LUNDS UNIVERSITET
Ekonomihögskolan

Institutionen för informatik

Analytical Infrastructure Under Resource Constraints

An Integrated Mixed-Methods Assessment of Organizational Requirements and Database Architecture Performance

Kandidatuppsats 15 hp, kurs SYSK16 i Informationssystem

Författare: Jack Magnell
Nils Törnqvist
Felix Bergman

Handledare: Markus Lahtinen

Rättande lärare: Björn Svensson
Paul Pierce

ENGELSK TITEL: Analytical Infrastructure Under Resource Constraints: An Integrated Mixed-Methods Assessment of Organizational Requirements and Database Architecture Performance

FÖRFATTARE: Jack Magnell, Nils Törnqvist, Felix Bergman

UTGIVARE: Institutionen för informatik, Ekonomihögskolan, Lunds universitet

EXAMINATOR: Osama Mansour, Docent

FRAMLAGD: augusti, 2026

DOKUMENTTYP: Kandidatuppsats

ANTAL SIDOR: 131

NYCKELORD: Analytical Infrastructure, Resource Constraints, Decision Support, Mixed Methods, HTAP, OLTP, OLAP, Columnstore Indexing, TPC-H

SAMMANFATTNING (MAX. 200 ORD):
This thesis examines the suitability of analytical infrastructure for decision support in organizations with limited dedicated analytics capacity. The study uses an integrated mixed-methods design combining semi-structured interviews with a controlled TPC-H benchmark. The interviews identify organizational requirements, infrastructure and data flows, resource constraints, and priorities shaping architectural suitability. The benchmark evaluates analytical performance across three database configurations: a normalized OLTP schema with columnstore indexing, a dimensional OLAP schema with columnstore indexing, and a dimensional OLAP schema with rowstore indexing.

The findings show that architectural suitability depends on technical performance alongside factors such as competence, data movement, coordination, maintainability, and analytical requirements. The benchmark shows that the columnstore-indexed OLTP configuration performed comparably to the dimensional columnstore configuration for many analytical queries, while some join-intensive workloads favoured the dimensional schema.

By interpreting benchmark performance in relation to the organizational requirements identified through the interviews, the study shows that integrated HTAP-oriented architectures may be suitable under some conditions, while separated OLAP architectures remain advantageous where workload or organizational requirements justify them. The study contributes to Information Systems research by demonstrating how organizational requirements, resource

constraints, and technical performance can be considered together when assessing analytical infrastructure for decision support.

Table of Contents

Tables

1 Introduction

1.1 Background

Business intelligence and analytics (BI&A) systems are used to transform organizational data into information and insights that can support decision-making. A recent literature review by Ain et al. (2025), synthesizing 173 BI&A studies published between 2000 and 2024, describes decision support as a central purpose of these systems and examines their success through outcomes extending beyond the technical operation of the system itself. BI&A also has a direct conceptual relationship with the Information Systems field of decision support systems (DSS). Phillips-Wren et al. (2021) argue that BI&A can be understood as part of the broader DSS research tradition. They classify BI&A and DSS publications into four categories: "conceptual framework, design & implementation, business value & organizational use, and cognition & decision making" (Phillips-Wren et al., 2021, p.4). Analytical infrastructure is therefore relevant to Information Systems not merely because it processes data, but because it forms part of the arrangements through which data are made available for organizational decision-making (Ain et al., 2025; Phillips-Wren et al., 2021).

The ability to get value from analytics, however, cannot be reduced to access to analytical technology. Gupta and George (2016) define big data analytics capability "as a firm's ability to assemble, integrate, and deploy its big data-specific resources" (p.1049). Their framework distinguishes between tangible resources, including data, technology, time, and investment; human resources, including technical and managerial skills; and intangible resources, including a data-driven culture and organizational learning. This multidimensional view remains influential in more recent research. Huynh et al.'s (2023) systematic review of the big-data analytics capability literature similarly treats analytics capability as an organizational capability composed of multiple complementary elements rather than as the possession of a particular technological artefact. Recent empirical IS research further supports the importance of this complementarity: Abubakar et al. (2025) found that business-analytics capabilities and the competencies of key personnel jointly contribute to the development of a data-oriented culture and subsequent organizational outcomes. Taken together, these studies provide a basis for distinguishing analytical technology from the broader organizational capacity to use and sustain it (Gupta & George, 2016; Huynh et al., 2023; Abubakar et al., 2025).

This distinction is important to the concept of resource constraints used in this thesis. Here, resource constraints do not refer exclusively to financial limitations or to a particular category of firm size. Instead, the term refers to limitations in the resources and capabilities that can be dedicated to establishing, operating, and maintaining analytical decision support. Depending on the setting, these may include limited specialist competence, personnel, time, technical capacity, organizational readiness, or financial resources. This definition is consistent with the multidimensional view of analytics capability described by Gupta and George (2016) and Huynh et al. (2023). It is also consistent with contemporary research on BI adoption. Al-Daraba et al. (2025), in a systematic review of 68 studies published between 2015 and 2024, identify organizational readiness, compatibility, complexity, management support, and

perceived ease of use among recurrent determinants of BI adoption. Their review therefore shows that the conditions surrounding an information system remain relevant alongside the characteristics of the technology itself.

Research on small and medium-sized enterprises provides one particularly visible example of such constraints, but it does not exhaust the phenomenon studied here. Justy et al. (2023) interviewed 35 respondents at 32 SMEs. They conclude that firms with fewer in-house data skills, no clear strategy for using data, and a culture resistant to it were the least likely to adopt analytics. Difficulties finding vendors or staff, and problems with data volume or quality mattered less by comparison. Earlier SME research has likewise examined BI adoption as the result of combinations of technological, organizational, and environmental conditions rather than firm size alone (Puklavec et al., 2018). These studies remain useful to this thesis because they demonstrate how constrained analytical capacity can affect the feasibility of analytics initiatives. However, this thesis does not use the formal SME category as its population boundary. Instead, it focuses on organizational settings in which the resources that can be dedicated to analytical work are limited relative to the analytical requirements that must be supported. SME research is therefore treated as evidence concerning one form of resource-constrained setting, rather than as a basis for assuming that resource constraints are unique to small firms.

Once analytical capability is understood in this way, the design of analytical infrastructure becomes more than a question of selecting the technically fastest database configuration. Infrastructure determines how organizational data are stored, transferred, prepared, and made available for analysis, and different architectural choices can place different demands on the organization that operates them. Foundational empirical IS research by Ariyachandra and Watson (2010) demonstrates this relationship directly in the context of data warehousing. Their survey of 400 organizations found that architecture selection varied with organizational conditions and that whether the warehouse was seen as strategic infrastructure or as a short-term solution had the most consistent effect on which architecture was chosen. Their results do not imply that technical characteristics are irrelevant, but they do demonstrate that architecture selection cannot be understood solely as a technical optimization problem. This continues to align with more recent BI research in which compatibility, complexity, and organizational readiness remain recurrent factors affecting whether analytical systems can be adopted and used (Al-Daraba et al., 2025). Architectural suitability can therefore be understood as a question of fit between what an analytical infrastructure provides and what an organization is able and needs to support (Ariyachandra & Watson, 2010; Al-Daraba et al., 2025).

One important architectural issue concerns the relationship between transactional and analytical processing. A table can be read along two dimensions: rows and attributes. Song et al. (2024) describe Online Transaction Processing (OLTP) and Online Analytical Processing (OLAP) as reading along these dimensions in opposite ways. Transactional work is narrow in rows and wide in attributes. Analytical work is the reverse. Song et al. (2024) treat this difference as the reason the two workload types require different optimizations. Transactional databases are commonly designed to support frequent operational updates, whereas analytical environments are designed for read-intensive decision-support workloads. Fotache et al. (2026) tested this assumption under TPC-H workloads and found the opposite of what the conventional trade-off predict: higher normal forms produced faster query execution and higher rates of successful query completion than lower normal forms even though lower normal forms require fewer joins. The distinction between OLTP and OLAP therefore reflects different

workload requirements rather than merely different labels for database systems (Song et al., 2024; Fotache et al., 2026).

A conventional response to these different requirements is to maintain operational and analytical processing in separate environments. In this arrangement, data generated by operational systems are transferred to an analytical environment where they can be prepared and queried for decision-support purposes. Extract-Transform-Load (ETL) processes have traditionally provided this connection. Separation allows transactional and analytical environments to be structured and optimized for their respective workloads, but the architecture also requires the organization to operate the process through which data are transferred and transformed. Song et al.'s (2024) recent survey of Hybrid Transactional/Analytical Processing characterizes traditional ETL as a time-consuming process and explains that periodic ETL can limit the freshness of the data available for analysis. The same survey notes that OLTP and OLAP have substantially different access patterns and optimization demands, which explains why simply combining the workloads is not technically trivial. Separation therefore offers genuine technical advantages, but it also creates an additional boundary that must be managed between operational data generation and analytical use.

The demands associated with such infrastructure are not limited to machine resources. Data warehousing research has long shown that organizational resources and competence affect implementation outcomes. Wixom and Watson's (2001) survey of 111 data warehouse projects found that resources and user participation predicted success with both organizational and project-level implementation issues, and that team skills predicted project-level success but not technical success. Source systems quality and development technology quality predicted technical implementation success (Wixom & Watson, 2001). Although the technological landscape has changed substantially since that study, contemporary BI research continues to identify organizational readiness, compatibility, complexity, and human or organizational conditions as relevant to system adoption and use (Al-Daraba et al., 2025). The older data-warehousing evidence is therefore not used here to describe the specific tools of current analytics environments, but to establish a persistent IS concern: analytical infrastructures depend on organizational work, skills, and coordination in addition to database technology.

This relationship is particularly relevant where dedicated analytical capacity is limited. A separated architecture may provide an analytically optimized environment, consolidated data, and workload separation, while simultaneously requiring an organization to sustain an additional analytical environment and the processes connecting it to operational systems. Whether these requirements constitute a meaningful burden cannot be determined from the architecture alone; it depends on the resources available and on what the organization actually needs from its analytical infrastructure. An organization that requires complex historical analysis across several source systems may derive substantial value from separation, while an environment with different analytical requirements may place greater value on reducing data movement or infrastructure complexity. These possibilities should therefore be treated as conditions to investigate rather than assumptions made in advance. The relevant IS question is therefore not whether integration or separation is universally preferable. Ariyachandra and Watson (2010) show that data warehouse architecture selection depends on organizational conditions while Song et al. (2024) show that no single database architecture performs best across all workload conditions.

Hybrid Transactional/Analytical Processing (HTAP) has developed in response to some of the limitations of strict transactional–analytical separation. Song et al. (2024) describe HTAP as running transactional and analytical workloads inside one system which removes the ETL

step that would otherwise move data into a separate analytical environment and allows analytical queries to run on recently written data. Their survey also emphasizes, however, that integration creates its own technical challenges. OLTP and OLAP have different access patterns and execution requirements, and an HTAP system must manage issues such as workload interference, data formats, resource isolation, and query execution. More integrated processing should consequently not be treated as an uncomplicated replacement for dedicated analytical systems. It represents an alternative architectural design with a different combination of potential advantages and trade-offs (Song et al., 2024).

One technological development that makes this question particularly relevant is the use of column-oriented storage structures for analytical workloads. Column-oriented processing can reduce the amount of unnecessary data accessed by queries that scan many records but require only selected attributes, making it relevant to analytical processing. Modern HTAP architectures consequently employ combinations of row- and column-oriented structures in different ways to reconcile transactional and analytical requirements (Song et al., 2024). At the same time, recent empirical research suggests that assumptions about database structure and analytical performance should not be taken for granted. Fotache et al. (2026), examining different degrees of normalization in decision-support databases using TPC-H workloads, note that empirical evidence on the performance implications of normalization in decision-support systems remains limited. Their results challenge a simple assumption that reducing joins through greater redundancy necessarily improves analytical performance, since the additional data volume can itself impose performance costs. Although their comparison does not answer the architectural question examined in this thesis directly, it demonstrates that the relationship between database design and decision-support performance remains empirically relevant rather than theoretically settled (Fotache et al., 2026).

The resulting architectural decision therefore contains both an organizational and a technical dimension. Organizational evidence is needed to establish what analytical infrastructure must support: the kinds of decisions for which data are used, the required freshness of that data, the nature of the analytical workloads, the difficulties associated with current infrastructure, and the resources and criteria that shape feasible alternatives. Technical evidence is then needed to establish how candidate architectures actually perform when subjected to analytical workloads. Either form of evidence on its own provides only part of the assessment. Practitioner accounts can identify organizational requirements and constraints but cannot establish the comparative query performance of alternative database configurations. Conversely, a controlled benchmark can establish performance differences but cannot determine whether those differences are important enough to justify the additional organizational requirements associated with a particular architecture. This distinction follows the broader IS position that BI&A involves design and implementation as well as organizational use and decision-making, and the empirical finding that architecture selection is affected by organizational context (Phillips-Wren et al., 2021; Ariyachandra & Watson, 2010).

This thesis addresses analytical infrastructure under resource constraints through these two interdependent perspectives. Semi-structured interviews are used to investigate how practitioners describe analytical decision support, existing infrastructure and data flows, current constraints, and priorities when analytical capacity is limited. These findings establish the organizational requirements and conditions against which architectural suitability can be interpreted. A controlled TPC-H benchmark then compares the analytical performance of an integrated, HTAP-oriented configuration with dedicated analytical configurations. The benchmark is not intended to determine architectural suitability independently. Instead, its performance evidence is interpreted in relation to the organizational requirements and constraints

established through the interviews. Architectural fit is therefore the common object of the study: the interviews establish what an appropriate analytical infrastructure must accommodate, while the benchmark evaluates one important technical dimension of whether the tested architectural alternatives can do so.

Because the study combines organizational and database-architectural concepts, Table 1.1 summarizes the central technical terms used throughout the thesis.

Table 1.1: Key concepts and technical terms used in this thesis

Term / Abbreviation	**Meaning**	**Use in this thesis**
Analytical infrastructure	Systems and processes that make organizational data available for analysis and decision support	Refers to the technical arrangements through which data are stored, prepared, transferred, and made available for reporting and analysis.
Architectural fit	Suitability of an architecture in relation to its context of use	Refers to how well an analytical architecture matches organizational requirements, resource constraints, and relevant technical performance characteristics.
Resource constraints	Limitations in resources and capabilities available for analytical work	Refers specifically to limitations in dedicated analytical capacity, including relevant human, organizational, technical, and financial resources, as defined in Section 1.1.
DSS	Decision Support System	An information system that provides information or analytical support for human organizational decision-making.
OLTP	Online Transaction Processing	Processing associated primarily with operational activities involving frequent transactions, inserts, updates, and retrieval of relatively small amounts of data.
OLAP	Online Analytical Processing	Processing associated primarily with analytical workloads such as aggregation, reporting, and analysis across larger volumes of data.
HTAP	Hybrid Transactional/Analytical Processing	An architectural approach designed to support transactional and analytical processing within a closely integrated database environment.

HTAP-oriented	Architecture incorporating characteristics associated with HTAP	Used for the integrated configuration evaluated in this thesis. It is described as HTAP-oriented because the benchmark evaluates analytical performance but does not test simultaneous transactional and analytical workloads.
ETL	Extract, Transform, Load	A process in which data are extracted from source systems, transformed into a suitable form, and loaded into another environment, commonly an analytical database or data warehouse.
TPC-H	TPC Benchmark H	A standardized decision-support benchmark containing a defined schema, generated data, and analytical business-oriented queries.
3NF	Third Normal Form	A normalized relational design form intended to reduce redundancy and certain data-modification anomalies.
Star schema	Dimensional schema structure	An analytical data model in which a central fact table is connected to dimension tables containing descriptive information.
Columnstore index	Column-oriented indexing/storage structure	Stores or indexes values by column rather than primarily by row and is designed to support efficient analytical processing over selected attributes and large numbers of records.

1.2 Problem Formulation

The suitability of analytical infrastructure cannot be determined solely by technical performance. Research on analytics capability shows that organizations depend on combinations of technological resources, human skills, and organizational capabilities to make effective use of analytics (Gupta & George, 2016; Huynh et al., 2023). Research on data warehouse architecture also indicates that architectural choices are influenced by organizational conditions such as available resources and IT capability (Ariyachandra & Watson, 2010). This means that a technically strong architecture may still be unsuitable if it requires resources or competencies that an organization cannot realistically provide.

This issue becomes particularly relevant in settings with limited dedicated analytical capacity. Conventional separation between transactional and analytical processing can provide workload isolation and structures optimized for decision support, but it also requires data

movement, transformation, maintenance, and coordination between operational and analytical environments. More integrated HTAP-oriented approaches may reduce some of these demands by bringing transactional and analytical processing closer together, although this can introduce technical trade-offs in performance and workload management (Song et al., 2024).

The central problem is therefore not simply whether an integrated architecture can execute analytical workloads, nor whether organizations experience limitations in their current analytical infrastructure. Rather, the problem is how technical performance should be interpreted in relation to the organizational requirements and resource constraints that shape architectural suitability. Organizational studies can identify what organizations need from analytical infrastructure and what limits their ability to sustain it, but they cannot determine the comparative performance of specific database configurations. Conversely, controlled benchmarks can demonstrate performance differences, but cannot establish whether those differences are meaningful in a particular organizational context.

This thesis addresses this problem by combining semi-structured interviews with a controlled TPC-H benchmark. The interviews identify analytical requirements, current infrastructure, constraints, and priorities, while the benchmark provides technical performance evidence for integrated and separated database architectures. The two forms of evidence are subsequently interpreted together in order to assess architectural fit for analytical decision support under resource constraints.

1.3 Research Question

RQ1: How do organizational requirements and resource constraints shape the suitability of analytical infrastructure for decision support?

RQ2: How do benchmarked performance differences between integrated HTAP-oriented and separated OLAP architectures inform their suitability under the conditions identified in RQ1?

1.4 Purpose

The purpose of this thesis is to assess the suitability of analytical infrastructure for decision support under resource constraints by integrating organizational requirements with technical evidence on database architecture performance. Through semi-structured interviews, the study identifies how organizational requirements, existing infrastructure, and resource constraints shape what organizations with limited dedicated analytics capacity need from their analytical infrastructure. These findings establish the conditions against which architectural suitability is assessed.

A controlled TPC-H benchmark is then used to compare the analytical performance of an integrated HTAP-oriented database architecture with separated dedicated OLAP alternatives. The integrated architecture is represented by a normalized OLTP schema enhanced with columnstore indexing, while the separated analytical architecture is represented by dimensional OLAP schemas using both columnstore and rowstore indexing. The benchmark is not treated as an independent evaluation of architectural suitability. Instead, its results are interpreted in relation to the requirements and constraints identified through the interviews. By combining

these two forms of evidence, the study aims to assess when a more integrated analytical architecture may constitute a suitable alternative to a separated analytical environment, and under which conditions the advantages of a dedicated OLAP architecture remain important.

1.5 Delimitations

In this thesis, analytical decision support is delimited to the use of data, reporting and analytical information to inform human organizational decision-making. This corresponds to the broader BI&A and decision-support literature, which treats analytical systems in relation to the production and use of information for decision-making (Phillips-Wren et al., 2021; Ain et al., 2025). Automated decision-making systems and the performance of machine-learning or artificial-intelligence models are therefore outside the scope of the study.

For the purposes of this thesis, resource constraints are delimited to limitations in the resources and capabilities that can be dedicated to analytical work, rather than being operationalized through organizational size. This delimitation is informed by analytics-capability research that conceptualizes capability in terms of multiple technological, human and organizational resources (Gupta & George, 2016; Huynh et al., 2023).

The architectural assessment is further delimited to relational database architectures and the degree of separation between transactional and analytical processing. RQ2 consequently concerns an integrated HTAP-oriented relational architecture in relation to a separated dedicated OLAP architecture. Alternative database paradigms, including NoSQL systems, data lakes, and distributed big-data platforms, are outside the architectural scope of the study.

2 Literature Review

2.1 Decision Support and Analytical Capability in Information Systems

Business intelligence and analytics (BI&A) is an established research area within Information Systems. Chen et al. (2012) published an MIS Quarterly special issue on this topic and described how the technology has developed through several generations. They also state that a BI&A project does not succeed based on technology alone. Projects also need people who understand the business and can communicate, and they see this combination of technology and business as a reason why IS departments should study the area.

Sharma et al. (2014) criticize two assumptions that are often taken for granted. Firstly, that better insights from data automatically lead to better decisions and that large amounts of data therefore lead to a large effect on the business. Secondly, that an organization can get value from analytics without changing anything else about how it works. They argue that the latter assumption failed for earlier technologies where both enterprise systems and knowledge management systems only delivered benefits after organizations changed their processes and structures too. Instead, Sharma et al. (2014) describe the path from data to value in three stages: (1) data to insight, (2) insight to decision, and (3) decision to value. Each stage can fail on its own (Sharma et al., 2014). In the first stage, they argue that an insight is not something a tool produces on its own. Before raw data turns into an insight, analysts and managers have to work on it together, and what determines how well that goes depends on the decision-making structures and routines that already exist in the organization.

Grover et al. (2018) note that it is still not clear how often big data analytics projects succeed and how much value they actually create. Their framework therefore has two separate processes. One for building analytics capability and one for realizing value from it (Grover et al., 2018). This separation is important because it means that simply buying and installing tools is only the first step, and value does not automatically follow.

Other studies point in the same direction. Popovic et al. (2012) surveyed 181 medium and large organizations and found that more mature BI systems mainly improved access to information, but access itself had no significant effect on whether the information was actually used in business processes. What did affect utilization was the organization's analytical decision-making culture, meaning how used the organization is to basing decisions on data (Popovic et al., 2012). Chen et al. (2015) found that value creation is connected to the actual use of analytics rather than simply having the technology, and that organizational readiness only affects use indirectly through support from top level management. Corte-Real et al. (2017) found that analytics affect performance through knowledge management and organizational agility.

Bozic and Dimovski (2019) interviewed respondents from nine medium and large organizations. According to their respondents, an organization ideally wants analysts who have both technical and business competence in one and the same person, but such people are difficult to find. The technical staff that organizations manage to hire tend to have their education in purely technical fields such as computer science, statistics, or mathematics rather than

business (Bozic & Dimovski, 2019). Because of this, they can work with the data, but struggle to say what it means for the organization (Bozic & Dimovski, 2019). The organizations in the study therefore handle this through how they design their teams: a team is comprised of both technical members and business managers (Bozic & Dimovski, 2019). In other words, they solve the problem by hiring more employees.

Looking at these studies, we can draw two conclusions. Firstly, two organizations can have the same tools and still get different results because the results also depend on whether the organization has both technical and business competence. Secondly, the solution described above, i.e. hiring one person of each kind, only works for organizations that can afford it. The studies above cover medium and large organizations that can often afford this. This is a gap that this thesis addresses.

The choice of analytics infrastructure is not only a technical question but also an organizational one, and IS research has tested this. Ariyachandra and Watson (2010) studied how organizations choose data warehouse architecture. They found that the three factors that best predicted the choice were how strategically the organization viewed the system, how many resources it had, and how capable it believed its own IT staff to be. Factors such as information interdependence between units and executive sponsorship only affected the choice indirectly through the strategic view (Ariyachandra & Watson, 2010). Their research model only contained organizational factors so it cannot show that technical factors are not important. But it does show that organizational factors have considerable weight in a decision that is often treated or perceived as technical. March and Hevner (2007) describe decision-support infrastructure as layers where each layer depends on the layers below it. Two of their points are especially relevant here. Firstly, that "effective integration decisions can only be made by those thoroughly familiar with the domain ontologies both internal and external to the business system" (March & Hevner, 2007, p.1041), in other words by people who know what the data means in the context of the business. Secondly, they place end-user training and support among the challenges of the use layer, i.e. the layer where the system is actually used. From these two studies, we can draw a conclusion that the choice of architecture also decides which in-house skills the organization must have, and an organization can choose an architecture that requires more skills than it has in practice.

The rest of this literature review builds on this in two steps: sections 2.2 and 2.3 examines what actually limits analytical capabilities in organizations with limited dedicated analytics capacity and whether it is the same limitations these organizations report themselves. Then, sections 2.4 to 2.7 examines which architectural response follows from this limitation and how it can be tested.

2.2 Analytical Capacity as the Binding Constraint: Cost, Competence, Capability

Tsiu et al. (2025) conducted a systematic review of 93 studies on data mining and business intelligence (BI) in SMEs on a global level, published in recent years (2014 to 2024). These tools allow SMEs to analyze large volumes of data resulting in competitive advantages through the discovery of actionable insights and competitive advantage (Tsiu et al., 2025). Kgakatsi et al. (2024) reviewing the same 93 studies report a similar finding: that big data adoption in SMEs is accompanied by measurable gains "in operational efficiency, revenue generation, and competitiveness". Because the two reviews share a corresponding author and

identical underlying study data, what may look like an independent confirmation of two different papers agreeing is actually more of a one research group agreeing with itself. Maroufkhani et al. (2020) working on a separate sample of SME survey data document similar advantages, specifically in financial and marketing performance.

Tsiu et al. (2025) also quantify the advantages across different performance dimensions ranked by certainty of evidence: with high certainty, BI utilization is associated with an average of 15% revenue growth and a 20% improvement in risk management outcomes; with moderate certainty, a 10% increase in customer retention, an 8% mean difference in new product success rates, a 12% mean difference in process completion times, and a 1.8 risk ratio for improved decision-making quality; and with low certainty, a hazard ratio of 1.5 for faster market adaptation through data mining. These findings suggest that financial and risk outcomes are the biggest advantages of analytics in SMEs. However, Champa and Segall (2026) complicate this. In their review, only a quarter of the studies reported no cost savings at all, and when savings did appear, they were rarely why the firm had invested in the first place. The value they did find was concentrated elsewhere: 69% of the reported outcomes had to do with how firms plan and decide, not with what they save (Champa & Segall, 2026). The two reviews aren't fully comparable since Tsiu et al.'s (2025) numbers are effect-size estimates while Champa and Segall's (2026) are proportions of studies reporting a given outcome. Still, this suggests that SMEs may get financial returns like Tsiu et al.'s (2025) 15% revenue growth mostly as a byproduct of perhaps making better strategic decisions rather than because cost reduction was ever the goal. This might also explain why cost is the most common reason that SMEs give for not adopting analytics even though it turns out to be one of the smaller benefits once they do adopt it (Champa & Segall, 2026).

The difference between owning analytics tools and getting value out of them has a name in IS research: *analytics capability*. Cosic et al. (2015) argue that it is capabilities rather than the technology itself that likely give a firm an advantage. In their framework, capability arises when tools are combined with the firm's people, culture, and governance, and it is that specific combination competitors struggle to copy. Mikalef et al. (2018) reviewed 84 studies on this and found that the research says a lot about which resources a firm needs but very little about how firms actually combine them. For an organization with no dedicated analytics staff, this is the hard part.

Kgakatsi et al. (2024) describes financial limitations as a "major hurdle" along with limited technical knowledge, cultural resistance to change, and low technological maturity. However, two other studies disagree about the financial limitations. Caldeira and Ward (2002) studied 12 Portuguese manufacturing SMEs and found that financial limitations did not emerge as the primary barrier to adoption. It was rather the top management's attitude towards IS/IT and the in-house IS/IT skills that separated the most successful firms from the least successful. Liu et al. (2020) had similar findings from a UK manufacturing sample where cost was noted as a barrier, but only half of the respondents were even aware that big data analytics even existed. Gupta and George (2016) explain that this pattern keeps coming back. They define big data analytics capability as a "firm's ability to assemble, integrate, and deploy its big data-specific resources" (Gupta & George, 2016, p.1049). In their model, three kinds of resources are needed at the same time: tangible resources like data, technology, money, and time; human skills, both technical and managerial; and intangible resources, meaning a data-driven culture and a habit of organizational learning. The point is that these do not substitute for each other. A firm can have all the data and technology in the world and still get nothing out of it if nobody knows what to do with it, or if decisions are made on a gut feeling anyway. Justy et al. (2023) conducted 35 interviews at 32 SMEs and found that internal factors such as mismatch

between business strategy and IS, shortage of analytics skills, lack of data-driven culture, hurt adoption more than external barriers such as shortage of specialized service providers or skilled labor.

This matches what Gupta and George (2016) predict: what organizations were missing were the human and intangible resources, and those are exactly the ones that a firm cannot just buy. Champa and Segall (2026) offer possible explanations for this pattern rather than treating it as settled. One is that the literature itself may be skewed: vendor-funded studies have reasons to emphasize growth stories over savings. Another explanation is that whatever savings BI does produce are spread out and hard to trace back to the system so they go unreported. Even with this insight, both old and new evidence suggests that competence and organizational readiness is the real constraint rather than cost alone even though cost is still the most mentioned barrier.

This shift away from cost also shows up outside of the SME literature too: Ain et al. (2019) reviewed 111 BI studies and found that most research focuses on organizational and IS-related factors while human factors have been largely ignored even through user resistance, low motivation, and missing technical skills keep appearing as the real reasons why systems succeed or fail. Al-Daraba et al. (2025) back this up at the firm level: across their review of 68 studies, top management's support is the most commonly cited reason BI adoption succeeds, ahead of any purely technical factor. They also found that adoption is best explained by several factors synergizing, for example technological, organizational, environmental, and individual, rather than by any of them in isolation. Stjepic et al. (2021) mostly agree but add a complication. In their survey of 100 Croatian SMEs, what predicted BI adoption was whether the firm had enough people, skills, and resources, and whether its data was in order. The properties of the technology itself barely mattered. But support from top management did not matter either in their results, which contradicts both Caldeira and Ward (2002) and Al-Daraba et al. (2025). So, the studies agree that the obstacle is organizational but not on exactly which part of the organization it sits in.

This does not mean that resource constraints do not matter. It just changes what kind of resource is actually missing. Champa and Segall (2026) draw a line between access gaps and outcome gaps. Obtaining the tools is no longer the main problem. The gap that remains is in what firms manage to do with them and it appears wherever organizational, cultural, or human skills are lacking. Chen and Lin (2021) found the same pattern outside of SME-specific literature. When surveying 231 Chinese firms they found that a BI system's driving capability, i.e. turning information and knowledge into real decisions and actions, did not produce the cumulative effect on firm performance that the earlier stages of the model showed, even though a weak direct effect was found. They attributed this to organizational structure and resistance to change rather than weaknesses in the BI system itself. For resource constrained SMEs, the takeaway is that analytical capability should probably be built up step-by-step rather than attempted all at once. A firm might start with cheap and basic descriptive tools layered onto systems it already has and only move on to predictive or prescriptive tools once each step yields real returns (Champa & Segall, 2026).

The shift from cost to competence is not just a theory but a claim about how SMEs actually experience their analytical systems on a day-to-day basis, and the literature gives additional related but separate evidence here. Iranmanesh et al. (2023) who surveyed 187 Malaysian SMEs found that organizational readiness, i.e. capital, IT infrastructure, analytical skills, and staff, affects whether a firm chooses to outsource big data analytics rather than building it in-house. Firms with lower readiness were more likely to outsource. So according to this

literature, outsourcing is the main way resource-constrained SMEs manage these limits, not falling back on the systems that they already have. But outsourcing is not always affordable or even possible, and the literature does not say what do when they can neither build their own in-house systems nor outsource them. A likely answer is that they just use their existing operational systems for analysis even though those systems were not necessarily built for that. Whether or not this actually happens and whether it helps or obstructs, and what SME staff themselves see as the real obstacle are questions that the literature cannot answer on its own. This is exactly what this thesis's interviews are meant to find out (RQ1). Only once we understand how SMEs actually experience their current systems does the second RQ make sense. If competence and organizational readiness rather than cost on its own are what really limits an SME's ability to run separate analytical infrastructure (OLAP), what kind of database design could lower that burden without needing the dedicated staff and infrastructure that separate systems have historically required (RQ2)?

This also decides which organizations this thesis is about. Of Gupta and George's (2016) three resource types, the tangible ones are the easiest to obtain. What tends to be missing is the rest, i.e. skills, culture, and data specialists. This thesis is about organizations in that situation, i.e. organizations where analytical work is done without a dedicated analytics function, by people whose main duties are something else, and without in-house data engineering capabilities. Firm size is a sub-optimal way of identifying these organizations. Gupta and George (2016) found no meaningful connection between how large an organization is and how good it is at analytics. The condition this thesis is interested in can exist inside a unit of a large group, and it can be absent in a small firm that happens to have the right people employed. Defining the population by size limits like Stjepic et al. (2021) do with headcount and turnover would therefore miss the point. This thesis instead defines its population by capacity: not how many employees a firm has but instead whether they have employees who can do this work. That is also why RQ2 matters. When there is no one to maintain a data pipeline and no one to interpret what comes out of it, every extra system is just one more thing that the organization cannot really support.

2.3 Analytical Practice Under Limited Capacity

Kasiri et al. (2024) interviewed 50 SMEs about what they use to analyze their data. The organizations ranged from micro (<10 employees), small (<50 employees), and medium (<250 employees), where the majority were micro (Kasiri et al., 2024). No matter the size, the most prevalent tool that was used was spreadsheets, more specifically Excel (Kasiri et al., 2024). Other tools such as "Google Analytics, Social Media Marketing, and Email Marketing Systems" were also widely used (Kasiri et al., 2024). All 50 organizations used at least one of seven data- and analytics their study examined (Kasiri et al., 2024).

The organizations mainly pointed out two constraints: expertise and money (Kasiri et al., 2024). The number of identified constraints did not show a consistent pattern relative to organization size (Kasiri et al., 2024). However, the type of constraint did correlate to organization size: money was most reported in micro organizations while medium organizations reported lack of expertise and were also the ones holding the most data (Kasiri et al., 2024). It is worth mentioning that Kasiri et al.'s (2024) study is exploratory and the authors themselves state the need for a larger sample of respondents and survey data before causal conclusions can be drawn. They note that more research is required to determine whether organization size matters.

