



## PROVINCIA DI MANTOVA

**COMMISSIONE GIUDICATRICE DEL CONCORSO PUBBLICO PER ESAMI PER LA COPERTURA DI N° 1 POSTO DI ISTRUTTORE DIRETTIVO TECNICO, CAT D, A TEMPO PIENO E INDETERMINATO DA ASSEGNARE AL SERVIZIO SISTEMI INFORMATIVI, INNOVAZIONE E SVILUPPO**

### VERBALE N.1

Il 13 maggio 2019, alle ore 11,00, presso la sede della Provincia di Mantova, via Principe Amedeo n.32, Mantova, si è riunita la Commissione giudicatrice del concorso pubblico, per esami, per la copertura di n.1 posto di Istruttore Direttivo Tecnico a tempo pieno e indeterminato da assegnare al Servizio Sistemi Informativi Innovazione Sviluppo.

Sono presenti i componenti della Commissione giudicatrice del concorso pubblico di cui all'oggetto nominata con determinazione n. 295 del 24/04/2019, nelle persone dei Signori:

- |                   |   |
|-------------------|---|
| - FARONI Barbara  | Dirigente dell'Area 1 – Funzioni delegate,<br>Istruzione, Edilizia scolastica<br>Provincia di Mantova<br>- PRESIDENTE DELLA COMMISSIONE |
| - ZANIBONI Matteo | Funzionario tecnico<br>Provincia di Mantova<br>- MEMBRO ESPERTO   |
| - CAPUCETTI Anna  | Istruttore direttivo amministrativo<br>Provincia di Mantova<br>- MEMBRO ESPERTO   |

Assiste con funzioni di Segretario verbalizzante il Dott. TELLINI Federico, istruttore direttivo amministrativo Provincia di Mantova.

Il Presidente dichiara aperta la seduta e preso atto della regolare costituzione della Commissione dichiara quindi insediata la Commissione stessa che inizia le operazioni di espletamento del concorso in oggetto.

La Commissione procede a prendere visione dei seguenti atti:

- Determinazione dirigenziale n.111 del 21 febbraio 2019 con la quale è stato indetto un concorso pubblico, per esami, per la copertura di n.1 posto di istruttore direttivo tecnico categoria D, a tempo pieno e indeterminato da assegnare al Servizio Sistemi Informativi Innovazione Sviluppo (Area Segreteria Generale);
- Bando del concorso pubblico per la copertura di n.1 posto di istruttore direttivo tecnico (Cat.D) a tempo pieno e indeterminato da assegnare al Servizio Sistemi Informativi Innovazione Sviluppo;

- Determinazione n.297 del 24.04.2019 con la quale l'Amministrazione ha provveduto ad approvare l'elenco dei partecipanti ammessi al concorso;

Il presidente e i componenti, unitamente al Segretario, dopo aver preso visione dell'elenco nominativo dei partecipanti (allegato "A" al presente verbale), dichiarano che non sussistono cause di incompatibilità ai sensi di legge nei confronti di nessun partecipante.

La Commissione, preso in esame il bando di concorso, rileva che:

A) il bando è stato pubblicato integralmente all'Albo Pretorio e nel sito internet della Provincia di Mantova dal 12 marzo 2019 alla data di scadenza; il bando è stato altresì pubblicato in estratto sulla Gazzetta Ufficiale della Repubblica Italiana, 4<sup>a</sup> serie Speciale Concorsi ed Esami n.20 del 12 marzo 2019 e sul quotidiano "Gazzetta di Mantova" del 17 marzo 2019;

B) il termine per la presentazione della domanda è scaduto l'11 aprile 2019;

C) il programma d'esame prevede le seguenti materie:

