

Vatic Technologies
Architecture Planning for Technology

Infrastructure Architecture

An Overview

Agenda

- Definition
 - What is meant by “Architecture”?

- Benefits
 - Why should “Architecture” be considered?

- Artifacts and Deliverables
 - How do you create an “Architecture”?

- Challenges
 - What is difficult about creating an “Architecture”?

- Discussion
 - Comments and Questions.

Infrastructure Architecture Definition

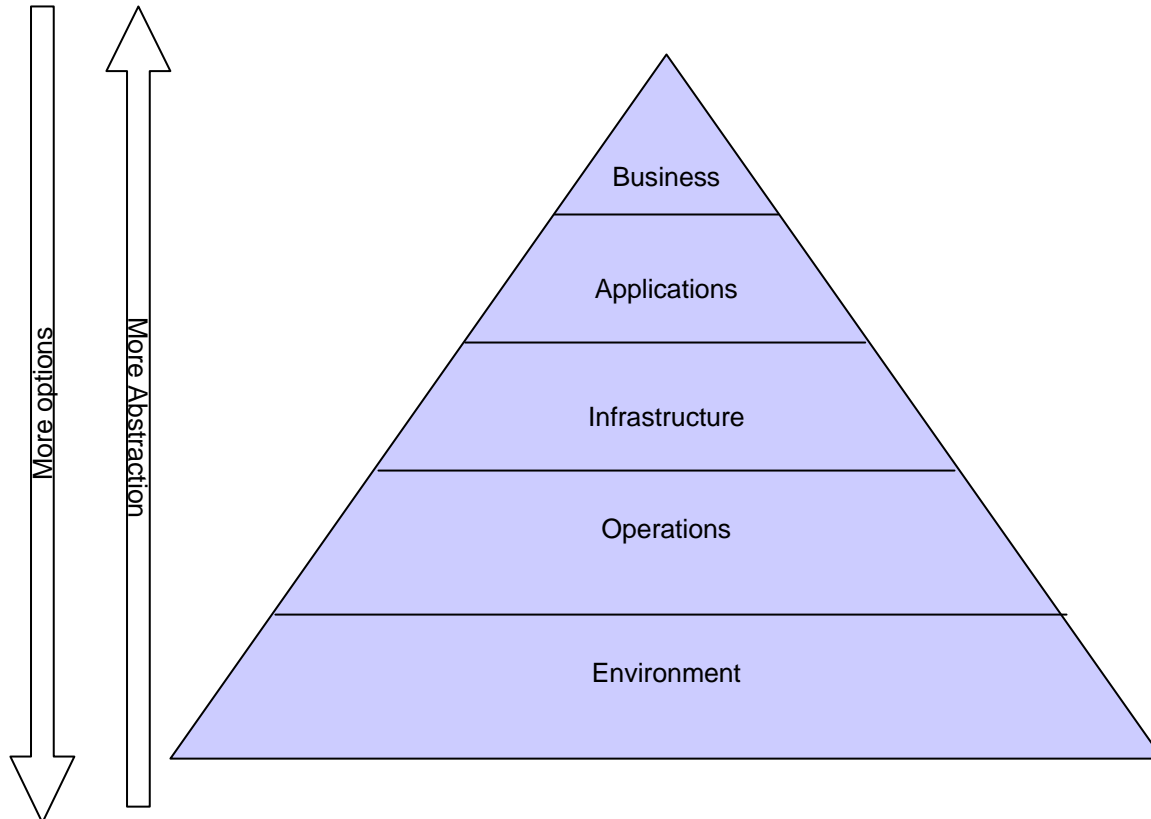
- Architecture is fundamentally about abstraction and re-use.
 - The infrastructure architecture serves to abstract the details of the technology from the application architecture.
 - Re-use is enabled at the appropriate abstraction layer.

- IT Architecture Framework
 - IT in the context of the enterprise.