There is a paradoxical choice to make when choosing analytics software between two shortcomings (Coleman et al., 2016). On the one hand, operating a product with real analytical power takes a specialist (Coleman et al., 2016). On the other hand, the products that a non-specialist can operate accomplish less (Coleman et al., 2016). Few products escape this trade-off and Coleman et al. (2016) include it in their list of what has held analytics adoption back.

The alternative this literature describes is a second system that sits apart from the operational ones. In a data warehouse, the data has already been made ready for decision support (Wixom & Watson, 2001). Watson and Wixom (2007) assign the work of filling it to a dedicated team which has to take data out of the systems that hold it and rework it until decision support can use it. They estimate that this stage takes around 80% of the time and effort in a BI project. Additionally, it takes knowledge specific to that system, along with coordination, to get the data held in those systems (Wixom & Watson, 2001). Across the 111 organizations that responded to Wixom and Watson's (2001) survey, there was an association between the rise of technical issues and the quality of the source systems and development technology available.

Puklavec et al. (2018) studied what happens when the BI system comes built into the *enterprise resource planning* (ERP) system instead of being bought separately. They describe an ERP system "as a business management software that a company can use to collect store, manage, and interpret data from many business activities" (Puklavec et al., 2018, p.259). Puklavec et al. (2018) surveyed 181 SMEs and measured three stages: evaluating BI, adopting it, and using it. Having the BI system built into the ERP was positively associated with all three stages (Puklavec et al, 2018). For use, the association holds once the indirect effects running through the two earlier stages are counted (Puklavec et al., 2018). The reason they give is that the integrated arrangement saves work for employees (Puklavec et al., 2018). Employees only have to learn one single system, nobody has to examine how the organization's own ERP system is built before installing, and preparing the data is simpler because the tools for it come with the product (Puklavec et al., 2018). Cost effectiveness was not deemed to be a determinant at all the three stages which the authors interpret as these organizations not being sensitive to it rather than as cost efficiency deterring them (Puklavec et al., 2018).

Purchasing analytical SaaS moves the purchase decision away from the IT function (Schneider & Sunyaev, 2016). Schneider and Sunyaev (2016) reviewed 88 empirical studies of IT sourcing and coded 625 relationships of which 542 have direct bearing on the sourcing decision. A single department can also purchase its own account since nothing has to be installed on anyone's machine when the service runs in a browser and a subscription sized for one department tends to stay inside what its manager is already permitted to spend (Schneider & Sunyaev, 2016). Schneider and Sunyaev (2016) call this *shadow IT*. Whether an organization still needs capability of its own is not conclusive in Schneider and Sunyaev's (2016) material because cloud studies disagree with one another on this while the outsourcing studies do not. The authors suggest that the disagreement exists because from what is being bought, with thin in-house capability drawing organizations toward hosted software while holding them back from infrastructure and platform services.

2.4 The Separated Architecture and the Cost of the Gap

This section aims to connect the limitations described in sections 2.2 and 2.3 to the architectural question that the rest of this chapter examines. The nature of the separated architecture is

already covered in the introduction (chapter 1) and in this literature review (chapter 2), so we only state its costs and downsides here. The first cost is infrastructure because the organization runs a second database along with the ETL process that loads it. March and Hevner (2007) place ETL at the stage of a data warehouse project that requires more time and money than any other stage does, and Watson (2002) describes it as a reason that some data warehousing projects have outright failed. The second cost is delay since analytical queries only see what the most recent ETL run returned, and Song et al. (2024) report that such runs "usually takes minutes to hours" (Song et al., 2024, p.1485).

The larger cost is competence because a separated architecture requires two kinds of work at the same time: (1) someone has to build and maintain the pipeline that moves data out of the operational systems, and (2) someone has to know what the data means well enough to decide what should be loaded and how the results should be interpreted. Jukic et al. (2017) show how much work the first kind is. According to them, ETL is where data warehousing implementations lose time more often than anywhere else, and what makes it expensive is creating surrogate keys and then matching them to incoming records, since every record has to be checked against the dimension tables before it can be loaded (Jukic et al., 2017). In their case at the Federal Reserve Bank of Kansas City, inserting 200 GB of data took twelve hours under this arrangement (Jukic et al., 2017). The two kinds of work are also interconnected and dependent on each other: Jukic et al. (2017) describes that the people using the data, and who may need a variable that the schema does not already contain cannot simply just add it since variable changes the target schema and therefore the ETL, and that change has to be made by whoever maintains the pipeline. Wixom and Watson (2001) who surveyed 111 organizations and found that the relationship between the two sides affects whether the project succeeds since user participation was associated with both organizational and project implementation success, and team skills, which they define as covering both technical ability and the ability to work with users, were associated with project implementation success. Their sample contained only operational data warehouses so it cannot show what happens when the two sides fail to work together with each other. However, it does show that skills and the working relationship between the technical staff and users are among the factors that separate more successful implementations from less successful ones. We conclude something that neither study states itself: that what a separated architecture consumes is exactly the human and intangible resources that section 2.2, following Gupta and George (2016) identifies as the ones an organization cannot buy, and an organization without dedicated analytics or in-house data engineering capacity has to supply both from people whose main duties are something else.

In section 2.1 we used Ariyachandra and Watson (2010) to show that the choice of data warehouse architecture is made on organizational grounds. The factors that best predicted which architecture an organization chose were the strategic view it took of the warehouse, the resources available to it, and how capable it believed its own IT staff to be. This is what allows the two research questions to be treated as one study rather than two separate ones. If architecture selection depends on organizational conditions, then an architecture cannot be assessed for a given population without first establishing what those conditions are. RQ1 characterizes the conditions through interviews, and RQ2 tests one architecture against them.

2.5 The Architectural Differences: Normalization vs. Dimensional Modelling

The choice between normalized and dimensional database schemas reflects a deeper question about how organizations structure their information for decision support. Two distinct design traditions have shaped how data is organized for different purposes. One originates from Codd's relational model and its emphasis on logical integrity, redundancy control, and data independence (Codd, 1970). The other is from Kimball's dimensional modeling approach and its emphasis on analytical simplicity and query performance (Kimball & Ross, 2013). While framing this contrast as a direct opposition between Codd and Kimball risks simplifying a more nuanced development in database theory and data warehousing practice, it still captures a difference in architectural priorities that is directly relevant to the questions examined in this thesis.

Normalization originates from the relational design logic proposed by Codd, where data is organized into relations and structured in ways intended to minimize redundancy and prevent inconsistency (Codd, 1970). In practical terms, normalization works through decomposition. Data is distributed across multiple related tables so that each fact is ideally stored once, thereby reducing update anomalies and preserving integrity (Fotache et al., 2026). This logic has made normalized schemas particularly suitable for OLTP environments, where frequent write operations require that insertions, deletions, and updates can be performed consistently without introducing conflicting values. As Kimball and Ross also acknowledge, normalized third normal form (3NF) structures are highly useful in operational processing because an update or insert transaction ideally touches the database in only one place (Kimball & Ross, 2013). Normalization should therefore be understood not only as a technical procedure, but as a design philosophy oriented toward logical correctness, consistency, and reliable operational data management.

Dimensional modeling emerged from a different set of priorities. Whereas normalization emphasizes integrity and decomposition, dimensional modeling focuses on analytical accessibility and performance. According to Kimball and Ross (2013), dimensional modeling is the preferred technique for presenting analytic data (OLAP) because it simultaneously supports understandability for business users and fast query performance. Its characteristic structure is the star schema, in which a central fact table containing quantitative measures is linked to surrounding dimension tables that provide descriptive context. This design reduces the number of joins required by typical analytical queries and creates a simpler and more symmetrical structure for reporting and decision support. Kimball and Ross (2013) argue that normalized schemas are often too complex for business intelligence queries because users struggle to understand and navigate them, while database optimizers may perform poorly when faced with large numbers of unpredictable joins. In this sense, dimensional modeling accepts controlled redundancy in exchange for simpler querying and greater analytical usability.

However, the performance trade-off between these two design logics is not as settled as the conventional contrast might imply. A central empirical contribution in this area is provided by Fotache et al. (2026) who note that there are hardly any empirical studies on the query performance implications of different degrees of normalization in decision support systems. Using the TPC-H benchmark across SQL Server, PostgreSQL, and MySQL, they found that higher normal forms were associated with better query completion rates and faster execution times than less normalized schemas (Fotache et al., 2026). A key explanation to this is that normalization can substantially reduce total data volume by eliminating redundancy, so that the gains

from scanning smaller tables may outweigh the costs introduced by additional joins. Importantly, Fotache et al. (2026) do not compare normalized schemas with dimensional star schemas directly, and their results should therefore not be taken as disproving the value of dimensional modeling. Nevertheless, their findings complicate the widespread assumption that normalization is inherently disadvantageous for analytical workloads. For this thesis, that complication is significant. If the relationship between normalization, join complexity, and analytical performance is more nuanced than the conventional narrative suggests, then the question of whether a normalized operational database can support decision support workloads becomes empirically open rather than theoretically settled in advance.

This distinction is particularly important from an Information Systems perspective because it is not only a matter of technical design, but also of architectural choice under organizational constraints. If acceptable analytical performance requires a dedicated dimensional OLAP environment, then organizations must also maintain separate infrastructures, ETL processes, and additional layers of technical complexity. Ariyachandra and Watson (2010) surveyed 400 organizations and found that how capable an organization believed its own IT staff to be was one of the biggest predictors of which architecture was selected alongside resource constraints and how strategically it viewed the warehouse. Organizations that rated their IT staff's capability lower were more likely to choose architectures such as independent data marts rather than enterprise data warehouses (Ariyachandra & Watson, 2010). The contrast between normalization and dimensional modeling therefore becomes highly relevant for organizations with limited dedicated analytics capacity, where the issue is not simply which schema is theoretically preferable, but whether analytical capability can be achieved in a way that is organizationally feasible and economically sustainable. In that sense, the architectural difference examined here is not merely a database design issue, but a practical decision-support problem at the intersection of technical performance, infrastructural complexity, and organizational fit.

2.6 Columnstore Indexing and HTAP

Hybrid Transactional/Analytical Processing (HTAP) was coined by Gartner Group in 2014 (Song et al., 2024) to describe database systems designed to support both transactional and analytical processes (Özcan et al., 2017). Attempts to design database systems that could process a combination of both OLTP and OLAP workloads were made prior to this but were to a high degree superseded by the introduction of HTAP (Song et al., 2024). However, some key remnants laid the groundwork for the progress of HTAP (Song et al., 2024). Historically, several relational database management systems (RDBMS) have attempted to implement support for both OLTP and OLAP workloads into one single system (Özcan et al., 2017). These mainly utilized rowstore rather than columnstore and were not efficient for either workload in most cases (Özcan et al., 2017).

Because of these inefficiencies and the incompatible nature of combining OLTP and OLAP, most database systems have historically attempted to satisfy only one of these two systems rather than combining them (Özcan et al., 2017). The main difference between how database systems handle OLTP and OLAP workloads is the data organization that is used (Özcan et al., 2017). The first type of data organization is called rowstore and is normally used in transactional database systems with OLTP workloads (Özcan et al., 2017). The second type is called columnstore and is mainly used in analytical database systems with OLAP workloads (Özcan et al., 2017). Rowstore, where data are stored row-by-row so that columns are stored contiguously and together represent a row in a table, are efficient for transactional operations that

affect single records, but less appropriate due the inefficiency for analytical operations that may read millions of rows just to aggregate a few columns (Angeles & Castro, 2013; Larson et al., 2011). This is because "the query processor retrieves the value of the attribute involved in the query from each record in the relation along with other attribute values stored next to it" which "wastes cache space to store data that will not be referenced" (Angeles & Castro, 2013, p.941). The reason for this is that data is not read at one attribute at the time. Disk and memory transfers happen in fixed units so the smallest amount that can be read is a whole sector or page, and under a row-by-row layout several complete rows sit inside such a unit which means individual attributes cannot be read on their own (Abadi, 2008). A columnstore instead keeps the values of each column on separate pages so that a query only reads pages holding the columns that it needs (Larson et al., 2011). Additionally, running analytical queries on a rowstore risk impeding transactional day-to-day operations which prolong response times for an organization's systems (Jukic et al., 2017).

Columnstore solves this type of analytical bottleneck by storing each column separately, meaning queries retrieve only the specific attributes they need rather than entire rows which directly reduces I/O costs (Angeles & Castro, 2013; Abadi et al, 2008). Because values within a single column share the same data type and often repeat, columnstore enables significant compression, for example, through "run-length encoding - where a sequence of repeated values is replaced by a count and the value" (Abadi et al., 2008, p.970) which reduces the volume of data that must be read from disk (Abadi et al., 2008; Jukic et al., 2017). Additionally, columnstore systems delay combining column data into rows until the very end of a query plan (Abadi et al., 2008). This technique is known as "late materialization" which allows predicates to be applied directly on compressed columns and irrelevant rows to be discarded before tuple reconstruction (Abadi et al., 2008). This yields approximately a factor of three improvements in performance (Abadi et al., 2008). There is an additional technique called "block iteration" that processes arrays of column values in a single operator call rather than one value at a time which reduces per-tuple overhead further (Abadi et al., 2008). However, columnstore performs poorly for transactional operations because retrieving or updating a single record requires reconstructing it from multiple separate column stores – a process Angeles and Castro (2013) describe as "extremely slow" for tuple reconstruction. This structural incompatibility is precisely why organizations have historically kept separate systems for each workload.

Additionally, the need to have two separate database systems with different responsibilities normally requires an ETL process where data is extracted from the OLTP system and then batch inserted into the OLAP system (Song et al., 2024; Özcan et al., 2017). The ETL process is described by Jukic et al. (2017) as "the most common time-consuming bottleneck in most implementations of data warehousing for quality decision support" (p.61). This process is expensive and often takes minutes or hours, resulting in delayed or outdated data in the OLAP system (Song et al., 2024). Additionally, it makes real-time analytics unattainable (Song et al., 2024). With the introduction of HTAP, which can handle both transactional and analytical workloads in one single database system concurrently, the expensive ETL process can be disposed of, and real-time analytics can be made possible (Song et al., 2024). Prior research on hybrid architectures demonstrated this principle directly: Angeles and Castro (2013) showed that by routing each query at execution time to whichever storage engine – rowstore or columnstore – minimizes I/O cost, organizations can serve both OLTP and OLAP workloads from a unified system while also reducing infrastructure costs since columnstore requires less disk capacity due to compression.

2.7 Normalization's Impact on Mixed-Workload Performance

The debate over database design for HTAP systems is closely connected to the relationship between data integrity and query performance. Highly normalized schemas generally reduce redundancy and strengthen consistency. Denormalized schemas, by contrast, are often used to improve analytical performance by reducing the number of joins required during query execution. In mixed transactional and analytical environments, this creates a central trade-off. Designs that preserve transactional reliability may make analytical queries more costly, while designs optimized for analysis may introduce redundancy and weaker control over data consistency (Bog et al., 2012).

This tension is visible in applied industrial database contexts. In their study of normalization and denormalization in Manufacturing Execution System (MES) databases, Zhou and Gao (2024) found that strict adherence to Third Normal Form (3NF) resulted in response times that were approximately 99% slower than a moderately denormalized design. The study also shows that fragmented data structures can lead developers to rely on inefficient application-layer processing, such as repeated loop-based operations. In their case, this increased response times to over 38 seconds (Zhou & Gao, 2024). After optimizing the implementation and using a denormalized structure, the study reported a throughput increase of nearly 97% (Zhou & Gao, 2024). These findings illustrate why denormalization has often been treated as a practical strategy for improving performance when analytical response time is important.

At the same time, research on modern mixed-workload systems complicates the assumption that denormalization is always necessary for performance. Bog et al. (2012) show that traditional disk-based database systems may struggle with normalized data in mixed OLTP and OLAP scenarios. However, systems designed specifically for mixed workloads can perform well with normalized schemas. In their study, a highly normalized schema performed best in the tested mixed-workload environment (Bog et al., 2012). Fotache et al. (2026) report the same direction of effect in a decision support setting where the 3NF schema completed more queries within a time limit and was faster than the 2NF and 1NF versions across three different database systems (SQL Server, MySQL, and PostgreSQL). Their tests only use data retrieval, so they say nothing about the transactional side of a mixed workload (Fotache et al., 2026). They also conclude that join reduction "do not compensate for the penalties associated with retrieving data from larger tables" (Fotache et al., 2026, p.2). This indicates that the performance effects of normalization depend on more than schema design alone. They are also shaped by the underlying database architecture, storage model, and execution mechanisms.

The storage model is the part of this that has been examined most directly in the Information Systems literature. Jukic et al. (2017) had a pre-joined analytical table in columnar form and found it faster than a row-based star schema on a dataset of around 100 million rows for queries that read fewer than roughly 20 columns. They also argue that the same design would not be usable in a row-based system at all because such a system has to read whole rows from disk and the pre-joined table is very wide (Jukic et al., 2017). Jukic et al.'s (2017) comparison changes the storage model and the schema at the same time which we interpret as a reason why the two effects cannot be separated in their result. Dreseler et al. (2020) add another limitation: TPC-H uses a normalized schema rather than the star schema that is more common when it comes to analytical workloads. However, they do emphasize that TPC-H is "relevant for the evaluation of relational database systems" (Dreseler et al., 2020, p.1206).

For this thesis, this is important because it supports the need to empirically test whether a normalized schema enhanced with columnstore indexing can perform sufficiently well for analytical decision-support workloads.

2.8 Theoretical Framework

In section 2.2 we showed that analytical capacity is not one resource but three. Gupta and George (2016) separate tangible resources, human skills, and intangible resources, and section 2.2 concluded that the tangible ones are the easiest to obtain while the other two usually are not as easy to obtain. The literature disagrees about which resources actually limit resource constrained organizations. Cost is the barrier that is often reported, but several studies find that competence and organizational readiness matter more once adoption is studied directly. The interviews aim to find out which one binds in practice.

Section 2.4 showed that a separated architecture consumes two resources that are difficult to get. It needs someone to build and maintain the ETL pipeline and someone who understands the data well enough to say what should be loaded and what the output means (Jukic et al., 2017; March & Hevner, 2007). Moving the data also risks losing or corrupting it along the way (Wixom & Watson, 2001). At the same time, a separated architecture (OLAP and OLTP) delivers data already prepared for decision support and pulled together from several source systems (Wixom & Watson, 2001) and a solution built into a system the organization already uses saves employee work (Puklavec et al., 2018). Architectural fit is therefore a question of what a configuration delivers weighed against what it costs to run. Section 2.6 adds another additional cost since analytical queries that run against a rowstore can slow down the transactional system they share (Jukic et al., 2017).

Existing research treats organizational barriers to analytics adoption and database performance under different schema designs as separate literatures. Neither side looks at how the technical alternatives it studies would fare against the organizational conditions that the other side describes. The framework below connects them and treats analytical need and organizational capacity as jointly determining architectural fit.

Which configuration an organization picks is in itself an organizational decision. Ariyachandra and Watson (2010) found that available resources and the organization's own belief in its own IT staff affected what architecture was chosen. This gives the framework used in this thesis: analytical need (which decisions are supported), how current the data must be, what kinds of queries are run, and which analysis are desirable but not possible or feasible, combined with organizational capacity, determines which configuration is viable. RQ1 examines these through interviews. RQ2 tests whether an integrated configuration performs well enough for that need, given the constraint's capacity places on it. The benchmark measures execution speed and not whether the organization's staff could write the query in the first place which section 2.5 already flags as a limitation of normalized schemas (Kimball & Ross, 2013).

Table 2.1: Constructs in the framework, basis in the literature, and how each is studied

Construct	Basis in literature	Research question	Data source
Analytical needs (decisions, freshness,	Kasiri et al. (2024); Song et al. (2024); Coleman et al. (2016)	RQ1 and defines the workload for RQ2	Interviews

query shape, unmet demand)			
Default tooling in practice	Kasiri et al. (2024); Coleman et al. (2016)	RQ1	Interviews
Organizational capacity (tangible, human, intangible resources)	Gupta & George (2016)	RQ1	Interviews
Division of work, ETL burden, and data integrity in transfer	Jukic et al. (2017); Wixom & Watson (2001); March & Hevner (2007)	RQ1	Interviews
Interference between analytical and transactional work	Jukic et al. (2017); Özcan et al. (2017)	RQ1 and RQ2	Interviews and benchmark
Selection criteria: cost, non-cost, resource availability, staff capability	Ariyachandra & Watson (2010); Puklavec et al. (2018)	RQ1	Interviews
Schema design and storage model under analytical workload	Fotache et al. (2026); Bog et al. (2012)	RQ2	Benchmark

3 Methodology

3.1 Overall Research Strategy

This thesis uses two different research methods, one for each research question. RQ1 is primarily addressed through interviews; the benchmark provides the technical evidence for RQ2, which is answered by interpreting that evidence against the conditions established through RQ1. Combining qualitative and quantitative methods in one project is called mixed-method research (Bell et al., 2019; Recker, 2021). This is not to be confused with multi-method research where there is more than one method, but it does not mix qualitative and quantitative strategies (Bell et al., 2019). This choice is made from the research questions themselves: it is important to match the strategy most suitable for answering it (Bell et al., 2019). RQ1 examines how practitioners experience and manage their analytical infrastructure which requires a method that can be used to access their insights. RQ2 examines measurable performance differences between database configurations which requires an amount of control and replicability that a controlled experiment can provide. Neither method could answer both research questions. Combining the two allows the strengths of each to compensate for the weaknesses of the other, although a mixed-method design is not automatically better than a single-method and must be justified by the problem at hand (Bell et al., 2019). Of the five purposes for mixed-method research that Recker (2021) lists – triangulation, complementarity, initiation, development, and expansion – this study is primarily motivated complementarity, because the interviews give organizational meaning to the benchmark results. It also has a developmental element because respondents' descriptions of their analytical workloads inform the business relevance of the TPC-H queries.

Two decisions follow from mixed-method designs: (1. Priority) how much weight each strategy carries, and (2. Sequence) in what order the methods are used (Bell et al., 2019). In this study they carry the same weight. The quantitative controlled experiment was conducted before the qualitative interviews. However, for the sake of logic and our research questions, the qualitative strategy precedes the quantitative strategy in this thesis. The reason for this is that the interview findings are required for interpreting the results of the experiment.

3.2 Qualitative Interview Study

3.2.1 Choice of Semi-Structured Interviews

RQ1 inquires how organizational requirements and resource constraints shape the suitability of analytical infrastructure for decision support. This question is about practitioners' own experiences and judgements and cannot be measured directly. Interviews are a suitable data generation method when the researcher needs detailed information, when questions are open-ended and may need follow-up or a different order for different respondents, and when the topic is about experiences that are difficult to obtain through predefined answer options

(Oates et al., 2022). The interview study therefore complements the experiment in section 3.3. This is because the experiment removes organizational context in order to isolate a technical effect while the interviews provide this context.

Interviews can be structured, semi-structured, unstructured (Oates et al., 2022). A structured interview uses identical and predetermined questions for all respondents which would have limited the study to obstacles the authors had already thought of. This makes it a poor fit, because the purpose of this study is to discover which obstacles actually exist. An unstructured interview on the other hand gives the respondent most of the control over the topics. This would have made it difficult to compare the interviews from different organizational settings. The semi-structured type was therefore chosen. In a semi-structured interview, the researcher has a list of themes and questions but can change their order, ask additional questions, and let respondents bring up issues that were not planned for (Oates et al., 2022; Bell et al., 2019). This flexibility did have concrete effect on the study: issues were raised by respondents outside the planned themes and questions contributed to the revised population definition described in section 3.2.2.

Semi-structured interviews are suitable when needing to discover rather than checking, and because they involve a small number of respondents, of whom all are not suitable to answer the same questions, they cannot support generalizations about a whole population (Oates et al., 2022). This limitation is accepted and discussed further in section 3.2.6. The interview study also has interpretive assumptions where the point is not to establish what the respondents' situations objectively are, but instead how the respondents themselves understand and value them, and two respondents in similar situations may describe them differently without either being wrong (Oates et al., 2022). Additionally, the interviews have a supporting role for RQ2. When respondents describe how current their data must be, what kind of questions they run against it, and whether reporting has ever slowed down their operational systems, they are describing real analytical workloads. These descriptions later provide an empirical basis for assessing how closely the standardized TPC-H workloads correspond to the analytical requirements described by respondents. This correspondence is evaluated when the benchmark findings are interpreted in Chapter 5.

3.2.2 Sampling and Respondents

The respondents were not chosen randomly. In qualitative studies, respondents are normally chosen on purpose. and the research question(s) determines which kinds of people can give useful answers, and the researcher then looks for people of those kinds, often building variation into the sample on characteristics that matter for the questions (Bell et al., 2019). A random sample was in any case not possible here since it would require something like a complete list of all organizations with limited dedicated analytics capacity, and no such list is likely to exist. Additionally, the goal was to get a breadth of perspectives rather than statistical representativeness (Bell et al., 2019). Initial selection criteria were established before data collection and were derived from the research questions, following the purposive sampling logic described by Bell et al. (2019). As described later in this section, these criteria were subsequently revised during data collection when the emerging interview material indicated that organizational size was a less appropriate population boundary than analytical capacity.

Recker (2021) uses the term key informant for a respondent whose role gives them access to knowledge about a setting that most people in that setting would not have. Both respondent

profiles were defined with this in mind: practitioners for their knowledge of a single organization, and consultants for insights that spans several.

Two main respondent profiles were defined. The first were practitioners who provide depth about a single setting since they work hands-on with data, reporting, or analytical infrastructure within an organization. The second profile is consultants who build or maintain analytical solutions for client organizations. A consultant can compare conditions across different organizational settings in a way that most in-house respondents cannot. Candidates were reached through LinkedIn. Availability and willingness to participate played a major role in who ended up in the sample, but they had to fit into one of the two profiles first. This is what separates the procedure from convenience sampling where availability alone decides who participates (Bell et al., 2019).

Table 3.1: Overview of respondents

ID	Role	Organizational context	Profile	Analytical capacity
R1	Analytics/customer success, also part of product R&D	About 20 employee sensor-data company, almost entirely engineers, IT competence distributed rather than held by a separate department	Practitioner	High: entire analytical pipeline built and maintained in-house by their own R&D team
R2	Consultant, data/analytics engineer	Recent assignments at two mid-sized firms, earlier also a smaller firm (about 200 employees)	Consultant	N/A: relays conditions across clients of varying capacity
R3	Controller and data analyst	About 100 employees in Sweden (food sector), subsidiary of a 6000-employee European group, much of the infrastructure sits with the parent	Practitioner	Low to medium: one-person, self-built BI function serving the whole organization
R4	Principal data scientist	About 35-person data science/analytics consultancy, no dedicated IT department	Consultant	Low to medium: technically strong as a firm but no dedicated BI/reporting function
R5	Consultant, Engineering AI and Data	Local office (about 3000 staff) of a global professional-services firm employing several hundred thousand worldwide	Consultant	Low: no dedicated analytics tooling on current engagement, office 365 is the default platform
R6	Part-time, news-feed data sourcing and	About 9-person early-stage startup, dedicated CTO plus two full-time data engineers, builds its own in-app analytics	Practitioner	Medium: small team but self-built,

	product/interface design	rather than adopting external BI tools		purpose-integrated tooling
R7	Partner consultancy, data platform support	Predominantly large clients, smaller-firm experience is earlier in career	Consultant	N/A: direct experience skews large, comments on smaller firms are general observation, not current assignments
R8	Software dev, team lead	Large bank: about 260 in department, about 3000 at the local office, tens of thousands group-wide, previously a consultant	Practitioner	High: strong in-house build capacity, interfaces with lower capacity, Excel-based business users downstream
R9	IT manager	Organization currently running about 40 systems, previously 15 years at a larger industrial group running about 200 systems	Practitioner	Medium: current part-time analytical capacity, through previously full-time in a similar role

The selection rules did not stay fixed: they were originally written around SMEs. As the interviews accumulated it became clear that the conditions the study is interested in – few specialists, unclear responsibility between those who produce data and those who analyze it, systems that do not connect – appear and disappear no matter the size of the organization. Respondents placed them inside units of large groups while the smallest company in the sample ran the most unified setup. The population criterion was therefore changed from size to capacity. The methodology literature accepts that a qualitative study may discover along the way that it needs participants of a kind the original plan did not consider (Bell et al., 2019). This is also treated as a result in itself since the mismatch between size and capability in the sample is part of what motivates the revised research question.

When it comes to the number of interviews, the methodological literature offers no fixed rule for how many are enough. What is required instead is an explicit argument for why the chosen number fits the specific study (Bell et al., 2019). The argument in this study is about purpose. This study does not build theory and does not try to establish how widespread the findings are. Rather, it looks for patterns that repeat across deliberately different organizational settings. The respondents in this study include startups, subsidiaries of large groups, public organizations, and consulting firms. A purposive sample says nothing about how common something is outside of the sample (Bell et al, 2019; Oates et al., 2022). What this study contributes instead is enough description of each case for readers to judge the findings' relevance to settings they know.

3.2.3 Interview Guide

The interview guide has four blocks. The first block, background, asks about the respondent's role, how they work with data or reporting, and the size and IT staffing of the organization. These questions are easy for the respondent to answer which helps the interview get going (Oates et al., 2022), and the answers are needed anyway. Collecting information about the respondent's position and organization is recommended because their remaining answers can only be understood in relation to it (Bell et al., 2019). The second block, decision support, asks how data and reports are utilized to support decisions, which decisions are the most important, which tools and systems are used, and whether there are analyses the respondent wants but cannot do. The third block, infrastructure and data flows, asks where data is stored, whether reporting runs in the operational systems or in a separate solution, what goes wrong when data is moved between systems, how current the data needs to be, whether analytical questions are quick lookups or reach into large amounts of history, and whether reporting has ever been seen to slow down the operational systems. The fourth block, constraints and priorities, addresses RQ1 by asking about the biggest obstacles in current reporting, whether reporting directly in the existing operational systems would be valuable if fast and reliable enough, and what matters most when choosing an analytics solution. Respondents with a technical background also got the question of whether OLTP, OLAP, and HTAP were familiar terms. The reason for this was to see how far these concepts have spread among practitioners.

The interview guide follows the recommendations of Bell et al. (2019): the blocks were given a natural order that the interviewer was free to depart from, the questions were tied to the research questions without steering the answers in a particular direction, everyday professional vocabulary was used rather than academic terms, and formulations that suggest desired answers were avoided. Most questions were built so that they cannot be answered with a yes-or-no or a single word which pushes the respondent toward explaining rather than confirming (Oates et al., 2022). During the interviews, interesting answers were followed up and the interviewer repeated back their interpretation of important answers so that the respondent had the chance to object (Oates et al., 2022). For consultants, every question was reformulated to be about the client organizations they had worked with instead of their employer. Every interview finished with the respondent being asked whether the questions had missed anything important (Oates et al., 2022).

3.2.4 Data Collection

The interviews were held one at a time over video calls in June and July. Video was chosen for practical reasons because it removed geography as a restriction on who could participate and cost the respondents no more time than the conversation itself (Oates et al., 2022). Interviewing over video changes little methodologically: the demands regarding information, consent, and preparation are the same as face-to-face, but building a comfortable atmosphere can take somewhat more work at a distance (Oates et al., 2022). Every session started the same way: we, the authors, introduced ourselves, repeated what the study is for, explained what would happen with the recording and the material, and asked for permission to record (Oates et al, 2022). Permission was given in all cases.

Audio was recorded because notes and memory lose too much for later analysis, and because a recording let us concentrate on the conversation, and because it leaves raw material that can be examined and fact-checked afterwards (Oates et al., 2022; Bell et al., 2019). Recording followed by transcription is also the normal way of working in qualitative interviewing since the

analysis depends on the exact formulations the respondents used which notes rarely preserve (Bell et al., 2019). The recorder can make some respondents careful about what they say (Oates et al., 2022). This was handled by asking for permission to record, offering anonymity, and placing the more evaluative questions later in the interview guide.

The recordings were transcribed using OpenAI Whisper which runs privately and locally on our computers, rather than running online in the cloud. In later chapters, quotes from the interviews may have been translated from another language, for example from Swedish to English, by us authors, with the context in mind so that the actual meaning was not lost in translation. The transcripts in the appendix keep the respondents' working verbatim as spoken. However, some disfluency may have been filtered away while still keeping the meaning and content intact which is common and accepted practice (Oates et al., 2022). In cases where audio could not be heard or made out properly, the transcripts mark these parts as inaudible without any guessed fillings. Each transcript can be identified by a respondent's code and date rather than identifiable information such as name in order to preserve anonymity (Oates et al., 2022).

3.2.5 Analysis of Interview Data

The interviews are analyzed thematically. In this study, a theme is defined as a category that we built from codes from the interviews and that relates to the research questions (Bell et al., 2019). The main indication that something may be a theme is that several respondents brought it up (Bell et al., 2019). However, repetition and recurrence are not enough (Bell et al., 2019). It should only become a theme if it also has a bearing on the research question (Bell et al., 2019), in our case this is RQ1.

The analysis was informed and inspired by the steps described by Bell et al. (2019) and Oates et al. (2022). We started by listening to and reading the transcripts in order to become familiar with the material (Bell et al., 2019; Oates et al., 2022). Then, we categorized the content of each transcript into three categories: (1) material outside the scope of the study, (2) background material about the respondent and their organization, and (3) material with bearing on the research question (Oates et al., 2022). The background material was used for the respondent overviews in table 3.1 and when presenting the respondents in chapter 4. The material in the third category which had bearing on the research questions was coded: we wrote a short code for each segment, and a segment could get several codes, and we did not limit the number of codes at this stage. We then reviewed the code list so that codes that described similar things were merged and codes that occurred infrequently were either combined with others or set aside (Bell et al., 2019; Oates et al., 2022). Lastly, the codes were grouped into the themes that chapter 4 presents. A theme was kept only if it had support in several transcripts (Oates et al., 2022).