Analisi delle attività sistemistiche per la gestione di un data center/ Gestione sistemistica server;  
 Architetture di rete interna ed esterna di media complessità aziendale, con particolare riferimento alla gestione di ambienti server farm;  
 Gestione della sicurezza dei dati, sotto diversi aspetti: sicurezza informatica, normativa sulla privacy ed in particolare sulla tematica della continuità operativa, sistemi di backup e tecniche di disaster recovery;  
 Integrazione, gestione e manutenzione dei principali applicativi software;  
 Individuazione, risoluzione delle problematiche hardware, software e di rete delle postazioni di lavoro;  
 Sistemi di posta elettronica, PEC e firma digitale e problematiche connesse;  
 Analisi di fasi progettuali connesse all'implementazione, miglioramento della digitalizzazione dei flussi documentali ed alla reingegnerizzazione dei processi della Pubblica amministrazione in un'ottica di integrabilità delle fasi operative della vita amministrativa dell'ente;  
 Tecniche di sviluppo e definizione di un sistema SIT;  
 Tecniche di integrazione banche dati;  
 Tecniche di sviluppo software Web e Mobile;  
 Sistemi Operativi Windows, Linux, VMWare;  
 Conoscenze di gestione di reti / apparati di rete / firewall;  
 Gestione database / Database Oracle PostgreSQL;  
 Conoscenza Linguaggi di Programmazione Java, HTML, XML, SQL;  
 Metodologie di sviluppo del software;  
 Ordinamento istituzionale, finanziario e contabile degli Enti locali (D.lgs. 267/2000 e s.m.i.);  
 Codice dell'amministrazione digitale, obiettivi fissati da AGID inerenti l'ente locale;  
 Disposizioni in materia di Amministrazione Trasparente e di Privacy e relativi adempimenti;  
 Attività contrattuale della Pubblica Amministrazione.

Il bando di concorso prevede che le prove d'esame consistano in due prove scritte e in una prova orale. In particolare:

### PRIMA PROVA SCRITTA

La prima prova scritta potrà consistere nella redazione di un elaborato o in domande a risposta sintetica su argomenti inerenti le materie oggetto del programma di esame

### SECONDA PROVA SCRITTA

La seconda prova scritta potrà consistere nella soluzione di casi concreti di lavoro e/o nella stesura di un atto amministrativo e/o tecnico riferito a un caso concreto, vertenti sulle materie del programma d'esame.

### PROVA ORALE si articolerà in:

- a) un colloquio sui temi oggetto del programma di esame;
- b) una verifica della capacità di utilizzo delle apparecchiature e delle applicazioni informatiche più diffuse (Word, Excel, Access, Posta elettronica, Internet);
- c) una verifica del livello di conoscenza della lingua INGLESE.

La Commissione, prende atto che il bando di concorso prevede che:

- 1) Sono ammessi a sostenere la prova orale solo i candidati che conseguano in ciascuna prova scritta una votazione di almeno punti 21/30.
- 2) Avranno superato positivamente la prova orale e potranno quindi essere inseriti nella graduatoria di merito solo i candidati che nella prova conseguano almeno punti 21/30, a condizione che abbiano conseguito almeno la sufficienza sia nell'accertamento delle conoscenze informatiche che nell'accertamento della conoscenza della lingua inglese.
- 3) Il punteggio finale è determinato dalla somma delle votazioni conseguite nelle due prove scritte e della votazione conseguita nella prova orale.

In caso di parità di merito si rimanda alle disposizioni vigenti.

La Commissione Giudicatrice predetermina come segue i criteri di massima per la valutazione delle prove d'esame:

- conoscenza delle materie oggetto d'esame e del quadro normativo di riferimento;
- capacità di analisi della situazione prospettata e del contesto;
- capacità di risolvere casi concreti;
- capacità di sintesi, intesa come capacità di organizzare le informazioni in maniera succinta, completa ed efficace;
- chiarezza e organicità espositiva (capacità di rendere evidente il percorso logico seguito);
- proprietà nell'uso del linguaggio tecnico-giuridico;
- correttezza formale.

