Business and IT Architecture Metamodel Alignment

Aligning Business and IT Architecture in a Shared Metamodel

Authors	**Adam Bajer, Lloyd Dugan, Bert Hooyman, Richard Lay, Hermann Schlamann, Pierre Wienke**
Contributors	**Business Architecture Guild Metamodel Team**
Reviewers	**William Ulrich**
Editor	**Jeanie Clapp**

Contents

Abstract

One of the major issues confronting most every organization with a sizeable technology investment and long list of goals and objectives is understanding how to move from strategy formulation through solution deployment in the most time- and cost-effective ways possible. This whitepaper lays the foundation for addressing these challenges. Using business architecture as a baseline for understanding an organization's business ecosystem, this whitepaper defines and formalizes the cross-impact points among what a business does, how it delivers stakeholder value, the information it relies on, and the technology involved. The end result provides business leaders and planning teams with insights into how and where technology is or should be impacted by change to the business ecosystem.

The overall approach leverages the Business Architecture Guild's business architecture metamodel[1], aligning it to corresponding IT architecture domains. The approach taken works for any organization in any industry, including the public sector, and is not methodology- or tool-constrained. Importantly, it offers organizations a way to focus their IT investments from an aggregate perspective on business and IT initiatives that span the gamut from simple updates to major technology transformation efforts.

1 Introduction

Genpact Research Institute found that, "of nearly $600 billion spent on digital projects, almost $400 billion of it was invested in projects that fell short of expectations and returns on investment".[2] The same study also found that "Two-thirds of the projects in the survey were either scrapped or ended up being underwhelming".

How do organizations prioritize technology investments? When leadership lays out its strategic plans and allocates initiative- and related-IT funding for the year, do business leaders and deployment teams understand the scope of those investments? Specifically, do deployment teams have precise visibility into what needs to be improved, added, rearchitected, scrapped, or otherwise changed? All too often, at least according to industry statistics and a good bit of anecdotal experience, the answer to these questions is a resounding no.

In today's rapidly evolving business landscape, the alignment between business architecture and IT architecture has become increasingly crucial. This whitepaper explores the intricate relationship between these two domains and provides insights into how organizations can more effectively execute their strategies and improve overall business performance. Specifically, the whitepaper focuses on aligning the metamodel perspective defined in *A Guide to the Business Architecture Body of Knowledge® (BIZBOK® Guide)* and commonly found representations of IT architecture to create a cohesive alignment that delivers a wide range of insights critical to planning and deployment teams.

To achieve their overall goals and objectives, organizations must focus on business and IT alignment in the broader context of end-to-end strategy execution. Introduced in *BIZBOK® Guide* section 1, the Strategy Execution Framework™ plays a crucial role in the efficient realization of business strategy, avoiding the duplication of work commonly found across business and IT initiatives. Business architecture additionally informs business strategy with new opportunities, which in turn directly impact IT architecture, as shown in figure 1.

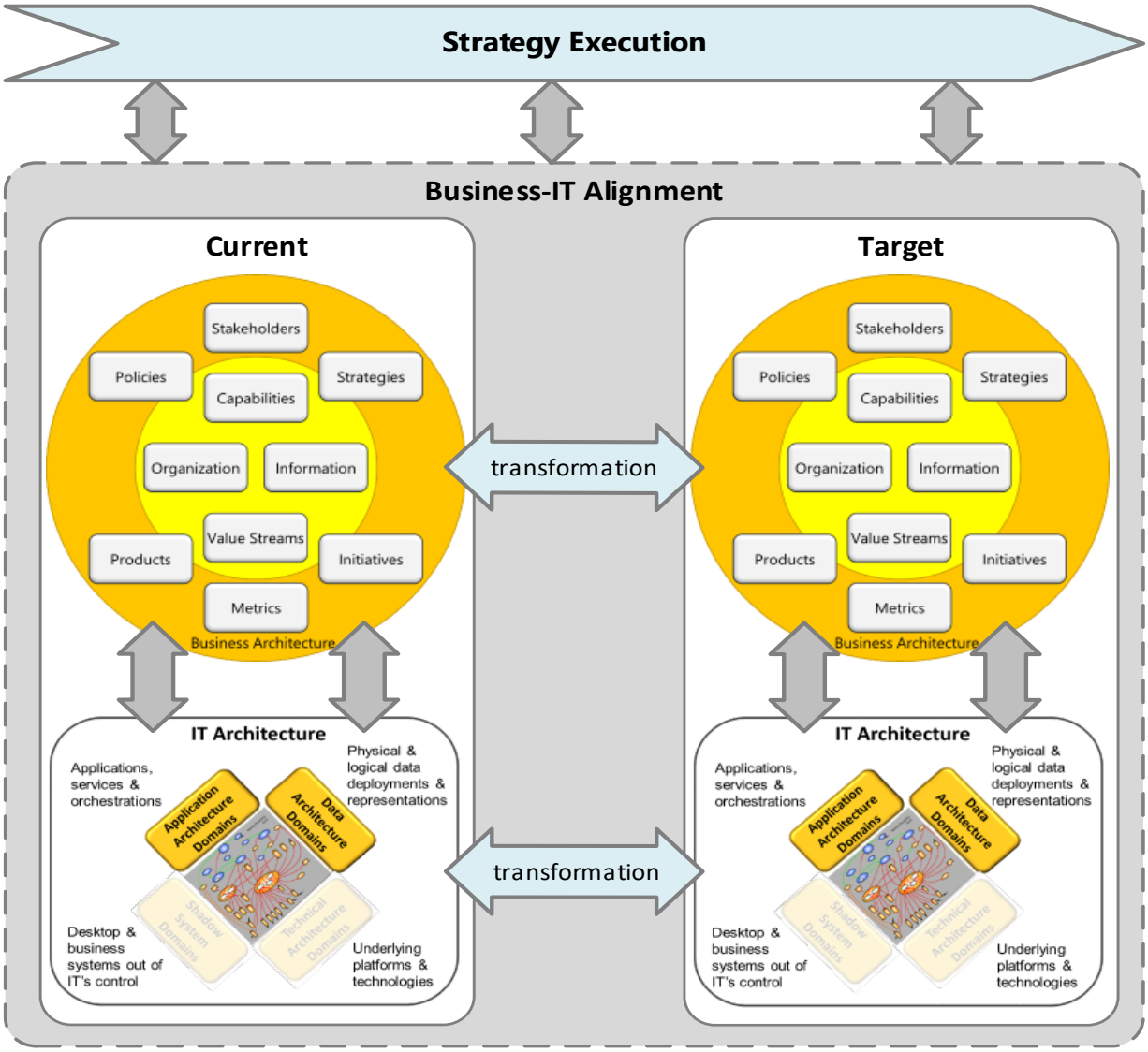

Figure 1: Business and IT Alignment and Strategy Execution[3]

As highlighted along the bottom of figure 1, this whitepaper primarily focuses on the data and application architecture domains as a basis for achieving efficient, successful business and IT alignment, which in turn enables organizations to achieve a wide variety of transformation initiatives. The bottom two perspectives shown as shadow systems that include spreadsheet and other desktop environments, and technical architecture that covers underlying platforms and technologies, are out of scope. This does not mean that these technologies should be ignored during business and IT transformation. It simply means that this whitepaper's focus is on application and data architectures, the primary capability automation and information deployment focal points for most organizations.

1.1 Business Architecture Overview

Business architecture represents holistic, multidimensional business views of capabilities, end-to-end value delivery, information, and organizational structure, and the relationships among these business views with strategies, products, policies, initiatives, stakeholders, and metrics.[4] Business architecture is, therefore, framed around the ten domains shown in figure 2. Each domain represents business ecosystem abstractions, which are formalized in the Business Architecture Metamodel Guide V3.0[5].

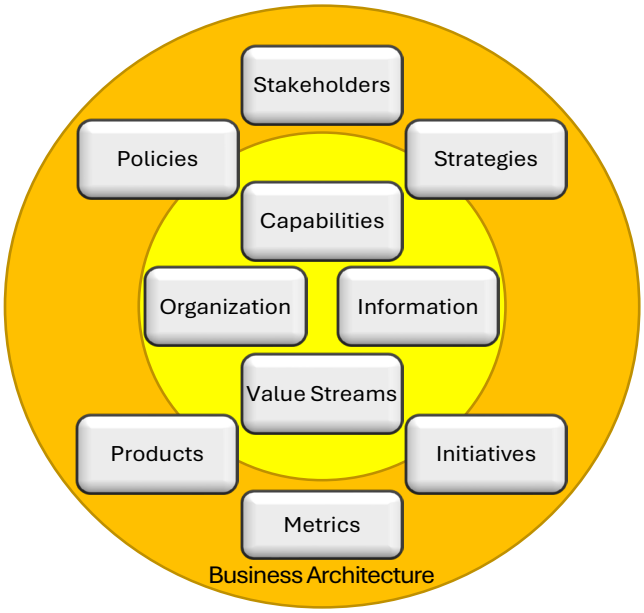

Figure 2: Business Architecture Domains

This whitepaper details relevant business architecture domains shown in figure 2, along with their constituent domain elements and corresponding relationships to IT architecture domains. The strategy domain, for example, incorporates goal, objective, and course of action domain elements, while the capability domain incorporates capability instance, capability behavior, and capability outcome domain elements.

Capability provides a critical link between the remaining core business architecture domains' value stream, information, and organization. As such, capability is also the central focal point for business architecture's link to IT architecture. If one wants to understand the information relevant to customer value delivery in a value stream, for example, those views would be derived from the association between capabilities and the information they require and modify. In addition, if one wants to understand the basis for value stream value delivery, one must look at the capabilities that enable those value streams. The following sections detail business and IT architecture alignment.

1.2 Business and IT Alignment Overview

Business and IT architecture alignment defines the state in which automated systems and data architectures fully enable business strategy, business capabilities, and stakeholder value.[6]

To understand the overview of business and IT architecture alignment, one should first be aware of the various architecture layers involving enterprise, business, and IT architecture. Figure 3 depicts enterprise architecture, business architecture, IT architecture, and IT architecture's decomposition into data, application, and technical architecture.