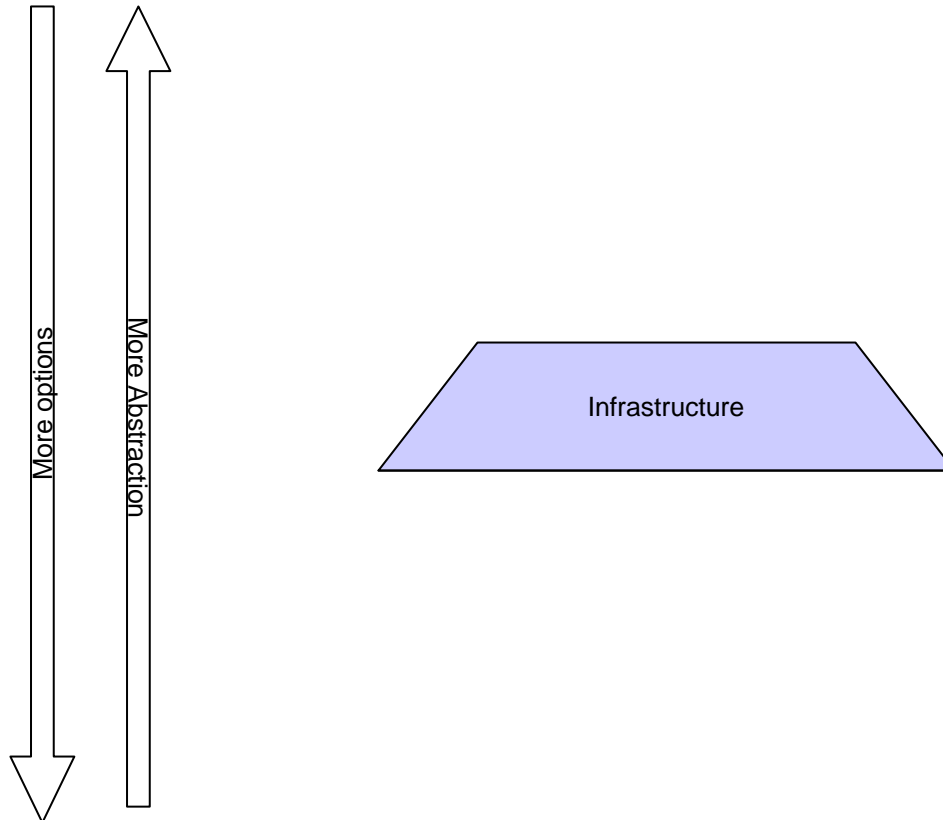
- Infrastructure Architecture Framework
 - Infrastructure in the context of IT

- Infrastructure Architecture Concepts
 - A set of conceptual tools for describing the infrastructure.

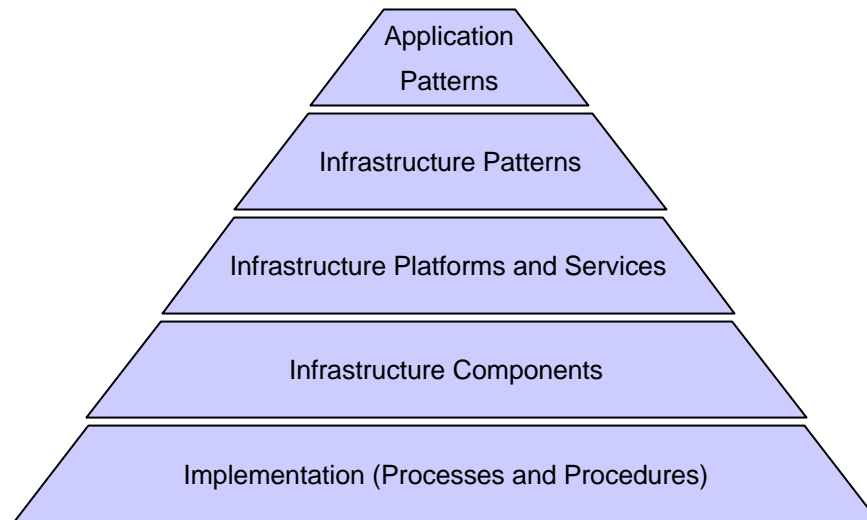
IT Framework (Enterprise)



IT Framework (Infrastructure)