Quindi la Commissione prende atto del seguente calendario delle prove:

**prove scritte:** 21 maggio 2019 – ore 9 – (1<sup>a</sup> prova) – ore 14 - (2<sup>a</sup> prova) presso Provincia di Mantova, Via Principe Amedeo n.32 Mantova, Sala Consiliare;

**prova orale:** 30 maggio 2019 – ore 9 – presso Provincia di Mantova, Via Principe Amedeo n.32 Mantova, Sala Orlandini.

Completate le operazioni di cui sopra, la Commissione alle ore 12,15 sospende i lavori e si aggiorna.

Letto, confermato e sottoscritto.

IL PRESIDENTE

F.to Barbara Faroni

I MEMBRI ESPERTI

F.to Roberto Piccinini

F.to Anna Capucetti

IL SEGRETARIO  
DELLA COMMISSIONE

F.to Federico Tellini



## PROVINCIA DI MANTOVA

**COMMISSIONE GIUDICATRICE DEL CONCORSO PUBBLICO PER ESAMI PER LA COPERTURA DI N° 1 POSTO DI ISTRUTTORE DIRETTIVO TECNICO, CAT D, A TEMPO PIENO E INDETERMINATO DA ASSEGNARE AL SERVIZIO SISTEMI INFORMATIVI, INNOVAZIONE E SVILUPPO**

### VERBALE N.2

Il 21 maggio 2019, alle ore 8,45, presso la sede della Provincia di Mantova, via Principe Amedeo n.32, Mantova, si è riunita la Commissione giudicatrice del concorso pubblico, per esami, per la copertura di n.1 posto di Istruttore Direttivo Tecnico a tempo pieno e indeterminato da assegnare al Servizio Sistemi Informativi Innovazione Sviluppo.

Sono presenti i componenti della Commissione giudicatrice del concorso pubblico di cui all'oggetto nominata con determinazione n. 295 del 24/04/2019, nelle persone dei Signori:

- |                   |   |
|-------------------|---|
| - FARONI Barbara  | Dirigente dell'Area 1 – Funzioni delegate,<br>Istruzione, Edilizia scolastica<br>Provincia di Mantova<br>- PRESIDENTE DELLA COMMISSIONE |
| - ZANIBONI Matteo | Funzionario tecnico<br>Provincia di Mantova<br>- MEMBRO ESPERTO   |
| - CAPUCETTI Anna  | Istruttore direttivo amministrativo<br>Provincia di Mantova<br>- MEMBRO ESPERTO   |

Assiste con funzioni di Segretario verbalizzante il Dott. TELLINI Federico, istruttore direttivo amministrativo Provincia di Mantova.

Il Presidente alle ore 8.45, preso atto della regolare costituzione della Commissione, dichiara aperta la seduta e quindi insediata la Commissione stessa.

La Commissione procede, all'unanimità, alla definizione e alla scelta delle tre tracce per ciascuna delle prove scritte.

Le tracce scelte dalla Commissione sono le seguenti:

#### **PRIMA PROVA SCRITTA**

**PRIMA PROVA SCRITTA - DOMANDE A RISPOSTA SINTETICA**

## Traccia 1

N	DOMANDA
1	La posta Elettronica Certificata (PEC).
2	Cosa si intende per Dati Aperti (Open Data), con riferimento a quanto previsto dal Codice dell'Amministrazione Digitale (CAD – D.Lgs. 82/2005).
3	Relativamente alla tenuta di un inventario dei dispositivi autorizzati e non autorizzati nella rete di un sistema Informativo, indicare sinteticamente le misure di sicurezza minime, standard e ottimali da adottare. E' possibile fare ad esempio riferimento alla circolare dell'AGID del 18 aprile 2017, n. 2/2017, recante "Misure minime di sicurezza ICT per le pubbliche amministrazioni" in attuazione della Direttiva 1 agosto 2015 del Presidente del Consiglio dei Ministri.
4	Spiegare brevemente cos'è in ambito informatico un DNS.
5	Le fasi gestionali della spesa nell'ordinamento contabile degli Enti Locali.