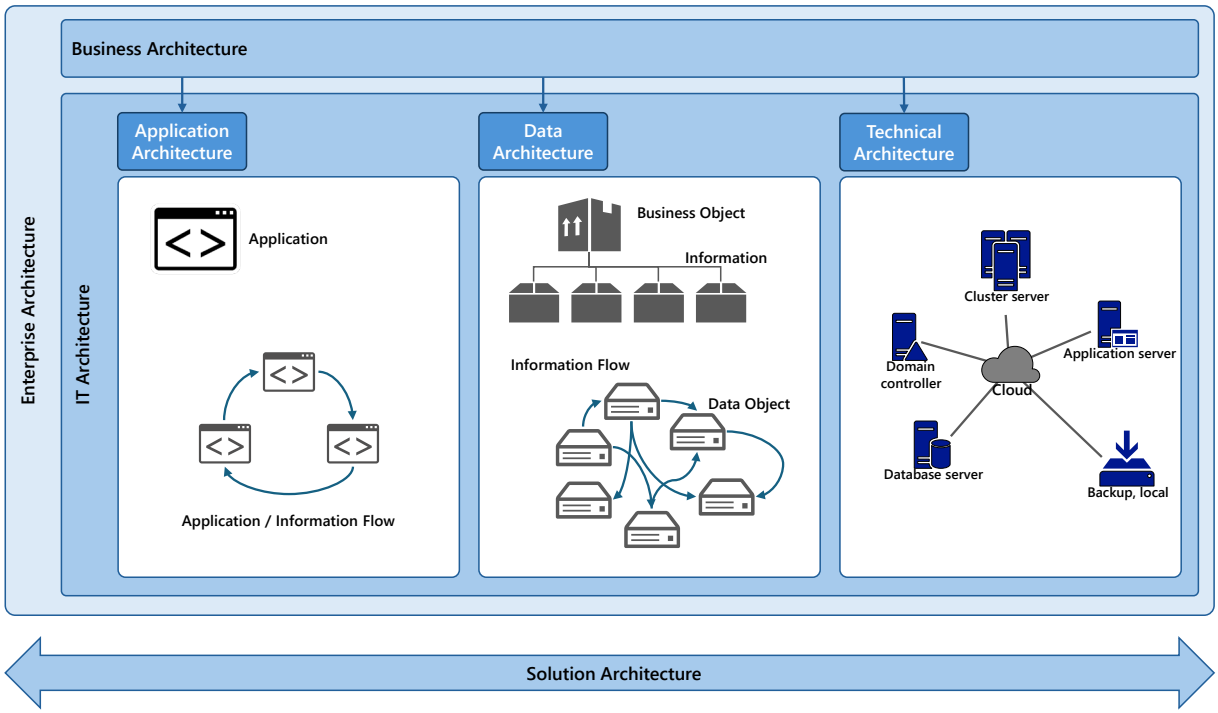

Figure 3: Architecture Discipline Overview

Business architecture domains are centered on and connected via business objects, which in turn form the basis for linking business architecture to data architecture, application architecture, and, within application architecture, software services. Business architecture's information concepts, for example, are based on business objects and have a direct association to data entities in data architecture. Similarly, capabilities, which are based on the same business objects coupled with actions that target those business objects, inform and establish the basis for identifying and specifying application architecture, application systems, and software services. With the main focus of business and IT alignment being on data and application systems, the technical architecture plays a limited role in this discussion and in this whitepaper.

2 Strategy Execution in the Context of Alignment

2.1 Business Architecture Enabling Strategy Execution

Business architecture enables business leaders to successfully execute business strategies based on complete business transparency, with full clarity and insights into the scope of business and technology impacts. While defining a clear path forward, business architecture provides a unique opportunity for executives to achieve strategic objectives, eliminate underperforming projects, and align investments. Business architecture provides the means; business leaders just need the will to leverage this increasingly important discipline, represented by the Strategy Execution Framework in figure 4.

Figure 4: Business Architecture Enables Strategy Execution

As described in the *BIZBOK® Guide*, the Strategy Execution Framework consists of five stages that apply to a wide variety of transformation initiatives, regardless of size, scope, or duration. Business architecture is a required element at each stage, from strategy formulation through solution deployment, with different aspects of the discipline leveraged along the way, based on the work being performed.

From the perspective of this whitepaper, stage 2, Assess Business Impact, and stage 3, Architect Business Solution, are the main focal points. While stages 2 and 3 are the primary focus, business architecture plays key roles in framing initiatives in stage 4 and solution deployment in stage 5.

2.2 Why Do We Need to Structure?

Structure is the way to understand how things are working together in complex organizations. While strategy execution may be understood by business leaders, business and IT architects must detail the role business and IT architecture play in achieving strategic goals and objectives. Consider that:

- Business architecture represents a holistic, rationalized view of the entire business ecosystem
- Business architecture may be engaged to visualize and facilitate organization design, providing insights into how a given business ecosystem might evolve to deliver on a given strategy
- Application architecture represents capability automation and related software solutions
- Data architecture represents a data view of the information vital to an organization's success and vitality
- Technical architecture represents the platform and enabling technologies needed for data and application architecture to work
- Solution architecture represents an investment portfolio-driven, cross-section of the business ecosystem to deliver a given solution

The remainder of this whitepaper will provide the detailed business and IT architecture mappings required to implement these transformation solutions.

3 IT Architecture Scope in Context of this Whitepaper

IT architecture establishes a blueprint of the application, data, and technical architectures that currently comprise or will comprise the information technology deployments that support the business.[7]

The main purpose of this whitepaper is to focus on the alignment between business architecture and delivering software solutions without going into the details of technical architecture and shadow system domains. As business architecture provides valuable perspective and insight into software design, data and application architecture play the crucial role in this alignment. Figure 5 frames this whitepaper's IT architecture scope in the rectangular red box at the top of the figure.

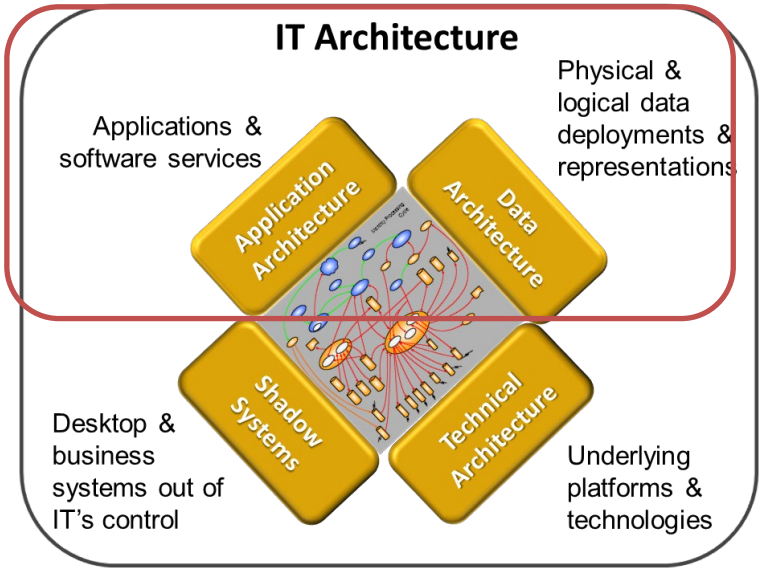

Figure 5: IT Architecture Overview[8]

Figure 5 conforms to the previous definition of IT architecture, with the addition of the shadow system domain. The discussion that follows provides all relevant application architecture and data architecture elements, definitions, and relationships covered by the in-scope portions highlighted along the top of figure 5.

Technical architecture, shown along the bottom right of figure 5, is out of scope for this white paper. The reasoning is that technical architecture relies on unique alignment factors including the latest state-of-the-art technologies, current best practices, and the evolution of industry standards, all of which may evolve independently from business-driven data and application architecture investments.

Shadow systems are not a focal point for this whitepaper but worth understanding. "Shadow systems represent business-developed software that typically resides on desktop environments, hidden from IT's line of sight in the 'shadows' of the business. Shadow systems support numerous critical business capabilities across an enterprise, yet do not exist as far as IT professionals are concerned". [9] Shadow systems in no way represent architectural perspectives and in fact are the antithesis of architecture. As a

result, they are not incorporated in the formal architecture discussions defined herein, but they do remain an important consideration in an organization's overall transformation strategy.

3.1 Application Architecture

Application architecture represents the specification and structural partitioning of technology-based automation into business logic, user experience, and data perspectives as an enabler of business architecture and strategy.[10]

Application architecture breaks down into IT the domains of application, software service, and software feature to align to capability, capability instance, capability behavior, and requirement. Requirement is not a business architecture domain, but it is relevant to the discussion of business-IT architecture alignment and the underlying metamodel. These domains, domain elements, and relationships allow practitioners to make better business and IT architecture alignment decisions and make those decisions actionable.

For reference:

- Capability instance is defined as "A specific realization of a capability, as it exists or is envisioned to exist, in context of a given business unit or situational context".[11]
- Capability behavior is defined as "The way in which a capability acts or conducts itself in certain circumstances or instances". [12]

Figure 6 identifies the targeted application architecture domains, which are based on standard industry practices.

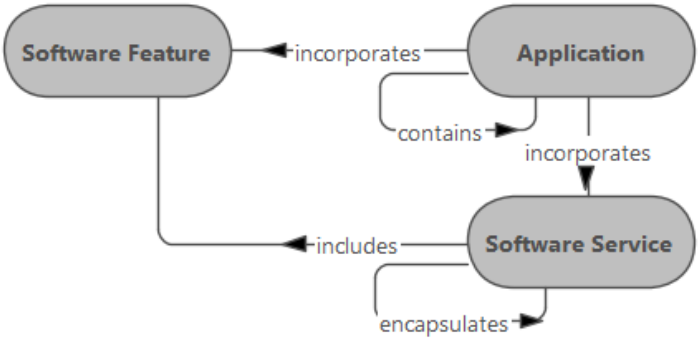

Figure 1: Application Architecture Domain Elements

While the terms application and software service in typical usage may seem somewhat similar, there are key differences at the conceptual level and in the associated behavioral semantics, as will be examined in more detail below. Notably, similar sounding but different terms used to name relationships between these IT domain elements indicate that there can be subtle – often technical – distinctions between how an application vs. a software service can be characterized in an application architecture.

Each can be classified according to a taxonomy used by the business community to distinguish between different types along technical, functional, or other lines, which facilitates the portfolio management of

applications and software services. Each term incorporates or includes one or more software features that are developed and deployed into production, which gives substance to those items under portfolio management from a technical as well as a functional perspective.

A software feature is defined as "A unit of functionality of a software system that satisfies a requirement, represents a design decision, and provides a potential configuration option".[13]

Aligning application architecture with business architecture equates to using capabilities to drive the build-out of software services for business-value-oriented portfolio management, where one or more software features implement one or more capability behaviors resulting in a capability automation. The relationships between the relevant business architecture domains and IT architecture domains are depicted in Figure 7.