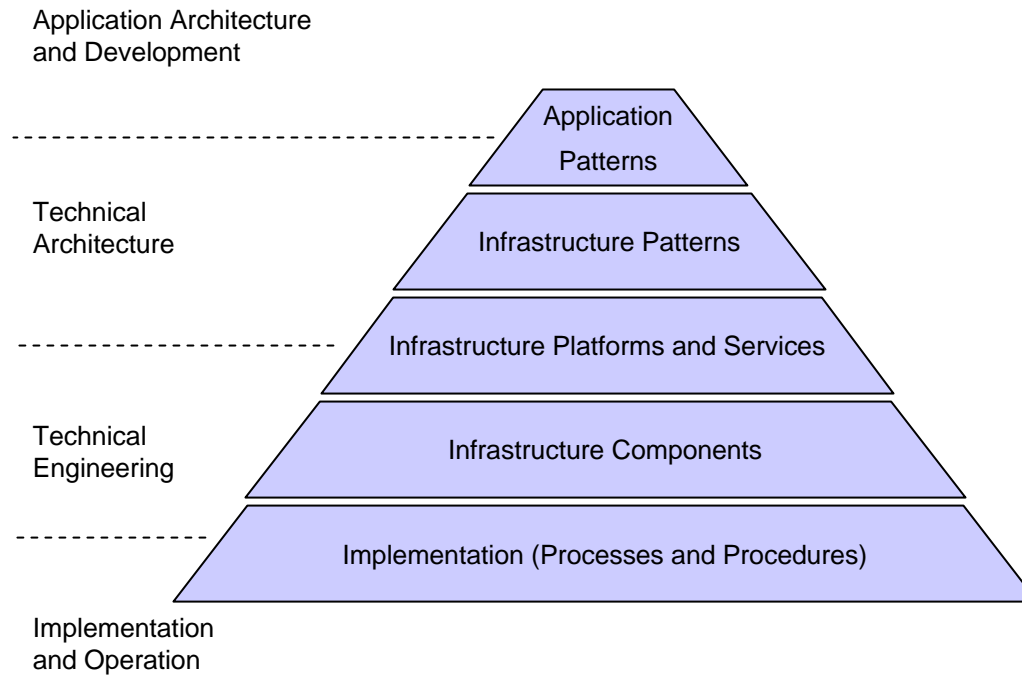


Infrastructure Architecture Framework

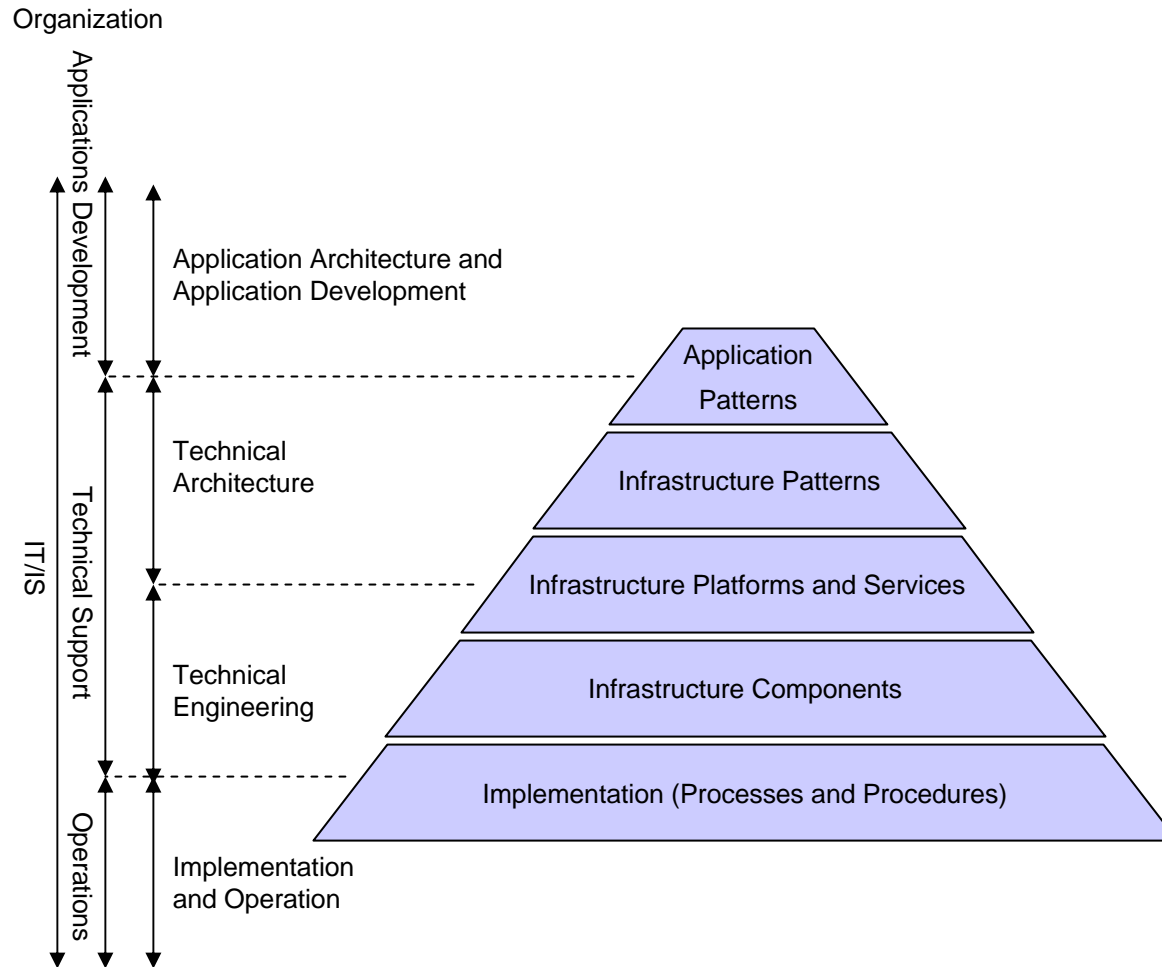


Infrastructure Architecture Framework

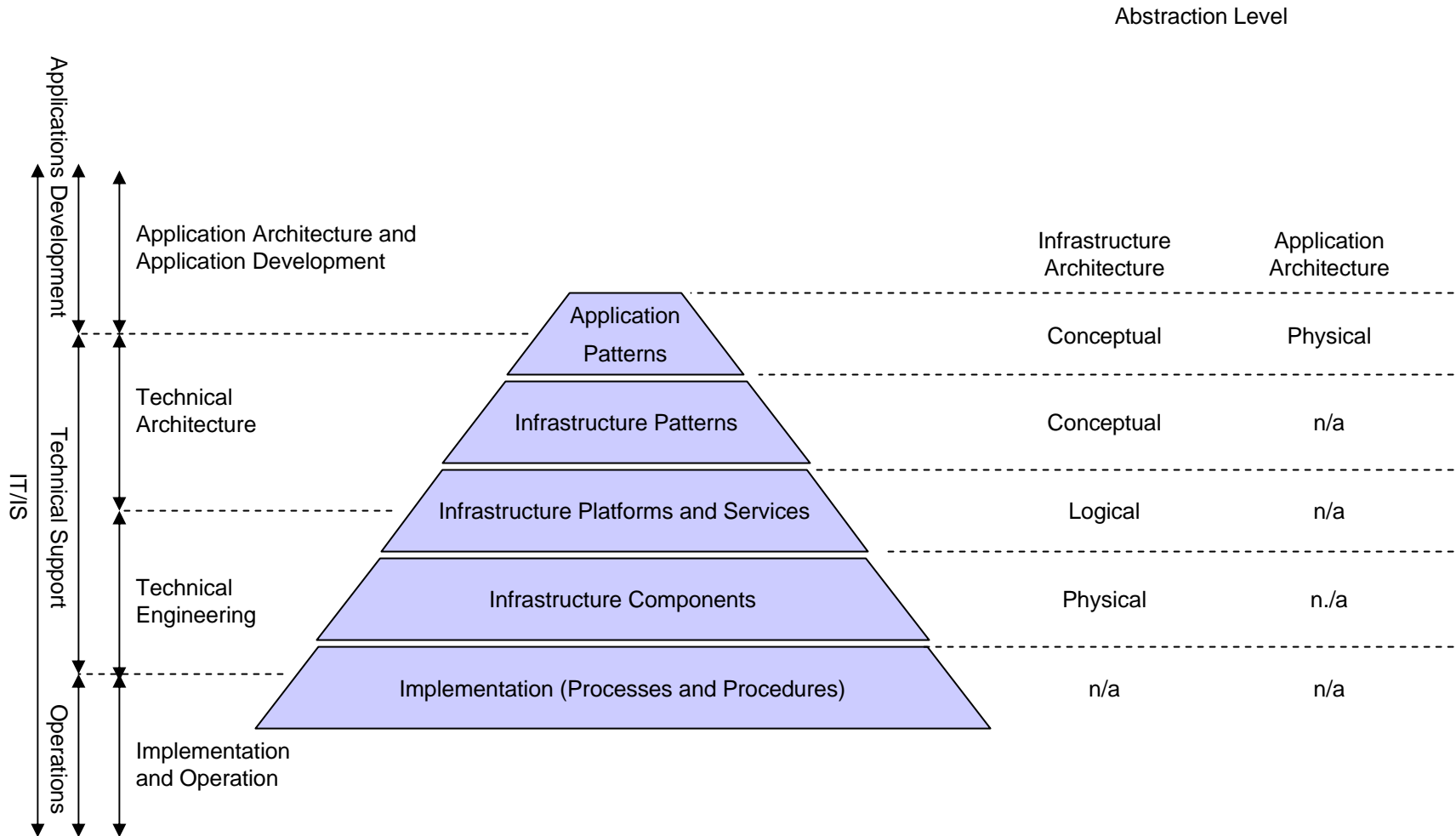
Functions



Infrastructure Architecture Framework



Infrastructure Architecture Framework



Infrastructure Architecture Concepts

Terminology

- **Domains**
 - Classify Platforms, Services, and Components.
 - Provide alignment with Technical Development groups.
 - Represent a particular “view” of a Pattern.
- **Patterns**
 - Provide an “abstracted” view of the infrastructure.
 - Are the primary communications tool for architects.
- **Platforms & Services**
 - Are standard infrastructure building blocks.
- **Components**
 - Are the standard products used to create the building blocks.