## PRIMA PROVA SCRITTA - DOMANDE A RISPOSTA SINTETICA

## Traccia 2

N	DOMANDA
1	La Firma digitale.
2	Differenze tra protocolli informatici IMAP e POP.
3	Relativamente alla tenuta di un inventario dei software autorizzati e non autorizzati nella rete di un sistema Informativo, indicare sinteticamente le misure di sicurezza minime, standard e ottimali da adottare. E' possibile fare ad esempio riferimento alla circolare dell'AGID del 18 aprile 2017, n. 2/2017, recante "Misure minime di sicurezza ICT per le pubbliche amministrazioni" in attuazione della Direttiva 1 agosto 2015 del Presidente del Consiglio dei Ministri.
4	Spiegare brevemente cos'è in informatica un servizio DHCP.
5	Le procedure di scelta del contraente nella Pubblica Amministrazione previste dal Codice dei contratti Pubblici (D.Lgs. 50/2016).

**PRIMA PROVA SCRITTA - DOMANDE A RISPOSTA SINTETICA**

**Traccia 3**

N	DOMANDA
1	Strumenti per l'accesso sicuro alla navigazione su Internet.
2	Cosa si intende in informatica per RAID e quali sono le modalità più diffuse.
3	Relativamente alle difese contro i malware nella rete di un sistema Informativo, indicare sinteticamente le misure di sicurezza minime, standard e ottimali da adottare. E' possibile fare ad esempio riferimento alla circolare dell'AGID del 18 aprile 2017, n. 2/2017, recante "Misure minime di sicurezza ICT per le pubbliche amministrazioni" in attuazione della Direttiva 1 agosto 2015 del Presidente del Consiglio dei Ministri.
4	Spiegare brevemente le funzioni di un Server di Dominio.
5	Illustrare i principali obblighi di pubblicità e trasparenza previsti per le Pubbliche Amministrazioni (D.Lgs 33/2013).

**SECONDA PROVA SCRITTA**

PROVA SCRITTA DI CONTENUTO TECNICO-PRATICO  
TRACCIA 1

Progettare la struttura della rete di un sistema informativo complesso distribuito su una sede centrale, due sedi periferiche nello stesso ambito urbano e 3 sedi remote a distanze superiori a 20 km. Dare indicazioni in merito all'hardware, alla connettività e alle configurazioni ottimali.

PROVA SCRITTA DI CONTENUTO TECNICO-PRATICO  
TRACCIA 2

Progettare le funzionalità e la struttura di un sistema informativo per la gestione delle risorse umane di un ente locale.

PROVA SCRITTA DI CONTENUTO TECNICO-PRATICO  
TRACCIA 3

Sicurezza informatica e privacy. Aspetti normativi, misure organizzative e soluzioni tecniche.



## PROVINCIA DI MANTOVA

**COMMISSIONE GIUDICATRICE DEL CONCORSO PUBBLICO PER ESAMI PER LA COPERTURA DI N° 1 POSTO DI ISTRUTTORE DIRETTIVO TECNICO, CAT D, A TEMPO PIENO E INDETERMINATO DA ASSEGNARE AL SERVIZIO SISTEMI INFORMATIVI, INNOVAZIONE E SVILUPPO**

### VERBALE N. 5

Il 30 maggio 2019, alle ore 8,55, presso la sede della Provincia di Mantova, via Principe Amedeo n.32, Mantova, si è riunita la Commissione giudicatrice del concorso pubblico, per esami, per la copertura di n.1 posto di Istruttore Direttivo Tecnico a tempo pieno e indeterminato da assegnare al Servizio Sistemi Informativi Innovazione Sviluppo.

Sono presenti i componenti della Commissione giudicatrice del concorso pubblico di cui all'oggetto nominata con determinazione n. 295 del 24/04/2019, nelle persone dei Signori:

Sono presenti i Sigg.:

- FARONI Barbara

Dirigente dell'Area 1 – Funzioni delegate,  
Istruzione, Edilizia scolastica  
Provincia di Mantova  
- PRESIDENTE DELLA COMMISSIONE

- ZANIBONI Matteo

Funzionario tecnico  
Provincia di Mantova  
- MEMBRO ESPERTO

- CAPUCETTI Anna

Istruttore direttivo amministrativo  
Provincia di Mantova  
- MEMBRO ESPERTO

E' altresì presente il Dott. PICCININI Roberto come membro aggregato per la prova in lingua inglese.