The coding approach was mainly inductive, but with some deductive elements. Most codes come from the interview material itself (Oates et al., 2022). We also used concepts from the literature review during the analysis, especially the contrast between cost and competence as constraints as well as the governance concepts. The risk with prior concepts that we as analysts only finds what the literature has prepared us for (Oates et al., 2022), but the free independent coding in step three limits this risk. Patterns without bearing on the research questions were not pursued further which Bell et al. (2019) describe as reasonable for projects with deadlines and constraints.

Each code was given a written definition that states which statement belongs to it. The definitions worked as decision rules where a segment only received a code if it matched the definition. The codes were not fixed in advance. Codes were added, merged, and renamed during the coding process described above. Table 3.2 shows the final version.

Table 3.2: Coding scheme

Theme	Code	Definition
Decision support and analytical demands	Division of analytical work	Analytical work is performed by people other than those running day-to-day operations.
	Freshness requirement	A stated requirement about how current the data must be.
	Query shape	Shape of analytical questions, for example quick lookups, aggregations etc.
Infrastructure and data flows	Default tooling	Excel or self-built tooling is named as the destination data ends up in regardless of what more capable systems exists.
	Data integrity in transfer	Data loses accuracy or requires substantial work when moved, for example corrections that fail, corruption during transfer, or cleaning and conversion effort.
	Value of the current arrangement	The existing separated setup is described as sufficient.
Constraints on analytical capacity	Competence as limiting resource	Constraint regarding the skill available in individuals, whether capability rests on one or more people, or the people lack the required understanding.
	Handoff breakdown	Knowledge fails to pass between those who move data and those who know what it

		means, or lack of responsibility for the transfer.
	Workload interference	Analytical activity degrades operational system performance, or scheduling is chosen to avoid degradation.
Priorities and direction	Cost criteria	Cost is named as a governing consideration when choosing analytical solutions.
	Non-cost criteria	Usability, compatibility, risk, lock-in, longevity, an existing technical ecosystem, or something else governs, instead of, or ahead of, cost.
	Integration stance	A stated position on merging operational and analytical systems.

3.2.6 Trustworthiness of the Interview Study

The criteria used for the experiment in section 3.3.9 is built on the assumption that repeating the same procedure should produce the same outcome. This assumption fails for interviews since a conversation cannot re-run under identical conditions (Bell et al., 2019), the situations the respondents describe change over time, and the answers depend partly on who is asking (Oates et al., 2022). The interview study is therefore instead judged against four criteria presented by Bell et al. (2019) and Oates et al. (2022): credibility, transferability, dependability, and confirmability:

- **Credibility** concerns whether the reader has reason to trust the picture of the respondents that the study gives. Three things support this here. Firstly, the interviews were carried out according to the documented procedures in sections 3.2.3-3.2.4. Secondly, everything was recorded and transcribed instead of reconstructed from notes. Thirdly, misunderstandings could be corrected on the spot since the interviewer repeated their interpretation of important answers during the interviews.
- **Transferability** concerns where else the findings might hold. This study cannot settle this question for the reader. Instead, what it can do is describe each case extensively enough so that readers can recognize, or fail to recognize, their own situation in it. Table 3.1 and the quotes in chapter 4 exist for this purpose (Bell et al., 2019; Oates et al., 2022).
- **Dependability** concerns whether the road from raw data to conclusions can be inspected afterwards. Everything needed to retrace this study exists in documented form: the interview guide, the anonymized transcripts, and this methodology's sections. An

outside reviewer can hence follow the chain from recording to finding (Oates et al., 2022; Bell et al., 2019).

- **Confirmability** asks whether the conclusions come out of the material or out of the researchers. This question is important in this study because we, the authors, also designed the benchmark experiment and could therefore have an interest in the interviews pointing a certain way. We have two countermeasures for this: every theme in chapter 4 stands on quotes that the reader can check for themselves in the transcripts, and secondly, the coding procedure includes deliberately searching for statements that contradicted or corroborated each theme rather than only statements that supported it.

The overall weaknesses of this method remain because an interview documents an account of practice rather than practice itself, and the two can differ; respondents know that they are speaking on the record which can affect what they choose to say; and the answers are affected by who is asking (Oates et al., 2022). Additionally, for this particular study, there are two more weaknesses that stand out: the consultants only relay their clients' conditions like secondary sources, and our purposive sample of nine respondents allowed no conclusions about how prevalent the identified constraints are (Bell et al., 2019).

3.2.7 Ethical Considerations in the Interview Study

This interview study processes personal data and the obligations that come with human participants therefore apply in this case. Participation had to be voluntary and possible to end at any point in time, consent had to be given with knowledge of what participation involved, and both the respondents' identities and what they divulged had to be protected (Oates et al., 2022). Bell et al. (2019) outlines the same obligations as principles of avoiding harm, ensuring informed consent, ensuring privacy, and preventing deception.

Before each interview, the respondent received written information about what the study was for, what participation means, and how the material would be handled. Consent was not treated as settled by that document alone. At the start of the session itself, the purpose, handling of the material, right to withdraw was explained and the recording started after the respondent had explicitly agreed to be recorded (Oates et al., 2022).

No respondent, organization, or third party named in an interview can be identified in this thesis. In a group of nine professionals in specialized roles, seemingly harmless details such as an unusual product, well-known client, niche industry can point out a person even when the name is gone and this risk is known to grow as samples shrink (Bell et al., 2019). Details of that kind are therefore rewritten or redacted in the transcripts and quotes wherever they risked someone's anonymity in a way that does not change what the quote means. The realistic harm in a study like this one is professional rather than physical: several respondents spoke openly about weaknesses in their organizations' data work and statements of that kind can damage someone's position or career which is something that the principle about avoidance of harm covers (Bell et al., 2019).

The original audio recordings were accessible only to the authors. Offline AI-based tools were used to assist with transcription. Interview material included in the thesis has been anonymized, and identifying details have been removed or generalized where necessary to protect respondents and third parties. The original recordings and identifiable working material will be deleted once the examination process is completed. Handling collected material about

people is part of the researcher's responsibilities when it comes to data collection and storage (Oates et al., 2022).

3.3 Quantitative Benchmark Experiment

3.3.1 Overall Research Strategy

The research strategy for RQ2 is a controlled experiment where a factor being tested, the *independent variable*, is varied while all other conditions are fixed so that any change in the measured outcomes, the *dependent variables*, can be traced to the factor that was deliberately varied (Oates et al., 2022). In this experiment, the independent variable is the database configuration (schema design and indexing strategy) varied across three types: a normalized 3NF schema with non-clustered columnstore indexing (HTAP), a dimensional star schema with rowstore indexing (OLAP-RS), and finally a dimensional star schema with columnstore indexing (OLAP-CS). The dependent variables are each query's execution time, CPU time, logical reads, and lob logical reads. Because everything else in the environment is held consistent, any difference in these measures can be attributed to the configuration rather than other factors (Oates et al., 2022; Bell et al., 2019).

Using research from the literature review, this experiment tests our predicted outcome that the dimensional configurations will achieve lower analytical query times than HTAP with the size of any difference being dependent on the workload. Having a testable prediction in advance rather than simply observing what happens is what separates a true experiment from an uncontrolled trial where the results support no actual conclusion because other explanations remain uncontrolled (Oates et al., 2022).

Case study and action research were considered but excluded. Both examine how systems operate in a real-world scenario by observing an existing case or studying a change while it is introduced into a live organization where many uncontrolled factors vary all at once (Oates et al., 2022). Neither one offers the variable control needed to isolate the performance effect of schema and indexing choices (Oates et al., 2022). Hence, a controlled experiment is the most appropriate way to examine this.

3.3.2 The TPC-H Benchmark as Research Instrument

Evaluating the performance of database architectures for decision support requires a standardized and reproducible benchmarking framework. In the context of this thesis, where three database configurations are compared under analytical workloads, the choice of benchmark directly affects the validity and interpretability of the results. The TPC-H benchmark is a decision support benchmark that consists of a suite of business-oriented ad-hoc queries designed to examine large volumes of data, execute queries with a high degree of complexity, and give answers to critical business questions (Transaction Processing Performance Council, 2022). Dreseler et al. (2020) describe TPC-H as "the most widely used benchmark for relational OLAP systems" (p.1206), and Fotache et al. (2026) similarly observe that TPC-H has retained its popularity for decision support benchmarking even as the benchmark itself may have grown dated. Part of the reason for this continued relevance is that standardized benchmarks for decision support systems are relatively scarce compared to those available for transaction processing, with TPC-H being one of a small number of established alternatives alongside benchmarks such as TPC-DS (Fotache et al., 2026).

From a measurement perspective, database performance in decision support systems is typically assessed through query execution latency rather than throughput, since the relevant concern is how quickly a given analytical query can return results to the end user rather than how many concurrent transactions the system can handle (Fotache et al., 2026). This aligns with the metrics used in this thesis, where execution time and CPU time serve as the primary performance indicators for comparing the schema configurations.

What makes TPC-H particularly appropriate for this study is that its queries are not abstract computational tasks but are designed to represent concrete business scenarios with broad industry-wide relevance (Transaction Processing Performance Council, 2022). The 22 queries simulate analytical tasks such as pricing summary reporting, supplier cost evaluation, shipping mode analysis, order priority monitoring, promotion effectiveness assessment, and more (Transaction Processing Performance Council, 2022). More broadly, the TPC-H queries represent standardized business-oriented decision-support tasks rather than abstract computational workloads. Using TPC-H therefore allows the experiment to compare the three database configurations under a common set of analytically relevant conditions. Their correspondence with the analytical requirements described by interview respondents is considered when the benchmark results are interpreted in Chapter 5. The specific mapping between individual TPC-H queries and their corresponding business decision contexts is presented in Section 3.3.7.

3.3.3 Experimental Environment

To isolate the effect of the independent variable, the experiment is conducted in a controlled local environment with dedicated hardware allocation. Doing this eliminates factors that could distort the experiment such as network latency and cloud-provider throttling, so that the observed differences in performance can be purely attributed to the configuration rather than the environment (Oates et al., 2022). Controlling the environment like this is a deliberate means to ensure internal validity of the experiment (Bell et al., 2019).

The hardware specification is also a deliberate control measure. The memory available to the database instance is capped at 4096 MB of memory so that all three configurations are evaluated under the same fixed resource conditions. The purpose of the cap is to improve comparability between configurations.

- **Hardware**: Intel i5 9300H (4 cores, 8 threads), 8 GB DDR4 RAM (limited to 4096 MB for the SQL server instance), SSD.

- **Software**: This experiment uses Microsoft SQL Server 2025 as the primary database management system (RDBMS) for its native support for columnstore indexing, installed natively on Windows without containerization and virtualization. SQL Server Management Studio (SSMS) is used to create the databases and execution of DDL. However, it is not necessary to execute the TPC-H queries and is hence not used for this purpose to minimize the amount of non-essential processes running in the operating system. Instead, PowerShell scripts are utilized for consistent query benchmarking.

- **Operating system**: Windows 11.

3.3.4 Design of Schema and Data Artifacts

The experiment compares three architecturally different database configurations populated with the same underlying test data. By purposefully keeping the test data consistent across all three configurations, a side-by-side comparison follows that any observed differences in performance during the execution of the queries can be attributed solely to the manipulated independent variable, i.e. schema design and indexing strategy, rather than to differences in data (Oates et al., 2022). This ensures the internal validity of this experiment which is the confidence that the measured effect is actually produced by the manipulation and not other factors (Bell et al., 2019). The following three configurations are used:

1. **HTAP (Normalized)**: A traditional schema designed for day-to-day transactional systems (OLTP). The schema strictly adheres to the third normal form (3NF). To enable analytical capabilities, non-clustered columnstore indexing is applied to specific columns and coexists alongside the default rowstore indexing used for transactions.

2. **OLAP (Dimensional, two versions)**: The OLAP candidate has a traditional denormalized star schema. It features a central fact table surrounded by dimension tables that each have a unique ID. The fact table contains the foreign keys of the surrounding dimension tables. Two versions of this schema are used: one with default rowstore, and the other with columnstore indexing applied.

3.3.5 Collection of Data/Results

The measurements of the dependent variable are the essence of an experiment and the time taken to process data is a standard experimental measure in IS and computing research (Oates et al., 2022). The dependent variables are execution time, CPU time, and to some extent logical- and lob logical reads. These are captured for each query across the three configurations through the following process:

1. **Generation**: Approximately 1 GB of synthetic machine-generated data is generated using the database population program DBGen tool, bundled with the TPC-H download, provided via the official TPC website (tpc.org) (Transaction Processing Performance Council, 2022). This translates to a scale factor of 1 which represents the minimum amount of data that DBGen can generate (Transaction Processing Performance Council, 2022). The outcome of using this tool will be eight .tbl files (Transaction Processing Performance Council, 2022) named according to their respective table name, i.e. [table name].tbl.

2. **HTAP schema and database creation**: A normalized 3NF schema is created based on the TPC-H schema found in the dss.ddl file bundled with the installation. The schema consists of eight tables: part, partsupp, supplier, customer, nation, lineitem, region, and orders. Additionally, primary keys, foreign keys, and referential integrity constraints will be enforced.

3. **Loading of data into HTAP**: Each of the generated .tbl files are loaded into the HTAP schema artifact using bulk insert to ensure identical and consistent data.

4. **Indexing of HTAP**: Non-clustered columnstore indexing is applied on columns in tables in the HTAP schema artifact. Specifically, the non-clustered columnstore index is applied to all foreign key columns as well as frequently aggregated numeric columns

within the three primary transaction-heavy tables. This approach results in an indexing strategy designed to accelerate analytical read-heavy and aggregation-heavy workloads without unnecessarily impeding transactional write performance.

Specifically, columnstore indexing is applied to the tables *lineitem*, *orders*, and *partsupp* while excluding columns in these tables that are not foreign keys or the subject of frequent aggregation and filtering. Additionally, large columns are excluded: in a real-world scenario, large columns lead to higher storage as well as declined performance for both transactional and analytical workloads. The columns *lineitem.l_comment*, *orders.o_comment*, and *partsupp.ps_comment* are all long strings and are not frequently used for aggregation or filtering. They have therefore been excluded from columnstore indexing.

5. **OLAP schema creation and ETL**: The normalized data from the HTAP artifact is manually designed and transformed into a star schema which represents the OLAP schema for this experiment. A central fact table and surrounding dimensions tables are designed and created. An ETL process from the normalized HTAP schema is then conducted to load the data into the OLAP schema.

6. **OLAP columnstore indexing variant**: A second variant of the OLAP schema artifact is created to reflect modern data warehousing environments where columnstore is commonly applied. Columnstore indexing is applied to columns in tables *fact_lineitem* and *dim_orders*, excluding the *comment* columns, which mirrors the equivalent columnstore application in the HTAP candidate.

7. **Query adaptation**: The standard TPC-H query templates are adapted for SQL Server compatibility. This includes replacing ANSI date syntax with DATEADD() and replacing EXTRACT() with YEAR. Placeholders in the templates are replaced with SQL code as per the TPC-H specification (Transaction Processing Performance Council, 2022).

   To support the interpretation of the benchmark results, each TPC-H query is mapped to the business decision context defined in the TPC-H specification, for example pricing summary reporting (Q1), national market share (Q8), and order priority checking (Q4) (Transaction Processing Performance Council, 2022). This mapping provides a business-oriented description of the standardized queries. Their correspondence with the analytical requirements identified through the interviews is assessed separately in Chapter 5.

8. **Automated query execution through script**: A PowerShell script is developed to automate query execution using sqlcmd. Each query is executed 6 times per schema artifact with SQL Server statistics enabled using SET STATISTICS TIME ON and SET STATISTICS IO ON. CPU time, execution time, logical reads, and lob logical reads are captured automatically for each run and written to an output .txt file for analysis. Automating execution like this is itself a control because measurement procedure is identical for each run and is not interfered with by the researcher which removes experimenter effects, i.e. variation introduced by the person conducting the test, as a threat to internal validity (Oates et al., 2022).

9. **Warm-up**: Each query is run at least once as a warm-up run within the PowerShell script to ensure tables are stored in the buffer/data cache (RAM) and to compile

execution plans to be stored in the plan/procedure cache. This first run is excluded from the reported results. Performance metrics are collected from the five subsequent executions only. This prevents cold reads from the disk and simulates real-world scenarios more appropriately where the database system has usually already been in active use.

10. **Execution of HTAP**: The HTAP schema artifact is benchmarked first. CPU time, execution time, logical reads, and lob logical reads are captured using the automated script described in step 8.

11. **Execution of OLAP schema artifact**: The experiment for the OLAP schema is conducted using the same script described in step 8. The buffer cache and plan cache are cleared before execution using *DBCC DROPCLEANBUFFERS* and *DBCC FREEPROCCACHE* to ensure an unpolluted baseline and consistency.

12. **Execution of OLAP columnstore indexing variant**: The OLAP columnstore variant is executed in the same way as the previous candidates. Cache is cleared in the same way.

3.3.6 Analysis of Data/Results

The CPU time, execution time, logical reads, and lob logical reads are compared through quantitative analysis to assess architectural efficiency, performance, and cost of running the queries through the three configurations.

- **Execution time**: Primary indicator of real-world performance which reflects whether HTAP is fast enough for organizations. From an organizational perspective, execution time is often the most important metric because it is directly experienced by users. For example, an analyst waiting for a report observes only the time it took to generate that report, not the underlying CPU time or logical reads.

- **CPU time**: Indicative of processing efficiency and hardware costs.

- **Logical reads & lob logical reads**: Measures I/O efficiency. Demonstrates how much unnecessary data is scanned during an operation.

The queries are grouped by their operational nature as follows with all the individual TPC-H queries predefined prior to execution:

- **Aggregation-intensive**: queries involving heavy aggregation, for instance COUNT(), SUM(), AVG(), MIN(), MAX().

- **Join-intensive**: queries involving compiling data from several tables into a single view through joins.

- **Filtering-intensive**: queries involving heavy filtering through keywords such as WHERE, AND, OR, and LIKE.

- **Multi-layered**: queries involving subqueries and/or correlated queries.

For each query, performance is measured by using the median of the five runs rather than the mean value. The median is the central value once all the five runs are ordered by size and unlike the mean, is not pulled up or down by a few unusually fast or slow runs (Bell et al., 2019). Because individual runs are vulnerable to OS background activity the median is a more stable and representative summary of each configuration's performance than the mean. Time metrics for each of the queries are compared between the artifacts using the formula below:

$$Speed = \frac{\text{Time (HTAP)}}{\text{Time (OLAP)}}$$

Speed < 1 (*Speed* less than one) indicates that HTAP is faster than OLAP whereas *Speed* > 1 (*Speed* greater than one) indicates that OLAP is faster than HTAP. The same formula is applied to CPU time for each of the queries. However, CPU time indicates processing efficiency rather than speed.

The analysis describes the results rather than testing them statistically. The claim that the differences in performance are caused by the schema and indexing strategy doesn't come from a statistical test but rather the control built into the experiment. Since the configuration is the only thing that changes between runs, it is the only factor left that could explain any difference in the results (Oates et al., 2022; Bell et al., 2019).

3.3.7 TPC-H Queries and Mapping/Categorization

Table 3.3 explains the characteristics of individual TPC-H queries according and interpreted through the official TPC-H specification (Transaction Processing Performance Council, 2022).

Table 3.3: TPC-H queries and their business scenarios and technical nature

Query number	Query name	Business scenario	Technical nature
Q1	Pricing Summary Report	Reports total and average sales and discounts metrics	Primary: aggregation-intensive Secondary: filtering-intensive
Q2	Minimum Cost Supplier	Identifies the supplier offering the lowest cost for specific parts under given conditions	Primary: multi-layered Secondary: join-intensive Tertiary: filtering-intensive
Q3	Shipping Priority	Identifies high-value orders that should be prioritized for shipping	Primary: join-intensive Secondary: aggregation-intensive

			Tertiary: filtering-intensive
Q4	Order Priority Checking	Monitors order priorities to identify delays in order fulfillment	Primary: multi-layered Secondary: aggregation intensive Tertiary: filtering-intensive
Q5	Local Supplier Volume	Measures revenue generated from local suppliers within a specific region	Primary: join-intensive Secondary: aggregation-intensive Tertiary: filtering-intensive
Q6	Forecasting Revenue Change	Estimates revenue impact under different pricing or discount conditions	Primary: filtering-intensive Secondary: aggregation-intensive
Q7	Volume Shipping	Analyzes shipping volume and revenue between regions over time	Primary: join-intensive Secondary: aggregation-intensive Tertiary: multi-layered
Q8	National Market Share	Measures market share of a product within a specific region or nation	Primary: join-intensive Secondary: aggregation-intensive Tertiary: multi-layered
Q9	Product Type Profit Measure	Analyzes profit by product type across regions and time	Primary: join-intensive Secondary: aggregation-intensive Tertiary: multi-layered
Q10	Returned Item Reporting	Identifies customers and revenue associated with returned items	Primary: join-intensive Secondary: aggregation-intensive

			Tertiary: filtering-intensive
Q11	Important Stock Identification	Identifies high-value inventory items that represent significant stock investment	Primary: multi-layered Secondary: aggregation-intensive Tertiary: join-intensive
Q12	Shipping Modes and Order Priority	Analyzes how shipping methods affect order priority and delivery performance	Primary: aggregation-intensive Secondary: filtering-intensive Tertiary: join-intensive
Q13	Customer Distribution	Analyzes customer distribution based on number of orders placed	Primary: multi-layered Secondary: aggregation-intensive Tertiary: join-intensive
Q14	Promotion Effect	Evaluates the impact of promotions on revenue share	Primary: aggregation-intensive Secondary: filtering-intensive Tertiary: join-intensive
Q15	Top Supplier	Identifies the supplier generating the highest revenue contribution	Primary: multi-layered Secondary: aggregation-intensive Tertiary: join-intensive
Q16	Parts/Supplier Relationship	Analyzes availability of suppliers for specific parts	Primary: multi-layered Secondary: filtering-intensive Tertiary: aggregation-intensive
Q17	Small-Quantity-Order Revenue	Analyzes revenue contribution from small-quantity orders	Primary: multi-layered Secondary: filtering-intensive

			Tertiary: join-intensive
Q18	Large Volume Customer	Identifies customers with high-volume orders and their revenue contribution	Primary: multi-layered Secondary: aggregation-intensive Tertiary: join-intensive
Q19	Discounted Revenue	Analyzes revenue generated from discounted product sales	Primary: filtering-intensive Secondary: join-intensive Tertiary: aggregation-intensive
Q20	Potential Part Promotion	Identifies parts suitable for promotion based on availability and supply conditions	Primary: multi-layered Secondary: filtering-intensive Tertiary: join-intensive
Q21	Suppliers Who Kept Orders Waiting	Identifies suppliers responsible for delays in order fulfillment	Primary: multi-layered Secondary: join-intensive Tertiary: aggregation-intensive
Q22	Global Sales Opportunity	Identifies potential customers and sales opportunities across regions	Primary: multi-layered Secondary: filtering-intensive Tertiary: aggregation-intensive

3.3.8 Delimitations and Scope

Several deliberate decisions regarding scope are included in this experiment in order to preserve control and focus. Each decision has an implication for external validity, i.e. the degree to which the findings generalize beyond the conditions tested (Oates et al., 2022; Bell et al., 2019).

Firstly, a fixed limited TPC-H scale factor is utilized to provide a standardized and consistent baseline. Specifically, the scale factor is set to 1 (approximately 1 GB) (Transaction Processing Performance Council, 2022). This is partly because of the memory cap described in

section 3.3.3: at 4096 MB, the working set stays within the resource envelope the study concerns, so that measured differences reflect schema and indexing rather than memory pressure. Fotache et al. (2026) also benchmarked TPC-H at 0.1 GB and 1.0 GB. The delimitation is about data volume, not organizational size. As established previously, resource constraints here are defined by the capacity that can be dedicated to analytical work, and organizations of any size may run analytical workloads at or well above this volume. The findings are therefore delimited to workloads of approximately this scale, and the relative performance of the configurations may differ at substantially larger volumes.

Secondly, the experiment is limited to a single DBMS, Microsoft SQL Server, so that observed differences can be attributed to schema and indexing rather than to differences between engines. SQL Server's columnstore divides rows into groups and stores each column as a separately compressed segment, with directory metadata that allows whole segments to be skipped before being read and can process scans in batch mode rather than row-at-a-time (Larson et al., 2011). Song et al. (2024) classify this as one of several HTAP architectures that organize storage differently. The execution times in chapter 4 are therefore engine-specific. The workload-dependent pattern is more likely to transfer, since it follows from a storage layout in which each column occupies its own segments and a scan touches only those belonging to attributes the query references (Larson et al., 2011) which is common to columnar implementations. Verifying this across engines is however outside the scope of this study.

Thirdly, this experiment mainly focuses on analytical queries rather than transactional queries. In other words, analytical and transactional queries are not tested concurrently to simulate the real-world dynamic workload of an HTAP database system. The purpose of this is to clearly isolate analytical performance from other system activities.

Additionally, machine-generated synthetic test data generated by TPC-H's DBGen tool is relied upon rather than real organizational data. This guarantees reproducibility while sacrificing the inconsistencies commonly found in real-world organizational data. Finally, the scope is strictly limited to relational database systems, excluding NoSQL and other big data architectures.

Recker (2021) warns that "the danger with experiments is that they become internally valid but not plausible in a realistic setting" (Recker, 2021, p.114), and instructs researchers to address this gap. The interview study is how we address that recommendation. Section 3.4 uses the workloads that the respondents report to assess how far the tested conditions correspond to real analytical demands.

3.3.9 Reliability and Replicability

The measures in this section are applied to address two quality criteria when it comes to quantitative research: *reliability*, concerned with whether the study would produce the same results if it were run again, and *replicability*, which demands that the procedure of the experiment have to be specified in enough detail for the study to be reproduced by others (Bell et al., 2019). Repetition of measurement under controlled conditions is also important in experimental research and is used to confirm that the measured outcomes are not the result of transient factors such as OS background activity (Oates et al., 2022). The procedures below ensure that these criteria are fulfilled:

3.3.9.1 Query execution and managing variability

- **Repetition**: Every query is executed a minimum of 6 times for each of the three configurations so that a single divergent run cannot determine the final reported result (Oates et al., 2022).
- **Warm up**: The first execution of a query is excluded from the results. This is to prevent cold reads from disk which is not realistic in a real-world operational production scenario.
- **Median time**: For the remaining 5 executions the median execution time is calculated rather than the mean. The reason for this is to prevent extremes and spikes caused by external noise such as OS background activity from distorting the results (Bell et al., 2019).
- **I/O aggregation**: For queries accessing multiple tables, total I/O reads are reported as the sum of all lob logical reads and standard logical reads across all tables accessed in that query execution. For columnstore-enabled configurations, I/O is primarily reflected as lob logical reads. For the rowstore configuration, I/O is reflected as standard logical reads. For queries accessing both columnstore-indexed and non-columnstore-indexed tables within the same configuration, both lob logical reads and standard logical reads are reported. By specifying this rule explicitly, it makes the I/O measurements replicable (Bell et al., 2019).

3.3.9.2 Environmental consistency

- **Clearing of cache**: Between each test, the SQL Server buffer cache and plan cache are cleared using *DBCC DROPCLEANBUFFERS* and *DBCC FREEPROCCACHE* to ensure an unpolluted baseline and prevent potentially cached data from influencing the experiment. This helps control a factor that would otherwise threaten the internal validity by introducing differences unrelated to the manipulated independent variable (Oates et al., 2022).
- **Fixed resource allocation**: SQL Server maximum server memory is capped at 4096 MB across all tests to ensure that each schema artifact is benchmarked under identical and consistent resource constraints.

3.3.10 Ethical Considerations

This experiment only uses synthetic machine generated data from TPC-H, and there is no involvement of personal data or humans. Ethical frameworks that concern research on humans, such as the participant rights by Oates et al. (2022) and the principles of avoiding harm, having informed consent, respecting privacy, and preventing deception described by Bell et al. (2019) therefore do not come into play in this experiment. The same holds for the legal duties that comes from holding personal data about individuals, as presented by Oates et al. (2022), since the test data does not describe real persons and that no personal data is collected, stored, or processed. This was one of the reasons for opting for synthetic benchmark data over real organizational data. The most important ethical requirement of this experiment is to be honest, accurate, and transparent (Oates et al., 2022) and describe the process so that others can replicate it (Bell et al., 2019) which is done in the sections above in this chapter.

3.4 Integration of Methods

This section describes where in the thesis the two methods are brought together. Sections 3.2 and 3.3 describe the interview study and the experiment separately because the two methods have different procedures and quality criteria. However, in a mixed-method study, the two methods are expected to integrate its findings rather than present them separately side-by-side (Bell et al., 2019).

Chapter 5 (discussion) is where we interpret the results of the experiment against the interview findings. The experiment can show under which workload conditions an integrated database configuration (HTAP) performs close to a dedicated analytical configuration (OLAP). It cannot show whether this performance difference really matters in practice. The interview findings are what provides that judgment because they show what burdens organizations with limited dedicated analytics capacity actually report, for example, lack of specialists, unclear responsibility, and problems with ETL. Chapter 5 therefore examines whether the measured performance differences between the configurations are consequential under the organizational requirements and constraints identified through the interviews. It also considers whether potential advantages associated with greater integration, such as reduced data movement, are relevant under those conditions, while recognizing that separated analytical arrangements can provide organizational benefits of their own. The interview findings on workload characteristics in table 3.3 are also used in chapter 5 to some extent to evaluate how well the workload conditions tested in the experiment match the workloads the respondents describe. Putting quantitative results in the context that qualitative findings provide is a recognized function of mixed-methods research (Bell et al., 2019). Chapter 6 (conclusion) then answers both research questions and states what the combined findings support and what they do not support.

4 Results and Analysis

This chapter presents the results of the qualitative interviews and the benchmark experiment. Sections 4.1-4.4 will present the results of the interviews, which will be further divided into subchapters based on the codes identified in the analysis process. Subsequently, 4.5-4.10 will present the results of the benchmark experiment and the performance of the different queries.

4.1 Decision Support and Analytical Demands

4.1.1 Division of Analytical Work

The division of analytical work can be defined as workloads being performed by different roles in an organization. Mainly, the analytical role which identifies long-term trends creates larger reports for presentation to managers, and processes larger amounts of data. And the operational role, the ones that treat real-time numbers and are responsible for inputting daily data into the system. The codified material shows how some respondents describe analytical work as divided between different roles. And these can depend on factors such as time horizons, technical responsibility, preparation of data, interpretation and decision-making.

R3 directly describes a division based on time horizon while R4 conversely describes his experience from a perspective outside of the direct context of analytical projects ([9] R4). R3 states clearly:

[5] R3: "*I mainly look at the long-term trends while many other departments work with the short-term, what needs to be done here and now."*

This shows how the division of analytical work can be divided by both time horizon and organizational roles, and how data is used for different purposes. R4 also notes that this experience of divided analytical work comes mainly from working with product companies, the respondent also states how it cannot speak generally about the issue.

A similar division is described by R7, where the preparation of data for BI-analysts and the analysis itself is not handled by the same role:

[3] R7: "[...] *often I sit and populate the data itself for the BI-analysts who maybe have looked at HR-information. Others look at product information, and so on. [...] You also want to look at where the data is coming from initially. That is what I have been doing mostly, not taking business related decisions based on the data.*"

Here R7 describes a clear bridge between the processing of the data itself, and the subsequent transfer of the data to the analysts who will use it for their own tasks. This shows a distinction between enabling analytical work and using data-based analysis for decision-making. Similarly, R6 gives insight into a further division within the analytical infrastructure:

[17] R6: *"[...] I am instead, like I said, getting the RSS-feed to Python. [...] There we have a website and then our application, if you will. I am a bit involved in the design. [...] So, I am designing it, and making sure our information from our Python, or internal database, ends up in the right place [on the website]. But the storage, or data management, I am not very involved in."*

Managing the data or maintaining the local database is also separate from the tasks R6 describes, making sure the data ends up in the right place in an application or on a website. This is a separation between technical responsibilities within the analytical infrastructure, although the company R6 is employed by is very small, only nine employees ([2] R6). The division of responsibilities was still present.

Similarly, R8 who, conversely, is employed at a much larger organization ([2] R8), also describes a division between data management and BI analysis, where the respondents work mainly involve combining and organizing data for the analysts so they do not have to sift through thousands of columns of data:

[7] R8: *"[...] we are working a lot with raw data, databases, direct integration and similar. And all the old case workers or financial posts work a lot with Excel."*

[18] R8: *"[...] We have a type diagram in QMC, but a lot more numerical data which is structured in a way that gives this overview like a graphical tool does, but in a numerical format. So, you work a lot with compiling a lot of data at the bottom, and gives you as a case worker for example, or financial analyst an insight without having to read through all these thousands of columns of data."*

The common pattern emerging is that analytical work is most often divided and further subdivided into different roles in organizations, and that these roles all have different functions and responsibilities. This division occurs both between the operational short-term and the long-term analytical roles, as well as in different stages of the analytical process, like collecting, integrating, and preparing data. Maintaining data quality, presenting or structuring information for other employees, interpreting that data in a business context, and using the data as basis for decision-making.