Roles

- **Architect**
 - Matches requirements to patterns in a conceptual design.
 - Maintains, documents, and socializes the architecture.
- **Engineer**
 - Creates logical and physical designs that deliver the functions of the conceptual design using standardized building blocks.

Domains

The Computing Domain includes:

- Nodes where code (application or system) is executed
- Software that abstracts the computing system (e.g., WAS, CICS)

The Data Domain includes:

- Nodes where data resides
- Software that manages data (e.g., DBMS)

The Communications Domain includes:

- Links between different nodes
 - Computing –to- Computing
 - Computing –to- Data
 - Data –to- Data
- Middleware that enables communications (e.g., MQ)

The Security Domain includes:

- Filters between nodes (e.g., firewall)
- Translation of communications links (e.g., encryption)
- Nodes providing authentication and/or authorization services

Patterns

Patterns are a high-level way to document reusable application and infrastructure designs. They are principally used as communication tools between architects, and between architects and engineers.

- **Application Patterns**

- Used to provide the application architect with a suite of pre-configured designs for their application.

- **Infrastructure Patterns**

- Used to provide the infrastructure architect with a suite of pre-configured designs for the infrastructure.

Platforms and Services

Platforms and Services provide a layer of abstraction from the actual products that deliver them. Either one, or both, may be combined in realizing a particular pattern.

- Platforms are characterized by:
 - A description of their function.
 - The service level objectives that are appropriate for the platform.
 - The components that are used to construct the platform.

- Services are characterized by:
 - A description of their function.
 - The service level objectives that are appropriate for them.

- The major distinction between Platforms and Services is that Services are entirely specified by their function and service level objectives (SLO).

Components

Components are the actual “bricks and mortar” of the infrastructure. They are the things that are most commonly thought of as infrastructure.

- Components are exemplified by:
 - Operating systems (Windows, z/OS, Solaris, HP/UX, AIX, etc.)
 - Hardware platforms (Intel, z/Series, Sparc, etc.)
 - Middleware software (WebSphere MQ Series, CICS Transaction Manager, WebSphere Application Server, etc.)
 - Network Devices (Switches, Routers, etc.)
 - Storage Devices (DMX1000, Direct Attach IDE, Direct Attach SCSI, etc.)

- Components are characterized by:
 - A description of their function.
 - Specific vendors, contracts, and products.
 - Service Level Objectives appropriate to the component.

Infrastructure Architecture Benefits

Implementing an Infrastructure Architecture can help reduce the impact of IT costs on the enterprise bottom line of the following industry trends.

- Rapid technology change (Moore's Law)
- Rapid business change (the business of the enterprise changes rapidly)
- Business demand for cost reduction (on-going search for low-cost)
- Business demand for new function (business opportunities are enabled by technology)
- Infrastructure is "subject to gravity" (application design is becoming more and more abstract)
- Heterogeneous technology (need to hook everything together)
- Ubiquitous technology (everyone "knows" how to use a PC)

Infrastructure Architecture Artifacts

The following are some of the deliverables created as a result of the implementation of an Infrastructure Architecture.

Infrastructure Architecture Repository

- Document and publish all of the elements (Patterns, Services, Platforms, Components) of the architecture.

Infrastructure Architecture Portfolio

- Documents all of the applications that use architecture elements and how those elements are used.

Infrastructure Architecture Governance

- Ensure that the architecture is followed correctly.

Infrastructure Roles & Responsibilities

- Clearly identify how the architecture aligns with the organizational structure.

Infrastructure Roadmaps

- Provide future predictions to facilitate planning.

Infrastructure Technology Lifecycle

- Clearly identify the location of each architecture element within the roadmap.