Assiste con funzioni di Segretario verbalizzante il Dott. TELLINI Federico, istruttore direttivo amministrativo Provincia di Mantova.

La Commissione, constatata la regolare costituzione della Commissione, dichiara riaperti i lavori ed inizia subito le operazioni di espletamento della prova orale.

La Commissione prende atto che sulla base dei verbali precedenti sono stati ammessi a sostenere la prova orale i seguenti candidati:

OMISSIS



La Commissione decide che a ciascun candidato verranno poste:

- quattro domande sulle materie oggetto del programma di esame previste dal Bando di Concorso;
- per la prova di informatica un quesito/problema da risolvere con l'utilizzo delle apparecchiature e applicazioni informatiche;
- per l'accertamento della conoscenza della lingua inglese un testo redatto in lingua inglese da leggere seguito da un breve colloquio in inglese.

Le domande predisposte dalla Commissione per la prova orale sono le seguenti:

### PROVA ORALE - TRACCIA 1

*Domande sul programma d'esame*

1. *Cos'è Active directory: struttura e servizi.*
2. *Cos'è il RAID? Spiegare le differenze fra il livello 0, 1, 5.*
3. *Metodologie di backup.*
4. *Il Protocollo informatico.*

*Prova Informatica*

*Rappresentare in forma tabellare in una sola tabella l'andamento della spesa nel triennio 2016- 2018 per RESPONSABILE DI PROCEDURA e per TITOLO DI SPESA*

*Rappresentare in forma grafica separatamente l'andamento della spesa nel triennio 2016-2018 per ogni RESPONSABILE DI PROCEDURA e per ciascun TITOLO DI SPESA*

*Prova di inglese*

*Brano in inglese tratto da un saggio di informatica (Software Architecture in Practice) (vedi allegato).*

---

### PROVA ORALE - TRACCIA 2

*Domande sul programma d'esame*

1. *Disaster recovery / Business Continuity.*
2. *Cosa sono OSPF e spanning tree?*
3. *Vincoli d'integrità in un database.*
4. *Le determinazioni dirigenziali.*

*Prova Informatica*

Conoscendo la data d'inizio, la data di conclusione e i giorni di sospensione di un procedimento calcolare il tempo medio per procedimento e darne rappresentazione grafica.

*Prova di inglese*

Brano tratto da un saggio di informatica in lingua inglese (Software Architecture in Practice) (v. allegato).

### PROVA ORALE - TRACCIA 3

*Domande sul programma d'esame*

1. Le principali component di una rete LAN / WAN.
2. La Java Virtual Machine.
3. Cosa sono le Viste su un Database e quali caratteristiche possono avere.
4. Il Piano Esecutivo di Gestione (PEG)

*Prova Informatica*

Calcolare e rappresentare graficamente il consumo medio dell'automobile in litri / km per anno per valutarlo in base all'invecchiamento dell'auto.

Calcolare e rappresentare graficamente il consumo medio dell'automobile in litri / km per mese (indipendentemente dall'anno) per valutarlo in base alla stagionalità.

*Prova di inglese*

Brano tratto da un saggio di informatica in lingua inglese (Software Architecture in Practice) (v. allegato).

### PROVA ORALE - TRACCIA 4

*Domande sul programma d'esame*

1. Principali Tipologie di Minacce Informatiche e contromisure.
2. Descrivere una tipica struttura di server per erogare un applicativo via web.
3. La gestione delle ridondanze in un database.
4. La conservazione dei documenti.