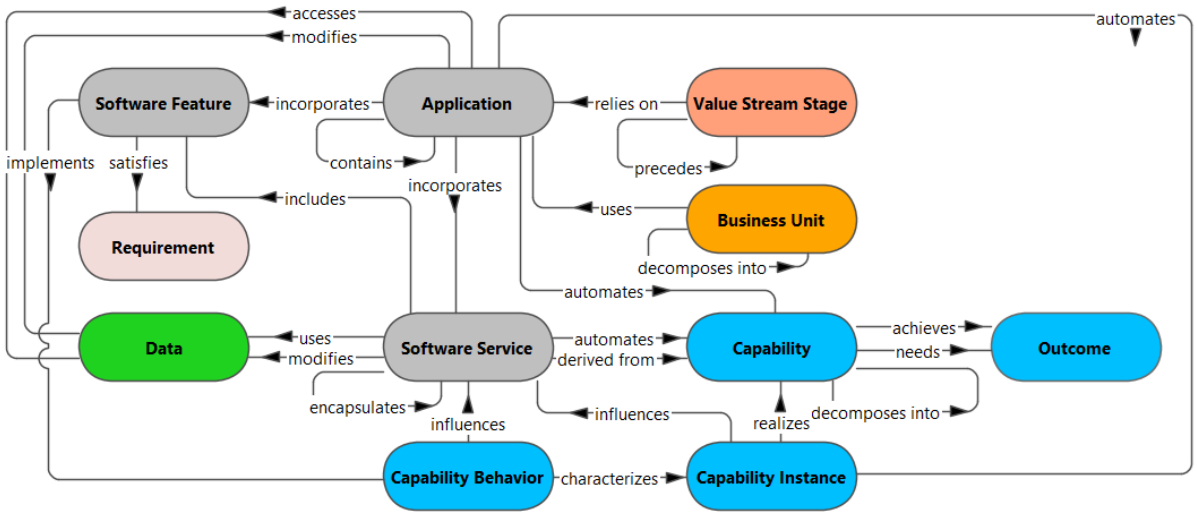

Figure 2: Application Architecture Relationships to Business Architecture

An application is defined in the *BIZBOK® Guide* as "a collection of software assets that automates and enables a bounded set of capabilities and is identifiable by name and other characteristics".[14]

The ISO/IEC defines an application as "a system for collecting, storing, processing, and presenting data" by means of software that runs on computers.[15]

Consistent with both definitions, an application:

1. Performs a scoped set of work bounded by one or more capabilities and behaviors.
2. Decomposes into one or more applications based on the granularity and scope of capability behaviors.
3. Is stateful (does persist state) or stateless (does not persist state) in its execution.
4. Accesses or modifies data during its behavioral execution and interchanges.
5. Automates capabilities (or capability instances).[16]

A software service is defined as:

"A self-contained unit of software that performs a specific task with three components: an interface, a contract, and an implementation".[17]

Consistent with both definitions, a software service:

1. Is derived from a capability and its corresponding instances and behaviors as a means of automating a well-bounded set of functionality and corresponding outcomes.
2. Uses and modifies data as an aspect of its execution.
3. Automates a capability, mirroring the corresponding capability outcome.
4. Encapsulates one or more software services to support its functionality.
5. Exchanges state between the provider and the requester, though it is typically stateless itself.
6. Has an interface that can be implemented via a service design pattern that manages the exchange of data over the interface.

An example of the most relevant relationships between these concepts is presented in figure 8.

Capability	Capability Behavior	Application	Software Service	Software Feature	Requirement
Conveyer State Management	Turns to "delayed" when an incident cannot be resolved in time allotted	Trip Update	Flight Status Update	Flight Status Check	Keep traveler updated on flight status
Trip State Management	Turns to "delayed" when a matched conveyor is delayed	Trip Update	N/A	Trip Status Check	Keep traveler updated on trip status
Message/Customer Matching	Text or call impacted traveler	Trip Update	Message Service	Notify traveler when delayed	Traveler should be notified when their flight is delayed

Figure 8: Example of Application Architecture and Business Architecture Concepts

The business architecture and application architecture terms in figure 8 create context for and enable the automation of real-world situations such as that summarized in the following example.

> *Example: A Trip Update application reports the status of a Trip to a traveler taking the Trip. It consists of a Trip Status Check feature that includes delivering the scoped functionality of conveying the status or condition of a Trip (e.g., text alerts on a cell phone) based on the capability behavior "text or call impacted traveler" that characterizes the Message/Customer Matching capability.*

The bottom line is that alignment of application architecture to business architecture requires using capability-driven perspectives for all aspects of application and software service portfolio management when applying a business-value-oriented approach.

3.2 Data Architecture

Data architecture represents integration of value specifications for qualitative and quantitative variables and their alignment with business architecture and strategy.[18]

Figure 9 represents the major elements of the data architecture according to best practices.

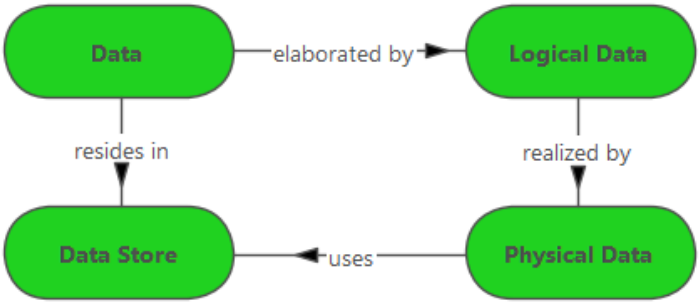

Figure 9: Data Architecture Elements

Data and its corresponding architecture domain elements are defined as follows.

- Data – represents a distinct concept that is recognized by business professionals without additional details. Data resides in data store and may be elaborated by logical data to reflect overall structure and relationships.
- Logical data – represents a detailed data structure called a logical data model that is composed of logically related data entities.
- Physical data – represents a detailed data structure called a physical data model that is derived from logical data and specifies the format or schema required by a particular technology.
- Data store – represents a data repository for storing and manipulating a collection of physical data. A data store is considered an integral part of a data architecture.

Figure 10 represents the primary business, data, and application architecture domain relationships. The business architecture relationships are described in the Business Architecture Metamodel Guide V3.0[19]. The mapping shown in figure 10 primarily focuses on data and application architecture and connections with core business architecture domains.

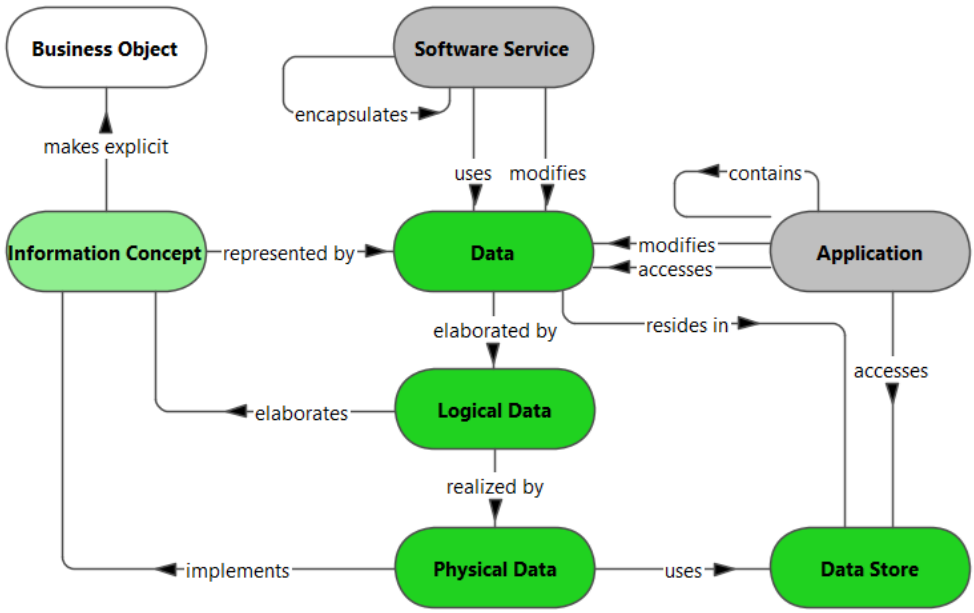

Figure 10: Data Architecture and Related Business Architecture Information Concept Domain

As shown in figure 10, information concept, a core business architecture domain, is directly linked to data in the data architecture. Application uses and modifies data by having direct access to one or more data stores. In addition to that, data during the analysis phase, is elaborated by logical data to cover more details in the form of required attributes. In this way, logical data elaborates information concept used to identify data and serves as a foundation for solution design where it is realized by physical data. As a result, the core technical representation of data as physical data implements the information concept from the solution technical perspective ensuring the alignment with business domain.

Data architecture typically includes three types of data models representing data at conceptual, logical, and physical levels as shown in figure 11.

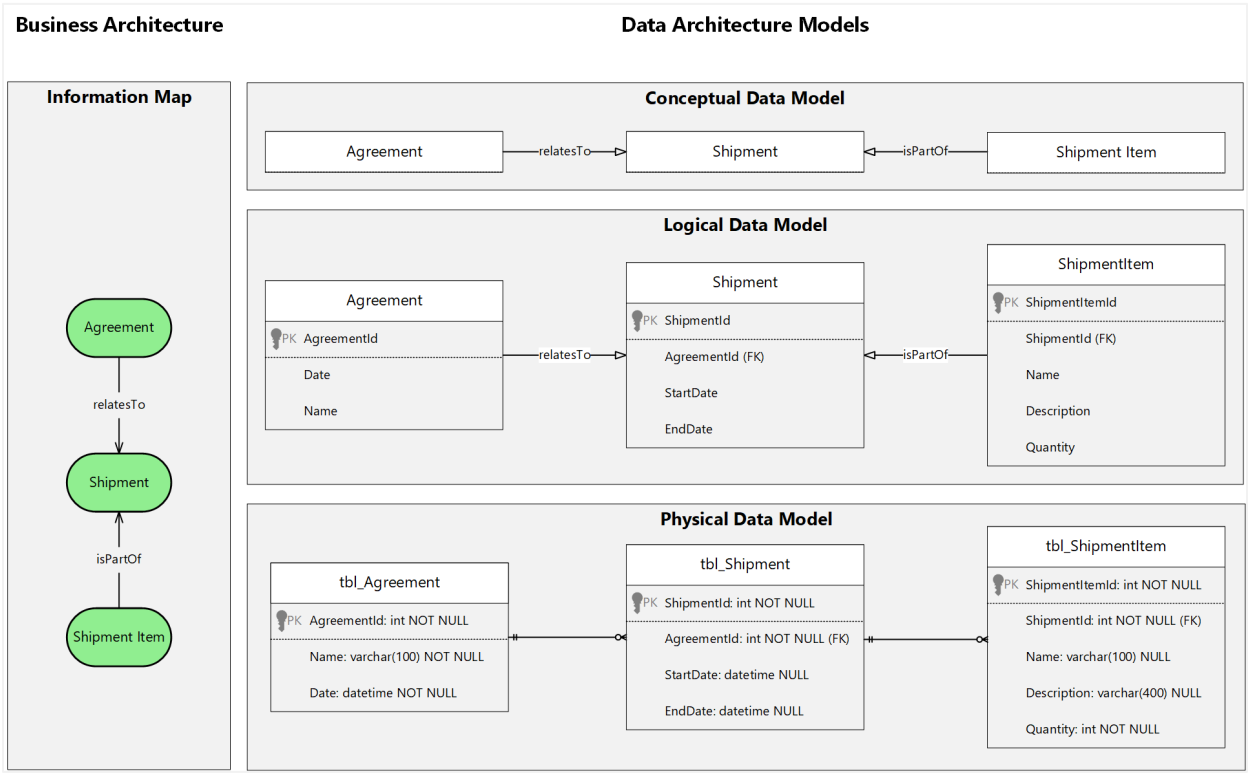

Figure 11: Conceptual, Logical, and Physical Data Model Perspective

All three models are recommended to formalize a data architecture. Depending on the specific needs, data models address different type of details at different levels.