Infrastructure Services

- Describe the infrastructure in terms of services rather than technology (Services Oriented Architecture)

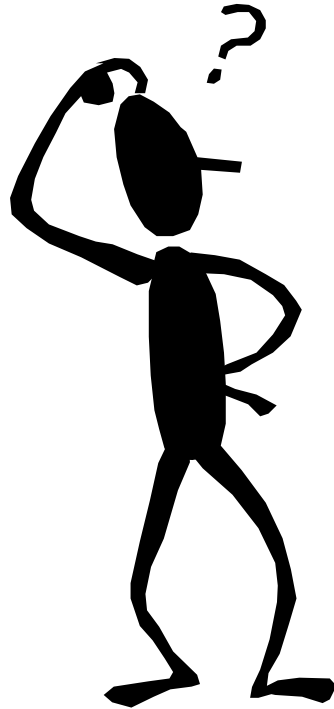
Some Challenges

- Requires senior management “buy-in”
 - Creating an architecture is a big project (12-18 months).
 - The benefits are long-term. They will show up as a better bottom-line for the business.
 - It cannot be solely justified based on cost reduction. It requires a business case that can refer to the enterprise “bottom line”, not just the technical support group.
 - It is likely to have an organizational impact, usually a shifting of some roles and responsibilities.
- Architects must not get disconnected from the technology or from the business.
 - The architects must continue to provide a bridge between the engineer and the application development organization.
- The architecture needs to be well understood throughout the IT organization.
 - The various artifacts need to be published and actively maintained otherwise the architecture can become “shelf-ware”.

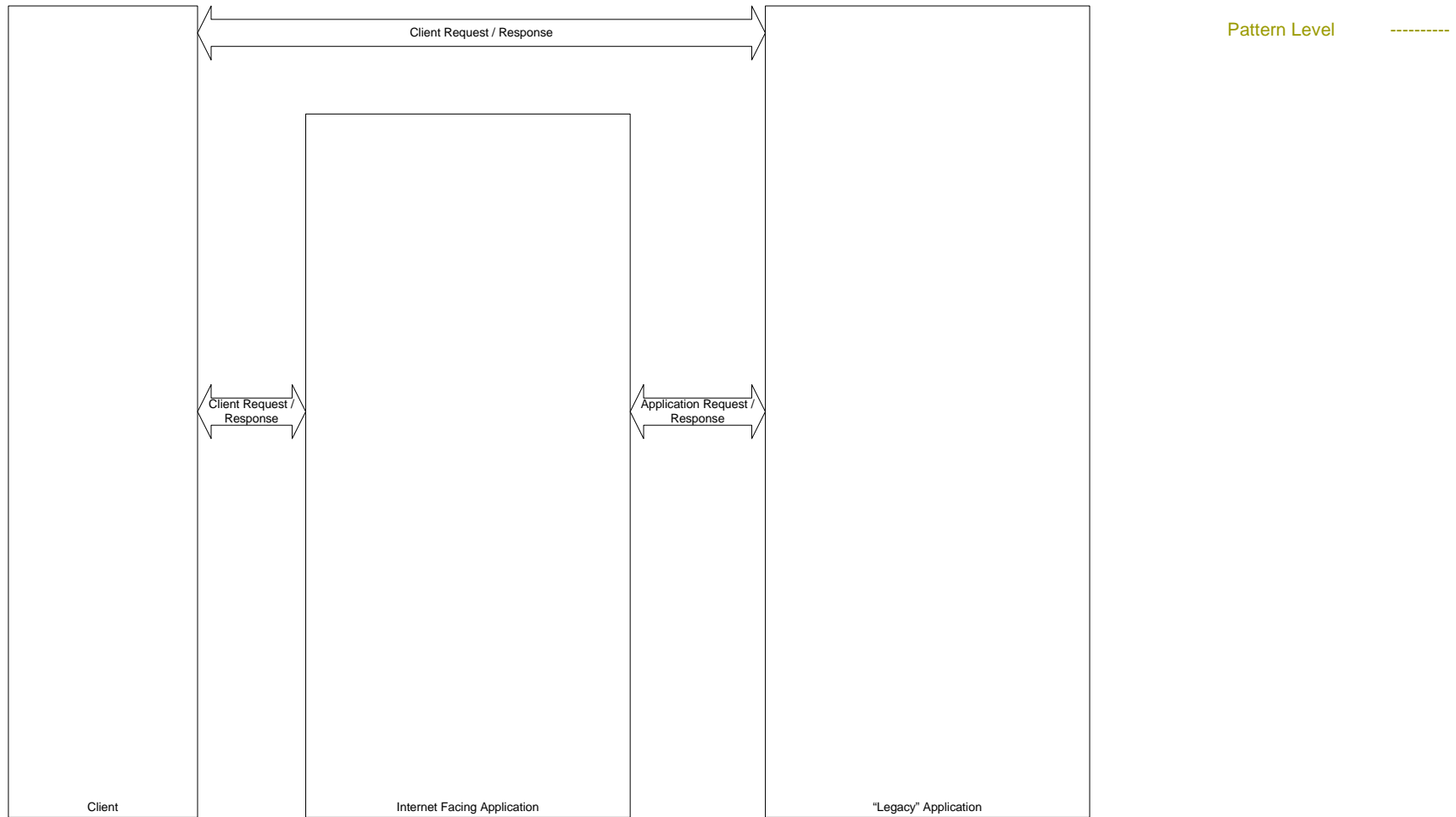
Summary

- ❑ Architecture is an organizational concept. Architects apply these concepts in connecting requirements and solutions.
- ❑ Architecture, if appropriately used, can improve the enterprise's bottom line. But, it may not always reduce the infrastructure cost.
- ❑ There are many tangible artifacts to the architecture, and they need to be tailored for each enterprise.
- ❑ Architecture requires management buy-in from the CIO down. It is difficult to implement successfully from the bottom up.