4.1.2 Freshness Requirement

Throughout the codified material, a topic which was identified was the current nature of data. The freshness requirement is therefore defined as a stated requirement about how current the data must be (section 3.2.5). As will be apparent in this section, freshness does not equate to a need for the data to be in real-time, or constantly updating live, rather it will be shown how different requirements depend on the purpose of the analysis, the type of information, and the activity being supported.

In the range of freshness requirements, there were some respondents who had a need for near real-time data. This data did not need to be updated every second but needed to be available within an hour of time. R1 described this as following:

[9] R1: *"It's not real time but it's near real time [...] between that acquisition and processing everything [...] they're usually available within an hour or something like that."*

Similarly, R2 describes:

[14] R2: *"[...] real time in terms of milliseconds, no, but within an hour it needs to be, so daily is not enough, but I think we are running our product jobs every 15th minute, and that's what I have seen, every 15th or 30th minute is what I have seen quite a lot lately [...]"*

Although, the respondent does not explicitly states that data needs to be real-time, R2, seems to suggest that data needing to be available within an hour is a type of real time data as the quote above is preceded by: *"Real time is always an interesting word because it depends on what you mean by it."* ([14] R2). Both R1 and R2 describe their freshness requirement to be available data within an hour. R1 characterizes this as near real-time, while R2 presses on the importance of context when defining the concept of real-time.

The freshness requirement also seems to be dependent on the purpose of the analysis. What kind of analysis needs to be done will determine how current the data must be. Both R3 and R5 describe how the requirement depends on the purpose. R3 states:

[12] R3: *"[...] daily is mostly enough. Weekly also. We have some reports we only take out monthly. But in general for the organization, daily. But for the more overarching questions, strategic questions it is weekly or monthly."*

Although R5 does not give as concrete of a time requirement as R3, the respondent nonetheless presses on the dependency of the analysis:

[10] R5: *"It is simply completely dependent on the purpose. If there is a certain engagement which needs very current data, then the reporting will occur according to that purpose. If it were to be based on a slightly longer period, the data might not need to be as detailed or as current."*

What both R3 and R5 describe is how their freshness requirement is dependent on decision horizon. Short-term operational decisions concerning day-to-day tasks have a certain requirement, and the longer-term strategic tasks have different requirements.

Another aspect which determines freshness requirements is the source of the data. In the same way R5 and R3 described varying requirements based on purpose, R6 also gives an insight into the variation of requirements coexisting within the same organization:

[9] R6: *"[...] if we want to look at profit in a company. [...] we use data from annual reports for example. And they only come out yearly. But if you take news, that is something we are live updating all the time. [...]. If we have an RSS-file on Aftonbladet, and a news article about, well Ikea. Then we retrieve it almost instantly into the system."*

What is described is the dependency of the data source itself. For R6's purposes, the freshness requirement is based on how frequently the data updates. Financial data gathered from annual reports can only be updated when a new report is published. News information, on the other hand, is gathered continuously and enters the system almost immediately after publication. What can be garnered from this is that freshness of data is not only an organizational preference but can also be determined or limited by the source of data.

A further insight into the coexistence of multiple freshness requirements within a technical environment is provided by R8:

[12] R8: *"The main system I work with. That is real time data. It is very important for it to be real-time [...] Everything that happens in real time in this QRM-system. And then we have a*

database in between in Amazon Dynamo which is polling in real time and updates every second, actually. [...] But then of course we have older and lower interval data. It is the type I mentioned earlier where we have monthly jobs which run and then everything is processed during the week. And then you see the result at the end of the week"

Within the same technical environment there is not a singular freshness requirement, some systems require continuous polling and updating, while others require data updated at longer intervals. R8 also notes, about the data updated at less frequency: *"It is not a major rush to retrieve this data, as long as you know it is treated correctly."* ([12] R8).

Freshness requirements can also be limited by system performance. As described by R9, the update schedule is determined by the technical conditions:

[9] R9: *"We usually ran it once per day, early in the morning, in order not to reduce the performance of the system. There was still quite a lot of data that needed to be processed and moved. In some cases you need to run tighter updates. Running it maybe 2-3 times per day, and you pick a suitable time when it is not very busy."*

The desired update frequency may need to be weighed against the performance impact of data-transfer and processing jobs. Update frequency is not solely based on a users ideal, but can be limited by the volume of data being moved, the processing load created by updates, and the need to avoid affecting the operational system.

The codified material shows how the freshness requirement for an organization can vary widely and be determined by several factors. The material has shown how the range of freshness can vary from real time polling to annual updates. Real-time updates were shown to be associated with rapidly changing operational information, transactions, news, and other time-sensitive activities. Within the same organization multiple freshness requirements can coexist at the same time. The required freshness depends on the purpose and time horizon of the analysis, the availability of the source data, and the processing demands placed on the operational system.

4.1.3 Query Shape

The code Query shape is defined as the shape of analytical questions; this can for example be quick lookups, aggregations, and other ways in which data is being searched or processed. The codified material shows that query shapes vary according to the purpose of the analytical task. This variation can include quick and simple lookups, historical trend analysis, continuous comparison against accumulated data, and large periodic calculations. Different factors such as historical scope and technical complexity should not initially be treated as the same thing.

An important concept is introduced by R2 who describes what was mentioned in the paragraph above, when asked about what analytical questions are most common, and whether it includes simple lookups or large amounts of historical data, the respondent answers:

[15] R2: *"Well, there exists all variants I would say. And I would say that simple queries and large amounts of historical data are not opposed to one another. There can be very simple queries which uses a lot of historical data. [...] The amount of history is rarely a parameter in how complex a query needs to be."*

R2 distinguishes between the complexity of a query and the historical scope of the data it accesses. As stated, simple queries can use a lot of historical data, and thus the Query shape has more than one dimension; the complexity of the operation being performed and the volume or historical scope of the data being examined.

R3 gives insight into two different query shapes based on organizational purposes. The respondent gives a distinction of the query shapes, where the respondent focuses on the long-term trends while his colleagues focus on daily operations:

[13] R3: *"Our salespeople are viewing our customers on a daily basis. [...] But also the order intake and when we put orders there are a lot of quick simple searches. While I who look more overarching organization can take historical data, and then I can use up to 3, 4 years back in time to find trends and identify patterns."*

The respondent distinguishes quick searches connected to customers, order intake, and purchasing from broader analyses using several years of historical data. Importantly, what cannot be garnered from this material is whether R3's analytical queries are considered more complex than those of his colleagues.

A similarity which can be identified between R1 and R6 is the way data accumulation over time affects query shape. As R1 states:

[10] R1: *"When we do our analytical, it considers the entire history of the asset. [...] we use what we call continuous learning in such a way that whenever each new measurement comes into place it's always compared towards the history but the history should remain static in every certain amount of time we just ensure the entire consistency like we revaluate that the entire consistency is there, especially whenever there's an overall update to any of our models."*

What emerges is an analytical pattern based on repeated comparison between current observations and accumulated history. This analytical pattern is similar to the one described by R6 who states:

[10] R6: *"Right now it is mostly simple searches. Then as I said, the longer this goes on, the more historical data there will be."*

Taken together, both respondents show that historical scope of an analytical workload can expand as new data accumulates, even when the basic analytical operation remains the same. The structure of the analytical task can remain while being applied onto an expanding historical dataset.

Some query shapes are large-scale and processes large amounts of data with many parameters. R8 describes how an aggregated analytical workload is used to produce a monthly financial decision value:

[5] R8: *"[...] for example our monthly calculations where we have a week when pretty much all IT is paused while we crunch [large amounts of data] in a database which includes parameters such as [...] All of this is finalized into a final number [...]. This is being done every month. Then also, this workload is being done daily, but in smaller parameters and smaller segments."*

What is described by the respondent is a clear example of an aggregation-oriented analytical workload. Large monthly total calculations of a variety of financial and economic parameters which need substantial processing, to the point where most of the organization's other IT activity is paused for a certain time period while this runs. This workload was also described as being performed in smaller segments more frequently.

In terms of historical scope. R9 gives some insight into why some Query shapes require a broader historical scope to be valuable.

[10] R9: *"historical data you want to be able to follow up maybe when you want to spot swings and tendencies, for example sales, how this has changed over a longer period of time. And from there act accordingly, then it is not enough to look at one day or week, instead you look at a quarter, half-year, year or even many years to see if there are any tendencies. Some products and markets are pretty slow. Viewing day-to-day you might not see any difference but looking over many years, you can notice, here we have gained shares, here we have lost shares."*

The respondent gives a thorough explanation for the benefit of the longer time interval Query shape. What can be gathered from the material is that the appropriate query depends partly on the rate of change of the underlying phenomenon. While the question of immediate operational conditions may require a certain historical scope, the questions concerning gradual changes in slow markets will require a broader historical comparison.

What the material therefore has shown is that most query shapes are heterogeneous and depend on many factors. Query shapes can include quick and simple lookups, searches across several years of data, repeated comparisons of new observations against accumulated data, large aggregation-oriented workloads, and smaller segments of broader calculations performed more frequently. As the respondents described, the shape of the analytical query depends on the question being answered, as well as the phenomenon being examined. The respondents also gave insight into how data volume, historical depth, and query complexity are distinct concepts in themselves, and not necessarily correlated. Some queries were described as simple lookups but nonetheless using large amounts of historical data. The material shows in summary that organizational analytical workloads vary in operational complexity, historical scope, scale, and purpose.

4.2 Infrastructure and Data Flows

4.2.1 Default Tooling

The code *Default tooling* is defined as cases where Excel or self-built tooling is named as the destination data ends up in, even when a more capable system exists. The material presented in this chapter will show a pattern of how many respondents describe two main forms of default tooling, firstly a familiar standard software, particularly Excel, and secondly, internally developed tools, script, or application-integrated analytical functions. The default tool is not necessarily the most technically capable tool, but it will be shown to be the tool where users most commonly receive, inspect, manipulate, document and present data. As well as how different employee groups use different tools within the same analytical process.

Although the material will show that there are self-built tools in which analysis is being done, many of the respondents mention Excel as the practical destination for data. R7 gives an account of what happens when analysts are about to receive data:

[4] R7: *"[...] what always happens in the end is that a BI-analyst will always say 'Can't I just get the data so I can run it in Excel?' [...] It is what you begin with and you want to use it all the time. It may not be that scalable, but that is where it ends up. Files easily end up on employer computers and they want to run their things in Excel."*

The main insight from this is that even though data can exist in a database, or a more specialized system, the endpoint of the analytical workflow is often Excel, since the BI-analysts are more familiar with that environment. This should, however, not be interpreted as technological inadequacy since the respondent also describes a consistent user preference and established working practice.

Similarly, R8 gives further insight into the use of Excel and expands it by describing its use by certain organizational roles:

[6] R8: "*All the case handlers and financial staff work a lot in standard tools like Excel and similar. As long as they can retrieve data from us and return data to use, we are satisfied.*"

Therefore, it can be gathered that not all organizational roles must work directly under the same database or within the same analytical environment. For employees outside of the technical data role, Excel appears to act as the working interface.

From R5 and R9 it can be gathered that Excel has several functions beyond being a destination for exported datasets. For example, R5 describes how documentation regarding insurance gets done in Excel:

[5] R5: *"I work with the K-system in insurance. [...] How all metadata looks within the system is most often documented in Excel I would say."*

R9, describes the use of Excel as a simple analytical option where the respondent creates macros for filtering and dividing data:

[4] R9: *"[...] in some places I have made it easy for myself and used Excel. Created some macros of my own to filter and divide the data in order to identify deviations."*

The respondent continues to identify some positive qualities Excel can have for his workflow while still pressing on the fact that it remains a simpler tool "*[...] there is good functionality like the use of symbols, like upward arrow, downward arrow, green and red color coding et cetera. But it is a bit simpler*" ([4] R9). Both R9 and R5 therefore provide evidence of how Excel can work as a lightweight tool for documentation and simpler analytical tasks, but one should be careful of interpreting this as primitive or ineffective even though R9 states how there are more advanced analytical tools like Qlik and PowerBI:

[4] R9: *"These particular systems I find better at doing more advanced calculations and analyses."*

This statement does not discredit the value of tools like Excel, instead it should be seen as how different tools are suited for different tasks.

Other respondents give insight into how self-built tools also provide value to the analytical process. R7 gives an account of how many places use tools built in Python:

[12] R7: *"But a lot of places have self-built solutions. Like in Python. Pandas Data Frames is something many people like since it is similar to Excel. They write long Python-scripts or Jupyter notebooks using Pandas and then run them as a form of cron job."*

Here we see the other side of the code *Default tooling,* the self-built tools, and it becomes apparent that the code does not necessarily mean non-technical. In some environments, the default tool is a collection of scripts and notebooks instead of a formal BI platform. The respondent compares Pandas Data Frames with Excel which suggests a continuity in the tabular way users interact with data, even when the implementation is more technical. Also, the scheduled execution means that relatively informal development artifacts can become part of recurring analytical infrastructure.

The described self-built tools above use a more un-integrated form of analysis. Conversely, R6 describes how the organization has developed their own integrated analytical solutions which integrate directly into the company's application:

[5] R6: "*[...] those who work more with it have built their own analytical solution that integrates with the applications. [...] We do not use anything external like Power BI, but it is similar to Power BI, you could say.*"

Here a clear distinction from the self-built tools described by R7 emerges, R6 shows how self-built tools can also be directly integrated into the organizational applications. Both are self-built but represent different levels of integration and user interaction. This material also shows how an organization does not have to depend on Excel to perform analytical tasks.

The material thus far has described default tooling and its uses. But many of the respondents also mention certain limitations certain tools can have. For example, R7 mentions how data often ends up in Excel environments which the respondent claims is *"Maybe not very scalable"* ([4] R7). This is also echoed by R9 who described how there are more suitable tools for advanced calculations and analytical tasks ([4] R9). R8 also mentions the limitations of Excel:

[11] R8: "*[...] they had stored all their archeological data in an Excel-file. This excel file contained, well maybe 50 million rows with 120-130 columns of data*"

Although the exact numbers of rows and columns should be taken with caution, it is still apparent from the respondent's account that this became an issue when the respondent was tasked with migrating this data into a different solution ([11] R8).

What becomes clear with these accounts taken together is that familiar tooling may continue to be used even when the volume or complexity of the data exceeds the use case for which the tool would ordinarily be expected to be useful.

In summary, the codified material shows that analytical data often ends up in familiar or locally adapted tools with an established working practice. Excel remains a recurring interface for business users, meanwhile self-built scripts and integrated applications provide customized options. More specialized systems exist, but they do not necessarily replace the tools which are already a part of an employee's established working practice.

4.2.2 Data Integrity in Transfer

The code *Data integrity in transfer* can be defined as how data loses accuracy or requires substantial work when moved, for example, corrections that fail, corruption during transfer, or cleaning and conversion efforts. What the codified material will show is how the integrity of data concerns more than just instances of visible corruption. The respondents will give accounts of inconsistency, outdated data, incompatible formats, and failed synchronization.

A problem described by R1 is conflicting versions and the "source of truth", or how the same information can exist in several places:

[8] R1: *"The main challenge [...] is handling what we call the source of truth. [...] So the main challenge is when they do one change in one side, they do another change in the other site, so just to make it match."*

The insight given is that when several systems can represent or modify the same information, problems of integrity can arise. Preserving this integrity then requires not only transferring data but determining which version should be seen as authoritative.

Another issue described by R1 is the heterogeneity of data, meaning that data is sometimes preprocessed or integrated in different ways by customers. What is described is mainly an issue of input consistency:

[12] R1: *"Not at all of them preprocess or integrate data in the same way. [...] The main challenge on our side is to ensure that [...] new features and characteristics are available or meaningful for all our customers independently of the sensor they have"*

The integrity of the data is therefore not only a question of organizational competency and policy, but it can also be a matter of input consistency, in what fashion does the organization receives data from customers. R1 works with raw wave format data and describes how they aim to provide their service regardless of what sensor the customer has ([12] R1). Data integrity can be affected before it enters the analytical system since the information of the same type may have already been processed differently by different sources.

R1 continues to describe another phenomenon, mainly how integration must be properly handled at both ends of the process, when data is received and enters the system and when providing analytical results to another user. The respondent states:

[18] R1: *"[...] there's always two different kinds of integration. One part is the receiving the data, and another one is producing the results. Those are two different sets of requirements."*

This gives a broader view of the concept of transfer. Data integrity must be preserved both at entry of the analytical environment and when it leaves as analytical results. The compatibility requirements at the input stage and output stage are different.

Another issue of data integrity is how the correction of data sometimes does not propagate through the analytical pipeline. R3 gives an insight into the inconsistency of different system layers:

[11] R3: *"[...] It is not that we lose historical data. But, say that it is wrong. And needs correction. Sometimes it can be difficult, because it is not always updated in the last stage. And there we sometimes have issues."*

What the respondent describes is how historical data which is in need in correction sometimes do not propagate this correction throughout the system, meaning the uncorrected versions of the historical data will remain in some places. This gives an account of how data integrity depends not only on identifying and correcting an error but also making sure that this correction is propagated through the complete data flow.

Before data is ready to enter a system, a certain groundwork has to take place to ensure high data quality. R4 describes the importance of this:

[13] R4: *"If you do good groundwork, you can usually achieve good data quality. Once you have reached that point and set up a pipeline that cleans the data and makes sure it is where it should be, everything flows after that. But the groundwork can be enormous. It can take years."*

Note that this is merely R4's own account, the respondent makes clear that this particular topic is not exactly his field ([13] R4), but the insight remains. Reliable transfer is partly produced by thorough preparatory work involving data cleaning rules, construction of pipelines, and decisions about where data should be located.

Similar to what was described by R1, the heterogeneity of data, R7 gives further insight into the combination of heterogenous sources of data, and the different structures these can have. The respondent states:

[8] R7: *"Transforming data that comes from many different sources and may have very different structures, and trying to put it into the same structure in the database in a way that feels logically understandable, is also a very large task."*

What this shows is how transformation is not necessarily a loss of integrity itself, but it does create work and opportunities for error. And the preservation of integrity does not necessarily mean the preservation of every source structure. Data must often be transformed, but this transformation must maintain the data's meaning while making different sources compatible.

The respondent continues his account of data integrity by giving an example of data passing through several systems. And how this can result in an end user receiving data from a database which has not been updated for months. Although this echoes the code *Freshness requirement* presented in section 4.1.2, the distinction here is still notable. Instead of being a case of the age of data, the issue also concerns the loss of reliability due to an inconsistent chain of transfer. R7 describes this as follows:

[9] R7: *"They may receive all their data from a database that has not been updated for six months. If they are sitting in an end system and it has passed through three different places before arriving there, they may make a completely incorrect decision."*

The integrity of the data can be compromised for each additional layer of transfer, which can make it difficult for the end user to determine the information's origin, when it was updated, and whether it still represents the source correctly.

R8 describes a problem of integrity when spreadsheet-data is moved into a database. The respondent describes how a supplier building a new system may need to migrate existing spreadsheet data into a database:

[11] R8: "*[...] if we as supplier build a new system and migrate the data into an existing database, we need to convert the data into the form we want.*"

The problem arising with this construction and migration often concerns cleaning data:

[11] R8: "*Cleaning the data so that it can actually be used is the hardest part, or at least what usually takes the most time in projects.*"

The claim here is not that databases cannot handle large datasets, instead, it is the transfer between tools which is not always a process of copying. Existing data may require extensive cleaning and restructuring before it can be used in the destination system.

The accounts above all describe issues with data integrity, but here follows the account of a respondent who did not experience any substantial issues in this regard. Although R2 acknowledges that data volumes and speed can be an issue in principle, in general the respondent claims these issues are not substantial:

[13] R2: *"Data volume can be a problem. Often with the case of financial data it is often not a big problem. There can be an issue with the speed of which the data needs to enter the system. [...] No, there have not been any major issues with that part."*

What the codified material has shown is how data integrity can be impacted on many levels and at several points in the data flow. The receiving of differently formatted or preprocessed data, deciding which system represents authoritative sources, propagations of corrections through every layer of the system, transforming heterogeneous sources into a shared structure, converting legacy or spreadsheet-based data. To maintain consistency, considerable cleaning, conversion, pipeline, and governance work is required. The material shows how integrity of data in transfer concerns more than just movement of information between locations. Reliable transfer requires the preservation of meaning, consistency, and traceability across systems with different formats, owners, and update processes. Each additional transformation or system boundary can introduce work or a potential point of integrity failure.

4.2.3 Value of the Current Arrangement

The code *Value of the current arrangement* is defined as how the existing separated setup is described as sufficient. With this it refers to the separation of the operational and analytical environments. Note however, that the codified material will represent a broader view and should not be interpreted as evidence for or against the separation of OLTP and OLAP architectures. The following accounts will instead show how respondents identify practical value in arrangements already fitted to their workflows, although those arrangements differ structurally. The common pattern which will emerge is how respondents describe their current arrangements as practically useful or sufficient, not technically optimal. What will be shown is how the perceived value is connected to how well the arrangement supports the organization's existing work.

R1 explains how their current arrangement minimizes movement by handling data in one container at a time, which the respondent associates with flexibility:

[6] R1: "*[...] it gives you plenty of flexibility because now we just need to deal with each container at a time. So, there is not that much movement from one place to another. And we keep everything within the single container.*"

The respondent values this arrangement due to its reduction of movement between environments and allows data to remain within a contained technical unit.

Meanwhile, R2 gives a positive account of a separated environment. When asked if it would be valuable if both operational and analytical tasks could be performed in the same place, the respondent answered:

[18] R2: "*Would it be valuable? Yes, it is already possible. Some of it is probably done, but often it is not sufficient. Something more is needed, and that is why one chooses to invest in a separate analytical solution.*"

Value is given to a combined architecture of operational and analytical systems. But the respondent presses on the fact that this is not often sufficient, and more often something additional in terms of analytical solution is needed. This account gives clear support for the code definition. Although not claiming that separation is universally necessary, the respondent gives a clear insight into the value of keeping the analytical solution separate and the insufficiency of the combined solution. This insufficiency should also not be taken as a universal truth, but the input is valuable.

Another value described by R3 is that of consolidation, how several existing operational systems which would require individual extraction from each system, instead can be gathered into one analytical solution, in this account Power BI:

[16] R3: "*There may be a department working with three systems, and to perform the analysis they must extract it from all three. Instead, we can have Power BI collect the three and provide an overall analysis more quickly.*"

The value of the existing analytical arrangement for the respondent is therefore the consolidation of data from several operational systems and the reduction of the need for users to collect information separately. Even though giving a positive account of the separated solutions, the respondent still gives some value to the idea of a combined approach: *"Yes. It would be a bit valuable on a daily basis [...]"* ([16] R3).

An account in favor of decentralization is given by R4 who describes how the centralization of data into one large data warehouse is often very costly, difficult, and slow. Instead, the respondent favors distributing responsibility across the organization:

[14] R4: "*The experience is that centralizing the data in one large data warehouse or some kind of data-lake solution managed by one team is very difficult, costly and slow. [...] The data probably benefits from remaining as close to its own business activity as possible.*"

The respondent does not outright reject central reporting, but instead questions the full centralization of data and responsibility. The value of the arrangement lies in the combination of local ownership of data with controlled exchange, and, where useful, centralized reporting. This account does not straightforwardly support the formal definition that a separated setup is sufficient. It can thus be seen as a distinct position for the benefit of a distributed or federated organizational arrangement.

R5 gives a brief account of the sufficiency of the current arrangement: "*It is a lot of Office 365. It feels like that is the majority, and it is very good.*" ([16] R5). What this shows is how

the value of an existing arrangement can lie in commonly available tools, instead of a specialized analytical infrastructure.

Moreover, R6 provides a positive account of the organization's own analytical solution. As the respondent describes the analysis is integrated in the application and the respondent considers the arrangement to work very well:

[8] R6: "*It is our own solutions that we use for that. [...] if you also think about analysis. It is integrated into the application.*"

[13] R6: "*I think it works very well instead of going through another external site. We have built it in the way we want it. […] There aren't that many clicks we have to perform compared to if we would have used a Power BI system. [...] It is only a few button presses, and then we get what we want quite quickly.*"

What the respondent seems to value with the current arrangement is the way the analytical functionality can be tailored to fit the organization's needs and how the functionality is embedded directly into an application employees already use. Although the respondent values their current arrangement above systems like Power BI, one should be cautious of interpreting this as evidence of self-built analytical solution being superior to Power BI. Instead, the superiority concerns workflow since external data import would be required, users would need to specify the analysis themselves, the current system already contains much of the required setup.

Beyond producing reports, R9 gives insight into further value of certain arrangements. The respondent gives an account of a case where stock was added to an inventory, but withdrawals were not registered. The value of the current analytical arrangement thus came from interpreting how reasonable this was:

[8] R9: "*It is of course also a part of the analysis to see if it is reasonable, if there is an error in the source data. [...] we had issues with inventory balances. [...] We just stocked up the inventory. It didn't register when we withdrew from the inventory. Finally, we realized this was not reasonable.*"

What becomes clear by this account is how the value of an analytical arrangement can be the interpretation of the data's reasonability. The value is not limited to decision support based on assume-correct data. It can also provide a means of validating operational information and detecting failures in source systems.

What can be seen from all these different accounts is how the value of a current arrangement is never due to one universal architectural principle. Instead, the value can come from different reasons. A contained setup can reduce movement, a separate analytical solution can consolidate data from many sources, the distribution of responsibility can keep the data close to its organizational context, standard tools can be considered sufficient for common tasks, a self-built application-integrated analysis can reduce user effort and fit established workflows, an analytical solution can also support the validation of data. Respondents also sometimes describe current operational capabilities as sufficient while deeming them insufficient for broader analytical requirements.

4.3 Constraints on Analytical Capacity

4.3.1 Competence as Limiting Resource

The code *Competence as limiting resource* can be described as the constraint regarding the skill available in individuals, whether the capability rests on one or more people, or the people lack the required understanding. Throughout the codified material there will not be any single one skill gap identified, instead the respondents will describe several forms of limited competence. What will become clear is how analytical capacity depends not only on whether a tool exists, but on whether the organization has people who can maintain, operate, understand, and critically assess it.

R7 gives an insight into the idea of key-person dependency. Meaning, when many roles are comprised by one individual, and the issues this can bring to the organization if this individual no longer has these roles, the respondent gives an illustrative example:

[6] R7: "*Lasse, who has worked here for 20 years, has kept track of this database. He is the only person who knows that it exists. Then he retires, resigns, starts somewhere else, or changes roles. Suddenly people are using a database, but nobody knows where it is, and nobody knows who is responsible for it.*"

It becomes clear with this account that the limiting resource is not the database itself, but the distribution and continuity of knowledge surrounding it. The respondent also assesses these types of issue to be quite common: "*I would say this collection of problems are quite common*" ([6] R7), note however that this is the respondent's perspective and should not be seen as a verified fact across organizations.

Apart from organizational dependence, there can also exist limited technical competency. R7 continues to describe limited competency in how employees have trouble working in enterprise systems like Salesforce and ServiceNow, how column names and relationships can be difficult to understand, and the difficulty BI-analysts can have with basic join-operations:

[7] R7: "*Problem is many of these column names in these products like Salesforce and ServiceNow [...] can be really difficult to figure out. It can be very difficult to determine the primary- and foreign-key relationships between these tables. How do I join them? Many BI analysts do not even understand a join. What is the difference between an outer and an inner join? They do not know that. Often, you have to explain it to them or help construct the query*."

This account is the respondent's perspective and experience and should not be seen as evidence that BI analysts as a professional group lack database knowledge in general. But what can be garnered from the material is how analytical capacity can be constrained by a mismatch between access to data and the technical knowledge to retrieve and combine it correctly. This account also echoes what was mentioned in the subsection *Division of analytical work* (section 4.1.1). Analysts may understand the business question but may also depend on people with deeper database knowledge to access the data they need.

R8 gives a description of limited competence as a problem of both adaptation and foundational understanding. The respondent juxtaposes two contrasting patterns, where the first is employees accustomed to older Excel-based practices may resist newer tools and methods,

and the second is employees using AI-based tools to generate outputs while not understanding the underlying software or process ([14] R8). The respondent presents the refusal to learn new technology, and the refusal to learn the technical foundation as a limitation on analytical work:

[14] R8: "*[...] there is an extremely large gap between those who completely refuse new technology and those who refuse to learn how it works at a fundamental level.*"

For the respondent, competence can be seen both as the willingness to adopt new technologies in the form of tools, as well as the willingness to learn the fundamentals of said technologies. Once again, the account should be taken as the respondent's own perception and be seen as a general statement for organizations.

Another perspective on competency as a limiting resource is provided by R9 who describes how it can also depend on the frequency of use of the analytical tools. The respondent recalls when working as a full-time analyst how the more advanced systems worked well, but notes that if these analyses only take place once a month it can be more difficult:

[11] R9: "*I worked full-time as an analyst, and then it works well. But if you are going to do this once a month and use an advanced system, it usually becomes more difficult.*"

What is described is not how skills necessarily disappear, instead that infrequent use can make the use of more advanced systems more difficult. An organization may technically have employees with access to an analytical system but lack the necessary frequency of use in order to maintain practical proficiency. R9 continues by describing how competency is not only limited to the technical system in use but also the need for employees to have enough knowledge to interpret and question analytical outputs. They must be able to judge if reported revenue, deliveries, inventory, or other values are plausible. And how a lack of this judgement, erroneous outputs can be accepted since the system successfully produced a number:

[12] R9: "*You need both competence and experience so that you can quickly question the result. Does this seem reasonable? Do we normally have this turnover? Do we normally deliver this much?*"

This shows how analytical competence also includes contextual judgment, not only the technical ability to execute a query. This contrasts with the previous account from R7 where competency was described as technical database understanding, while R9 presses on organizational and domain understanding.

What the presented codified material has shown is how competency has many dimensions, it can include technical competence, operational competence, domain competence, and organizational continuity. Analytical capacity is shown to depend on both employees' technical ability to access and process data and also on their contextual ability to understand and critically evaluate the results.

4.3.2 Handoff Breakdown

This code refers to a situation when knowledge fails to pass between those who move the data and those who know what it means, or lack responsibility for the transfer. In contrast to the previous code which concerns whether the required knowledge exists, this code is about whether existing knowledge is successfully communicated and coordinated between roles.

A first account is given by R1 who describes how the handoff of knowledge can occur at both ends of an analytical process. This account was already visited in the subsection *Data integrity in transfer* (section 4.2.2), there the focus was format compatibility and preserving data, here the focus is whether the result is defined and presented in a way that another individual can actually use. As R1 states:

[18] R1: "*[...] there's always two different kinds of integrations. One part is the receiving the data, and another one is producing the results. And those are two different set of requirements.*"

The respondent then goes on to explain how the input stage requires a certain robustness toward different formats and technical methods, while the output stage requires deciding what information should be provided ([18] R1). The main takeaway from the respondent's description is that a successful handoff requires more than technically transmitting data. The ones producing the results must also properly define its content and adapt the format for the end user.

A clear description of the code is provided by R2 who identifies two distinct groups, firstly the data engineers or other employees who build pipelines and move data, and secondly platform or domain specialists who understand what the data means. R2 states this as follows:

[13] R2: "*A lot of it is about understanding what the data means.*"

And when asked what the biggest hurdles with the current reporting were, the respondent answers:

[17] R2: "*It is probably the knowledge transfer between the data engineers and the platform specialists, the people who know what the data means and those who need to build the data pipelines. I would say that is the biggest problem.*"

When a pipeline built by technical staff depends on knowledge held by employees outside of the pipeline-building role, problems can arise. The breakdown occurs when the data can be technically transferred, but its meaning, context, or correct use is not successfully transferred.

A brief account of how the same data can be interpreted differently is provided by R3:

[15] R3: "*The data is actually correct, but it is viewed through different eyes.*"

According to the respondent, data can technically be correct but by viewing it from another perspective the knowledge fails to properly transfer. This account is preceded by an explanation of how there can be different working practices in other countries where the business operates ([15] R3). Transferring the same dataset does not guarantee a shared understanding of what it means.

A rather detailed example of handoff breakdown in an organizational context is provided by R4. The respondent describes how since the responsibility is dispersed throughout the organization, and since this often leads to each team modifying the data to their own liking without notifying the data scientists, it can lead to failures between departments:

[13] R4: "*The data may be somewhat dispersed in the organization, perhaps between different IT departments responsible for different parts of the complete dataset.*"

The description of how data may be spread out in the organization is followed by:

[13] R4: "*If there is no shared responsibility for the data […] and they do not communicate that they are making updates, things will break repeatedly because contracts that do not exist are being broken between the data-science department and the IT department.*"

An important note with the previous quote is that the word "contracts" does not seem to imply formal legal contracts, the respondent instead seems to refer to technical or organizational agreements between data producers and users. But nonetheless, the account gives a clear description of how the handoff breakdown is not only a failure to explain individual data fields. Instead, it can result from unclear data ownership and responsibility between the individuals producing, maintaining, and analyzing the data.

R7 describes how certain domain-roles can act as bottlenecks in an analytical pipeline. The necessary domain knowledge may exist, but its communication and transfer fail due to being handled by employees who already have a full workload and who would need to take time to explain the proper meaning of the data:

[8] R7: "*The bottleneck is usually the business. You have to find people who are already fully occupied who can give you information and explain what the data means. That is what takes the most time.*"

The knowledge exists, but to obtain it employees must find the right individual and get a full explanation of what it means in order to prevent the handoff breakdown. And these individuals might have their primary responsibility elsewhere. The handoff is therefore constrained by availability and organizational priorities rather than by a total absence of knowledge. While R2 identified the knowledge transfer boundary, R7 gives insight into why that transfer can be slow in practice.