- Conceptual data model defines the “what” – the core data entities and relationships as abstract forms that cover business context – but avoids details
- Logical data model defines the “how” – the structure and details of the data that represents problem domains independent of a technology and include tables, data attributes, and relationships
- Physical data model defines the “how” of implementation – specific database structures specifying clusters of logical data components as implemented or to be implemented

It needs to be noted that there are also additional possible relationships between business architecture domains and data architecture elements as shown in figure 12.

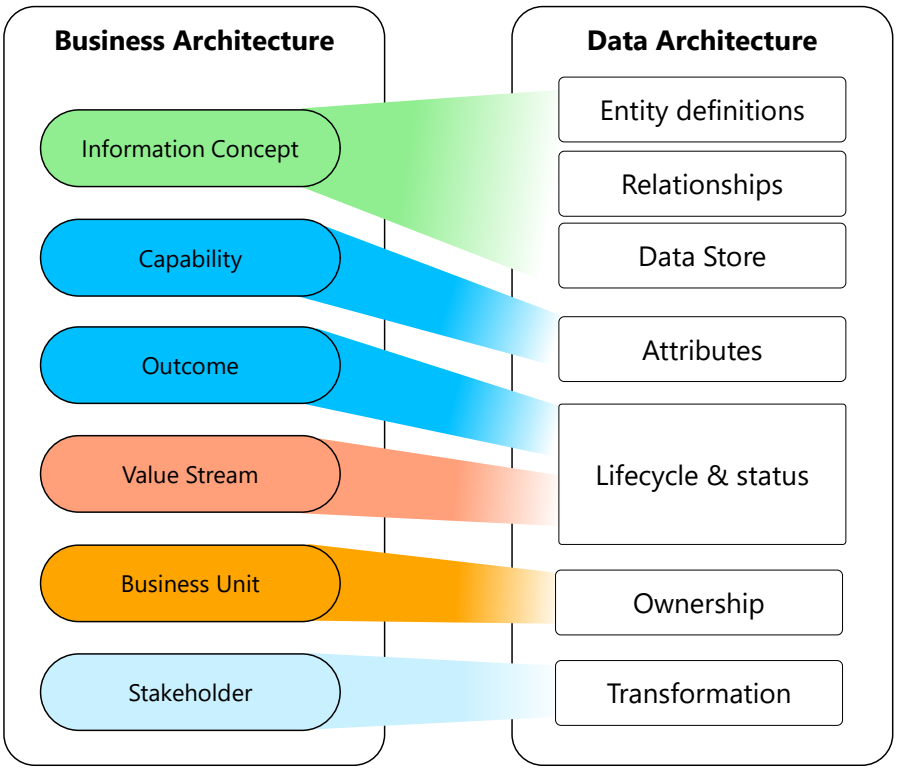

Figure 12: Additional Relationships from Business Architecture to Data

These relations are valuable as a way to derive additional information to provide the comprehensive design of data. As presented in figure 12, core elements defined in the business architecture domain could be leveraged during design of the data architecture as following:

- Information concept serves as the foundation for the existence, name, and definition of data entities and relationships among those data entities
- Capabilities are used as input to data attribute definition for each data entity, adding necessary details for logical and physical data models
- Capability outcomes, value streams, and value stream stages bring an overview of a data lifecycle and corresponding states as a basis for designing state machines and flow diagrams
- Business units provide additional information about data ownership, sharing, responsibilities, and actions performed on the data
- Stakeholders provide insights into the benefits and resources needed for larger transformations

Data is the final expression of the meaning of business terms at the detailed level, where information concepts inform the need for and completeness of that data and capability decisions are articulated and communicated.

4 Business and IT Alignment

The general alignment between business and IT architecture is presented in figure 13, with connections among business architecture domains and IT architecture domains at an aggregate level. Note that associations among business architecture domains, such as those between capability instance and capability behavior, or information concept and capability, were omitted to simplify the view. For these more detailed perspectives, readers should refer to the Business Architecture Metamodel Guide V3.0.[20]

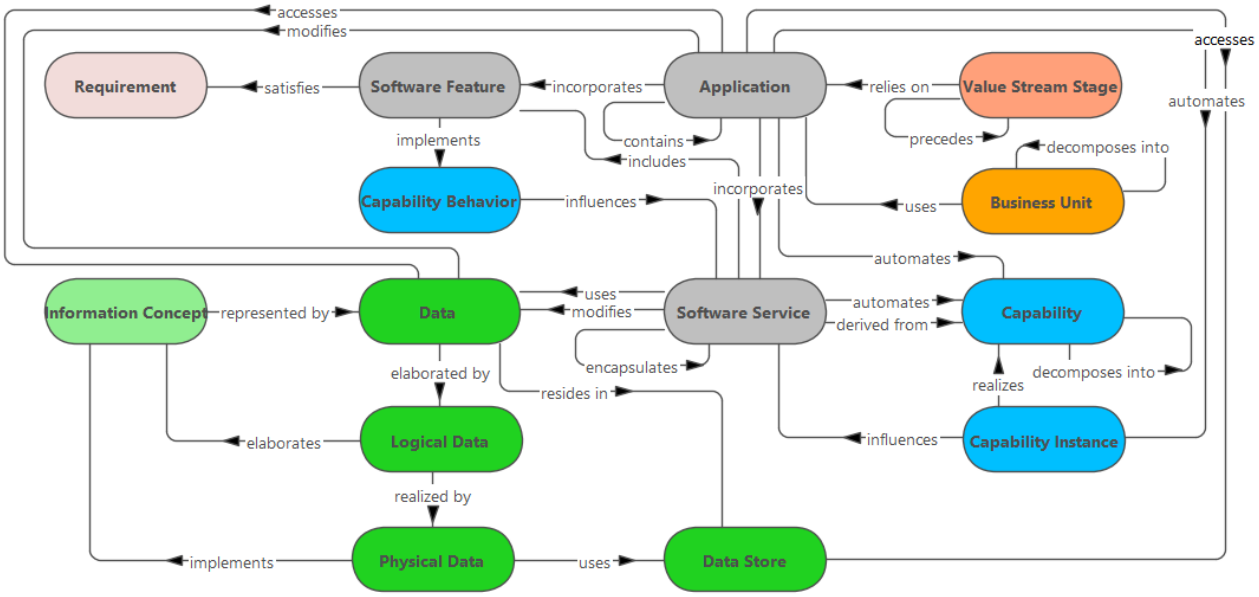

Figure 13: Business and IT Architecture Relationships

Figure 13 provides a structured view of how capability, value stream stage, business unit, information concept, and requirement interact with data, application, software service, and software feature. Populating and leveraging these views to represent a business ecosystem, which spans an enterprise and more, allows leadership to manage the evolution, transformation, and related investments across every aspect of end-to-end strategy execution.

At the heart of the model, a business unit, which is an element of an organization, represents where specific work is performed and where that work relies on one or more applications to support its operational needs. An application itself is modular, as it contains other applications, reflecting enterprise system landscapes with decomposed and interrelated software components.

From a strategic perspective, an application plays a vital role in stakeholder value delivery. A value stream stage, which represents a subset of a value stream and delivers value items that accrue to a value stream's value proposition, relies on one or more applications to automate the capabilities that enable that stage. While the direct relationship is between an application and the capabilities it automates, aggregating the applications that collectively automate a value stream stage's enabling capabilities provides planning and investment insights at a value stream level.

Applications also incorporate software services, which encapsulate discrete functionality that automate a capability and, in turn, are essential to business operations. Additionally, applications modify data,

demonstrating their role in processing the information that capabilities rely on and modify. In addition, applications incorporate software features, which represent enhancements to those applications by adjusting capability behaviors that those applications automate.

A software service plays a central role in the IT landscape by including software features, which represent capabilities and corresponding behaviors, and encapsulating other software services, allowing for a modular and reusable solution architecture.

Capability instance represents a specific realization of a capability in practice, such as a business unit or partner. Capability instances and related behaviors influence software services, highlighting the interdependency and alignment between business execution and corresponding automations.

Data plays a crucial role in both business and corresponding automations. An information concept, which defines business terms and semantics, is represented by data in an automated environment, ensuring that business knowledge is captured, structured, operationalized, and automated. Data resides in a data store, providing structured persistence for enterprise data management.

4.1 Guild Metamodel Key with IT Architecture Elements

The figure 14 diagram color codes the combined business domains defined in the Business Architecture Metamodel Guide V3.0 and the IT architecture domains defined in this whitepaper.

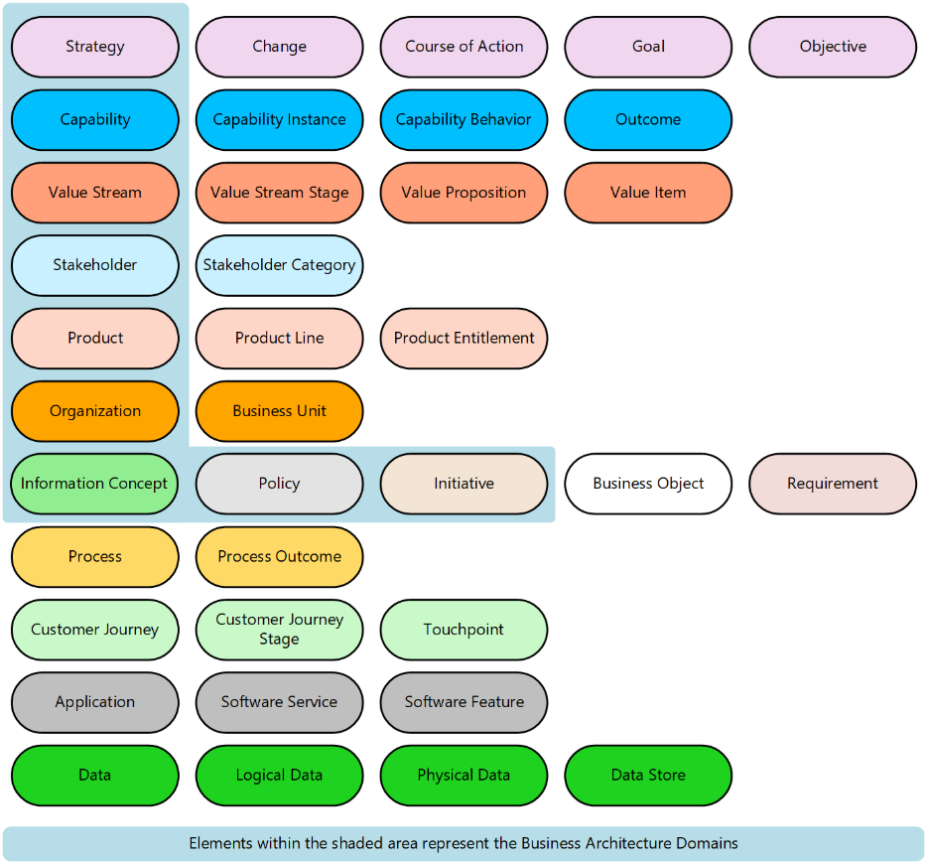

Figure 14: Business and IT Domains and Domain Elements in Scope

The business architecture domain elements in figure 14 are shown to the right of the shaded business architecture domains referenced in this whitepaper. Each domain element shares a corresponding color shading with its domain. The product domain, for example, shares the same color coding as product line and product entitlement. Note that business object is not a business architecture domain, but capability, information concept, and stakeholder category are all based on or derived from business object. Finally, while requirement, process, and customer journey are business domains and defined in the metamodel guide, they belong to separate modeling disciplines and are not part of business architecture.