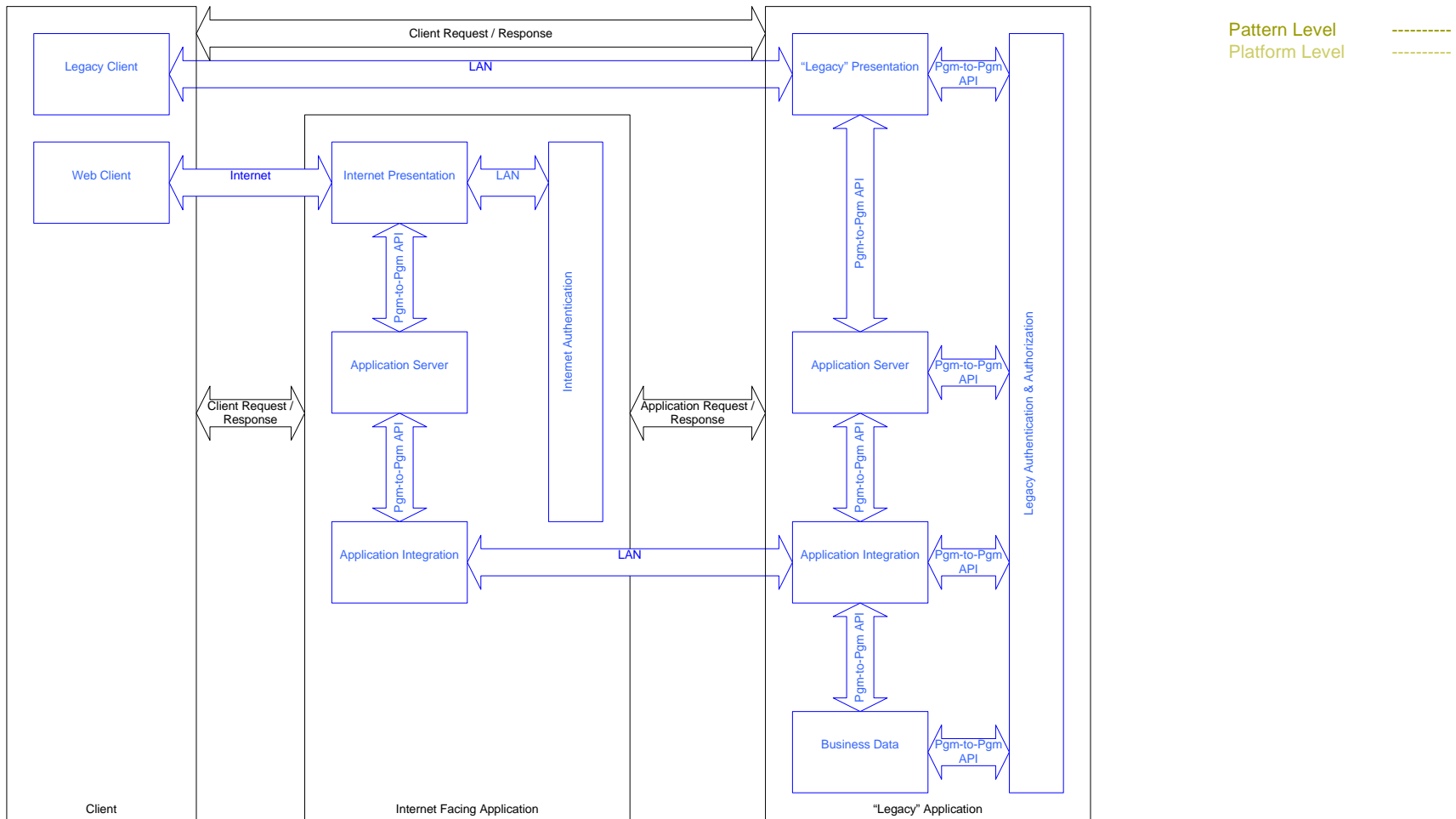
Discussion



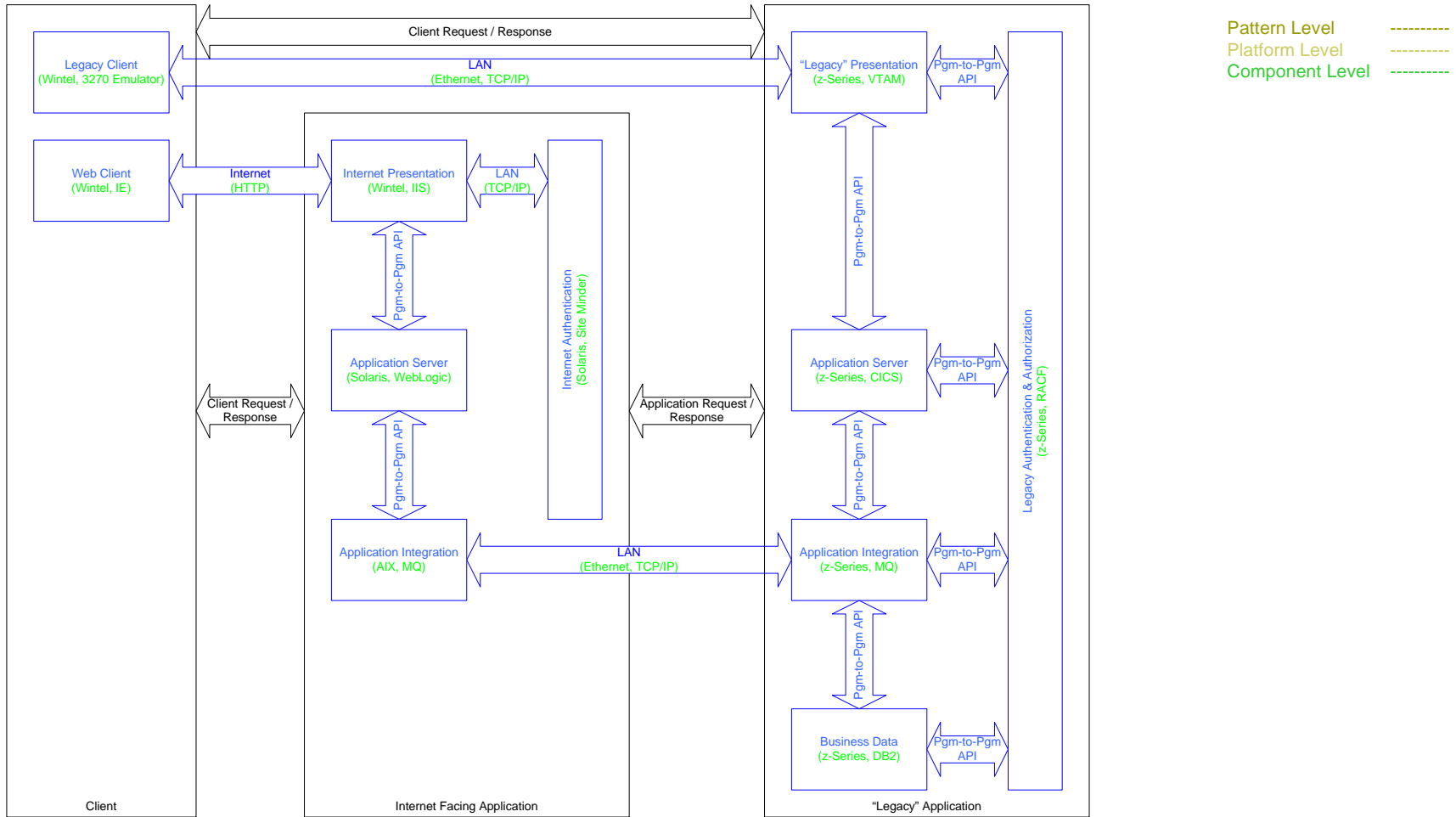
Example: 3-Tier Internet – Pattern Level



Example: 3-Tier Internet – Platform Level



Example: 3-Tier Internet – Component Level



References and Contact Info

- The Adaptive Enterprise, IT Infrastructure Strategies to Manage Change and Enable Growth.
 - Author: Bruce Robertson and Valentin Sribar
 - Publisher: Addison-Wesley
 - ISBN: 0-201-76736-8

- Building Operational Excellence, IT People and Process Best Practices
 - Author: Bruce Allen and Dale Kutnick
 - Publisher: Addison-Wesley
 - ISBN: 0-201-76737-6

- Don Melton
 - Vatic Technologies Limited
 - <mailto:meltond@acm.org>