Similarly, R8 also describes a handoff breakdown in the organizational context. The respondent gives an account of how different departments working according to different needs and routines can give rise to issues:

[7] R8: "*We currently have a very large mismatch where the data we want and the data they want do not really align, because we work in two different ways.*"

And the respondent follows up with:

[7] R8: "*There is always some problem between departments. One person works in one way and someone else works in another way, and then it becomes difficult.*"

The mismatch described by the respondent should not be taken as a mismatch of data formats. The account has broader wording and may concern requirements, processes, or priorities.

What has been presented in this subsection is the different types of handoff breakdown. In summary these types can be grouped into three categories. Firstly, the breakdown can be semantic where the understanding of the data fails to transfer during the handoff. Secondly, technical handoffs which concern the defining of formats, structures, and outputs that another system or user can receive. And thirdly, organizational handoffs which concern the establishment of ownership, communication responsibilities, and agreements between departments. The codified material shows how analytical work can depend on the coordination between two groups of individuals. Individuals who understand the business meaning of the data, and

the individuals who technically collect, transform, or deliver it. The handoff breakdown occurs when meaning, requirements, changes, or responsibility do not properly pass successfully across the boundary between these two groups.

4.3.3 Workload Interference

The code *Workload interference* is defined as how analytical activity degrades operational system performance, or scheduling is chosen to avoid degradation. This subsection will introduce a pattern showing how interference in workloads in organizations can be seen as a conditional constraint, and how it is dependent on architecture, size and timing of the analytical workloads.

R3 starts by providing an example of how a separate solution can limit interference, and gives an account of how older tools used could be more cumbersome:

[14] R3: "*Since we have a separate solution for most of it, it does not affect us very much. Then we also have some older tools from which we previously extracted many reports, and those were somewhat more cumbersome. At certain times, such as the end of the month, when many reports run automatically, you notice that it becomes slower.*"

Although it is not explicitly stated that the older solution had more interference than the new separate solution. The respondent still associates the limited interference in the current environment with the use of the separate analytical solution. The interference in terms of slowdown is experienced with the older solutions and during concentrated periods of increased reporting, like at the end of the month. This account from the respondent does not explicitly prove that a separate solution will reduce workload interference, instead it should be seen as the respondent's description of this effect in a particular setting.

A clear example of how operational and analytical activity interfere with each other comes from R6 who states:

[11] R6: "*There were a number of occasions when we were producing a report. We use some data in real time, and sometimes, particularly in the beginning, when we generated a report, the report was produced, but the real-time data coming in stopped and we had to restart the process.*"

Important contextual note about this account is that what is referred to as the "beginning" is the time period in which they were building the application ([11] R6). But the insight is still relevant. The respondent describes how reporting did not only become slower, but it was also interrupting the ongoing real-time flow of data and required operational recovery.

Another example of workload interference is provided by R8, who describes this interference at a much larger organizational scale:

[5] R8: "*One example is our monthly calculations, where we have a week during which almost all IT at the bank is paused while we process [large amounts of data] in a database.*"

While it is not claimed that all IT at the bank was shut down, there was still a major disruption by pausing "almost all" of it while generating the monthly calculations. In contrast to R6's account, this instance of workload interference was scheduled and deliberate, instead of

unintended. The respondent describes an analytical workload large enough to affect the scheduling and availability of other IT activities across the organization.

R9 gives us a further insight into how organizations manage interference instead of merely experiencing it. The respondent describes how the organization scheduled report generation in order to not affect the organization's workflow:

[9] R9: "*We usually ran it once per day, early in the morning, in order not to reduce the performance of the system. There was still quite a lot of data that needed to be processed and moved.*"

The respondent also describes how sometimes updates closer in time are necessary, how you run these updates maybe two to three times per day. These updates would also be performed in hours when organizational activity was slower, like in the morning ([9] R9). The respondents account shows how workload interference can shape the timing of analytical processing. The description is clear in how the organization adjusted the update frequency and execution time in order to limit competition with operational use.

There are also instances in the codified material where workload interference is not described as a problem, or where it is outright denied. R1, for example gives a brief comment about how sometime a large dataset can lead to additional loading time, but how the analytical process itself was not a major source of interference ([11] R1). R5 describes a minor slowing down of workflow when working with large Excel files but the respondent did not describe this as significant ([12] R5). In the same fashion does R7 describes how the bottleneck is rarely the performance of the IT-systems themselves ([8] R7).

From R8 we can gain an insight into how the performance of the IT system can be compromised. The respondent describes how certain jobs can start competing for resources:

[13] R8: "*Another job may be running and taking up too many resources, so your job fails after four hours of calculations because the memory runs out.*"

The context of this quote is in terms of running traditional data storage in Microsoft SQL ([13] R8). For the cloud-based solutions the respondent instead mentions network problems as the main issue ([13] R8).

Taken together the described instances of workload interference can be presented as following. Firstly, there is the performance degradation where systems become slower during intensive reporting periods ([14] R3, [9] R9). Secondly, the operational interruption, where reporting or analytical processing stops an ongoing real-time data process ([11] R6). And thirdly, the resource and scheduling constraints where large jobs occupy shared processing resources or are scheduled during quiet periods to avoid affecting daily operations ([5] R8, [13] R8, [9] R9). The extent of the identified instances of workload interference was varied, and several respondents did not identify system performance as a significant limitation.

4.4 Priorities and Direction

4.4.1 Cost Criteria

The code *Cost criteria* is defined as where cost is named as a governing consideration when choosing analytical solutions. This criteria of cost are discussed in several ways by the respondents, some will describe the solving of as many analytical needs as cost-effective as possible, others will refer to the direct operating cost of a solution. The material will show how organizations do not always choose the cheapest alternative, and one respondent will explicitly state how certain improvements may justify a greater expense.

R2 gives an account of cost-effectiveness, the respondent describes the goal of solving problems as cost-effective as possible as well as emphasizing the satisfaction of as many needs as possible. The respondent states:

[19] R2: "*We should solve the problem as cost-effectively as possible. Meet as many of the needs as possible in the most cost-effective way.*"

Instead of interpreting this as a demand for the lowest-priced system, it should be seen as an instance of where cost is a criterion for evaluating how much analytical capability can be obtained for the resources spent.

When asked about cost criteria, R7 clearly states cost as the primary criteria. The respondent specifies this by referring to the cost of running a solution:

[12] R7: "*Number one is probably cost, how much it costs to run something. Number two is what you are accustomed to.*"

The respondent immediately ranks another criterion which is not cost. This is included merely to show that while cost was deemed important by the respondent, it was not the sole criteria for choosing an analytical solution.

Another account is provided by R8 who describes how cost can be experienced as a governing organizational constraint. The respondent connects the company's size with the prioritization of cost. The respondent also emphasizes this priority in a negative manner:

[15] R8: "*Unfortunately, when you are a company of this size, cost always becomes the first priority.*"

This statement should not be generalized as a fact of how organizations of a particular size always prioritize cost. Nonetheless, the respondent describes how cost is not merely one technical evaluation criteria. It also functions as an organizational condition that structures which alternatives can be considered. The respondent will also identify other factors, but this will be handled in the subsequent subsection.

In contrast to the previous accounts where cost is described merely as a criterion. R4 provides an important qualification to the value of this criterion. Sometimes, organizational improvement is not defined as reducing cost. A solution can also create value by improving the user experience, and how this improvement can justify higher expenditure:

[15] R4: "*It is not always necessarily about cost efficiencies. It can definitely concern improvements that provide a better user experience, even if it costs a little more.*"

The respondent does not reject cost as irrelevant but instead provides insight into why it is not always the decisive criterion. Additional expenditure may be accepted where another organizational benefit is considered sufficiently valuable. In relation to the code, this should be taken as a boundary case which can also be included in the next subsection *Non-cost criteria,* it does concern cost as a criterion but also juxtaposes it to the value certain higher cost decisions can provide.

In summary, the respondents all name the importance of cost as a criterion when choosing an analytical solution. Some respondents describe cost in regard to optimization, obtaining sufficient analytical capability for the expenditure, others describe it as a constraint which limits which solutions can be realistically selected. Respondents also describe cost as one criterion among many, which will be explored in the next subsection.

4.4.2 Non-Cost Criteria

This code is defined as criteria like usability, compatibility, risk, lock-in, longevity, an existing technical ecosystem, or something else governing, instead of, or ahead of, cost. This subsection will explore the criteria for determining the value of an analytical solution beyond cost. Respondents will emphasize the importance of employees being able to practically use the system, and how some criteria concern immediate use while others concern long-term organizational risk.

Respondents R3 and R4 provide a similar insight into the importance of usability as the primary reason for change. R3 describes how previous arrangements were not considered user-friendly enough and how this was changed, and that this is described as the main reason for said change:

[17] R3: "*I would probably say usability, actually. […] I did not think it was very user-friendly for the end user, so we needed to make it more generally accessible. I would say that is the main reason for us.*"

As explored in previous subsection, R4 also presses on the importance of usability, even in situations where costs will increase:

[15] R4: "*It is not always necessarily about cost efficiencies. It can definitely concern improvements that provide a better user experience, even if it costs a little more.*"

What the respondents share is their prioritization of usability over other criteria. A solution can be valuable because it is easier or better for employees to use, even when it is not the least expensive alternative.

R5 also emphasizes the importance of usability when choosing an analytical solution but also highlights the importance of being compatible with the surrounding technical environment:

[15] R5: "*It should be user-friendly and easy to use. It should be compatible with other systems, and it should simply be easy to work with.*"

The respondent mentions both technical compatibility and simplicity. But these should not be seen as interchangeable. Compatibility concerns the system's ability to operate alongside existing technologies, while simplicity concerns the user's interaction with the system. The respondent combines a user criterion with a technical one. Employees must understand a system while at the same time this system functions alongside the organization's other systems.

R6 also gives the criterion of usability high priority. And the respondent also describes how usability means in practice:

[13] R6: "*We have built it in the way we want it. There are not many clicks required. […] Almost everything is already set up since before, so it is only a few button presses and then we obtain what we want quite quickly.*"

The organization's solution is directly integrated into its existing application, and so users do not need to move through external platforms. The system is configured according to the organization's needs and user can therefore perform the analyses with relatively few actions. These benefits are contrasted by the association the respondent has to tools such as Power BI where there would be a requirement of additional imports and configuration work. This association should not be seen as a general disqualification of tools such as Power BI; this is the respondent's comparison based on the organization's particular workflow. The respondent values the solution since it reduces the number of steps between the user and the required analytical result.

R7 names another important criterion besides what was mentioned in the previous subsection. The respondent also describes how previous technological choices will constrain or guide later one:

[12] R7: "*Number two is what you are accustomed to. If you are already deeply invested in Azure and the Microsoft ecosystem, you may not choose something from AWS, for example.*"

The respondent describes how an analytical solution is not evaluated independently of the organization's existing infrastructure, skills, or vendor relationships. If the organization is invested in a certain solution, employees are accustomed to its workflow, and the solution is compatible with current services, it is valued higher, and can also determine any future changes of this analytical solution.

R8 provides a rather detailed account of long-term non-cost criteria and how risk also influences the choice of an analytical solution. The respondent phrases this as longevity, meaning how a solution will hold up in the long run, and what happens if the use of the solution is suddenly disrupted due to licensing issues or supplier disruption:

[15] R8: "*The second priority is longevity. Licensing and longevity. […] We conduct a substantial risk analysis concerning how quickly we could withdraw from the software we use.*"

The respondent also puts emphasis on a solutions replaceability:

[15] R8: "*Excel, for example, is a fantastic tool, and there are hundreds of similar tools, so the risk is very low because it can always be converted away from.*"

Note that the respondents mention of hundreds of tools similar to Excel should not be taken as a solid fact, this is the respondents' justification for considering it low risk. The respondent thus provides three distinct criteria. Longevity, or if the current solution is likely to remain

viable in the future. Licensing risk, how the organization will handle a change in conditions or increase in costs from the supplier. Exist possibility, whether data and processes can be migrated to another platform. The respondent evaluates a solution in part after the organization's ability to continue using the platform or the migration away from it if the underlying commercial or technical conditions change.

While most of the respondents have prioritized one or two non-cost criteria. R9 describes how the choice of an analytical solution may involve several requirements simultaneously:

[13] R9: "*It should be efficient, secure, affordable, user-friendly, and not create a lock-in effect. It should be flexible enough that we can change it when circumstances change.*"

Instead of ranking criteria in priority like previous respondents have, R9 instead describes how the desired solution must satisfy a combination of operational, technical, financial, and strategic requirements. The respondent does not claim that the organization currently possesses a solution which satisfies all these requirements, it is merely the respondent's description of a solution's desired characteristics.

The respondents have all mentioned several criteria which they deem important besides cost. These criteria have shown to operate at several levels. One important criterion is at the user- and workflow level, this criterion concerns user-friendliness, simplicity, fewer steps, faster access to prepared analyses. Then there is the technical level where the concern is about compatibility with other systems, how the solution fits with established vendor systems, and how the solution integrates with existing applications. Finally, there is the strategic level, this concerns aspects such as security, longevity, licensing risk, possibilities of migration, flexibility, and the avoidance of lock-in effects. The implication of these stated criteria is not that cost as a criterion disappears; many respondents rank non-cost factors alongside cost. Analytical solutions are assessed according to their fit with users, workflow, existing technologies, and future organizational needs. If conditions change, the solution will depend on factors such as longevity, flexibility, migration possibility, and a limited lock-in effect. Meanwhile, usability and compatibility determine whether a system works in the present environment.

4.4.3 Integration Stance

The code Integration stance is defined as a stated position on merging operational and analytical systems. The codified material will show how this will not be a uniform preference for either complete integration or strict separation of systems. Some respondents will describe benefits of having an analytical solution separate when the operational solution is insufficient. Other respondents will describe a desire for a shared analytical layer which combines data from several operational systems. In instances of a desire for a single solution, there must be caution about what this entails. Respondents often discuss integration at the level of applications, workflows, interfaces, or reporting rather than database architecture.

R1 provides an insight into how integration is not always a binary choice. Although explicitly stating a hesitancy of answering the question ([16] R1), the respondent sees advantages and disadvantages in different arrangements, the organization wants a certain independence while also facilitating the work:

[16] R1: "*I can see some pros and cons in different situations. At the end of the day, we want a certain level of independence, but at the same time we want to facilitate the work.*"

The respondent also describes how there is now an ongoing discussion about placing certain analytical functions on the operational side:

[16] R1: "*Normally, we have tried to keep everything on the analytical side, but we see that there may be an increasing need to put some things on the operational side.*"

Integration is therefore described as being a question of deciding which functions should remain on the analytical side, and which should move closer to operational use. The respondent's hesitancy to answer the question means a certain caution of the interpretation of this material is necessary. The respondent describes this as an ongoing discussion with an unresolved balance.

A more explicit preference is provided by R2 who describes how a separate analytical environment is still preferable. Even if some analytical functions can be performed by the operational system, this is often insufficient. When asked about the benefits if a combined solution was possible, the respondent answers:

[18] R2: "*It is already possible, and some of it is probably done, but it is often not sufficient. Something more is needed, and that is why one chooses to invest in a separate analytical solution.*"

This statement does not outright reject integration in all circumstances. Instead, it supports separation when the existing capability is insufficient, not as a universal principle. The respondent's position is based on analytical sufficiency.

From R3, a middle position between separation and integration is presented. The respondent describes how a department may work in several operational systems, and instead of extracting from each of these individually, a shared layer in Power BI can collect the data and provide a combined view of these sources more quickly. The respondent states:

[16] R3: "*There may be a department working with three systems. To perform the analysis, they must extract data from all three. Instead, Power BI can collect the three and provide an overall analysis more quickly.*"

This material does not describe an integrated source system, instead it concerns how data is brought together for analysis, and how this data originates from systems which are not necessarily merged. The shared analytical solution provides an integration from the user's perspective. The evidence supports consolidated reporting through Power BI.

Further support for separation is provided by R4 who describes the challenges a full centralized data warehouse can bring. The challenge is described as following:

[14] R4: "*Centralizing the data in one large data warehouse or some kind of data-lake solution managed by one team is very difficult, costly and slow.*"

The respondent continues by giving benefit to a central solution for reporting, but presses on the benefits of having data remain close to its immediate business context:

[14] R4: "*There can absolutely be a central system that handles reporting […] but I think the data probably benefits from remaining as close to its own business activity as possible.*"

This position described by the respondent can be seen as selective integration. A clear benefit for a central system for reporting is provided, but the respondent favors distributing responsibility for the underlying data.

A positive stance for integration is provided by R5, who describes how working in two different systems can have its limits. In response to the question of whether it would be valuable if reporting could be done directly from an existing operational system, the respondent answers:

[14] R5: "*Absolutely. As I said, I work in two different systems for the implementation work and for the documentation, which is in Office 365. But it is easier said than done for many different reasons.*"

Since the respondent does not specify what kind of integration would be beneficial, this statement should be interpreted with caution. Nonetheless, the respondent expresses a positive stance towards integration while acknowledging that the existing division between systems cannot be easily removed.

R8 provides a clear desire for an integrated solution. The respondent first describes how users of the financial department currently use Excel, and how the respondent wants them to use the organization's own web portal instead. This web portal, developed by the respondent, can perform functions such as calculations, starting jobs, extracting data, uploading rules, and related activities ([7] R8). The respondent expresses the goal of improving user experience and the removal of any need to process the data through numerous intermediate steps:

[7] R8: "*What they currently do in Excel should instead be possible directly in the web portal, so that they receive a better user experience and we avoid processing their data through ten steps before we can actually use it.*"

The respondent supports integration due to its effect of reducing movement between tools and allowing operational users and analytical employees to work through the same platform. The respondent continues to describe the broader goal and the expected organizational value:

[20] R8: "*My goal is to combine it into one system so that we can perform our analyses while they can view their transactions and everything in one place.*"

The respondent further emphasizes the benefits:

[20] R8: "*We gain better insight, more control over the system, and we know how the data is handled, while above all reducing the risk from the human factor.*"

The respondent's main argument for the benefit of an integrated solution is that of workflow simplification, control over data handling, and the removal of a transfer boundary between the internal platform and Excel. Although the respondent in this full quote refers to terms such as OLTP and OLAP, this should not be interpreted literally, the respondent appears to use these terms in a more conceptual fashion ([20] R8). The use of these terms does not in itself establish that the proposed integrated platform constitutes a technically implemented HTAP database.

Another account which describes integration in a favorable way comes from R9. The respondent describes how integration is desirable but difficult at scale. The respondent had a plan for consolidating data with Power BI with the purpose of connecting financial outcomes with service and maintenance tasks:

[7] R9: "*My plan was to begin using Power BI to combine data from the different systems and obtain an overall view.*"

Previous experience with organizations containing approximately 40 to 200 systems makes the respondent uncertain of if a unified solution can cover everything:

[15] R9: "*I am interested in seeing a unified solution, but from experience I have seen the challenges and problems involved when data still has to be extracted from different systems.*"

Analytical integration is favored by the respondent due to it enabling comparisons across functional areas. The respondent also recognizes that the number and diversity of source systems constrain how complete that integration can become.

From all the respondents accounts we can gather that integration can be desirable, and how integration can occur on different levels. Integration can be at the level of data, where data is combined from different source systems, it can be at the level of application and workflow, allowing operational and analytical users to work through the same interface or platform. Integration also concerns architecture, running both operational and analytical workloads within the same database environment. The positions taken by the respondents depend on analytical sufficiency, the number of source systems, organizational responsibility, workflow complexity, and the risk created by transferring data between tools. In summary, integration is favored where it supports a unified view of workloads and removes unnecessary boundaries. Separation on the other hand is favored where dedicated analytical capability or organizational dependence is required.

4.5 Overview of Benchmark Findings

This chapter presents the findings of the TPC-H benchmark experiment which compares three configurations: HTAP (a normalized 3NF schema with non-clustered columnstore indexing applied), OLAP-CS (a dimensional star schema with columnstore indexing applied), and OLAP-RS (a dimensional star schema with default rowstore indexing). Performance is evaluated using two primary metrics: execution time and CPU time. These metrics are directly comparable across all three configurations. Execution time reflects the response time experienced by the end user while CPU time reflects processing efficiency and hardware utilization.

I/O activity is reported as lob logical reads for tables accessed through its columnstore index and standard logical reads for tables accessed through its rowstore index. These are presented as complementary descriptive metrics within each category subchapter. Because these two read types are not directly comparable when considering unit size, no collective I/O formula is applied.

For each query, the first run of the six executions is excluded since they are considered warm-up runs. Hence, all reported values represent the median of runs 2-6. Since all three configurations operate on identical datasets under controlled conditions, the observed performance differences are attributed to schema design and indexing strategy, rather than external factors. The findings in this chapter are descriptive and form the empirical base for the discussion in chapter 5. At the same time, the results are presented not only as isolated performance measurements, but as empirical evidence for assessing whether a more integrated data architecture may be sufficient for resource constrained organizations in certain decision-support tasks.

Table 4.1 presents the complete findings regarding execution time and CPU time for all three configurations.

Table 4.1: Median execution time and CPU time across all three configurations

Query Number	Query Name	Primary technical nature	HTAP exec (ms)	OLAP-CS exec (ms)	OLAP-RS exec (ms)	HTAP CPU (ms)	OLAP-CS CPU (ms)	OLAP-RS CPU (ms)
Q1	Pricing Summary Report	Aggregation-intensive	91	85	1,737	173	172	11,750
Q2	Minimum Cost Supplier	Multi-layered	36	363	811	31	359	6,030
Q3	Shipping Priority	Join-intensive	392	132	328	391	250	1,816
Q4	Order Priority Checking	Multi-layered	71	70	564	63	63	3,172
Q5	Local Supplier Volume	Join-intensive	110	101	301	189	173	2,110
Q6	Forecasting Revenue Change	Filtering-intensive	71	71	219	63	78	1,359
Q7	Volume Shipping	Join-intensive	150	64	131	141	125	798
Q8	National Market Share	Join-intensive	145	90	100	141	157	639
Q9	Product Type Profit Measure	Join-intensive	253	205	290	391	328	2,029
Q10	Returned Item Reporting	Join-intensive	97	95	140	94	94	1,047

Q11	Important Stock Identification	Multi-layered	35	99	228	31	173	1,484
Q12	Shipping Modes and Order Priority	Aggregation-intensive	134	135	194	125	125	1,437
Q13	Customer Distribution	Multi-layered	271	443	1,084	530	843	7,249
Q14	Promotion Effect	Aggregation-intensive	52	53	169	79	94	1,314
Q15	Top Supplier	Multi-layered	54	54	169	47	62	1,249
Q16	Parts/Supplier Relationship	Multi-layered	167	238	447	125	375	2,685
Q17	Small-Quantity-Order Revenue	Multi-layered	54	59	75	110	124	558
Q18	Large Volume Customer	Multi-layered	9,651	442	2,535	9078	438	18,577
Q19	Discounted Revenue	Filtering-intensive	73	82	83	141	156	611
Q20	Potential Part Promotion	Multi-layered	62	95	172	63	94	1,170
Q21	Suppliers Who Kept Orders Waiting	Multi-layered	209	208	318	391	375	2,393

Q22	Global Sales Opportunity	Multi-layered	87	116	401	156	203	2,998

Exec = median elapsed time in milliseconds, runs 2-6. CPU = median CPU time in milliseconds, runs 2-6.

4.6 Aggregation-Intensive Queries (Q1, Q12, Q14)

The aggregation-intensive category is comprised of three queries: Q1, Q12, and Q14. Each of these queries perform large-volume column scanning with summary computations. These queries can represent the type of frequent financial and operational monitoring that managers and analysts rely on, for example, financial statistics, pricing performance, shipping mode comparisons, and promotion effectiveness. The results of these types of queries are highly relevant to the day-to-day decision-making processes rather than strategic or periodic analysis.

Table 4.2: Execution times and speed ratios for aggregation-intensive queries

Query Number	Query Name	HTAP exec (ms)	OLAP-CS exec (ms)	OLAP-RS exec (ms)	HTAP/OLAP-CS	HTAP/OLAP-RS
Q1	Pricing Summary Report	91	85	1,737	1.07	0.05
Q12	Shipping Modes and Order Priority	134	135	194	0.99	0.69
Q14	Promotion Effect	52	53	169	0.98	0.31

Speed ratio < 1 indicates HTAP is faster. Speed ratio > 1 indicates OLAP is faster.

Table 4.3: CPU time and ratios for aggregation-intensive queries

Query Number	Query Name	HTAP CPU (ms)	OLAP-CS CPU (ms)	OLAP-RS CPU (ms)	HTAP/OLAP-CS	HTAP/OLAP-RS

Q1	Pricing Summary Report	173	172	11,750	1.01	0.01
Q12	Shipping Modes and Order Priority	125	125	1,437	1.00	0.09
Q14	Promotion Effect	79	94	1,314	0.84	0.06

Speed ratio < 1 indicates HTAP is faster. Speed ratio > 1 indicates OLAP is faster.

4.6.1 Q1: Pricing Summary Report

The Pricing Summary Report query aggregates shipped lineitems up to a recent cutoff date by return flag and line status. It summarizes revenue, discounted revenue, tax-adjusted totals, average quantities, and average discounts. Q1 reported a median execution time of 91 ms for HTAP and 85 ms for OLAP-CS. With a speed ratio of 1.07 this indicates nearly identical performance between the two columnstore configurations with OLAP-CS being marginally faster. OLAP-RS reported a median of 1,737 ms, resulting in a speed ratio of 0.05, meaning HTAP completed the query in approximately 5% of the time required by OLAP-RS. CPU time followed the same pattern: 173 ms for HTAP, 172 ms for OLAP-CS, and 11,750 ms for OLAP-RS. Q1 accesses only the columnstore-indexed lineitem table which produces lob logical reads for both the HTAP and OLAP-CS configurations, as well as standard logical reads for OLAP-RS which is consistent with the large page-level scanning required by rowstore access on the fact table.

4.6.2 Q12: Shipping Modes and Order Priority

The Shipping Modes and Order Priority query supports logistics decisions by counting the amount of lineitems received by customers in a given year. It compares two shipping methods by how many of their deliveries arrive after the committed date. Q12 reported 134 ms for HTAP and 135 ms for OLAP-CS, with a speed ratio of 0.99, indicating almost identical performance. OLAP-RS reported 194 ms, with a speed ratio of 0.69. CPU time was identical for HTAP and OLAP-CS at 125 ms (speed ratio 1.00), compared to 1,437 ms for OLAP-RS which resulted in a speed ratio of 0.09. Out of all three aggregation-intensive queries, Q12 shows the smallest execution time difference between columnstore and rowstore configurations which is consistent with its heavy filtering in the WHERE clause which narrows the dataset significantly before the aggregation even begins. This drastically reduces the volume advantage that the columnstore indexing provides.

4.6.3 Q14: Promotion Effect

The Promotion Effect query calculates the share of total monthly revenue that can be attributed to promotional products. Q14 reported 52 ms for HTAP and 53 ms for OLAP-CS with a speed ratio of 0.98. OLAP-RS reported 169 ms with a speed ratio of 0.31. CPU time was 79 ms for HTAP and 94 ms for OLAP-CS, resulting in a speed ratio of 0.84, compared to 1,314 ms for OLAP-RS with a speed ratio of 0.06. The lower CPU time for HTAP relative to OLAP-CS despite near identical execution times displays a difference in processing efficiency that does not demonstrate as an end-user-observable speed difference.

4.6.4 Summary for Aggregation-Intensive Queries

Across the aggregation-intensive category, HTAP and OLAP-CS reported execution times with a maximum gap of 6 ms of each other in all three queries while OLAP-RS required between 1.45 and 19.09 times more time. CPU time differences between the two columnstore configurations were minimal, and in some cases congruent, whereas OLAP-RS required between 11.50 and 68.31 times more CPU time. This category displays the narrowest performance gap between HTAP and OLAP-CS out of any category in this benchmark experiment.

4.7 Join-Intensive Queries (Q3, Q5, Q7, Q8, Q9, Q10)

The join-intensive query category consists of six queries: Q3, Q5, Q7, Q8, Q9, and Q10. These queries combine data from several tables in order to produce analytical outputs involving customers, suppliers, orders, regions, products, and revenue. Examples include shipping prioritization, supplier evaluation, regional sales analysis, market share assessment, product profitability, and returned item reporting. Unlike aggregation-intensive queries, where performance is mainly driven by scanning and summarizing large volumes of data, join-intensive queries place greater demands on how efficiently each architecture reconstructs relationships between tables.

Table 4.4: Execution times and speed ratios for join-intensive queries

Query Number	Query Name	HTAP exec (ms)	OLAP-CS exec (ms)	OLAP-RS exec (ms)	HTAP/OLAP-CS	HTAP/OLAP-RS
Q3	Shipping Priority	392	132	328	2.97	1.20
Q5	Local Supplier Volume	110	101	301	1.09	0.37
Q7	Volume Shipping	150	64	131	2.34	1.15

Q8	National Market Share	145	90	100	1.61	1.45
Q9	Product Type Profit Measure	253	205	290	1.23	0.87
Q10	Returned Item Reporting	97	95	140	1.02	0.69

Speed ratio < 1 indicates HTAP is faster. Speed ratio > 1 indicates OLAP is faster.

Table 4.5: CPU time and ratios for join-intensive queries

Query Number	Query Name	HTAP CPU (ms)	OLAP-CS CPU (ms)	OLAP-RS CPU (ms)	HTAP/OLAP-CS	HTAP/OLAP-RS
Q3	Shipping Priority	391	250	1,816	1.56	0.22
Q5	Local Supplier Volume	189	173	2,110	1.09	0.09
Q7	Volume Shipping	141	125	798	1.13	0.18
Q8	National Market Share	141	157	639	0.90	0.22
Q9	Product Type Profit Measure	391	328	2,029	1.19	0.19
Q10	Returned Item Reporting	94	94	1,047	1.00	0.09

Speed ratio < 1 indicates HTAP is faster. Speed ratio > 1 indicates OLAP is faster.

4.7.1 Q3: Shipping Priority

The Shipping Priority query retrieves the highest-revenue unshipped orders and their shipping priority.

Q3 reported a median execution time of 392 ms for HTAP and 132 ms for OLAP-CS. This produces a speed ratio of 2.97, meaning that HTAP required almost three times as much execution time as OLAP-CS. OLAP-RS reported 328 ms, which means that HTAP was also slower than the rowstore dimensional schema for this query, with a speed ratio of 1.20. CPU time followed a similar pattern: HTAP used 391 ms, OLAP-CS used 250 ms, and OLAP-RS used 1,816 ms.

This result indicates that Q3 is one of the clearer cases where the dimensional columnstore configuration outperformed the integrated HTAP configuration. Although HTAP still completed the query in less than half a second, the relative difference compared with OLAP-CS is quite substantial.

4.7.2 Q5: Local Supplier Volume

The Local Supplier Volume query measures revenue volume by nation for transactions where customers and suppliers are located in the same nation within a selected region, supporting evaluation of local distribution decisions.

Q5 reported a median execution time of 110 ms for HTAP and 101 ms for OLAP-CS. The resulting speed ratio of 1.09 indicates that OLAP-CS was slightly faster, but the absolute difference between the two configurations was only 9 ms. OLAP-RS reported 301 ms, meaning that HTAP completed the query in approximately 37 % of the time required by OLAP-RS. CPU time showed a similar pattern: HTAP used 189 ms, OLAP-CS used 173 ms, and OLAP-RS used 2,110 ms.

Unlike in Q3, the performance gap between HTAP and OLAP-CS is small. Although OLAP-CS remained the faster configuration, the difference here is unlikely to be meaningful from an end-user perspective in this experimental setting. For this type of supplier-volume analysis, the result suggests that a columnstore-indexed normalized database can approximate the performance of a dedicated dimensional columnstore schema to a high degree.

4.7.3 Q7: Volume Shipping

The Volume Shipping query measures discounted revenue from goods shipped between two nations over time, supporting evaluation of shipping relationships and contract decisions. The query is join-intensive because it combines supplier, customer, order, and lineitem data in order to produce a regional revenue view.

Q7 reported a median execution time of 150 ms for HTAP and 64 ms for OLAP-CS. This gives a speed ratio of 2.34, indicating that HTAP required more than twice the execution time of OLAP-CS. OLAP-RS reported 131 ms, which means that HTAP was also slower than the rowstore dimensional schema for this query, with a speed ratio of 1.15. CPU time was closer between HTAP and OLAP-CS: HTAP used 141 ms, OLAP-CS used 125 ms, and OLAP-RS used 798 ms.

The result shows that OLAP-CS had a clear execution-time advantage for Q7, even though the CPU-time difference between HTAP and OLAP-CS was relatively limited. This suggests that the dimensional structure may have reduced execution-time overhead for this regional shipping analysis.

4.7.4 Q8: National Market Share

The National Market Share query measures how a nation's market share within a region changes over two years for a selected product type.

Q8 reported a median execution time of 145 ms for HTAP and 90 ms for OLAP-CS. The resulting speed ratio of 1.61 indicates that OLAP-CS was faster in execution time. OLAP-RS reported 100 ms, meaning that HTAP was also slower than OLAP-RS, with a speed ratio of 1.45. However, CPU time showed a different pattern: HTAP used 141 ms compared with 157 ms for OLAP-CS and 639 ms for OLAP-RS. This means that HTAP used slightly less CPU time than OLAP-CS, even though its elapsed execution time was higher.

This result is important because it shows that execution time and CPU time do not always point in exactly the same direction. From the user's perspective, OLAP-CS produced the faster response. From a processing-efficiency perspective, however, HTAP used slightly less CPU time.

4.7.5 Q9: Product Type Profit Measure

The Product Type Profit Measure query analyzes profit for a selected line of parts, broken down by supplier nation and year. The query is join-intensive because profitability must be calculated by combining product, supplier, order, and lineitem information.