4.2 Strategy-to-IT Architecture

The model for alignment between the business architecture strategy domain and IT architecture domains is presented in figure 15.

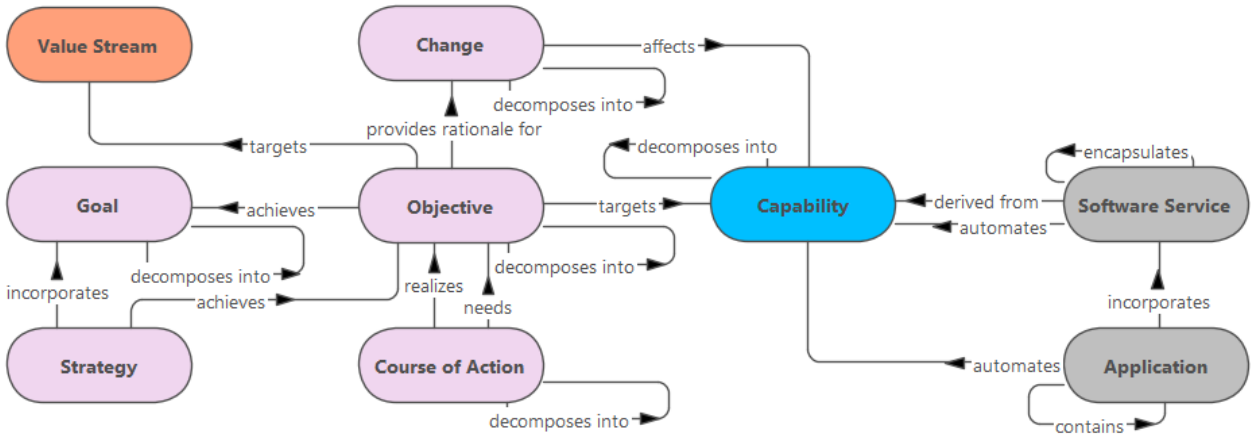

Figure 15: Strategy and IT Architecture Relationships

The figure 15 diagram depicts aspects of strategy, capability, and value domains, with the application architecture and software service domains representing the technical components. This distinction visually separates the business and technical domains, highlighting their interdependence.

1. Capabilities serve as the primary link between business intent and technical implementation. The diagram shows that applications directly automate capabilities, demonstrating how what an organization does is realized in technical systems.
2. Applications, defined as collections of software assets that automate capabilities, can contain other applications. This allows for a modular and scalable IT architecture adaptable to business complexity.
3. Software services, derived from the capabilities they automate, tighten the alignment between technology and business needs. Deploying software services derived from capabilities is key to rigorously defining what the organization does into a technical implementation. Software services also encapsulate other software services, enabling complex service architectures that support business capabilities while maintaining modularity.

4. Applications incorporate software services, demonstrating how larger units of software assets are composed of more granular software services. This composition allows for flexible, scalable capability automation and service reuse across applications.

The absence of direct connections between strategy and the application architecture is deliberate, underscoring that translating strategy into technical implementations should be mediated through capabilities, ensuring that IT solutions are driven by business needs.

The approach to connecting strategy through business architecture to technical implementations offers several advantages:

1. A framework for IT governance based on what the organization does and what it needs.
2. Alignment of technical solutions with business capabilities and strategic objectives.
3. Flexibility in responding to changes in business capabilities.
4. End-to-end traceability from strategy, through capabilities and related business architecture domains, to requirements, software features, and corresponding technology deployments.

The connection between business architecture and IT architecture provides an essential link in the enterprise architecture landscape, illustrating how business strategy is translated into technical solutions and ensuring that IT deployments support strategic goals and objectives.

4.3 Value Stream-to-IT Architecture

The alignment between business architecture value stream and IT architecture is shown in figure 16.

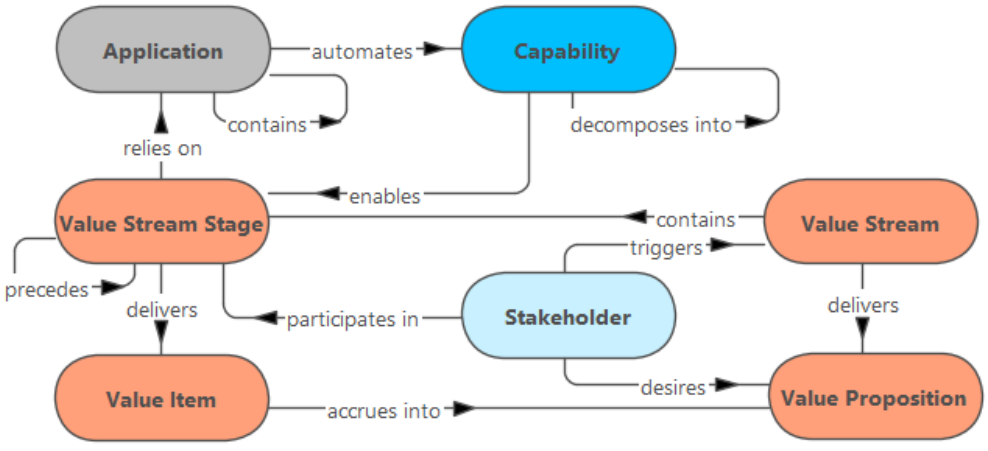

Figure 16: Value Stream and IT Architecture

Figure 16 illustrates how value creation, represented by value stream, value stage, value item, and value proposition, connect to one or more applications through capability.

Value stream represents an end-to-end collection of activities that create value for a customer. Figure 16 shows that value streams are composed of value stream stages, which are discrete value item-producing phases in the value creation process. Each value stream stage relies on applications that automate stage-enabling capabilities.

The direct link between value stream stage and application provides an important shortcut association, which is required because the vast majority of legacy applications are not software-service based and redundantly automate capabilities. The shortcut provides architects, planning teams, and deployment teams with greater specificity when performing value-based, application architecture investment planning and deployment.

Value stream and value stream stages play a central role in connecting various business architecture elements to technology implementations. Specifically, value stream stages:

1. Are enabled by capabilities, demonstrating how what the organization does supports value creation and value consumption.
2. Deliver value items, which accrue to value propositions delivered to stakeholders.
3. Are impacted by initiatives showing how strategic changes affect value delivery.
4. Serve as a primary target of business objectives, ensuring that value creation is aligned with strategic goals and objectives.
5. Enable customer journey stages, aligning value delivery with customer experiences.
6. Frame requirements, ensuring that technical specifications are derived from value-creating activities associated with enabling capabilities and participating stakeholders.

Value stream stages rely on applications by way of capabilities, indicating that each step in the value-creation process is supported by a variety of technical solutions. This relationship ensures that applications are aligned with value delivery, rather than existing in isolation. The association also provides architects with insights into the technical implications of a given strategy and related initiatives that require modifications to supporting applications.

In summary, figure 16 demonstrates how value streams and value stream stages, as representations of value creation, connect to applications. By establishing these relationships, organizations can ensure that their technical implementations are directly aligned with value delivery, customer needs, and strategic objectives. This alignment is crucial for creating an IT landscape that effectively supports an organization's value propositions to its customers.

4.4 Capability-to-IT Architecture

Figure 17 highlights the alignment between capability and IT architecture. The capability associations shown in figure 17 provide the foundation for connecting business and IT architecture to enable strategy execution.

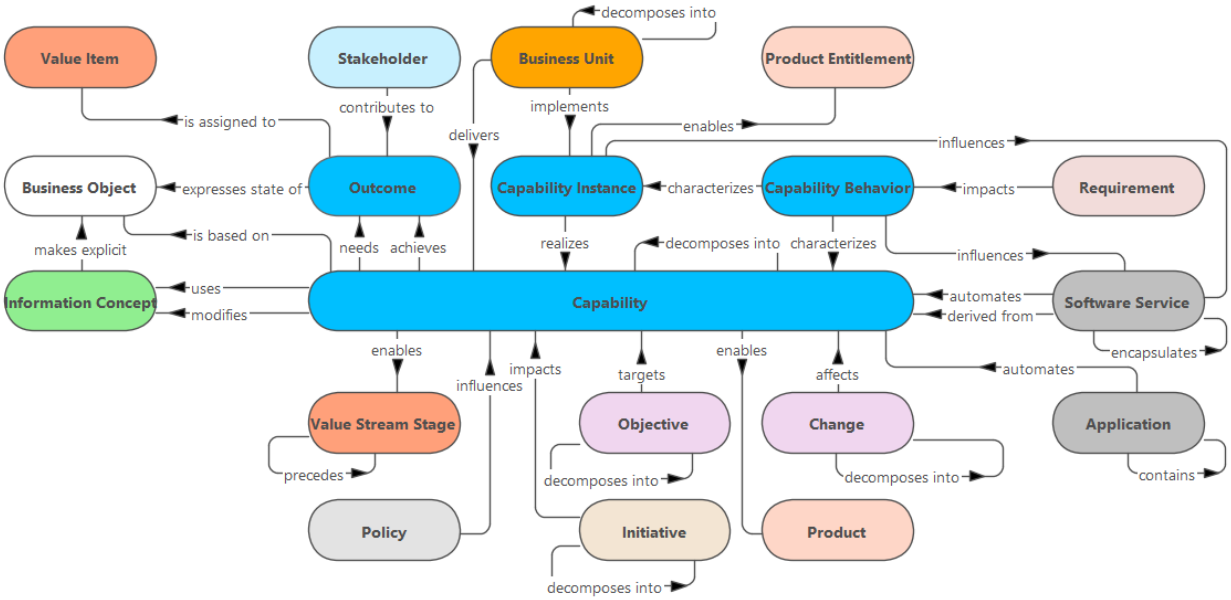

Figure 17: Capability and IT Architecture

Two key relationships shown in figure 17 establish the primary link between the capability and IT domains.

1. Application automates capability.
2. Software service automates and is derived from capability.

Capabilities decompose into capabilities, allowing for a hierarchical structure that reflects the organization's ability to deliver a wide variety of outcomes at various and increasing levels of granularity. One major benefit of leveraging capability decompositions as input to software service design is that software service boundary definition and corresponding decomposition may be largely derived by those capabilities and related decompositions.

There are multiple aspects to capability-derived relationships. Capability instance, coupled with capability behavior, influences the design and evolution of software services, allowing for more contextual, nuanced, and specialized design approach.

Figure 17 also illustrates how capabilities interact with other domains.

1. Information concept: Capabilities use and modify information concepts, showing the data dependencies of business functions.
2. Value stream: Capabilities enable value stream stages, demonstrating how organizational competencies support value creation processes.
3. Product: Capabilities enable products, and capability instances enable product entitlements, showing how organizational abilities translate into customer-facing offerings.
4. Organization: Business units deliver capabilities and implement capability instances, highlighting the operational aspect of capabilities.