*Prova Informatica*

Nella tabella sono presenti i seguenti dati:

Matricola: matricola di un dipendete

Categoria: Livello economico del dipendente

Attività: Classificazione dell'attività svolta

% Utilizzo = percentuale del tempo lavorato sulla specifica attività

Lavorato: Ore e minuti TOTALI lavorati nel 2018

Produrre un riepilogo in cui sia riportato per ogni categoria e ogni attività la somma di ore e minuti lavorati dal totale dei dipendenti.

*Prova di inglese*

*Brano tratto da un saggio di informatica in lingua inglese (Software Architecture in Practice) (v. allegato).*



## 1.1. What Software Architecture Is and What It Isn't

There are many definitions of software architecture, easily discoverable with a web search, but the one we like is this one:

The software architecture of a system is the set of structures needed to reason about the system, which comprise software elements, relations among them, and properties of both.

This definition stands in contrast to other definitions that talk about the system's "early" or "major" design decisions. While it is true that many architectural decisions are made early, not all are—especially in Agile or spiral-development projects. It's also true that very many decisions are made early that are not architectural. Also, it's hard to look at a decision and tell whether or not it's "major." Sometimes only time will tell. And since writing down an architecture is one of the architect's most important obligations, we need to know now which decisions an architecture comprises.

Structures, on the other hand, are fairly easy to identify in software, and they form a powerful tool for system design.

Let us look at some of the implications of our definition.

### Architecture Is a Set of Software Structures

This is the first and most obvious implication of our definition. A structure is simply a set of elements held together by a relation. Software systems are composed of many structures, and no single structure holds claim to being *the* architecture. There are three categories of architectural structures, which will play an important role in the design, documentation, and analysis of architectures:

1. First, some structures partition systems into implementation units, which in this book we call *modules*. Modules are assigned specific computational responsibilities, and are the basis of work assignments for programming teams (Team A works on the database, Team B works on the business rules, Team C works on the user interface, etc.). In large projects, these elements (modules) are subdivided for assignment to subteams. For example, the database for a large enterprise resource planning (ERP) implementation might be so complex that its implementation is split into many parts. The structure that captures that decomposition is a kind of module structure, the *module decomposition structure* in fact. Another kind of module structure emerges as an output of object-oriented analysis and design—class diagrams. If you aggregate your modules into layers, you've created another (and very useful) module structure. Module structures are static structures, in that they focus on the way the system's functionality is divided up and assigned to implementation teams.
2. Other structures are dynamic, meaning that they focus on the way the elements interact with each other at runtime to carry out the system's functions. Suppose the system is to be built as a set of services. The services, the infrastructure they interact with, and the synchronization and interaction relations among them form another kind of structure often used to describe a system. These services are made up of (compiled from) the programs in the various implementation units that we just described. In this book we will call runtime structures *component-and-connector* (C&C) structures. The term *component* is overloaded in software



Aa

### 1.3. Architectural Patterns

In some cases, architectural elements are composed in ways that solve particular problems. The compositions have been found useful over time, and over many different domains, and so they have been documented and disseminated. These compositions of architectural elements, called *architectural patterns*, provide packaged strategies for solving some of the problems facing a system.

An architectural pattern delineates the element types and their forms of interaction used in solving the problem. Patterns can be characterized according to the type of architectural elements they use. For example, a common module type pattern is this:

- *Layered pattern*. When the *uses* relation among software elements is strictly unidirectional, a system of layers emerges. A layer is a coherent set of related functionality. In a *strictly* layered structure, a layer can only use the services of the layer immediately below it. Many variations of this pattern, lessening the structural restriction, occur in practice. Layers are often designed as abstractions (virtual machines) that hide implementation specifics below from the layers above, engendering portability.

Common component-and-connector type patterns are these:

- *Shared-data (or repository) pattern*. This pattern comprises components and connectors that create, store, and access persistent data. The repository usually takes the form of a (commercial) database. The connectors are protocols for managing the data, such as SQL.
- *Client-server pattern*. The components are the clients and the servers, and the connectors are protocols and messages they share among each other to carry out the system's work.