Q9 reported a median execution time of 253 ms for HTAP and 205 ms for OLAP-CS. This gives a speed ratio of 1.23, meaning that OLAP-CS was faster, but the difference was less pronounced than in Q3 and Q7. OLAP-RS reported 290 ms, meaning that HTAP was faster than OLAP-RS with a speed ratio of 0.87. CPU time followed the same general pattern: HTAP used 391 ms, OLAP-CS used 328 ms, and OLAP-RS used 2,029 ms.

Q9 therefore shows a moderate advantage for OLAP-CS, while HTAP still remained clearly faster and more CPU-efficient than OLAP-RS.

4.7.6 Q10: Returned Item Reporting

The Returned Item Reporting query identifies customers associated with returned items and ranks them by revenue, which can indicate customers who may be experiencing problems with shipped parts.

Q10 reported a median execution time of 97 ms for HTAP and 95 ms for OLAP-CS. The speed ratio of 1.02 indicates near-identical execution performance between the two configurations. OLAP-RS reported 140 ms, meaning that HTAP completed the query in approximately 69% of the time required by OLAP-RS. CPU time was identical for HTAP and OLAP-CS at 94 ms, while OLAP-RS required 1,047 ms.

This is one of the strongest join-intensive results for HTAP. Despite the query requiring joins across several business entities, the HTAP configuration performed almost identically to OLAP-CS in both execution time and CPU time.

4.7.7 Summary for Join-Intensive Queries

Across the join-intensive category, OLAP-CS was the fastest configuration in execution time for all six queries. However, the size of the performance gap varied considerably. In Q5 and Q10, HTAP performed almost identically to OLAP-CS, with execution time ratios of 1.09 and 1.02 respectively. In Q9, the gap was moderate, with a ratio of 1.23. By contrast, Q3 and Q7 showed clearer OLAP-CS advantages, with HTAP/OLAP-CS ratios of 2.97 and 2.34. Q8 also favored OLAP-CS in execution time, although HTAP used slightly less CPU time.

Compared with OLAP-RS, the HTAP configuration performed better in Q5, Q9, and Q10, but worse in Q3, Q7, and Q8. This indicates that the rowstore dimensional schema did not consistently provide a useful middle ground between HTAP and OLAP-CS. Instead, the category demonstrates that performance is strongly query dependent.

The join-intensive category provides a more cautious result than the aggregation-intensive category. HTAP can approximate OLAP-CS for some join-intensive business analyses, especially returned item reporting and supplier-volume analysis. However, for queries involving more demanding cross-table reconstruction, such as shipping priority and regional volume shipping, the dedicated dimensional columnstore schema retains a clearer performance advantage.

4.8 Filtering-Intensive Queries (Q6, Q19)

The filtering intensive queries category consists of two queries: Q6, and Q19. These queries mainly test how efficiently each configuration can reduce a large dataset to the rows that match specific conditions. The filters include dates, discounts, quantities, product attributes, shipping modes, and delivery instructions.

Table 4.6: Execution times and speed ratios for filtering-intensive queries

Query Number	Query Name	HTAP exec (ms)	OLAP-CS exec (ms)	OLAP-RS exec (ms)	HTAP/OLAP-CS	HTAP/OLAP-RS
Q6	Forecasting Revenue Change	71	71	219	1.00	0.32
Q19	Discounted Revenue	73	82	83	0.89	0.88

Speed ratio < 1 indicates HTAP is faster. Speed ratio > 1 indicates OLAP is faster.

Table 4.7: CPU time and ratios for filtering-intensive queries

Query Number	Query Name	HTAP CPU (ms)	OLAP-CS CPU (ms)	OLAP-RS CPU (ms)	HTAP/OLAP-CS	HTAP/OLAP-RS
Q6	Forecasting Revenue Change	63	78	1,359	0.81	0.05
Q19	Discounted Revenue	141	156	611	0.90	0.23

Speed ratio < 1 indicates HTAP is faster. Speed ratio > 1 indicates OLAP is faster.

4.8.1 Q6: Forecasting Revenue Change

The Forecasting Revenue Change query estimates the revenue effect of discount conditions during a specific time period. In practical terms, it represents a "what if" revenue analysis.

Q6 reported a median execution time of 71 ms for HTAP and 71 ms for OLAP-CS. This gives a speed ratio of 1.00, meaning that the two columnstore configurations performed identically in execution time. OLAP-RS reported 219 ms. HTAP therefore completed the query in approximately 32% of the time required by OLAP-RS. CPU time followed the same general pattern. HTAP used 63 ms, OLAP-CS used 78 ms, and OLAP-RS used 1,359 ms. This means that HTAP was slightly more CPU-efficient than OLAP-CS and substantially more compared to OLAP-RS.

The results show that HTAP did not suffer any execution time disadvantage in Q6. For this type of targeted revenue query, the integrated configuration performed just as well as the dedicated dimensional columnstore schema.

4.8.2 Q19: Discounted Revenue

Q19 calculates discounted revenue for selected products shipped under specific conditions. The query filters by brand, container type, quantity, part size, shipping mode, and delivery instruction.

Q19 reported a median execution time of 73 ms for HTAP and 82 ms for OLAP-CS. This gives a speed ratio of 0.89, meaning that HTAP was slightly faster. OLAP-RS reported 83 ms, which means that HTAP was also slightly faster than the rowstore dimensional configuration. CPU time showed the same direction. HTAP used 141 ms, OLAP-CS used 156 ms, and OLAP-RS used 611 ms. The difference between HTAP and OLAP-CS was small, but HTAP was still the most efficient configuration on both primary metrics.

Q19 is therefore one of the cases where the HTAP configuration performed best. The absolute differences are not large, but the result is still relevant. It shows that the normalized,

columnstore-indexed configuration can handle this type of selective revenue analysis without losing performance compared with the dimensional alternatives.

4.8.3 Summary for Filtering-Intensive Queries

Across the filtering-intensive query category, HTAP performed strongly. In Q6, HTAP and OLAP-CS had identical execution times, and in Q19, HTAP was slightly faster than both OLAP configurations. On top of this, HTAP also used less CPU time than both alternatives in both queries.

These queries represent targeted revenue and discount analyses rather than broad enterprise reporting. In this category, the integrated HTAP configuration provided performance that was equal to, or slightly better than the dedicated dimensional columnstore, as well as the rowstore schemas. The result should be limited to the queries tested here, but the pattern clearly supports the viability of HTAP for this type of workload.

4.9 Multi-Layered Queries (Q2, Q4, Q11, Q13, Q15, Q16, Q17, Q18, Q20, Q21, Q22)

The multi-layered query category consists of eleven queries: Q2, Q4, Q11, Q13, Q15, Q16, Q17, Q18, Q20, Q21, and Q22. These queries are characterized by nested conditions, subqueries, correlated subqueries, common table expressions, and multi-stage aggregation. In practice, they represent analytical tasks where the database must do more than scan and aggregate data. It must also evaluate intermediate results before producing the final output. These include procurement analysis, customer segmentation, stock identification, supplier evaluation, promotion planning, and sales opportunity analysis. Such queries are often less routine than basic revenue reports, but they can still be important for managerial decision-making. The category is therefore useful for assessing whether an integrated HTAP-oriented configuration remains viable when the analytical workload becomes structurally more complex.

Table 4.8: Execution times and speed ratios for multi-layered queries

Query Number	Query Name	HTAP exec (ms)	OLAP-CS exec (ms)	OLAP-RS exec (ms)	HTAP/OLAP-CS	HTAP/OLAP-RS
Q2	Minimum Cost Supplier	36	363	811	0.10	0.04
Q4	Order Priority Checking	71	70	564	1.01	0.13
Q11	Important Stock Identification	35	99	228	0.35	0.15

Q13	Customer Distribution	271	443	1,084	0.61	0.25
Q15	Top Supplier	54	54	169	1.00	0.32
Q16	Parts/Supplier Relationship	167	238	447	0.70	0.37
Q17	Small-Quantity-Order Revenue	54	59	75	0.92	0.72
Q18	Large Volume Customer	9,651	442	2,535	21.84	3.81
Q20	Potential Part Promotion	62	95	172	0.65	0.36
Q21	Suppliers Who Kept Orders Waiting	209	208	318	1.00	0.66
Q22	Global Sales Opportunity	87	116	401	0.75	0.22

Speed ratio < 1 indicates that HTAP is faster. Speed ratio > 1 indicates that OLAP is faster.

Table 4.9: CPU time and ratios for multi-layered queries

Query Number	**Query Name**	**HTAP CPU (ms)**	**OLAP-CS CPU (ms)**	**OLAP-RS CPU (ms)**	**HTAP/OLAP-CS**	**HTAP/OLAP-RS**
Q2	Minimum Cost Supplier	31	359	6,030	0.09	0.01
Q4	Order Priority Checking	63	63	3,172	1.00	0.02

Q11	Important Stock Identification	31	173	1,484	0.18	0.02
Q13	Customer Distribution	530	843	7,249	0.63	0.07
Q15	Top Supplier	47	62	1,249	0.76	0.04
Q16	Parts/Supplier Relationship	125	375	2,685	0.33	0.05
Q17	Small-Quantity-Order Revenue	110	124	558	0.89	0.20
Q18	Large Volume Customer	9,078	438	18,577	20.73	0.49
Q20	Potential Part Promotion	63	94	1,170	0.67	0.05
Q21	Suppliers Who Kept Orders Waiting	391	375	2,393	1.04	0.16
Q22	Global Sales Opportunity	156	203	2,998	0.77	0.05

Speed ratio < 1 indicates that HTAP is faster. Speed ratio > 1 indicates that OLAP is faster.

4.9.1 Q2: Minimum Cost Supplier

Minimum Cost Supplier identifies the minimum cost supplier for parts of a given type and size within a selected region. It also reports supplier information such as account balance, name, nation, address, phone number, and comment information.

Q2 reported a median execution time of 36 ms for HTAP and 363 ms for OLAP-CS. This gives a speed ratio of 0.10, meaning that HTAP completed the query in about one tenth of the time required by OLAP-CS. OLAP-RS reported 811 ms, making HTAP substantially faster than both dimensional configurations. CPU time followed the same pattern: HTAP used 31 ms, compared with 359 ms for OLAP-CS and 6,030 ms for OLAP-RS.

This result is one of the clearest advantages for HTAP in the multi-layered category. Despite the query's nested supplier-cost logic, the normalized columnstore-indexed configuration handled the query efficiently.

4.9.2 Q4: Order Priority Checking

Order Priority Checking counts orders from a given quarter where at least one lineitem was received later than its committed date. The result is grouped by order priority. This makes the query relevant for delivery performance monitoring and operational follow-up.

Q4 reported a median execution time of 71 ms for HTAP and 70 ms for OLAP-CS. The speed ratio of 1.01 indicates practically identical execution performance. OLAP-RS reported 564 ms, meaning that both columnstore configurations were substantially faster than the rowstore dimensional schema. CPU time was identical for HTAP and OLAP-CS at 63 ms, compared with 3,172 ms for OLAP-RS.

The result shows that HTAP performed almost exactly like OLAP-CS for this query.

4.9.3 Q11: Important Stock Identification

Important Stock Identification identifies parts that represent a significant share of supplier stock value in a given nation. It does this by calculating stock value from supply cost and available quantity.

Q11 reported a median execution time of 35 ms for HTAP and 99 ms for OLAP-CS. This gives a speed ratio of 0.35, meaning that HTAP was notably faster. OLAP-RS reported 228 ms. CPU time showed an even larger relative difference: HTAP used 31 ms, compared with 173 ms for OLAP-CS and 1,484 ms for OLAP-RS.

This result suggests that HTAP handled the stock-value calculation particularly well. The query includes aggregation and a threshold condition, but the normalized HTAP configuration still outperformed both dimensional alternatives.

4.9.4 Q13: Customer Distribution

Customer Distribution analyzes the distribution of customers by the number of orders they have placed. It also excludes orders with a specified comment pattern.

Q13 reported a median execution time of 271 ms for HTAP and 443 ms for OLAP-CS. This gives a speed ratio of 0.61, meaning that HTAP was faster than OLAP-CS. OLAP-RS reported 1,084 ms, which again placed it clearly behind the two columnstore configurations. CPU time followed the same direction: HTAP used 530 ms, OLAP-CS used 843 ms, and OLAP-RS used 7,249 ms.

The result is relevant because Q13 is not a simple scan-and-aggregate query. It includes an outer join and a nested aggregation step. Even so, HTAP performed better than the dedicated dimensional columnstore schema.

4.9.5 Q15: Top Supplier

Top Supplier identifies the supplier or suppliers that contributed the most revenue from parts shipped during a given quarter.

Q15 reported a median execution time of 54 ms for both HTAP and OLAP-CS. This gives a speed ratio of 1.00, indicating identical execution-time performance. OLAP-RS reported 169 ms. CPU time was 47 ms for HTAP, 62 ms for OLAP-CS, and 1,249 ms for OLAP-RS.

The results show that HTAP and OLAP-CS were equivalent in user-observable response time for this query. HTAP also used slightly less CPU time. For this type of supplier-revenue analysis, the integrated configuration therefore produced performance comparable to the dedicated dimensional columnstore schema.

4.9.6 Q16: Parts/Supplier Relationship

Parts/Supplier Relationship counts distinct suppliers for combinations of part brand, type, and size. It also excludes suppliers whose comments indicate customer complaints.

Q16 reported a median execution time of 167 ms for HTAP and 238 ms for OLAP-CS. This gives a speed ratio of 0.70, meaning that HTAP was faster. OLAP-RS reported 447 ms. CPU time showed an even stronger HTAP advantage: 125 ms for HTAP, compared with 375 ms for OLAP-CS and 2,685 ms for OLAP-RS.

This result suggests that the normalized HTAP configuration handled the supplier-counting and exclusion logic efficiently.

4.9.7 Q17: Small-Quantity-Order Revenue

Small-Quantity-Order Revenue estimates the average yearly revenue that would be lost if small-quantity orders were no longer accepted for selected parts.

Q17 reported a median execution time of 54 ms for HTAP and 59 ms for OLAP-CS. The speed ratio of 0.92 indicates that HTAP was slightly faster, although the absolute difference was only 5 ms. OLAP-RS reported 75 ms. CPU time was 110 ms for HTAP, 124 ms for OLAP-CS, and 558 ms for OLAP-RS.

The result shows that all three configurations completed Q17 relatively quickly, but HTAP remained the fastest on both primary metrics. The practical difference between HTAP and OLAP-CS is small, yet it still supports the broader pattern that HTAP can be highly competitive for several multi-layered queries.

4.9.8 Q18: Large Volume Customer

Large Volume Customer ranks the top customers who have placed large-quantity orders. It reports customer, order, date, total price, and quantity information.

Q18 is the major outlier in the multi-layered category. It reported a median execution time of 9,651 ms for HTAP, compared with 442 ms for OLAP-CS and 2,535 ms for OLAP-RS. This

gives a speed ratio of 21.84 compared with OLAP-CS and 3.81 compared with OLAP-RS. In other words, HTAP was more than twenty-one times slower than the dimensional columnstore schema and almost four times slower than the dimensional rowstore schema. CPU time showed the same general pattern. HTAP used 9,078 ms, while OLAP-CS used 438 ms and OLAP-RS used 18,577 ms. Compared with OLAP-CS, HTAP required more than twenty times as much CPU time. Compared with OLAP-RS, however, HTAP used less CPU time, even though its execution time was slower than OLAP-RS. This indicates that Q18 created a severe elapsed-time problem for the HTAP configuration, but not the highest CPU load among the three alternatives.

The result is important because it limits any broad claim that HTAP can generally approximate OLAP-CS for multi-layered queries. Q18 first identifies high-quantity orders and then joins back to customer, order, and lineitem data. Under the tested configuration, this structure was much less favorable for the normalized HTAP schema than for the dimensional columnstore schema.

4.9.9 Q20: Potential Part Promotion

Potential Part Promotion identifies suppliers in a selected nation who have sufficient available quantity of selected parts to support potential promotion.

Q20 reported a median execution time of 62 ms for HTAP and 95 ms for OLAP-CS. This gives a speed ratio of 0.65, meaning that HTAP was faster. OLAP-RS reported 172 ms. CPU time was also lower for HTAP: 63 ms compared with 94 ms for OLAP-CS and 1,170 ms for OLAP-RS.

The result suggests that HTAP handled this promotion-planning query efficiently, despite the use of nested conditions.

4.9.10 Q21: Suppliers Who Kept Orders Waiting

Suppliers Who Kept Orders Waiting identifies suppliers in a given nation whose product was part of a multi-supplier order where they were the only supplier who failed to meet the committed delivery date.

Q21 reported a median execution time of 209 ms for HTAP and 208 ms for OLAP-CS. The difference was only 1 ms, producing a speed ratio of 1.00 when rounded to two decimals. OLAP-RS reported 318 ms. CPU time was also close between the two columnstore configurations: 391 ms for HTAP and 375 ms for OLAP-CS, compared with 2,393 ms for OLAP-RS.

The result shows near-identical performance between HTAP and OLAP-CS. Given the query’s relatively complex logic, this is a meaningful finding. For supplier-delay analysis, the integrated HTAP configuration performed at practically the same level as the dedicated dimensional columnstore schema.

4.9.11 Q22: Global Sales Opportunity

Global Sales Opportunity identifies country codes where customers have not placed orders for seven years but have above average positive account balances. The query also reports the number of such customers and the total account balance for each country code.

Q22 reported a median execution time of 87 ms for HTAP and 116 ms for OLAP-CS. This gives a speed ratio of 0.75, meaning that HTAP was faster. OLAP-RS reported 401 ms. CPU time followed the same pattern: HTAP used 156 ms, OLAP-CS used 203 ms, and OLAP-RS used 2,998 ms.

This result indicates that HTAP performed well for this customer-opportunity query. The query includes filtering, aggregation, and exclusion logic, yet the normalized columnstore-indexed configuration outperformed both dimensional alternatives.

4.9.12 Summary for Multi-Layered Queries

The multi-layered query category produced the most varied results in the benchmark. In most queries, HTAP either outperformed OLAP-CS or performed very close to it. HTAP was faster than OLAP-CS in Q2, Q11, Q13, Q16, Q17, Q20, and Q22. It performed identically or near-identically in Q4, Q15, and Q21. The only clear exception was Q18, where HTAP was drastically slower than both OLAP-CS and OLAP-RS in execution time.

When comparing with OLAP-RS, HTAP was faster in all multi-layered queries except Q18. This confirms the broader pattern that the rowstore dimensional schema was generally less competitive than the two columnstore-based configurations. However, Q18 also shows that columnstore indexing alone does not guarantee strong performance when the query structure interacts unfavorably with the normalized schema.

This category gives a nuanced result. Many complex analytical tasks, including supplier-cost analysis, stock identification, customer distribution, promotion planning, and sales opportunity analysis were handled effectively by HTAP. This supports the potential viability of an integrated architecture. At the same time, Q18 demonstrates that certain high-volume customer and order analyses may still create serious performance problems for the HTAP configuration.

The main finding in this category is therefore not that HTAP is generally better or worse than OLAP-CS. Rather, the result shows that performance depends strongly on the specific query structure.

4.10 Summary of Observed Patterns Across Query Categories

The results show that the performance relationship between HTAP, OLAP-CS, and OLAP-RS was strongly dependent on query type. No single configuration was uniformly superior across all queries. Instead, the benchmark indicates that the viability of the integrated HTAP configuration depends on the structure of the analytical workload.

The clearest pattern is that OLAP-RS was generally the weakest configuration. In most queries, the rowstore dimensional schema required substantially more execution and CPU time

than the two columnstore-based configurations. This was especially visible in aggregation-intensive queries, where OLAP-RS was much slower than both HTAP and OLAP-CS. The result suggests that dimensional modeling alone was not enough to produce strong analytical performance in this experiment. Columnstore indexing appears to have been a more decisive factor.

The aggregation-intensive category produced the narrowest performance gap between HTAP and OLAP-CS. Across Q1, Q12, and Q14, the two columnstore configurations performed almost identically in execution time. This is an important result because these queries represent recurring financial and operational monitoring tasks, such as pricing summaries, shipping mode analysis, and promotion evaluation. This category provides the strongest evidence that a columnstore-indexed normalized OLTP database can approximate the performance of a dedicated dimensional OLAP schema.

The filtering-intensive category showed a similar pattern. In Q6, HTAP and OLAP-CS had identical execution times, while in Q19, HTAP was slightly faster than both OLAP alternatives. These queries represent targeted revenue and discount analyses, where the database filters data according to specific business conditions. The results suggest that this type of analytical workload can be handled effectively by the HTAP configuration.

The join-intensive category produced a more cautious result. OLAP-CS was the fastest configuration in execution time for all six join-intensive queries, but the size of the gap varied. In Q5 and Q10, HTAP performed very close to OLAP-CS. In Q9, the difference was moderate. In Q3 and Q7, however, OLAP-CS had a clearer advantage. Q8 also favored OLAP-CS in execution time, even though HTAP used slightly less CPU time. This indicates that join-intensive workloads cannot be assessed as one uniform category. Some join-heavy business analyses may be suitable for an integrated HTAP approach, while others still benefit from the dimensional structure of OLAP-CS.

The multi-layered category was the most varied. In several queries, HTAP outperformed OLAP-CS, including Q2, Q11, Q13, Q16, Q17, Q20, and Q22. It also performed identically or near-identically in Q4, Q15, and Q21. These results show that multi-layered query structure alone does not necessarily disadvantage the normalized HTAP configuration. However, Q18 was a major exception. In that query, HTAP was substantially slower than both OLAP-CS and OLAP-RS in execution time. This makes Q18 an important warning case. It shows that some customer- and order-focused analyses can interact unfavorably with the normalized HTAP schema, even when columnstore indexing is applied.

Taken together, the results suggest that the main distinction is not simply between normalized and dimensional schemas. Instead, performance appears to depend on the interaction between schema design, indexing strategy, and query structure. HTAP performed particularly well in aggregation-intensive, filtering-intensive, and several multi-layered queries. OLAP-CS retained clearer advantages in parts of the join-intensive category and especially in Q18. OLAP-RS was generally less competitive, which indicates that the dedicated OLAP schema only became consistently strong when combined with columnstore indexing.

The results do not support a general conclusion that resource constrained organizations always need a separate dimensional OLAP schema for analytical performance. For several types of analytical tasks, especially recurring monitoring, targeted revenue analysis, stock identification, supplier evaluation, and promotion planning, the integrated HTAP configuration delivered performance that was comparable to or better than OLAP-CS. At the same

time, the results also show that HTAP should not be treated as a universal replacement for dedicated OLAP environments. Certain workloads, particularly those resembling Q18, may still justify having a dedicated analytical schema.

The overall empirical pattern therefore supports a workload-dependent interpretation. A columnstore-indexed normalized OLTP database can approximate the analytical performance of a dedicated dimensional OLAP schema in several decision-support scenarios, but not across all query types. The practical implication is therefore that architectural decisions should be based on the specific analytical workload the organization needs to support, rather than on a general assumption that either integrated HTAP-oriented architectures or separated OLTP/OLAP architectures are always preferable.

5 Discussion

5.1 What Constrains Analytical Capacity in Practice

This section uses the interview material to answer a question that section 2.2 left open. The earlier research in the literature review agrees that the obstacle to analytical capability is organizational rather than technical. The literature does not agree exactly on which part of the organization the obstacle is in. Kgakatsi et al. (2024) call financial limitations a major hurdle. Caldeira and Ward (2002) found that in-house IS/IT skills rather than money were what separated the more successful firms from the less successful ones. Kasiri et al. (2024) reported both expertise and money and in their material the balance between the two shifted depending on the size of the organization. The interviews were a way to establish which of these constraints in practice.

The interview material supports a distinction that the literature does not draw. Cost is a real criterion, but it applies at a different point in the process than competence does. Respondents named cost when we asked what governs the choice regarding the analytical solution. R7 ranked cost first among the selection criteria, and R8 described it as an organizational condition that decides which alternatives can be considered at all. When we asked what obstructs their analytical work, the same respondents described competence rather than budget. Cost therefore applies at the point of purchase where this criterion decides which solutions an organization can consider. Competence applies when the solution is in use where it decides whether the solution produces anything useful.

This matches the distinction that Champa and Segall (2026) draws between an access gap and an outcome gap. It also fits their observation that cost is the barrier firms report most often while being one of the smaller benefits once adoption occurs. The interviews suggest a reason for this pattern. Cost is the criterion that respondents are asked about when they buy something so it is the criterion they can name. Competence only becomes visible once the tools are in practical use. This explanation is our own interpretation from the interview material and neither Champa and Segall (2026) nor the respondents explicitly state it.

The material also adds something to Gupta and George's (2016) resource model. Their model treats human skills as one resource class covering technical and managerial ability. Respondents instead described four different things an organization can be missing, and an organization can be missing one of them while having the three others. The first is knowing how the database works. R7 said that BI analysts sometimes cannot code a join (in SQL) or figure out the primary and foreign key relationships between tables. The second is knowing the business well enough to see when a number is wrong. R9 said that a user should be able to look at a reported figure and ask whether the organization normally has that turnover and whether it normally delivers that much.

The third is practice. R9 said that an advanced system works well for someone who analyses data full time, but becomes harder for someone who only uses it once a month. The fourth is

having more than one person who knows how something works. R7 gave the example of an employee who had worked there for twenty years who is the only person who knows that a particular database exists, and described what happens when that person leaves. These accounts are the respondents' own perceptions and should not be read as evidence about any professional group in general. Together, they still show that a description such as a shortage of analytics skills is too coarse for what limits these organizations. The relevant material is presented in section 4.3.1.

R2 addressed the division of work that section 2.4 derived from the literature. When we asked what the biggest obstacle in the current reporting was, R2 identified the knowledge transfer between data engineers who build the pipelines and the platform specialists who know what the data means. Section 2.4 derived the same roles from Jukic et al. (2017) who showed that the two kinds of work depend on each other. An analyst who needs a variable that the schema does not contain cannot simply add it because the change needs to be reflected in the ETL pipeline and has to be made by whoever maintains it. R1 described the same division from the other direction and separated the requirement for receiving data from the requirements for producing results. Wixom and Watson (2001) found that team skills, which they define as covering both technical ability and the ability to work with users, were associated with implementation success. Their sample contained only operational data warehouses so it cannot show what happens when two sides fail to work together. The interviews describe cases where that happened and these are presented in section 4.3.2.

The material also shows that the transfer between two roles fails in more than one way and not always just because knowledge is missing. R3 described data that is technically correct but understood differently across different contexts, e.g. across countries and different working practices which is a failure of shared meaning rather than a technical failure. R4 described data ownership spread across several IT departments and described how things break when teams change the data without telling the data science function. This is an organizational failure rather than a failure of any specific individual. R7 described a third case where the knowledge exists but cannot be obtained. The people who hold it are already fully occupied and explaining the meaning of the data takes time that they do not have.

R7's case has a consequence for the argument in section 2.2. It indicates that the constraint is not always an absence of the human resource but its unavailability. This means that an organization can fail the capability test Gupta and George (2016) describe while technically having all three resource classes. Ain et al. (2019) argue that BI research has focused mostly on organizational and IS factors and has neglected human factors even though these keep appearing as the reason why systems fail or succeed. From this we can infer something that Ain et al. (2019) does not state themselves: that factors such as availability and coordination belong in that neglected category alongside skill. However, one respondent's account of this is a thin basis for a claim so it is more of a proposition than an actual finding.

The burden described above about separate systems did not necessarily make the respondents dissatisfied with their separated arrangements. R2 stated that reporting directly from the operational systems is already possible but not often sufficient, which is why a separate analytical solution gets purchased. R3 valued Power BI because it collects data from three operational systems that would otherwise have to be extracted one-by-one. R6 described an analytical solution built into the organization's own application and considered it to work well. These accounts do not describe organizations struggling under the burden of separation. They rather describe organizations that have found arrangement appropriate for what they do. The

interviews therefore do not settle whether the burden is heavy enough to motivate a change in architecture, and we will return to this in section 5.5.

The constraints that have been described in this section are also not primarily about query execution. If competence and coordination are what constrains then an architecture that runs faster but is harder for the staff to operate does not improve matters. Section 2.8 already noted that the benchmark measures execution speed and not whether an organization's staff could write the query in the first place. The material in this section shows why that limitation matters for the interpretation of the benchmark.

5.2 The Shape of Analytical Work Under Limited Capacity

In the previous section, it was established what factors limit and constrain analytical capacity (section 5.1). This section, on the other hand, will establish what analytical work actually looks like under those conditions and also attempt to identify what workload requirements an appropriate analytical infrastructure should support. These requirements will be interpreted against the literature findings, particularly those in section 2.3 *Analytical Practice Under Limited Capacity*, and the theoretical framework established in section 2.8. This section will, together with 5.1, help define the organizational and workload conditions against which architectural fit is later assessed in section 5.5.

In section 2.3, Kasiri et al. (2024) described how organizations across the size categories in their SME sample commonly used simple spreadsheet-like analytical tools, mainly Excel. Furthermore, Coleman et al. (2016) describes how more powerful analytical tools often require expertise. Taken together, these two accounts suggest how constrained analytical practice can favor more accessible tooling. The interview material adds a workload perspective which is less explicitly stated in the literature. For example, R2 and R3 both describe a variety of analytical demands, from quick simple lookups to multi-year analyses (section 4.1.3). R2 further emphasizes how even technically simple queries can require large amounts of historical data. Furthermore, R6 describes how the current analytical operation consists mainly of simple searches, while the underlying historical dataset to which those searches are applied will continue to grow over time. R9 also gives a perspective on the shape of analytical demand, and its importance for decision support by describing the need for longer historical horizons in order to identify slow-moving trends (section 4.1.3). An interpretation of the literature taken together with the interview material is how limited analytical capacity should not be assumed to imply limited analytical demand. The limitations concern the resources available to support analytical work, while the analytical questions that still need to be answered can remain varied in their purpose, historical scope, and scale.

Song et al. (2024) describe in section 2.4 how freshness can be reduced due to analytical queries being dependent on the most recent ETL-run, which can take minutes to hours to complete. Reducing this delay can therefore provide architectural value. The interview material further qualifies this benefit by showing how respondents do not share one universal requirement for highly current, or real-time data. For example, some respondents described a relatively high demand for current data, R1 describes how data is most often required within an hour, R2 similarly describes how daily updates are insufficient, and subsequently describes how intervals such as 15 to 30 minutes are common examples of what may constitute sufficiently current data (section 4.1.2). Other respondents instead describe a need for less current data, R3 for example describes how daily updates are most often enough, while weekly or monthly updates are required for more overarching questions (section 4.1.2). Similarly, R6

also describes requirements ranging from rapidly updated news data to information that only updates annually. An interpretation of this is how freshness is partly inherent to the analytical purpose and source. Furthermore, different requirements for freshness can also coexist within the same organization. R8 describes how data can be polled second-by-second, and conversely how some tasks run once a month and need an entire week of processing (section 4.1.2). Organizations should therefore not simply be identified as “real-time” or “batch-oriented”. Instead, the implied suggestion is that one organization can have different requirements for freshness dependent on the analytical work it supports. R9 also gives an insight into how freshness can partly be a business requirement, and how achievable freshness can also be negotiated against processing and operational constraints, as the respondent described it, updates needed to be scheduled during quieter working hours to reduce the risk of negatively impacting operational performance (section 4.1.2). The literature and interviews, taken together, suggest how architectural suitability should not be simply evaluated by the reduction in delay the infrastructure can achieve, but instead whether the provided freshness is appropriate for the decisions and the analytical activities which the infrastructure supports. The potential advantage of reducing ETL-related delays therefore becomes more important for the workloads requiring current data, compared to the workloads using daily, weekly, or monthly data.

The literature reviewed in sections 2.5-2.7 indicates that analytical performance can be influenced by both join requirements and the data that must be processed. While Kimball and Ross (2013) press on the benefits of reducing joins through dimensional modelling, Fotache et al. (2026) describes how less normalized structures can increase data volume which itself can create performance costs. The interview material adds another distinction by showing how historical scope does not necessarily correspond to technical query complexity. As previously mentioned, R2 describes how simple queries can use large amounts of historical data, and that the amount of history accessed does not necessarily determine the complexity of the query (section 4.1.3). Furthermore, respondents R3 and R9 suggest how the purpose of the analysis and the phenomena it examines also can broaden historical scope (section 4.1.3). Workload scale can also vary over time; respondents R1 and R6 both describe how the historical scope changes through the accumulation of data, while the underlying analytical operation remains the same (section 4.1.3). Another dimension of analytical workload is provided by R8, who describes a variation in processing scale and recurrence (section 4.1.3). The respondent gives an example with a comparatively large aggregation-oriented process performed monthly alongside smaller versions of the same workload being performed daily. Taken together, these accounts provide a more detailed characterization of the analytical workload described by the respondents. The workload varies in query operation, historical scope, data volume, and execution frequency. The material also indicates that these characteristics do not necessarily change together.

As previously mentioned, in section 2.3, Kasiri et al. (2024) showed how spreadsheets, more specifically Excel, were the most prevalent analytical tools among the 50 SMEs included in their study, although other analytical tools were also used. Separately from this, Coleman et al. (2016) identified a trade-off in analytical software, between the analytical capability of a tool and the specialized expertise required to operate it. A similar pattern to what is shown by Kasiri et al. (2024) is provided by the interview material. Both R7 and R8 describe the use of Excel or similar tools by certain user groups, where R7 describes it as a frequent endpoint requested by BI analysts, and R8 describes it as a working interface for case handlers and financial staff (section 4.2.1). Furthermore, respondents R5 and R9 give an input into the practical analytical function of Excel like documenting metadata, filtering and dividing data, and identifying deviations, while R9 also contrasts these uses from more advanced calculations

performed by tools such as Qlik or Power BI. Although this distinction does not directly establish the trade-off described by Coleman et al. (2016), it does show how different tools are regarded as suitable for different analytical tasks. What is shown therefore is how Excel appears to be an established working interface for certain analytical tasks and for certain user groups. Taken together, these accounts provide a user-facing dimension of analytical workload. For an analytical infrastructure to be appropriate, it needs to accommodate workflows in which analytical results are accessed through established user-facing interfaces. The user-facing aspect identified here is not measured by the benchmark, which evaluates database-level execution performance, but it remains part of the organizational context against which said performance is later assessed.