5. Strategy: Objectives target capabilities, ensuring alignment between strategic goals and organizational competencies, with change targeting those same capabilities.
6. Initiatives: Initiatives impact capabilities, showing how strategic decisions influence the capability landscape.
7. Stakeholder: Participating stakeholders in value stream stages contribute to capability outcomes, demonstrating the role of capabilities in meeting stakeholder expectations.
8. Policy: Policy impacts capability, along with strategy and other domains, but the association was abbreviated to simplify the model.

This comprehensive view of capabilities and their relationships emphasizes the central role capabilities play in bridging business strategy and technology. By deriving software services from capabilities and using applications to automate capabilities, organizations can ensure that their IT landscape is closely aligned with their business needs and competencies.

The model in figure 17 additionally supports several key enterprise architecture practices.

1. Capability-based planning for IT investments.
2. Alignment of service-oriented architecture with business capabilities.
3. Traceability between strategic objectives and IT implementations through capabilities.
4. Impact analysis of changes and initiatives on both business capabilities and IT systems.

Finally, the model in figure 17 provides a robust framework for aligning IT systems with capabilities, ensuring that technology investments directly support the organization's core competencies and contribute to achieving strategic objectives.

4.5 Information Concept-to-IT Architecture

Figure 18 highlights a model for alignment between information concept and IT architecture.

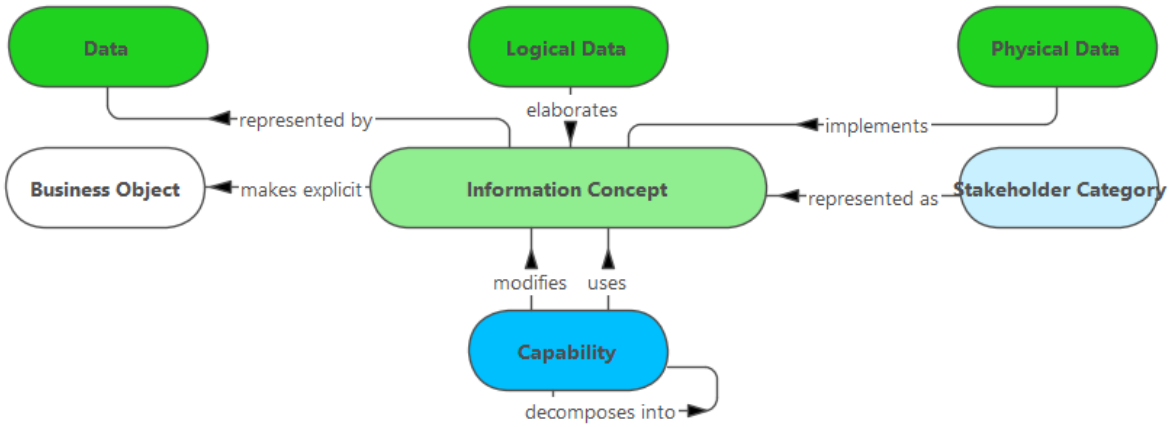

Figure 18: Information Concept and IT Architecture

Complete, well-organized, and high-quality information is critical to an organization's ongoing success and viability. Information concepts provide the ideal perspective for a given organization in terms of the information it requires for all capabilities to work effectively. Information concepts:

1. Make business terms, definitions, and relationships explicit and transparent.
2. Represent the totality of information used across a business ecosystem, which is often a superset of an organization's data.
3. Are used and modified by capabilities to generate the outcomes.
4. Represent stakeholder categories, where all stakeholders are represented in the information map.

The connection of information concept-to-IT architecture domains is mainly focused on the data architecture. Specifically, information concepts:

1. Ensure that business critical information is available to manifest effective, optimized capability outcomes that ultimately deliver stakeholder value.
2. Provide a framework that helps structure, design, and organize data so that it accommodates business understanding, desires, and success.
3. Ensures that technical implementations use and access the data that reflects the information an organization requires from a strategic as well as an operational point of view.
4. Delivers comprehensive input and insights for business-driven data architecture, data management, and data governance.

4.6 Organization-to-IT Architecture

Figure 19 depicts a model that aligns business architecture's organization domain with IT architecture domains.

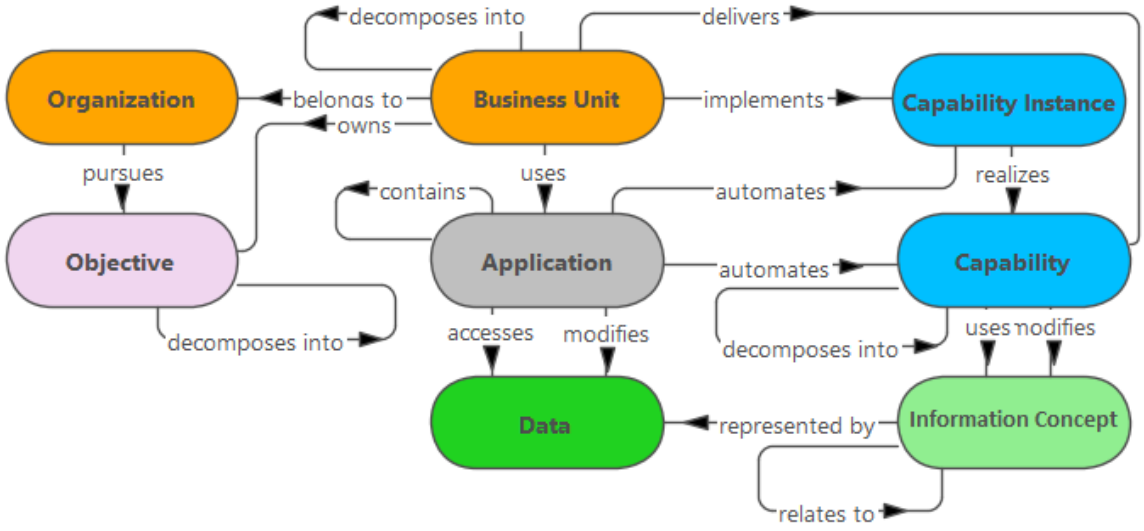

Figure 19: Organization and IT Architecture

In order for an organization and its constituent parts to excel, it needs to have effective deployments of its capability instances. Realizations of capability instances are automated by applications defined in

application architecture. In other words, business units use applications to process and deliver data required to perform essential business operations.

The linkage between organization and IT architecture provides the valuable insights into:

1. Where various business units have certain capabilities that use and rely on various applications.
2. The importance of various aspects of an application portfolio to business units and the organization as a whole.
3. Potential application and related data-ownership responsibilities.
4. Traceability between strategic objectives and IT implementations from an individual business unit perspective via capability instances within those business units.
5. The impact of changes and initiatives on business units, applications, and corresponding data.

4.7 Requirements-to-IT Architecture

One major area that organizations invest in involves the implementation of various types of requirements, which impact IT architectures in a variety of ways. Figure 20 highlights the relationships among various business architecture domains, IT architecture domains, and requirements.

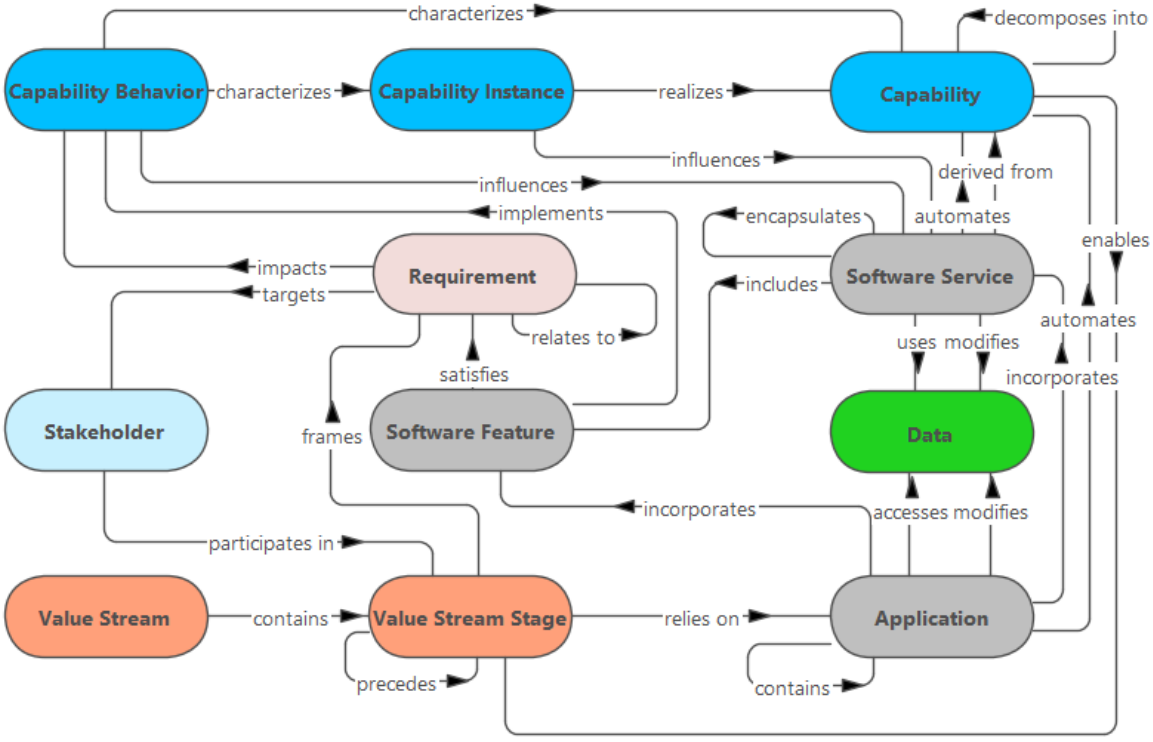

Figure 20: Business Architecture, IT Architecture, and Requirements

When a value stream stage is identified as limited in its contribution to the value proposition, then capabilities, capability instances, and capability behaviors that enable that value stream stage should be evaluated for improvement. Where there are issues, it signals a need to specify requirements to improve the behavior of those capability behaviors and corresponding capability instances. These requirements,

in turn, target the stakeholders that participate in those same value stream stages by contributing to the corresponding capability outcomes within those stages.

From the requirements, software features, aligned to capability behaviors, would be specified to satisfy those requirements. The software features are ultimately incorporated into the application or software service that ultimately automates the capability or capabilities in question.

Based on the associations shown in figure 20, requirements ensure that:

1. Anticipated changes are articulated in requirements so that software build decisions for services and features add value to the capability being addressed.
2. Where resources are applied, the progress toward the benefits of the change can be assessed and adjusted.
3. The impact of changes can be fully articulated so that initiative and stakeholder scope is comprehensive and accurate.
4. Dependencies between the software and data are reflected in the ordering and sequencing of changes and mirror the dependencies defined by the associations between corresponding capabilities and information concepts.

5 Business and IT Alignment's Role in Strategy Execution

Figure 21 presents the overview of the connections between business architecture and IT architecture across the Business Architecture Guild's Strategy Execution Framework[21].

