Architecture Management**
   1. Architecture Implementation – SDLC, Waterfall, Iterative/Incremental, DSDM, Agile, Transition, ISO9001 (Quality Management Std)
   2. Architecture Change Management
   3. Baseline Configuration, Configuration Item, Agile, Change Management, Change Control, Request fir Change, Impact Analysis, Configuration Management
   4. Architecture Governance – Architecture Board, Architecture Contract, Governing Architect, Compliance Review, Review Checklist, Conformance Level, Compliance Level, Dispensation, Capability Maturity Model
   5. Architecture In Operations - COBIT, IT Service, IT Service Management, ITIL. ISO/IEC 20000Service Management Std, CMDB, Asset Management System, Common Information Model (CIM)

## Foundation and Intermediate Learning Objectives None.

**Practitioner Learning Objectives (4 questions)  
Architecture implementation**Distinguish three varieties of Software Development Life Cycle (SDLC): Waterfall, Iterative Development and Agile Development.  
**Architecture change management**Recognise the need for and concepts used in architecture change management: baseline configuration, configuration item, change management, change control, request for change (rfc), impact analysis, configuration management.   
**Architecture governance**Recognise the need for and concepts used in architecture governance: architecture board, architecture contract, governing architect, architecture compliance review, architecture review checklist, architecture conformance level, architecture compliance level, dispensation, capability maturity model.   
**Architecture in operations**Recognise approaches used to manage the architecture in operations: COBIT, IT services management (ITSM), Information technology Infrastructure Library (ITIL), IT Configuration Management Database (CMDB), Common Information Model (CIM), Problem and help-desk management. Systems management.

1. **Enterprise Technology Classification (Not Examinable)**
   1. Concepts that may be used – Client/user access component, Client Device, Peripheral, Generic User Application, Application Platform, FTTP Server, App Server, Transaction Manager, Distributed Transaction Manager, Web Services Stack, Remote Database Access, Software Development, Screen Scrapper, Integration Tools, Middleware, Point to Point Integration Tool, RPC, Web Services (SOAP / IIOP / HTTP/HTTPS / SMTP), ORB, CORBA, MQ, DBMS, Data Warehouse, Router/Hub/Switch