As described in section 2.3, a study conducted by Puklavec et al. (2018) found that BI functionality which was built into an existing ERP (Enterprise Resource Planning) system had a positive association with evaluation, adoption, and use. The authors explained this positive association partly by the fact that this integration of functionality resulted in reduced employee work, that employees needed to learn only one system, and a simplification of data preparation (section 2.3). An important takeaway from this is where analytical functionality is situated relative to existing systems can matter for its practical use. Note however, that this takeaway should not be seen as a general statement for integration in all contexts. A case relating to the study previously mentioned - although not identical - is given by R6 who describes a similar configuration. The similarity does not lie in the integration of BI functionality and an ERP system, instead the similarity lies in the takeaway; R6's organization also directly integrated analytical functionality into their existing system (section 4.2.1). As a contrast to the previous paragraph, the example of R6 provides an insight into how analytical work can be delivered as part of an existing application and workflow rather than through a separate recognizable BI interface. R6 also further expands this point by describing how the integration results in fewer steps needing to be taken by the users (section 4.2.3). A similar and yet distinct example from R6 is that of R7 who also describes a self-built configuration, where scripts and notebooks are built using frameworks such as Pandas - who the respondent describes being similar to Excel – and Jupyter which were sometimes executed as cron jobs, or scheduled jobs (section 4.2.1). This shows how recurring analytical work can also be supported through locally developed and relatively informal technical artifacts. What both R6 and R7 together show is how locally adopting tools, from scheduled scripts to functionality embedded directly into applications, can let organizations operationalize recurring analytical work. When specifying what analytical workload means for the benchmark, what emerges from this is a dimension of workflow-embedment. Analytical workloads are not only defined by what data they process or how frequently they run, but also how their execution and outputs are incorporated into recurring organizational workflows. The interview material therefore complements the study by Puklavec et al. by showing how analytical functionality can be incorporated into existing work through practical forms, while not establishing that one degree or form of integration is generally preferable. For the benchmarking, this also adds a workflow context for when the measured performance is later interpreted as architectural suitability.

Returning to the theoretical framework established in section 2.8 we can see a distinction between analytical need and organizational capacity. The framework also proposes how a combination of the two determines the viability of an analytical configuration. The interview material in this section has shown how analytical demand varies across organizations, freshness ranges substantially, historical horizons can vary, workloads also vary in operation, scale and recurrence, and how relatively lightweight and substantial analytical tasks can coexist. From the literature it can be shown why organizational capacity and practical burden matter. Ariyachandra and Watson (2010) found, in section 2.1, the selection of architecture to be associated

with available resources and the organization's view of its IT staff. Coleman et al. (2016) identified the trade-off between capability and specialist expertise, while Puklavec et al. (2018) showed how integration of BI functionality could save employee work through one system to learn and simplify data preparation. These studies do not collectively prove any conclusions, but once again they establish specific reasons for why organizational capacity and practical burden matter. They also show how available resources, staff capability, and the burden imposed on employees influence which analytical arrangements are adopted by organizations. The interviews give practical examples of this relationship, how analytical work is frequently routed through familiar tools, locally developed scripts, or functionality integrated into existing applications. In section 5.1, it was shown how competence, availability, and coordination were identified as practical constraints, and these patterns, along with the ones presented in this section, suggest how capacity places conditions on how analytical requirements can realistically be serviced. In this study, limited analytical capacity can be interpreted primarily as constraining the means through which analytical demand can be met, instead of defining the analytical demand itself. The analytical capacity affects which tools can be properly maintained, how much burden, organizational or technical can realistically be sustained, and how analytical functionality can be integrated into existing workflows. What emerges in this section, in relation to the benchmark, is a profile for the workload which later will be interpreted. This profile has specific demands like freshness, query operation, historical scope, data volume, and recurrence, along with a support context which includes user-facing accessibility, embedment of workflow, and limited capacity to sustain additional complexity.

In summary, the findings in this section provide a profile of the analytical work that must be supported under the capacity conditions identified in section 5.1. The first thing to be included in this profile is differing freshness requirements, ranging from highly current information to daily, monthly, or annual data. And how this depends on analytical purpose. It also includes workloads that vary in historical scope, query operation, data volume, and execution frequency; these characteristics do not necessarily change together. At the same time, the analytical workload will be taking place in certain user-facing environments, like locally developed solutions, and functionality embedded into existing workflows. These analytical requirements, when under limited analytic capacity, must be supported by arrangements which the organization can realistically operate and maintain. When relating to the theoretical framework in section 2.8 which distinguishes between analytical need and organizational capacity, where the first specifies what the infrastructure must support, and the second constraints which ways of supporting it are feasible. What has been tested in the benchmark should therefore be interpreted against the analytical requirements identified here, and if the measured performance differences are consequential under the organizational conditions described in sections 5.1 and 5.2. Subsequent sections 5.3 and 5.4 can now establish the technical performance differences between the configurations tested.

5.3 Performance in Relation to Schema Design

The benchmark analysis shows that columnstore indexing had a considerably stronger effect on performance than schema design alone. The two columnstore-indexed configurations, HTAP and OLAP-CS, outperformed OLAP-RS on most of the 22 queries. In the aggregation-intensive category, for instance, OLAP-RS required between 11.50 and 68.31 times more CPU time than either columnstore configuration for the same TPC-H workload (see Table 4.3). This indicates that columnstore indexing was a decisive factor for efficient analytical performance, even when applied to a dimensional schema. Schema design was not irrelevant,

but the results suggest that it should be interpreted together with indexing strategy rather than as the primary determinant of performance.

For organizations with limited analytical capacity, this distinction between schema design and indexing strategy is important. If meaningful analytical performance gains can be achieved through indexing within an existing operational environment, the need for a separate OLAP environment may be reduced for some workloads. For organizations without dedicated analytics staff or in-house data engineering capacity, this suggests that improved analytical capability may be achievable without immediately operating a separate data warehouse. Optimizing an existing operational database may therefore be sufficient for some of the analytical workloads examined here.

These results are consistent with the architectural logic established in the literature review. As Abadi (2008) explains, columnstore indexing allows the database to access only the columns needed by a query. Predicates can also be applied efficiently to reduce unnecessary row processing. This reduces unnecessary data access and improves both performance and efficiency. The benchmark results are consistent with this mechanism. The columnstore-indexed configurations, both normalized (HTAP) and dimensional (OLAP-CS), performed better overall than OLAP-RS. The major exception is Q18, where HTAP performed substantially slower than OLAP-CS. This shows that query structure remains relevant and can affect the performance of a particular schema design.

The value of these results for the organizations this thesis concerns where database expertise or technical resources are limited lies not only in query speed but also in processing requirements. They may therefore benefit from configurations where processing demands are lower. Columnstore indexing reduces the amount of data processed for common analytical tasks and is therefore well suited to this kind of environment. Within a smaller technical environment, several decision-support workloads may become feasible on a columnstore-indexed configuration. This fits some of what the respondents described as their limitation: R7 talked about BI analysts who could not write a join or identify primary and foreign key relationships across enterprise systems, and R9 described advanced systems becoming harder to use when the work is done monthly rather than on a full-time basis. In both of these accounts the difficulty is in retrieving and combining data correctly rather than the lack of a tool. The results suggest that columnstore indexing may allow such organizations to extract greater analytical value from systems they already operate. The Q18 result indicates, however, that this should not be treated as a guarantee for all query types.

The results also indicate that the difference between a normalized schema and a dimensional star schema mattered less than expected for several workloads. Kimball and Ross (2013) describe how normalization can lower the performance of analytical queries through join complexity. As shown in Chapter 4, this disadvantage was substantially reduced when columnstore indexing was applied. Although the HTAP schema is highly normalized into 3NF and therefore introduces join complexity, it performed very close to OLAP-CS in many queries. For example, in Q15 both configurations completed the query in 54 milliseconds. This suggests that, for several analytical workloads, columnstore indexing can reduce the performance gap between normalized and dimensional schemas.

This finding is especially relevant for organizations that have limited analytical capacity, where architectural decisions are not only a matter of technical optimization. A dedicated OLAP environment requires additional infrastructure, ETL processes, data duplication, and

ongoing maintenance. R2 named the knowledge transfer between data engineers who build the pipelines and the specialists who know what the data means as the biggest obstacle in current reporting which is a cost that does not appear in any performance measurement. If many of an organization's decision-support tasks can be served by a normalized columnstore-indexed database, the value of a dedicated OLAP configuration must be weighed against its organizational cost. The question is therefore not whether HTAP is theoretically optimal. The more relevant question is whether it is practically sufficient for the analytical tasks the organization actually performs.

The results are also supported by Fotache et al. (2026), who found that higher normal forms did not necessarily worsen the performance of analytical queries. Their study did not, however, compare normalized and dimensional schemas directly. Similarly, Bog et al. (2012) showed that normalized schemas can perform well when the database can manage the join complexity that normalization introduces. In this study, the SQL Server query optimizer and the applied columnstore indexing appear to have reduced the expected performance penalty from normalization. This is particularly evident in the filtering-intensive category, where HTAP performed similarly to, or in some cases outperformed, the dimensional configurations. Applying filters directly on compressed column data and eliminating irrelevant rows early in query execution, as described by Abadi (2008), reduces the impact of schema-level differences for certain workloads.

These results diverge from those of Zhou and Gao (2024), who found that strict adherence to 3NF resulted in response times approximately 99% slower than a denormalized design in an industrial manufacturing context. A likely explanation is that their study was conducted on a disk-based system without columnstore indexing. The present experiment, by contrast, benefits from SQL Server's columnstore implementation and query optimizer. This suggests that the suboptimal performance traditionally associated with normalized schemas may be significantly reduced when a modern system with columnstore indexing is used. This distinction is relevant for organizations evaluating whether to denormalize an operational database schema.

For organizations, this challenges the assumption that operational databases are inherently unsuitable for analytical tasks. Several respondents described extracting data from the operational systems and working on it elsewhere. R7 said that a BI analyst will eventually ask to get the data so it can be run in Excel, and R8 described case handlers and financial staff working in standard tools and being satisfied as long as the data can be retrieved and returned. Other respondents did not work this way: R6 described analytics built directly into the organization's own application. Enhancing those systems to support analytical workloads could therefore reduce the need for fully implemented data warehouses in some cases. This should not be read as a claim that normalized schemas are generally better for analytics. Rather, the results suggest that normalized operational databases may be more capable than commonly expected when supported by columnstore indexing and a query optimizer that can handle additional joins.

Schema design nonetheless remains relevant for join-intensive queries. For Q3, Q7, and Q8, OLAP-CS was substantially faster than HTAP. In Q3, the difference was several hundred milliseconds, with OLAP-CS completing the query in 132 ms compared with 392 ms for HTAP. This points to a structural advantage of the dimensional star schema when fewer table operations are required. Schema design is therefore still a relevant performance factor, even if it was not the primary one across the benchmark as a whole. The extent to which columnstore

indexing can compensate for the join complexity of a normalized schema depends on the structure of the specific query.

The question of whether a particular organization should adopt an HTAP-oriented configuration therefore depends on the organization's actual analytical workload. If the workload resembles those tested in this study, such as continuous monitoring, filtering, aggregation, supplier evaluation, inventory visibility, or promotion analysis, the results suggest that an HTAP-oriented implementation may be sufficient. A dedicated OLAP schema may still be valuable if the organization frequently performs complex analytical tasks that reconstruct relationships across customers, orders, suppliers, products, and regions. An organization should therefore evaluate whether the performance benefits provided by OLAP-CS justify the additional architectural complexity.

It is also important to clarify the meaning of "HTAP" as used in this thesis. The HTAP literature defines HTAP systems as capable of supporting transactional and analytical workloads concurrently in a single environment (Özcan et al., 2017; Song et al., 2024). The benchmark in this study evaluates analytical performance in isolation rather than under concurrent mixed workloads. The configuration tested is therefore best understood as HTAP-oriented. It is a normalized OLTP schema augmented with columnstore indexing to support analytical workloads. It is not a full HTAP system operating under concurrent transactional and analytical pressure. This distinction is important for interpreting the results and is revisited in section 5.6. Additionally, it matters because the respondents described the interference that this benchmark does not reproduce: R9 ran the update once per day early in the morning in order to not reduce the performance of the system, and R8 described monthly calculations during which almost all IT is paused.

5.4 Workload-Dependent Performance Differences

The benchmark results show that performance differences between HTAP and OLAP-CS depended strongly on the type of query tested. No single configuration was uniformly superior across the workload categories. Instead, performance depended on how each query accessed, filtered, joined, and aggregated data. This matters for organizations because analytical decision support is not one uniform activity. Different business questions place different demands on the database architecture. R3 described salespeople viewing customers on a daily basis and running quick simple searches on order intake while the respondent themself used up to three or four years of historical data to find trends, both inside the same organization. Rather than restating the categorical results from Chapter 4, this section discusses the mechanisms that appear to explain the workload-dependent pattern.

The strongest similarity between HTAP and OLAP-CS occurred in the aggregation-intensive and filtering-intensive categories. The shared mechanism in both cases is that columnstore execution can read only the relevant columns, apply predicates efficiently to compressed column data, and eliminate rows early in the query plan (Abadi, 2008; Larson et al., 2011; Jukic et al., 2017). When most of the work in a query consists of scanning and summarizing a limited number of columns, schema layout contributes less to total execution time. This helps explain why HTAP, despite its normalized structure, performed close to OLAP-CS in Q1, Q12, Q14, Q6, and Q19. R8's monthly calculation is aggregation of this kind, running over a set of parameters in full each month and in smaller segments daily.

The join-intensive category requires more caution. OLAP-CS was the fastest configuration for all six join-intensive queries, but the size of the advantage varied. In Q5 and Q10, the gap was small. This suggests that the optimizer was able to exploit columnstore execution effectively even on the normalized schema. In Q3, Q7, and Q8, the gap was larger. This suggests that the dimensional structure produced more favorable join plans for these queries. This is consistent with Kimball and Ross's (2013) argument that dimensional modelling can simplify analytical access by organizing data around fact and dimension tables. The mechanism is not that normalization is always penalized by joins. Rather, join order, foreign-key cardinality, and the optimizer's ability to use columnstore rowgroup elimination interact with schema design in ways that depend on the specific query.

The multi-layered category produced the most variable results. In several queries, including Q2, Q11, Q13, Q16, Q17, Q20, and Q22, HTAP outperformed OLAP-CS. Q4, Q15, and Q21 produced near-identical performance. This indicates that nested conditions and multi-stage logic do not, in themselves, disadvantage a normalized schema. This pattern is consistent with Bog et al. (2012) and Fotache et al. (2026), who argue that normalized schemas can perform competitively for analytical workloads when the database system handles join complexity efficiently. It also fits R2's point that simple queries and large amounts of historical data are not opposed to one another and that the amount of historical data is rarely a parameter in how complex a query needs to be. The four categories separate queries by operation type, not by difficulty.

Q18 is the principal exception. HTAP was more than twenty times slower than OLAP-CS in execution time, while using less CPU time than OLAP-RS for the same query. The combination of high elapsed time and lower CPU time relative to OLAP-RS suggests that the bottleneck is unlikely to be raw computation alone. A more plausible explanation is that the query plan chosen for the normalized schema involved costly intermediate result handling. Q18 first identifies orders whose total lineitem quantity exceeds a threshold. It then joins that result back to customer, orders, and lineitem data. On the dimensional schema, this pattern aligns naturally with the fact and dimension structure and benefits from columnstore access on the fact table. On the normalized schema, the optimizer appears to have produced a less favorable join order or used columnstore rowgroup elimination less effectively. This resulted in significantly longer execution time despite lower CPU consumption than OLAP-RS. This is consistent with Kimball and Ross's (2013) observation that dimensional schemas can simplify analytical access for queries involving extensive customer- and order-level reconstruction. Q18 therefore illustrates a type of workload where the normalized schema's structural disadvantage is not fully compensated for by columnstore indexing.

Taken together, the patterns suggest a single overall interpretation. HTAP-oriented configurations perform well when the dominant cost of a query is column scanning, filtering, and aggregation. They perform less consistently when the dominant cost lies in reconstructing relationships across multiple large tables. Whether organizations can adopt an HTAP-oriented architecture therefore depends less on whether the architecture is generally superior to OLAP-CS. It depends more on whether the organization's analytical workload tends toward the first or the second pattern.

5.5 Architectural Fit Under Limited Analytical Capacity

Sections 5.3 and 5.4 established what the three configurations do under different query types, and sections 5.1 and 5.2 established what constraints these organizations and what their analytical work looks like. This section looks at whether the performance differences measured here matter under the conditions the interviews describe. Section 2.8 states that architectural fit depends on analytical need together with organizational capacity, and this section applies that.

The tested workload matches part of what respondents described. R8 described a monthly financial calculation that aggregates a large number of parameters into one figure and is also run daily in smaller segments, and R3 and R9 described analyses reaching back three to four years or longer (section 4.1.3). Q1 performs the same kind of aggregation over the lineitem table. The aggregation-intensive category is where HTAP and OLAP-CS came closest, with a maximum gap of 6 ms (section 4.6.4). The match is partial in three ways. None of the 22 queries in section 3.3.7 is a single-record lookup, but R3 and R6 described such lookups as a regular part of the work, and R3 noted that different people perform them (section 4.1.3). Several business scenarios in section 3.3.7, such as supplier evaluation in Q15 and promotion planning in Q20, have no counterpart in the material. And R1 and R6 described a workload where the operation stays the same while the historical scope grows and data increases, which a fixed scale factor of 1 GB does not measure.

The differences between HTAP and OLAP-CS are small and neither of the two is consistently faster than the other. In 18 of the 22 queries the gap is under 100 ms, the exceptions being Q2 at 327 ms, Q3 at 260 ms, Q13 at 172 ms, and Q18 at 9,209 ms (table 4.1). HTAP was faster in Q2, Q11, Q13, Q16, Q17, Q19, Q20, and Q22, and by 1 ms in Q12 and Q14. R3 reported that most reporting runs daily and that strategic questions are answered weekly or monthly (section 4.1.2). Against reports that run once a day or once a week a few hundred milliseconds do not change what the decision that the report is used for. The delay that matters in a separated architecture is a different one. Song et al. (2024) describe the ETL process as taking minutes or hours and leaving the analytical system with delayed data, and Jukic et al. (2017) call it the most common bottleneck in data warehousing for decision support. Section 4.2.2 shows a related cost in the interviews, where respondents described inconsistency, outdated data, incompatible formats, and failed synchronization when data is moved between systems. Neither source compares these two delays; that comparison is ours. But if moving the data costs hours and the schema choice costs milliseconds, the schema choice is not what an organization would notice. R1 and R2 needing data within the hour and R8 describing a system polled every second (section 4.1.2) are requirements about how fresh the data have to be (ETL), not about how long a query takes to run.

Q18 is the one query a user would notice, at 9,651 ms for HTAP against 442 ms for OLAP-CS. Its scenario is the identification of high-volume customers, but it ranks the largest orders by total price across the whole dataset with no time restriction, while R3 and R9 described following customers and sales across several years. The query with the worst HTAP result is therefore also one of the queries with the weakest support in the interviews, though the interview guide asked about the shape of analytical questions rather than for a list of the queries an organization runs (section 3.2.3), and section 4.9.8 treats Q18 as a warning case rather than an anomaly.

The benchmark did not test what the interviewees had most to say about, which is whether analytical work slows down operational work. Jukic et al. (2017) note in section 2.6 that running

analytical queries against a rowstore can prolong response times for the transactional operations sharing it, and Özcan et al. (2017) and Song et al. (2024) define HTAP by the concurrent handling of both workloads. Analytical and transactional queries were not run concurrently here (section 3.3.8), which is why section 5.3 described the configuration as HTAP-oriented. Section 4.3.3 shows this happening in practice. R3 attributed the low level of interference to the analytical solution being separate, R6 described report generation stopping the incoming real-time data during the period when the application was being built, R8 described a monthly calculation that pauses almost all other IT activity, and R9 scheduled updates for early morning to protect system performance. R1, R5, and R7 did not describe interference as a significant problem, so the material does not settle the question. It does show that the cost Jukic et al. (2017) identify is real for some of these organizations, and that this study says nothing about whether columnstore indexing reduces it.

Respondents did not judge their arrangements on speed. Most described their setup as sufficient, and the reasons were consolidation of several operational systems (R3), fit to an established workflow and fewer steps to reach a result (R6), and the sufficiency of common tools (R5). R2 stated that reporting from operational systems is already possible but that something more is usually needed (section 4.2.3). R6's account matches what Puklavec et al. (2018) found across 181 SMEs, that a BI system delivered inside the system the organization already runs was associated with evaluation, adoption, and use, though for use the explanation only holds once counting the indirect effects that run through the two earlier stages. The explanation for the association is that the arrangement saves the employee’s work. Performance does not enter their model at all, so the comparison with response time is ours rather than theirs. The non-cost criteria named in section 4.4.2, usability, compatibility, longevity, licensing risk, and the possibility of migrating away, are of the same kind, and Ariyachandra and Watson (2010) found architecture selection predicted by the strategic view of the system, available resources, and the organization's belief in its own IT staff. None of these is a measure of query response time.

Respondents disagreed about whether operational and analytical systems should be merged, and they were mostly not talking about the database when they answered. R8 wanted the financial users' work moved out of Excel and into one platform so that analyses and transactions sit in the same place, while R2 preferred a separate analytical solution and R4 favoured keeping data close to the business activity that produces it (section 4.4.3). Section 4.4.3 also notes that respondents discussed integration at the level of applications, workflows, and reporting rather than database architecture, and that R8's use of the terms OLTP and OLAP was conceptual. These accounts show what respondents want from an arrangement, not whether an HTAP-oriented configuration would give it to them.

Section 5.1 concluded that competence rather than money was the resource that is in short supply, and one part of that applies specifically to the schema tested here. Kimball and Ross (2013) argue in section 2.5 that normalized schemas are unsuitable for business intelligence work because users struggle to understand and navigate them, which is a problem relating to the structure the HTAP configuration uses. R7 described enterprise products where the primary and foreign key relationships are difficult to determine, and described BI analysts, as R7 encountered them, who could not tell the difference between an inner join from an outer join (section 4.3.1). Gupta and George (2016) give the reason this matters for the comparison: tangible resources do not substitute for human and intangible ones, so a technical gain that raises the skill requirement does not by itself raise capability. Coleman et al. (2016) describe the same trade-off from the product side, where tools with real analytical power need a specialist and tools a non-specialist can operate accomplish less.

How much this matters depends on who writes the queries. Section 4.1.1 showed that analytical work is most often divided between roles, and where a query is written once by someone with database competence and then run through a reporting layer, the schema is only a problem for the person who writes the query, not for the people who later just runs it. Section 4.2.1 showed that the work often ends in Excel, which is what Kasiri et al. (2024) found across 50 SMEs, or in local Python scripts. Section 2.8 splits this into analytical need and organizational capacity. The analytical need these organizations described are achievable in both configurations, and what separates them is organizational capacity: whether the person who can operate the schema is available, and whether the analytical jobs are small enough to prevent disturbing the daily operations. If an organization has neither of these, i.e. no one to write the queries and analytical jobs are too disruptive to the operational systems, then the choice between HTAP and OLAP-CS stops being the relevant question. Then it becomes a question whether to move off Excel at all rather than which database configuration is faster. This is however our own conclusion, rather than something the literature or respondents said.

5.6 Limitations and Considerations

This study combines qualitative interview evidence with a controlled database benchmark, and the limitations of both methods affect the conclusions that can be drawn. The interviews provide evidence about organizational requirements, constraints, and analytical practices, while the benchmark provides controlled evidence about one technical dimension of architectural suitability. Neither method provides a complete assessment independently, and their integration should therefore be interpreted as conditional evidence about architectural fit rather than as a general prescription for analytical infrastructure.

Limitations of the interview study. The qualitative findings are based on a purposive sample of nine respondents and cannot establish how widespread the identified requirements or constraints are across organizations. This follows from the purpose of qualitative sampling in this study: respondents were selected because their roles provided relevant experience rather than to produce a statistically representative sample (Bell et al., 2019, p. 389; Oates et al., 2022, p. 195). The findings concerning competence, coordination, data movement, freshness requirements, and infrastructure priorities should therefore be interpreted as patterns identified within the material rather than estimates of their prevalence in a wider population.

The sample is also deliberately heterogeneous. It contains practitioners and consultants and includes organizational settings with low, medium, and high analytical capacity. Some consultants primarily describe experiences from client organizations, while some practitioners work in organizations with considerably greater technical capacity than the condition at the center of this thesis. This variation is useful because it allows contrasting arrangements to be compared, but it also means that the nine interviews should not be treated as nine direct observations of organizations with limited dedicated analytics capacity. The relevance of particular accounts depends on the organizational conditions from which they were drawn.

A related limitation concerns the development of the population definition. Recruitment initially began with SMEs as the intended population, but the interview material contributed to changing the criterion from organizational size to dedicated analytical capacity. This revision is documented in Section 3.2.2 and is consistent with the flexible character of qualitative research (Bell et al., 2019, p. 397). Nevertheless, because the final population definition was not

used from the beginning of recruitment, the sample was not systematically constructed around that definition. The interviews therefore provide evidence for understanding limited analytical capacity, but they cannot establish that every respondent or organizational setting represented that condition to the same degree.

The interviews also capture respondents' accounts of practice rather than direct observation of organizational practice. Respondents may interpret the same situation differently, may not have access to all relevant information about their organization, and may be affected by the interview situation itself (Oates et al., 2022, pp. 207, 301–303). Consultants additionally describe client environments retrospectively rather than providing evidence collected directly from those client organizations. Some interview material was translated into English by the authors and lightly edited for readability, which introduces an additional interpretive step even though the original recordings and transcripts were retained. These limitations are partly addressed through recorded interviews, documented coding procedures, respondent quotations, and the deliberate consideration of material that contradicted emerging interpretations, as described in Section 3.2.6.

Limitations of the benchmark experiment. The benchmark uses synthetic TPC-H data rather than data collected from actual organizational systems. This allows all three configurations to be evaluated using identical and reproducible data, but means that the experiment measures performance under the structure and distributions defined by TPC-H rather than under the data conditions of a particular organization. The results should therefore be interpreted as comparative performance evidence under standardized decision-support workloads rather than as predictions of execution times in real organizational databases.

The experiment also uses a single TPC-H scale factor of SF=1, corresponding to approximately 1 GB of generated data. This provides a fixed baseline for comparing the configurations, but the study does not establish how the observed performance relationships change as data volume increases. The scale factor should therefore not be interpreted as representative of organizations with limited analytical capacity, since the population in this thesis is defined by analytical capacity rather than organizational size or data volume. In particular, the interview findings show that relatively simple analytical questions may still operate over growing or multi-year datasets. Whether the relative performance of HTAP, OLAP-CS, and OLAP-RS remains similar at larger scales cannot be determined from this experiment.

The experiment is further limited to Microsoft SQL Server 2025 running in one controlled local hardware environment. The observed results are consequently influenced by SQL Server's query optimizer, columnstore implementation, memory management, and execution strategies. The experiment supports conclusions about the tested configurations under these conditions, but it does not establish that the same performance relationships will occur in other relational database management systems, hardware environments, or indexing implementations.

A particularly important limitation concerns the use of the term HTAP. Transactional and analytical workloads were not executed concurrently in the experiment. The benchmark isolates analytical query performance and therefore does not measure workload interference, transaction latency, or resource competition between OLTP and OLAP activity. The tested configuration is consequently described as **HTAP-oriented** rather than as a complete HTAP system in the concurrent sense described by Özcan et al. (2017) and Song et al. (2024). This limitation is particularly relevant because several interview respondents described analytical processing interfering with operational work or being scheduled specifically to avoid such

interference. The experiment cannot determine whether columnstore indexing would reduce, preserve, or increase that interference under concurrent workloads.

The benchmark also measures a restricted set of technical outcomes: execution time, CPU time, and logical reads. These measures allow controlled comparison of analytical performance and resource use, but do not measure other dimensions of architectural suitability identified through the interviews, such as usability, maintainability, licensing, governance, compatibility, required expertise, or the organizational effort involved in maintaining data pipelines. The benchmark should therefore be understood as evidence about **performance**, not as a complete technical or economic evaluation of either architectural approach.

Limitations of the integration between methods. Finally, the relationship between the interview workloads and TPC-H is only partial. The TPC-H queries were not constructed directly from the interview material. They constitute a standardized decision-support workload whose relevance is subsequently assessed against the analytical activities respondents described. As Section 5.5 shows, some correspondence exists, particularly for aggregation and analyses over historical data, but respondents also described single-record lookups that are not represented by the benchmark, while several TPC-H business scenarios have no direct counterpart in the interviews. The query categories used in Chapter 4 are therefore useful analytical groupings but should not be interpreted as a complete classification of analytical work in organizations with limited dedicated analytics capacity.

The integration also depends on interpretation by the same researchers who designed the benchmark and analyzed the interviews. This creates a risk that organizational evidence could be interpreted in ways that favor the architectural comparison being tested. The use of explicit coding rules, respondent quotations, contradictory cases, and clear separation between respondent statements and the authors' own interpretations reduces this risk but cannot remove it completely. Accordingly, the combined findings support a **conditional assessment of architectural fit** rather than a general claim that either an integrated HTAP-oriented architecture or a separated OLAP architecture is preferable. The study can identify conditions under which each alternative appears more or less suitable; determining suitability for a particular organization would still require evaluation of that organization's workloads, capabilities, existing infrastructure, and operational requirements.

6 Conclusion

6.1 Answer to the Research Questions

This thesis addressed two research questions concerning the relationship between organizational conditions and the performance of alternative analytical database architectures. The first research question examined how organizational requirements and resource constraints shape the suitability of analytical infrastructure for decision support. The second examined how benchmarked performance differences between integrated HTAP-oriented and separated OLAP architectures inform their suitability under the conditions identified in RQ1.

RQ1: How do organizational requirements and resource constraints shape the suitability of analytical infrastructure for decision support?

The findings show that the suitability of analytical infrastructure is shaped by the relationship between what an organization requires from analytics and the resources it can dedicate to supporting those requirements. Resource constraints were not expressed through one single factor. Cost influenced which analytical solutions respondents considered feasible, but the interview material indicated that competence, access to relevant expertise, coordination between technical and business roles, and the ability to understand and maintain data flows were important constraints once analytical work was being performed. In particular, the findings distinguish between possessing analytical technology and having the organizational capacity required to use and maintain it effectively.

The findings also show that limited dedicated analytical capacity should not be equated with limited analytical demand. Respondents described analytical activities ranging from simple lookups and recurring reports to aggregation over large amounts of historical data and analyses extending several years into the past. Requirements for data freshness also differed substantially. Some analytical activities required data within minutes or hours, whereas daily, weekly, or monthly updates were sufficient for others. Different freshness requirements could also coexist within the same organization. Analytical infrastructure must therefore be assessed against the particular decisions and workloads it supports rather than against a general assumption that more current data or more sophisticated infrastructure is always preferable.

Existing infrastructure and organizational work practices further shaped suitability. Respondents did not evaluate their current arrangements primarily in terms of database query speed, and separated analytical environments were not universally experienced as undesirable. Some respondents valued separate solutions because they consolidated data from several operational systems or fitted established reporting workflows. Others described more integrated arrangements as effective because they reduced steps between operational activity and analysis. The interviews therefore do not support a general preference for either integration or separation. Instead, they indicate that suitability depends on factors such as analytical requirements, data movement, competence, maintainability, compatibility with existing workflows, and the organizational capacity available to operate the infrastructure.

The answer to RQ1 is consequently that **organizational requirements define what the analytical infrastructure must provide, while resource constraints determine which ways of providing it are practically feasible**. Architectural suitability emerges from the fit between these two conditions rather than from the technical characteristics of an architecture in isolation.

RQ2: How do benchmarked performance differences between integrated HTAP-oriented and separated OLAP architectures inform their suitability under the conditions identified in RQ1?

The benchmark shows that an integrated HTAP-oriented configuration can provide analytical query performance comparable to a separated dimensional OLAP configuration for many of the tested workloads. The normalized OLTP configuration with columnstore indexing performed particularly close to the dimensional columnstore configuration for aggregation- and filtering-intensive workloads, while it also performed competitively in several multi-layered queries. Across 18 of the 22 TPC-H queries, the execution-time difference between HTAP and OLAP-CS was below 100 milliseconds. These results indicate that normalization alone does not necessarily make an operational database technically unsuitable for analytical processing when columnstore indexing is available.

The performance differences were nevertheless workload-dependent. The separated dimensional columnstore configuration was faster in all six queries categorized as join-intensive, although the magnitude of the advantage varied considerably. Q18 represented the clearest warning case, where the HTAP-oriented configuration was substantially slower than OLAP-CS. The benchmark therefore does not support treating an integrated architecture as a universal replacement for a dimensional analytical schema. Where analytical workloads depend heavily on reconstructing relationships across several tables, the dimensional structure can retain an important performance advantage.