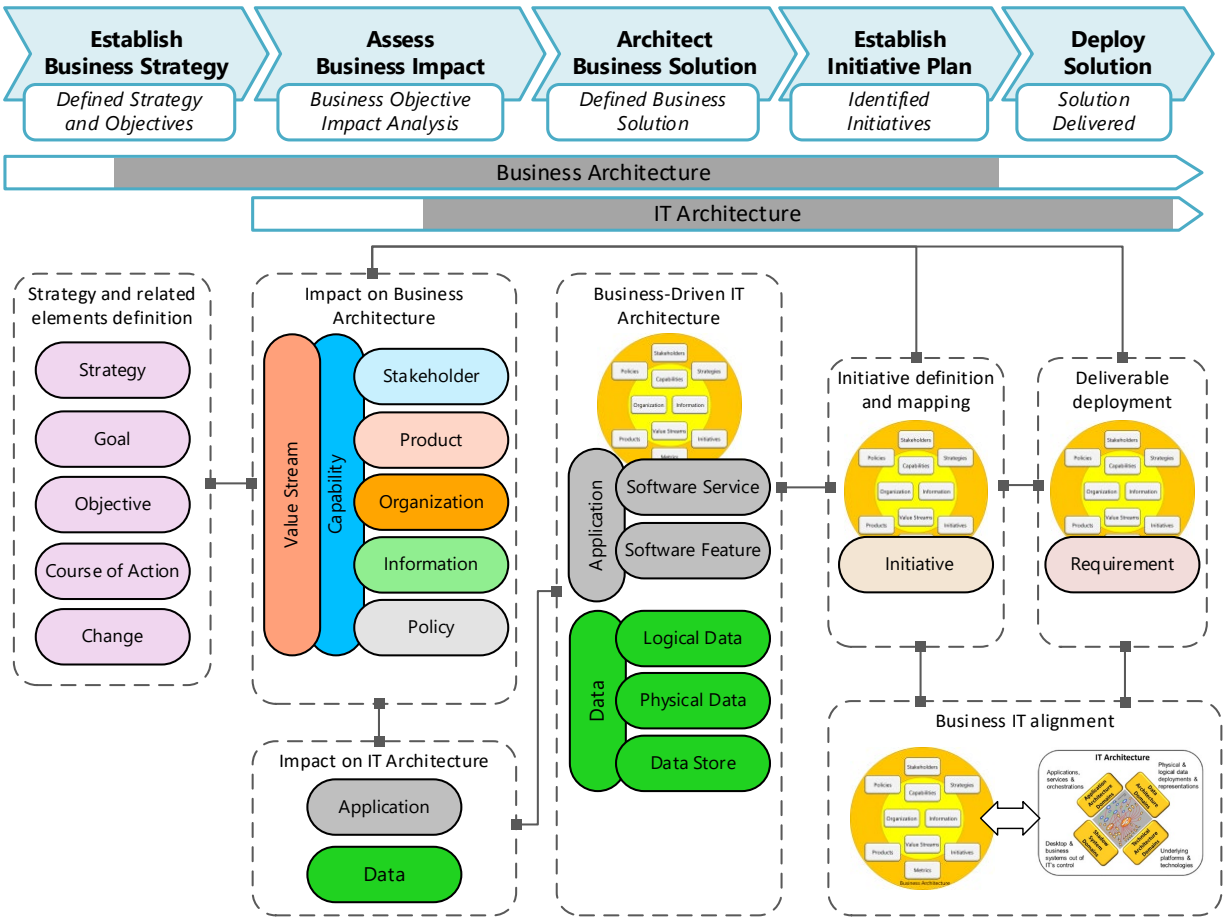

Figure 21: Strategy Execution in The Contest of Business and IT Architecture

As indicated in figure 21, business architecture identifies and maintains key elements of the business ecosystem that facilitate end-to-end strategy formulation and through solution deployment as articulated by the Strategy Execution Framework. The framework breaks down as follows.

Stage 1: "Establish Business Strategy" establishes strategy, goals, objectives, and courses of actions that drive all aspects of organizational change, transformation, and corresponding investment.

Stage 2: "Assess Business Impact" exposes and formalizes the scope of change across the business and IT architecture.

Stage 3: "Architect Business Solution" formulates desired future states, including organization and technology architecture and design.

Stage 4: "Establish Initiative Plan" formulizes rationalized, optimized business initiatives, funding, and corresponding roadmaps to achieve the targeted strategy.

Stage 5 "Deploy Solution" implements the established strategy through successful initiative delivery of all targeted business and technology requirements in an optimized target state.

6 Business and IT Alignment - Summary of Relationships

Summarizing the business architecture and IT architecture relationships defined within this paper provides a basis for planning and execution teams to leverage the metamodel in strategy execution.

6.1 Business and IT Alignment Relationships - Application Domain

To summarize the details captured in this whitepaper, figure 22 presents all relationships between business architecture domains and IT architecture domains.

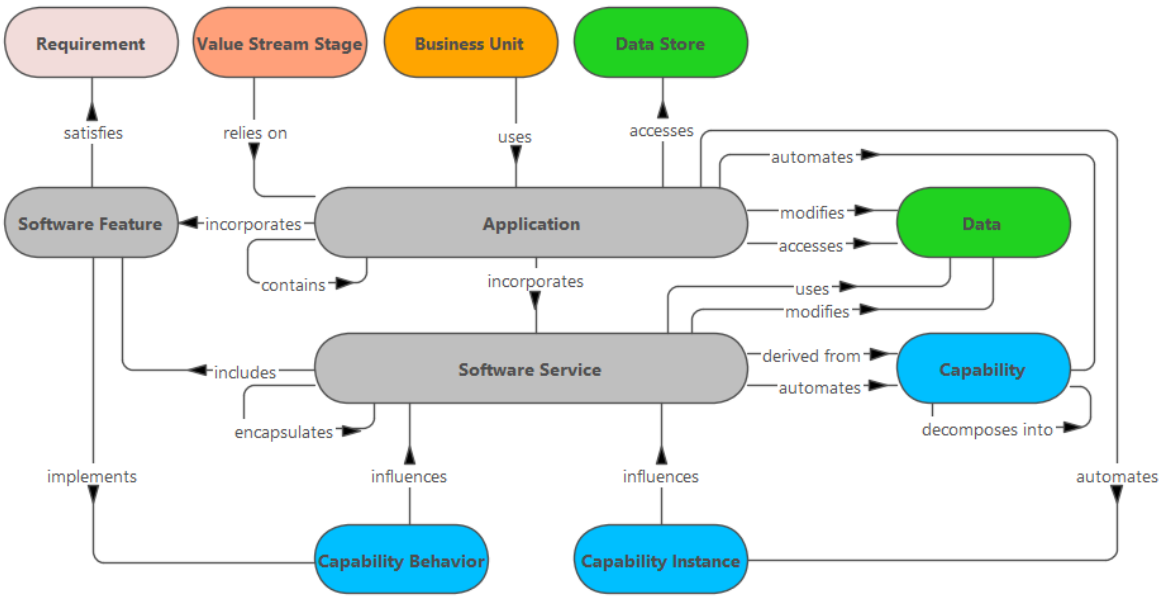

Figure 22: Business and IT Alignment Relations - Application Domain

The points below recap the business architecture and application architecture domain associations presented herein.

APPLICATION

Application accesses Data
Application accesses Data Store
Application automates Capability
Application automates Capability Instance
Application contains **Application**
Application incorporates Software Feature
Application incorporates Software Service
Application modifies Data
Business Unit uses **Application**
Value Stream Stage relies on **Application**

SOFTWARE FEATURE

Application incorporates **Software Feature**
Software Feature implements Capability Behavior
Software Feature satisfies Requirement
Software Service includes **Software Feature**

SOFTWARE SERVICE

Application incorporates **Software Service**
Capability Behavior influences **Software Service**
Capability Instance influences **Software Service**
Software Service automates Capability
Software Service derived from Capability
Software Service encapsulates **Software Service**
Software Service includes Software Feature
Software Service modifies Data
Software Service uses Data

6.2 Business and IT Alignment Relationships - Data Domain

Figure 23 summarizes the business architecture and data architecture domain associations captured in this whitepaper.

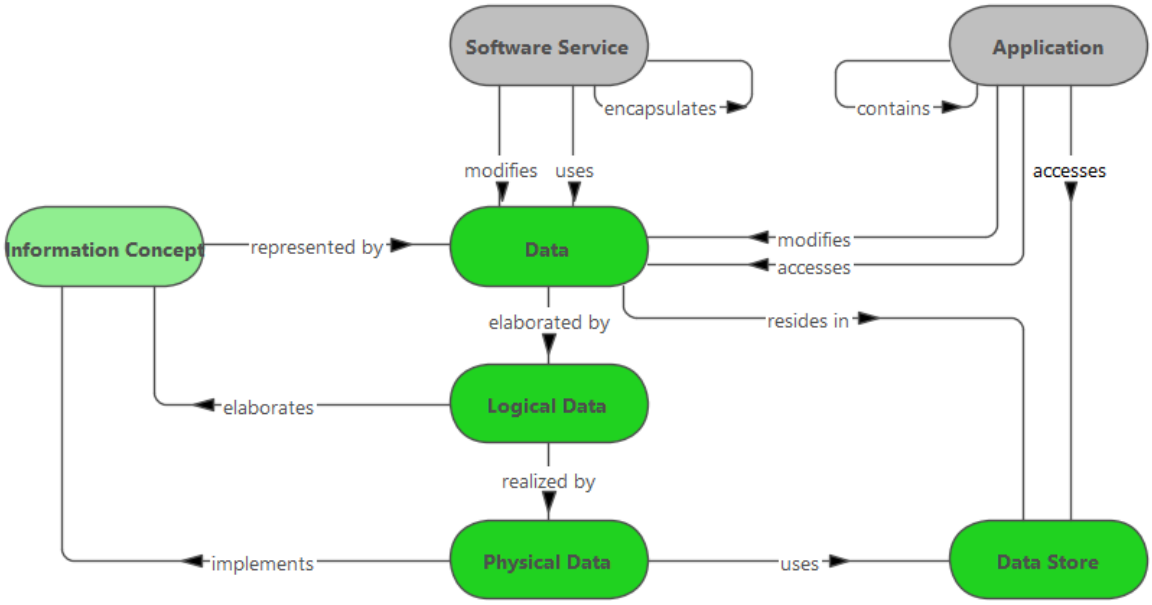

Figure 23: Business and IT Alignment Relations - Data Domain

The description below specifies comprehensively all outcoming and incoming relations for specific elements from data architecture with other elements mentioned in this whitepaper.