Common allocation patterns include the following:

- *Multi-tier pattern*, which describes how to distribute and allocate the components of a system in distinct subsets of hardware and software, connected by some communication medium. This pattern specializes the generic deployment (software-to-hardware allocation) structure.
- *Competence center* and *platform*, which are patterns that specialize a software system's work assignment structure. In *competence center*, work is allocated to sites depending on the technical or domain expertise located at a site. For example, user-interface design is done at a site where usability engineering experts are located. In *platform*, one site is tasked with developing reusable core assets of a software product line (see [Chapter 25](#)), and other sites develop applications that use the core assets.

Architectural patterns will be investigated much further in [Chapter 13](#).



## 1.5. Summary

The software architecture of a system is the set of structures needed to reason about the system, which comprise software elements, relations among them, and properties of both.

A structure is a set of elements and the relations among them.

A view is a representation of a coherent set of architectural elements, as written by and read by system stakeholders. A view is a representation of one or more structures.

There are three categories of structures:

- Module structures show how a system is to be structured as a set of code or data units that have to be constructed or procured.
- Component-and-connector structures show how the system is to be structured as a set of elements that have runtime behavior (components) and interactions (connectors).
- Allocation structures show how the system will relate to nonsoftware structures in its environment (such as CPUs, file systems, networks, development teams, etc.).

Structures represent the primary engineering leverage points of an architecture. Each structure brings with it the power to manipulate one or more quality attributes. They represent a powerful approach for creating the architecture (and later, for analyzing it and explaining it to its stakeholders). And as we will see in [Chapter 18](#), the structures that the architect has chosen as engineering leverage points are also the primary candidates to choose as the basis for architecture documentation.

Every system has a software architecture, but this architecture may be documented and disseminated, or it may not be.

There is no such thing as an inherently good or bad architecture. Architectures are either more or less fit for some purpose.



## 2.1. Inhibiting or Enabling a System's Quality Attributes

Whether a system will be able to exhibit its desired (or required) quality attributes is substantially determined by its architecture.

This is such an important message that we've devoted all of [Part 2](#) of this book to expounding that message in detail. Until then, keep these examples in mind as a starting point:

- If your system requires high performance, then you need to pay attention to managing the time-based behavior of elements, their use of shared resources, and the frequency and volume of inter-element communication.
- If modifiability is important, then you need to pay careful attention to assigning responsibilities to elements so that the majority of changes to the system will affect a small number of those elements. (Ideally each change will affect just a single element.)
- If your system must be highly secure, then you need to manage and protect inter-element communication and control which elements are allowed to access which information; you may also need to introduce specialized elements (such as an authorization mechanism) into the architecture.
- If you believe that scalability will be important to the success of your system, then you need to carefully localize the use of resources to facilitate introduction of higher-capacity replacements, and you must avoid hard-coding in resource assumptions or limits.
- If your projects need the ability to deliver incremental subsets of the system, then you must carefully manage intercomponent usage.
- If you want the elements from your system to be reusable in other systems, then you need to restrict inter-element coupling, so that when you extract an element, it does not come out with too many attachments to its current environment to be useful.

The strategies for these and other quality attributes are supremely architectural. But an architecture alone cannot guarantee the functionality or quality required of a system. Poor downstream design or implementation decisions can always undermine an adequate architectural design. As we like to say (*mostly in jest*): The architecture giveth and the implementation taketh away. Decisions at all stages of the life cycle—from architectural design to coding and implementation—affect system quality. Therefore, quality is not completely a function of an architectural design.

A good architecture is necessary, but not sufficient, to ensure quality. Achieving quality attributes must be considered throughout design, implementation, and deployment. No quality attribute is entirely dependent on design, nor is it entirely dependent on implementation or deployment. Satisfactory results are a matter of getting the big picture (architecture) as well as the details (implementation) correct.

For example, modifiability is determined by how functionality is divided and coupled (architectural) and by coding techniques within a module (nonarchitectural). Thus, a system is typically modifiable if changes involve the fewest possible number of distinct elements. In spite of having the ideal architecture, however, it is always possible to make a system difficult to modify by writing obscure, tangled code.