DATA
Application accesses **Data**
Application modifies **Data**
Data elaborated by Logical Data
Data resides in Data Store
Information Concept represented by **Data**
Software Service modifies **Data**
Software Service uses **Data**

LOGICAL DATA
Data elaborated by **Logical Data**
Logical Data elaborates Information Concept
Logical Data realized by Physical Data

PHYSICAL DATA
Logical Data realized by **Physical Data**
Physical Data implements Information Concept
Physical Data uses Data Store

DATA STORE
Application accesses **Data Store**
Data resides in **Data Store**
Physical Data uses **Data Store**

The metamodel definitions and associations defined in this whitepaper lay the foundation for planning and deployment teams to invest in business and IT transformation initiatives with greater transparency and confidence, ideally shifting the wasted IT investment dollars cited in the introduction into a more positive category. Organizations should seek out and implement the degree of formality defined herein to realize the benefits of this whitepaper.

7 Glossary of Terms

The following glossary defines certain terms used in this whitepaper. These terms are drawn from the multiple white papers, independent sources where appropriate, and *BIZBOK® Guide, Appendix D:* Glossary, which is an unrestricted, downloadable Business Architecture Guild® document at: https://cdn.ymaws.com/www.businessarchitectureguild.org/resource/resmgr/bizbok_14/v14_Glossary.pdf

Application

A collection of software assets that automates and enables a bounded set of capabilities and is identifiable by name and other characteristics.

Source: BIZBOK® Guide v14, Appendix D: Glossary.

Business Object

A representation of a thing, including at least its business name and definition, attributes, behavior, relationships, and constraints, which may represent (for example) a person, place, or concept.

Source: IMIS,

https://help.imis.com/enterprise/features/rise/business_objects/understanding_business_objects.htm#What_is_a_business_object.

Business Unit

A logical element or segment of a company (such as Accounting, Production, or Marketing) representing a specific business function and a definite place on the organizational chart under the domain of a manager. Also called Department, Division, or Functional Area.

Source: BIZBOK® Guide v14, Appendix D: Glossary.

Capability

A particular ability or capacity that a business may possess or exchange to achieve a specific purpose or outcome.

Source: A Business-Oriented Foundation for Service Orientation, Ulrich Homann, 2006,

https://cdn.ymaws.com/www.businessarchitectureguild.org/resource/resmgr/homann_article_on_capabiliti.pdf.

Capability Behavior

The way in which a capability acts or conducts itself in certain circumstances or instances.

Source: BIZBOK® Guide v14, Appendix D: Glossary.

Capability Instance

A specific realization of a capability, as it exists or is envisioned to exist, in the context of a given business unit or another situational context.

Source: BIZBOK® Guide v14, Appendix D: Glossary.

Change

A tangible alteration or modification in context of a business ecosystem.

Source: BIZBOK® Guide v14, Appendix D: Glossary.

Course of Action

A sequence of activities that an organization plans to pursue to achieve an objective.

Source: BIZBOK® Guide v14, Appendix D: Glossary.

Data

Value specifications for qualitative and quantitative variables.

Source: Definition sourced to Federation of Enterprise Architecture Professional Organizations (FEAPO), Taxonomy Working Group, 2017, https://cdn.ymaws.com/www.businessarchitectureguild.org/resource/resmgr/docs/feapo_adopted_architecture_d.pdf.

Data Store

A repository for persistently storing and managing collections of data, which include not just repositories like databases, but also simpler store types such as simple files, emails, and more.

Source: "Glossary D: Data Store". Information Management. Archived from the original on 2013-01-14. Retrieved 2011-04-04. A place where data is stored; data at rest. A generic term that includes databases and flat files.

Goal

An end toward which effort is or should be directed.

Source: Merriam-Webster Dictionary, https://www.merriam-webster.com/dictionary/goal.

Information Concept

The way to represent business terms and semantics within the context of business architecture.

Source: BIZBOK® Guide v14, Appendix D: Glossary.

Initiative

A course of action that is being executed or has been selected for execution.

Source: BIZBOK® Guide v14, Appendix D: Glossary.

Logical Data

Represents a detailed data structure composed of logically related data entities.

Objective

A quantitative, measurable result that defines strategy.

Source: BIZBOK® Guide v14, Appendix D: Glossary.

Organization

A social unit of people, systematically structured and managed to meet a need or to pursue collective goals on a continuing basis.

Source: BIZBOK® Guide v14, Appendix D: Glossary.

Outcome

An end-result or final product that is a consequence of an event, action, or a series of events and actions.

Source: BIZBOK® Guide v14, Appendix D: Glossary.

Physical Data

Represents the data structure derived from logical data and specifies the format or schema required by a particular technology.

Policy

A course or principle of action adopted or proposed by a government, party, business, or individual.

Source: Oxford Dictionaries, Oxford University Press, s.v.

Product

The overall experience provided by the combination of goods and services to satisfy the customer's needs.

Source: Greg Geracie and Stephen Eppinger, The Guide to the Product Management and Marketing Body of Knowledge® (ProdBOK®) (Association of International Product Marketing and Management, 2013).

Product Entitlement

A specified aspect of a product that represents an inherent commitment made by an organization to a customer that is realized as an immediate, on demand, or continuing obligation of the organization to the customer that acquired that product.

Source: BIZBOK® Guide v14, Appendix D: Glossary.

Product Line

Products may belong to a product line with similar characteristics or that target a particular buyer.

Source: BIZBOK® Guide v14, Appendix D: Glossary.

Requirement

A condition needed by a stakeholder to solve a problem or achieve an objective.

Source: BIZBOK® Guide v14, Appendix D: Glossary.

Software Feature

A *feature* is a unit of functionality of a software system that satisfies a requirement, represents a design decision, and provides a potential configuration option.

Source: Introduction of Overview of Feature-Oriented Software Development - Sven Apel, Christian Kästner.

Software Service

A self-contained unit of software that performs a specific task with three components: an interface, a contract, and an implementation.

Source: Service Oriented Architecture Field Guide for Executives, Wiley, Jan. 2012, Kyle Gabhart and Bibhas Bhattacharya.

Stakeholder

An internal or external individual or organization with a vested interest in achieving value through a particular outcome.

Source: BIZBOK® Guide v14, Appendix D: Glossary.

Stakeholder Category

A fundamental and distinct classification to which a stakeholder is assigned, and for which information must be managed.

Source: BIZBOK® Guide v14, Appendix D: Glossary.

Strategy

The pattern or plan that integrates an organization's major goals, policies, and action sequences into a cohesive whole.

Source: J. B. Quinn, Strategies for Change: Logical Incrementalism (Homewood: Richard D. Irwin, Inc., 1980).

Value Item

The judgment of worth, made by an individual or organization, attached to something tangible or intangible, and attained in the course of a particular interaction with one or more parties.

Source: BIZBOK® Guide v14, Appendix D: Glossary.

Value Proposition

An innovation, service, or feature intended to make a company, product, or service attractive to customers or related stakeholders.

Source: BIZBOK® Guide v14, Appendix D: Glossary.

Value Stream

An end-to-end collection of activities that create a result for a customer, who may be the ultimate customer or an internal end-user of the value stream.

Source: The Great Transition: Using the Seven Disciplines of Enterprise Engineering to Align People, Technology, and Strategy, Martin, James, 1995, AMACOM, ISBN-10: 0814403158.

Value Stream Stage

A distinct, identifiable phase or step within a value stream that has a unique name, entrance criteria, exit criteria, and identifiable participating stakeholder(s).

Source: BIZBOK® Guide v14, Appendix D: Glossary.

8 About the Business Architecture Guild®

The Business Architecture Guild® is an international, not-for-profit, member-driven professional association that provides valuable resources to business architecture practitioners and others interested in the profession. Formed in 2010, the Guild's primary purpose is to promote best practices and expand the knowledgebase of the business architecture discipline. The Guild is the source for *A Guide to Business Architecture Body of Knowledge®* (*BIZBOK® Guide*), the go-to guide for business architecture practitioners and other professionals seeking to leverage the discipline, and home of the Certified Business Architect® (CBA®) certification program.

The Guild is active in industry standards programs and partners with related professional associations to further its purpose. In addition to the *BIZBOK® Guide*, the Guild offers a Business Architecture Maturity Model® (BAMM®), Business Architecture Tool Evaluator™, and business architecture reference models for various industry sectors, which include financial services, government, healthcare, insurance, manufacturing, telecommunications, transportation, and common industry reference models. All Guild-produced content, including the industry reference models, is developed by its members. In addition to these resources, the Guild has a vendor partner program, a Guild Accredited Training Partner® (GATP®) program, and an academic program.

For more information and more details, visit www.businessarchitectureguild.org.

9 Endnote References

[1] Business Architecture Metamodel Guide V3.0, www.businessarchitectureguild.com.

[2] Javier David, Nearly 70 Percent of Tech Spending Is Wasted, October, 2015, https://www.vox.com/2015/10/31/11620222/study-nearly-70-percent-of-tech-spending-is-wasted.

[3] Business/IT Transformation Framework, Source: TSG, Inc., extracted from *BIZBOK® Guide* section 6.8.

[4] Federation of Enterprise Architecture Professional Organizations (FEAPO), https://cdn.ymaws.com/www.businessarchitectureguild.org/resource/resmgr/docs/feapo_adopted_architecture_d.pdf.

[5] Business Architecture Metamodel Guide V3.0, www.businessarchitectureguild.com.

[6] *BIZBOK® Guide* Version 14 Appendix A: Glossary, page 771.

[7] William Ulrich and Neal McWhorter, *Business Architecture: The Art & Practice of Business Transformation*, Tampa: MK Press, 2011, p. 133. Note that a comma was added after the term "data" in this definition.

[8] Business/IT Transformation Framework, Source: TSG, Inc., extracted from *BIZBOK® Guide* section 6.8.

[9] William Ulrich and Neil McWhorter, Business Architecture: The Art and Practice of Business Transformation, January 2011, Meghan Kiffer Press.

[10] Federation of Enterprise Architecture Professional Organizations (FEAPO), Taxonomy Working Group, 2017. https://cdn.ymaws.com/www.businessarchitectureguild.org/resource/resmgr/docs/feapo_adopted_architecture_d.pdf.

[11] *BIZBOK® Guide* Version 14, section 2.2, p 64.

[12] *BIZBOK® Guide* Version 14, section 2.2, p 64.

[13] An Overview of Feature-Oriented Software Development, Sven Apel and Christian Kästner, https://www.jot.fm/issues/issue_2009_07/column5/index.html.

[14] *BIZBOK® Guide* Version 14, Appendix A: Glossary, p 779.

[15] See Normative Reference for the term Application in ISO/IEC 16350:2015 and updated in ISO/IEC 24570:2018.

[16] *BIZBOK® Guide* Version 14, v14, p 501.

[17] Service Oriented Architecture Field Guide for Executives, Wiley, Jan. 2012, Kyle Gabhart and Bibhas Bhattacharya.

[18] Federation of Enterprise Architecture Professional Organizations (FEAPO), Taxonomy Working Group, 2017. https://cdn.ymaws.com/www.businessarchitectureguild.org/resource/resmgr/docs/feapo_adopted_architecture_d.pdf.

[19] Business Architecture Metamodel Guide V3.0, www.businessarchitectureguild.com.

[20] Business Architecture Metamodel Guide V3.0, www.businessarchitectureguild.com.

[21] *BIZBOK® Guide* Version 14, section 2.1.