However, RQ1 changes how these technical differences should be interpreted. For analytical activities that run daily, weekly, or monthly, a difference of tens or hundreds of milliseconds may have little practical relevance compared with other requirements such as data freshness, movement between systems, maintainability, or the availability of specialist competence. The interview material showed that data transfer between systems could involve outdated data, incompatible formats, synchronization problems, and additional organizational coordination. Consequently, where an integrated architecture provides sufficient performance for the required workload, avoiding or reducing separation may have organizational value that is not visible in benchmark execution times alone.

The opposite also applies. A dedicated OLAP environment may provide organizational value that cannot be reduced to query performance. Respondents described benefits from consolidating several operational sources, separating analytical activity from operational systems, and fitting analytical infrastructure to established workflows. Furthermore, the benchmark did not test simultaneous transactional and analytical activity, even though several respondents described workload interference as a practical concern. The experiment therefore cannot determine whether the HTAP-oriented configuration would preserve acceptable transactional performance under concurrent use.

The answer to RQ2 is therefore that the benchmarked performance differences **inform architectural suitability by establishing whether the tested alternatives are technically**

credible for particular analytical workload characteristics, but performance alone does not determine which architecture is suitable. An integrated HTAP-oriented architecture becomes a credible alternative where its analytical performance is sufficient for the required workload and where reducing separation between operational and analytical processing addresses meaningful organizational constraints. A separated OLAP architecture remains advantageous where its performance characteristics, workload isolation, data consolidation, or analytical structure justify the additional infrastructure and organizational resources required to maintain it.

Taken together, the two research questions lead to one central conclusion: **analytical infrastructure under resource constraints should be evaluated as a question of architectural fit rather than architectural superiority**. Neither integrated HTAP-oriented nor separated OLAP architectures are inherently more suitable for organizations with limited dedicated analytics capacity. Suitability depends on whether the architecture can support the required analytical workloads and freshness requirements while remaining feasible in relation to available competence, coordination, maintenance capacity, existing infrastructure, and operational demands. Technical performance is therefore an important condition of architectural suitability, but it is not the architectural decision itself.

6.2 Contributions to IS Research

This thesis contributes to Information Systems research by examining analytical infrastructure as a question of fit between organizational requirements, resource conditions, and technical system characteristics. This framing is situated within the broader Information Systems tradition of decision support, in which BI&A and DSS are treated as closely related areas concerned with supporting organizational decision-making (Phillips-Wren et al., 2021). Rather than treating database architecture as an isolated performance problem, the study combines evidence about how analytical work is organized and constrained with controlled evidence about the performance of alternative database configurations. The contribution therefore lies not only in the individual interview and benchmark findings, but in how these findings jointly explain architectural suitability.

The first contribution concerns the understanding of resource constraints in analytical work. Existing analytics-capability research conceptualizes analytical capability as dependent on combinations of tangible resources, human skills, and intangible resources (Gupta & George, 2016). The interview findings provide a more detailed view of how such constraints can appear in practice. In particular, the findings suggest a distinction between cost and competence. Cost was primarily described as a condition affecting which analytical solutions could be considered, whereas competence became more visible when respondents discussed whether analytical work could actually be performed effectively after technology was available. The findings also indicate that competence itself should not necessarily be treated as a single resource. Respondents distinguished between technical database knowledge, understanding of the business meaning of data, continued practice in analytical work, and the organizational risk created when important knowledge is concentrated in a single person. Given the qualitative and limited sample, these patterns should not be interpreted as a general model of analytical capability. They nevertheless provide a more specific account of the forms that limited analytical capacity can take and suggest dimensions that future IS research could investigate further.

The second and central contribution concerns architectural fit. The study shows that the suitability of analytical infrastructure cannot be inferred from technical performance alone. Previous IS research has established that combinations of organizational factors influence the selection of data warehouse architectures (Ariyachandra & Watson, 2010). The benchmark demonstrates that a normalized, columnstore-indexed HTAP-oriented configuration can provide performance close to a dedicated dimensional columnstore configuration for many of the tested analytical workloads, while dedicated OLAP retains advantages for some join-intensive workloads. However, the interview findings show that organizations assess analytical arrangements through additional concerns, including competence, data movement, freshness, maintainability, compatibility with existing work practices, and workload interference. The same measured performance difference can therefore have different significance under different organizational conditions.

By integrating these two forms of evidence, the thesis contributes a conditional view of analytical architecture. An integrated architecture is not suitable simply because its benchmark performance approaches that of a dedicated analytical system, and a separated architecture is not unsuitable simply because it introduces additional infrastructure and data movement. Instead, technical performance becomes one dimension of a broader assessment in which the advantages and demands of an architecture are evaluated against the analytical needs and organizational capacity of the setting in which it will operate. This extends the architectural comparison from the question of which configuration performs best to the Information Systems question of which configuration provides an appropriate fit between technology, organizational capability, and decision-support requirements.

The third contribution is methodological. The study illustrates how qualitative organizational evidence and a controlled technical benchmark can be combined when investigating infrastructure choices in Information Systems. The benchmark provides internal control and allows performance differences between configurations to be isolated, but it cannot determine whether those differences matter to an organization. The interviews provide the organizational context required for that interpretation but cannot establish the comparative technical performance of the architectures themselves. The mixed-methods design therefore allows each method to address a limitation of the other. This does not constitute a new mixed-methods methodology, but it demonstrates the usefulness of integrating technical performance evidence with practitioner accounts when system suitability depends simultaneously on technological and organizational conditions.

Taken together, these contributions support a view of analytical infrastructure in which technical capability, organizational capacity, and analytical requirements are interdependent. For Information Systems research, the implication is that architectural choices for decision support should not be evaluated solely through technical optimization or solely through organizational adoption conditions. The relationship between the two is itself part of the phenomenon being studied.

6.3 Future Research

The findings and limitations of this study suggest several directions for future research. First, the organizational patterns identified through the interviews should be examined in broader and more systematically selected samples. This study suggests that analytical capacity can be

constrained in different ways, including through technical competence, business knowledge, availability of knowledgeable personnel, coordination between roles, and dependence on individual employees. It also suggests that cost and competence may operate at different stages, with cost shaping which solutions can be considered and competence affecting whether those solutions can be used effectively. Because these findings are based on nine purposively selected respondents, future studies could examine whether the same distinctions appear across larger samples, particular industries, or organizations selected specifically according to different levels of dedicated analytical capacity.

A second direction concerns the relationship between organizational requirements and technical workloads. The TPC-H benchmark provided a standardized analytical workload, but its correspondence with the activities described in the interviews was only partial. Respondents described, for example, both simple lookups and analyses over several years of historical data, while some TPC-H business scenarios had no direct counterpart in the interview material. Future research could therefore derive benchmark workloads directly from case organizations or combine standardized benchmarks with queries collected from real analytical environments. This would allow technical performance to be assessed against organizational requirements with a closer empirical connection between the two forms of evidence.

Third, the technical comparison should be extended beyond the conditions tested in this study. The experiment used a single TPC-H scale factor of SF=1 and one Microsoft SQL Server environment, meaning that the observed performance relationships cannot be assumed to persist at larger data volumes or in other database systems. Future experiments could test larger scale factors and alternative relational database management systems to determine whether the observed workload-dependent relationship between normalized columnstore and dimensional OLAP configurations remains stable under different technical conditions.

A particularly important extension would be to evaluate concurrent transactional and analytical workloads. The present benchmark measured analytical performance in isolation and therefore cannot establish how the HTAP-oriented configuration affects transactional performance when both workloads compete for the same resources. This is especially relevant because several respondents described analytical or data-processing activities that interfered with operational systems or had to be scheduled during quieter periods. Future studies using mixed-workload benchmarks or controlled combinations of OLTP and OLAP activity could therefore examine whether the analytical performance advantages observed here can be maintained without unacceptable operational interference.

Finally, future research could study organizations before and after an architectural change. Case studies of organizations moving from separated analytical infrastructure toward a more integrated arrangement, or in the opposite direction, could examine consequences that a benchmark cannot capture directly, including maintenance effort, competence requirements, data-pipeline complexity, usability, governance, and changes in analytical work. Such longitudinal evidence would provide a stronger basis for understanding whether the architectural fit identified in this thesis translates into sustained organizational value over time.

Appendix 1: AI declaration

In this thesis, AI-based tools have been used for language refinement, structural suggestions, brainstorming, translation, and support during the planning and drafting process. AI tools have also assisted with the formulation of a PowerShell script, feedback on database schemas and query adaptation, and the summarization and organization of working material. In relation to the qualitative interview study, AI-based tools were used to support transcription.

All substantive academic decisions regarding the research questions, research design, coding and interpretation of interview data, benchmark design, argumentation, conclusions, and final written content have been made by the authors. All sources cited in the thesis have been selected, reviewed, and incorporated by the authors.

The AI-based tools used in the thesis were ChatGPT, Claude, GitHub Copilot, Gemini, and OpenAI Whisper.

Appendix 2: OLAP DDL

```
CREATE TABLE dim_customer (
   dc_customerkey   INT NOT NULL,
   dc_name          VARCHAR(25) NOT NULL,
   dc_address       VARCHAR(40) NOT NULL,
   dc_phone         CHAR(15) NOT NULL,
   dc_acctbal       DECIMAL(15,2) NOT NULL,
   dc_mktsegment    CHAR(10) NOT NULL,
   dc_comment       VARCHAR(117) NOT NULL,
   dc_nationkey     INT NOT NULL,
   dc_nation        CHAR(25) NOT NULL,
   dc_regionkey     INT NOT NULL,
   dc_region        CHAR(25) NOT NULL,
   PRIMARY KEY (dc_customerkey)
);

CREATE TABLE dim_supplier (
   ds_suppkey       INT NOT NULL,
   ds_name          CHAR(25) NOT NULL,
   ds_address       VARCHAR(40) NOT NULL,
   ds_phone         CHAR(15) NOT NULL,
   ds_acctbal       DECIMAL(15,2) NOT NULL,
   ds_comment       VARCHAR(101) NOT NULL,
   ds_nationkey     INT NOT NULL,
   ds_nation        CHAR(25) NOT NULL,
   ds_regionkey     INT NOT NULL,
   ds_region        CHAR(25) NOT NULL,
   PRIMARY KEY (ds_suppkey)
);

CREATE TABLE dim_part (
   dp_partkey       INT NOT NULL,
   dp_name          VARCHAR(55) NOT NULL,
   dp_mfgr          CHAR(25) NOT NULL,
   dp_brand         CHAR(10) NOT NULL,
   dp_type          VARCHAR(25) NOT NULL,
   dp_size          INT NOT NULL,
   dp_container     CHAR(10) NOT NULL,
   dp_retailprice   DECIMAL(15,2) NOT NULL,
   dp_comment       VARCHAR(23) NOT NULL,
   PRIMARY KEY (dp_partkey)
);

CREATE TABLE dim_orders (
   do_orderkey      INT NOT NULL,
```

```
    do_orderstatus   CHAR(1) NOT NULL,
    do_totalprice    DECIMAL(15,2) NOT NULL,
    do_orderdate     DATE NOT NULL,
    do_orderpriority CHAR(15) NOT NULL,
    do_clerk         CHAR(15) NOT NULL,
    do_shippriority  INT NOT NULL,
    do_comment       VARCHAR(79) NOT NULL,
    PRIMARY KEY (do_orderkey)
);

CREATE TABLE fact_lineitem (
    fl_orderkey      INT NOT NULL,
    fl_linenumber    INT NOT NULL,
    fl_partkey       INT NOT NULL,
    fl_suppkey       INT NOT NULL,
    fl_customerkey   INT NOT NULL,
    fl_quantity      DECIMAL(15,2) NOT NULL,
    fl_extendedprice DECIMAL(15,2) NOT NULL,
    fl_discount      DECIMAL(15,2) NOT NULL,
    fl_tax           DECIMAL(15,2) NOT NULL,
    fl_supplycost    DECIMAL(15,2) NOT NULL,
    fl_availqty      INT NOT NULL,
    fl_returnflag    CHAR(1) NOT NULL,
    fl_linestatus    CHAR(1) NOT NULL,
    fl_shipdate      DATE NOT NULL,
    fl_commitdate    DATE NOT NULL,
    fl_receiptdate   DATE NOT NULL,
    fl_shipinstruct  CHAR(25) NOT NULL,
    fl_shipmode      CHAR(10) NOT NULL,
    fl_comment       VARCHAR(44) NOT NULL,
    PRIMARY KEY (fl_orderkey, fl_linenumber),
    FOREIGN KEY (fl_orderkey) REFERENCES dim_orders(do_orderkey),
    FOREIGN KEY (fl_partkey) REFERENCES dim_part(dp_partkey),
    FOREIGN KEY (fl_suppkey) REFERENCES dim_supplier(ds_suppkey),
    FOREIGN KEY (fl_customerkey) REFERENCES dim_customer(dc_customerkey)
);
```

Appendix 3: OLAP Queries

```
-- Q1 - Pricing Summary Report
SELECT
fl_returnflag,
fl_linestatus,
SUM(fl_quantity) AS sum_qty,
SUM(fl_extendedprice) AS sum_base_price,
SUM(fl_extendedprice * (1 - fl_discount)) AS sum_disc_price,
SUM(fl_extendedprice * (1 - fl_discount) * (1 + fl_tax)) AS sum_charge,
AVG(fl_quantity) AS avg_qty,
AVG(fl_extendedprice) AS avg_price,
AVG(fl_discount) AS avg_disc,
COUNT(*) AS count_order
FROM
fact_lineitem
WHERE
fl_shipdate <= DATEADD(day, -90, '1998-12-01')
GROUP BY
fl_returnflag,
fl_linestatus
ORDER BY
fl_returnflag,
fl_linestatus;

-- Q2 - Minimum Cost Supplier Query
SELECT TOP 100
ds_acctbal,
ds_name,
ds_nation,
dp_partkey,
dp_mfgr,
ds_address,
ds_phone,
ds_comment
FROM
dim_part,
dim_supplier,
fact_lineitem
WHERE
dp_partkey = fl_partkey
AND ds_suppkey = fl_suppkey
AND dp_size = 15
```

```
AND dp_type LIKE '%BRASS'
AND ds_region = 'EUROPE'
AND fl_supplycost = (
SELECT
MIN(fl_supplycost)
FROM
fact_lineitem,
dim_supplier,
dim_part
WHERE
dp_partkey = fl_partkey
AND ds_suppkey = fl_suppkey
AND ds_region = 'EUROPE'
)
ORDER BY
ds_acctbal DESC,
ds_nation,
ds_name,
dp_partkey;

-- Q3 - Shipping Priority Query
SELECT TOP 10
fl_orderkey,
SUM(fl_extendedprice * (1 - fl_discount)) AS revenue,
do_orderdate,
do_shippriority
FROM
dim_customer,
dim_orders,
fact_lineitem
WHERE
dc_mktsegment = 'BUILDING'
AND dc_customerkey = fl_customerkey
AND fl_orderkey = do_orderkey
AND do_orderdate < '1995-03-15'
AND fl_shipdate > '1995-03-15'
GROUP BY
fl_orderkey,
do_orderdate,
do_shippriority
ORDER BY
revenue DESC,
do_orderdate;

-- Q4 - Order Priority Checking Query
SELECT
do_orderpriority,
COUNT(*) AS order_count
FROM
dim_orders
```

```
WHERE
do_orderdate >= '1993-07-01'
AND do_orderdate < DATEADD(month, 3, '1993-07-01')
AND EXISTS (
SELECT *
FROM fact_lineitem
WHERE
fl_orderkey = do_orderkey
AND fl_commitdate < fl_receiptdate
)
GROUP BY
do_orderpriority
ORDER BY
do_orderpriority;

-- Q5 - Local Supplier Volume Query
SELECT
ds_nation,
SUM(fl_extendedprice * (1 - fl_discount)) AS revenue
FROM
dim_customer,
dim_orders,
fact_lineitem,
dim_supplier
WHERE
dc_customerkey = fl_customerkey
AND fl_orderkey = do_orderkey
AND fl_suppkey = ds_suppkey
AND dc_nation = ds_nation
AND ds_region = 'ASIA'
AND do_orderdate >= '1994-01-01'
AND do_orderdate < DATEADD(year, 1, '1994-01-01')
GROUP BY
ds_nation
ORDER BY
revenue DESC;

-- Q6 - Forecasting Revenue Change Query
SELECT
SUM(fl_extendedprice * fl_discount) AS revenue
FROM
fact_lineitem
WHERE
fl_shipdate >= '1994-01-01'
AND fl_shipdate < DATEADD(year, 1, '1994-01-01')
AND fl_discount BETWEEN 0.05 AND 0.07
AND fl_quantity < 24;

-- Q7 - Volume Shipping Query
SELECT
```

```
supp_nation,
cust_nation,
l_year,
SUM(volume) AS revenue
FROM
(
SELECT
ds_nation AS supp_nation,
dc_nation AS cust_nation,
YEAR(fl_shipdate) AS l_year,
fl_extendedprice * (1 - fl_discount) AS volume
FROM
dim_supplier,
fact_lineitem,
dim_orders,
dim_customer
WHERE
ds_suppkey = fl_suppkey
AND do_orderkey = fl_orderkey
AND dc_customerkey = fl_customerkey
AND (
(ds_nation = 'FRANCE' AND dc_nation = 'GERMANY')
OR (ds_nation = 'GERMANY' AND dc_nation = 'FRANCE')
)
AND fl_shipdate BETWEEN '1995-01-01' AND '1996-12-31'
) AS shipping
GROUP BY
supp_nation,
cust_nation,
l_year
ORDER BY
supp_nation,
cust_nation,
l_year;

-- Q8 - National Market Share Query
SELECT
o_year,
SUM(CASE
WHEN nation = 'BRAZIL' THEN volume
ELSE 0
END) / SUM(volume) AS mkt_share
FROM
(
SELECT
YEAR(do_orderdate) AS o_year,
fl_extendedprice * (1 - fl_discount) AS volume,
ds_nation AS nation
FROM
dim_part,
```

```
dim_supplier,
fact_lineitem,
dim_orders,
dim_customer
WHERE
dp_partkey = fl_partkey
AND ds_suppkey = fl_suppkey
AND fl_orderkey = do_orderkey
AND fl_customerkey = dc_customerkey
AND dc_region = 'AMERICA'
AND do_orderdate BETWEEN '1995-01-01' AND '1996-12-31'
AND dp_type = 'ECONOMY ANODIZED STEEL'
) AS all_nations
GROUP BY
o_year
ORDER BY
o_year;

-- Q9 - Product Type Profit Measure Query
SELECT
nation,
o_year,
SUM(amount) AS sum_profit
FROM
(
SELECT
ds_nation AS nation,
YEAR(do_orderdate) AS o_year,
fl_extendedprice * (1 - fl_discount) - fl_supplycost * fl_quantity AS amount
FROM
dim_part,
dim_supplier,
fact_lineitem,
dim_orders
WHERE
ds_suppkey = fl_suppkey
AND dp_partkey = fl_partkey
AND do_orderkey = fl_orderkey
AND dp_name LIKE '%green%'
) AS profit
GROUP BY
nation,
o_year
ORDER BY
nation,
o_year DESC;

-- Q10 - Returned Item Reporting Query
SELECT TOP 20
dc_customerkey,
```

```
dc_name,
SUM(fl_extendedprice * (1 - fl_discount)) AS revenue,
dc_acctbal,
dc_nation,
dc_address,
dc_phone,
dc_comment
FROM
dim_customer,
dim_orders,
fact_lineitem
WHERE
dc_customerkey = fl_customerkey
AND fl_orderkey = do_orderkey
AND do_orderdate >= '1993-10-01'
AND do_orderdate < DATEADD(month, 3, '1993-10-01')
AND fl_returnflag = 'R'
GROUP BY
dc_customerkey,
dc_name,
dc_acctbal,
dc_phone,
dc_nation,
dc_address,
dc_comment
ORDER BY
revenue DESC;

-- Q11 - Important Stock Identification Query
SELECT
fl_partkey,
SUM(fl_supplycost * fl_availqty) AS value
FROM
fact_lineitem,
dim_supplier
WHERE
fl_suppkey = ds_suppkey
AND ds_nation = 'GERMANY'
GROUP BY
fl_partkey
HAVING
SUM(fl_supplycost * fl_availqty) > (
SELECT
SUM(fl_supplycost * fl_availqty) * 0.0001
FROM
fact_lineitem,
dim_supplier
WHERE
fl_suppkey = ds_suppkey
AND ds_nation = 'GERMANY'
```

```
)
ORDER BY
value DESC;

-- Q12 - Shipping Modes and Order Priority Query
SELECT
fl_shipmode,
SUM(CASE
WHEN do_orderpriority = '1-URGENT'
OR do_orderpriority = '2-HIGH'
THEN 1
ELSE 0
END) AS high_line_count,
SUM(CASE
WHEN do_orderpriority <> '1-URGENT'
AND do_orderpriority <> '2-HIGH'
THEN 1
ELSE 0
END) AS low_line_count
FROM
dim_orders,
fact_lineitem
WHERE
do_orderkey = fl_orderkey
AND fl_shipmode IN ('MAIL', 'SHIP')
AND fl_commitdate < fl_receiptdate
AND fl_shipdate < fl_commitdate
AND fl_receiptdate >= '1994-01-01'
AND fl_receiptdate < DATEADD(year, 1, '1994-01-01')
GROUP BY
fl_shipmode
ORDER BY
fl_shipmode;

-- Q13 - Customer Distribution Query
SELECT
c_count,
COUNT(*) AS custdist
FROM
(
SELECT
dc_customerkey,
COUNT(DISTINCT fl_orderkey) AS c_count
FROM
dim_customer LEFT OUTER JOIN fact_lineitem
ON dc_customerkey = fl_customerkey
AND fl_orderkey NOT IN (
SELECT do_orderkey FROM dim_orders
WHERE do_comment LIKE '%special%requests%'
)
```

```
GROUP BY
dc_customerkey
) AS c_orders
GROUP BY
c_count
ORDER BY
custdist DESC,
c_count DESC;

-- Q14 - Promotion Effect Query
SELECT
100.00 * SUM(CASE
WHEN dp_type LIKE 'PROMO%'
THEN fl_extendedprice * (1 - fl_discount)
ELSE 0
END) / SUM(fl_extendedprice * (1 - fl_discount)) AS promo_revenue
FROM
fact_lineitem,
dim_part
WHERE
fl_partkey = dp_partkey
AND fl_shipdate >= '1995-09-01'
AND fl_shipdate < DATEADD(month, 1, '1995-09-01');

-- Q15 - Top Supplier Query
WITH revenue AS (
SELECT
fl_suppkey AS supplier_no,
SUM(fl_extendedprice * (1 - fl_discount)) AS total_revenue
FROM
fact_lineitem
WHERE
fl_shipdate >= '1996-01-01'
AND fl_shipdate < DATEADD(month, 3, '1996-01-01')
GROUP BY
fl_suppkey
)
SELECT
ds_suppkey,
ds_name,
ds_address,
ds_phone,
total_revenue
FROM
dim_supplier,
revenue
WHERE
ds_suppkey = supplier_no
AND total_revenue = (
SELECT
```

```
MAX(total_revenue)
FROM
revenue
)
ORDER BY
ds_suppkey;

-- Q16 - Parts/Supplier Relationship Query
SELECT
dp_brand,
dp_type,
dp_size,
COUNT(DISTINCT fl_suppkey) AS supplier_cnt
FROM
fact_lineitem,
dim_part
WHERE
dp_partkey = fl_partkey
AND dp_brand <> 'Brand#45'
AND dp_type NOT LIKE 'MEDIUM POLISHED%'
AND dp_size IN (49, 14, 23, 45, 19, 3, 36, 9)
AND fl_suppkey NOT IN (
SELECT
ds_suppkey
FROM
dim_supplier
WHERE
ds_comment LIKE '%Customer%Complaints%'
)
GROUP BY
dp_brand,
dp_type,
dp_size
ORDER BY
supplier_cnt DESC,
dp_brand,
dp_type,
dp_size;

-- Q17 - Small Quantity Order Revenue Query
SELECT
SUM(fl_extendedprice) / 7.0 AS avg_yearly
FROM
fact_lineitem,
dim_part
WHERE
dp_partkey = fl_partkey
AND dp_brand = 'Brand#23'
AND dp_container = 'MED BOX'
AND fl_quantity < (
```

```
SELECT
0.2 * AVG(fl_quantity)
FROM
fact_lineitem
WHERE
fl_partkey = dp_partkey
);

-- Q18 - Large Volume Customer Query
SELECT TOP 100
dc_name,
dc_customerkey,
do_orderkey,
do_orderdate,
do_totalprice,
SUM(fl_quantity)
FROM
dim_customer,
dim_orders,
fact_lineitem
WHERE
do_orderkey IN (
SELECT
fl_orderkey
FROM
fact_lineitem
GROUP BY
fl_orderkey
HAVING
SUM(fl_quantity) > 300
)
AND dc_customerkey = fl_customerkey
AND do_orderkey = fl_orderkey
GROUP BY
dc_name,
dc_customerkey,
do_orderkey,
do_orderdate,
do_totalprice
ORDER BY
do_totalprice DESC,
do_orderdate;

-- Q19 - Discounted Revenue Query
SELECT
SUM(fl_extendedprice * (1 - fl_discount)) AS revenue
FROM
fact_lineitem,
dim_part
WHERE
```

```
(
dp_partkey = fl_partkey
AND dp_brand = 'Brand#12'
AND dp_container IN ('SM CASE', 'SM BOX', 'SM PACK', 'SM PKG')
AND fl_quantity >= 1 AND fl_quantity <= 11
AND dp_size BETWEEN 1 AND 5
AND fl_shipmode IN ('AIR', 'AIR REG')
AND fl_shipinstruct = 'DELIVER IN PERSON'
)
OR
(
dp_partkey = fl_partkey
AND dp_brand = 'Brand#23'
AND dp_container IN ('MED BAG', 'MED BOX', 'MED PKG', 'MED PACK')
AND fl_quantity >= 10 AND fl_quantity <= 20
AND dp_size BETWEEN 1 AND 10
AND fl_shipmode IN ('AIR', 'AIR REG')
AND fl_shipinstruct = 'DELIVER IN PERSON'
)
OR
(
dp_partkey = fl_partkey
AND dp_brand = 'Brand#34'
AND dp_container IN ('LG CASE', 'LG BOX', 'LG PACK', 'LG PKG')
AND fl_quantity >= 20 AND fl_quantity <= 30
AND dp_size BETWEEN 1 AND 15
AND fl_shipmode IN ('AIR', 'AIR REG')
AND fl_shipinstruct = 'DELIVER IN PERSON'
);

-- Q20 - Potential Part Promotion Query
SELECT
ds_name,
ds_address
FROM
dim_supplier
WHERE
ds_suppkey IN (
SELECT fl_suppkey
FROM fact_lineitem fl_outer
WHERE
fl_outer.fl_partkey IN (
SELECT dp_partkey FROM dim_part WHERE dp_name LIKE 'forest%'
)
AND fl_outer.fl_availqty > (
SELECT 0.5 * SUM(fl_quantity)
FROM fact_lineitem l2
WHERE l2.fl_partkey = fl_outer.fl_partkey
AND l2.fl_suppkey = fl_outer.fl_suppkey
AND l2.fl_shipdate >= '1994-01-01'
```

```
AND l2.fl_shipdate < DATEADD(year, 1, '1994-01-01')
)
)
AND ds_nation = 'CANADA'
ORDER BY
ds_name;

-- Q21 - Suppliers Who Kept Orders Waiting Query
SELECT TOP 100
ds_name,
COUNT(*) AS numwait
FROM
dim_supplier,
fact_lineitem l1,
dim_orders
WHERE
ds_suppkey = l1.fl_suppkey
AND do_orderkey = l1.fl_orderkey
AND do_orderstatus = 'F'
AND l1.fl_receiptdate > l1.fl_commitdate
AND EXISTS (
SELECT *
FROM fact_lineitem l2
WHERE
l2.fl_orderkey = l1.fl_orderkey
AND l2.fl_suppkey <> l1.fl_suppkey
)
AND NOT EXISTS (
SELECT *
FROM fact_lineitem l3
WHERE
l3.fl_orderkey = l1.fl_orderkey
AND l3.fl_suppkey <> l1.fl_suppkey
AND l3.fl_receiptdate > l3.fl_commitdate
)
AND ds_nation = 'SAUDI ARABIA'
GROUP BY
ds_name
ORDER BY
numwait DESC,
ds_name;

-- Q22 - Global Sales Opportunity Query
SELECT
cntrycode,
COUNT(*) AS numcust,
SUM(dc_acctbal) AS totacctbal
FROM
(
SELECT
```

```
SUBSTRING(dc_phone, 1, 2) AS cntrycode,
dc_acctbal
FROM
dim_customer
WHERE
SUBSTRING(dc_phone, 1, 2) IN
('13', '31', '23', '29', '30', '18', '17')
AND dc_acctbal > (
SELECT
AVG(dc_acctbal)
FROM
dim_customer
WHERE
dc_acctbal > 0.00
AND SUBSTRING(dc_phone, 1, 2) IN
('13', '31', '23', '29', '30', '18', '17')
)
AND NOT EXISTS (
SELECT *
FROM fact_lineitem
WHERE fl_customerkey = dc_customerkey
)
) AS custsale
GROUP BY
cntrycode
ORDER BY
cntrycode;
```

Appendix 4: PowerShell Script

```
$PSDefaultParameterValues['Out-File:Encoding'] = 'utf8'

$server = "<server-name>"
$database = "<database-name>"
$queryFolder = "<path-to-folder>"
$outputFile = "<path-to-file>"
$queries = 1..22

"" | Out-File $outputFile

foreach ($q in $queries) {
    $queryFile = "$queryFolder\$q.sql"

    if (-not (Test-Path $queryFile)) {
        "Query file not found: $queryFile" | Out-File $outputFile -Append
        continue
    }

    "========================================" | Out-File $outputFile -Append
    "QUERY $q" | Out-File $outputFile -Append
    "========================================" | Out-File $outputFile -Append

    $queryContent = Get-Content $queryFile -Raw

    $wrapper = @"
SET NOCOUNT ON;
SET STATISTICS TIME ON;
SET STATISTICS IO ON;
GO
$queryContent
GO
SET STATISTICS TIME OFF;
SET STATISTICS IO OFF;
GO
"@

    $tempFile = "$env:TEMP\tpch_q$q.sql"
    $wrapper | Out-File $tempFile -Encoding utf8

    for ($run = 1; $run -le 6; $run++) {
        "--- Run $run ---" | Out-File $outputFile -Append
```

```
        $statsFile = "$env:TEMP\tpch_stats_q$q_r$run.txt"

        $elapsed = Measure-Command {
            sqlcmd -S $server -d $database -E -C -r 1 -i $tempFile -o $statsFile
        }

        if (Test-Path $statsFile) {
            Get-Content $statsFile | Where-Object {
                $_ -match "SQL Server Execution Times" -or
                $_ -match "CPU time" -or
                $_ -match "elapsed time" -or
                $_ -match "logical reads" -or
                $_ -match "Table '"
            } | Out-File $outputFile -Append
        }

        $ms = [math]::Round($elapsed.TotalMilliseconds)
        "Wall clock time: $ms ms" | Out-File $outputFile -Append
        "" | Out-File $outputFile -Append

        "." | Write-Host -NoNewline
    }
    Write-Host " Q$q done"
}

Write-Host "Benchmark complete. Results saved to $outputFile"
```

Appendix 5: Interview guide

Background

1. What is your role, and how do you work with data or reporting?
2. How many employees does the company have, and is there dedicated IT staff?
    1. (For consultants): What did this look like at the companies you have worked with?

Decision support in practice

3. How are data and reports used to support decisions, and which decisions are most important?
4. What tools or systems do you use for reporting and analytics?
5. Are there analyses you'd like to do but currently can't?

Infrastructure and data flows

6. Where is the data used for reporting stored, and do you need to move or combine data from different systems to create reports?
7. Do you use your existing operational systems for reporting, or do you have a separate analytics solution?
    1. (If separate systems are used): What problems arise when data is moved between the systems?
8. How current does the data need to be − are daily reports sufficient, or is more real-time data needed?
9. Are analytical questions usually simple searches, or do they involve large amounts of historical data?
10. Have you noticed that reports or analyses slow down the system?

Constraints and priorities

11. What are the main obstacles with your current reporting − are they related to cost, technical complexity, lack of competence, data quality, or the maintenance burden of having separate systems?

12. Would it be valuable if reporting could be done directly from existing operational systems, if it was fast and reliable enough?

13. What is most important when choosing an analytics solution?

Closing

14. (For technical respondents): Are you familiar with terms such as OLTP, OLAP, or HTAP?

15. Is there anything important we have not covered?

16. May we follow up if we need clarification?

References

Abadi, D. J. (2008). Query Execution in Column-Oriented Database Systems. PhD thesis, Massachusetts Institute of Technology (MIT)

Abadi, D. J., Madden, S. R., & Hachem, N. (2008). Column-Stores vs. Row-Stores: How different are they really?, in Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, 9 June 2008, Vancouver Canada: ACM, pp.967–980, https://doi.org/10.1145/1376616.1376712

Abubakar, A. M., Türkmen, A., Işik, V., Mikalef, P., & Turel, O. (2025). Exploring the Complementary Effects of Business Analytics Capabilities and π-shaped Skills on Innovation Outcomes, *European Journal of Information Systems*, vol. 34, no. 1, pp.146–163, https://doi.org/10.1080/0960085X.2024.2304030

Ain, N., DeLone, W. H., & Vaia, G. (2025). Measuring the Success of Business Intelligence and Analytics Systems: A literature review, *Technovation*, vol. 146, p.103277, https://doi.org/10.1016/j.technovation.2025.103277

Ain, N., Vaia, G., DeLone, W. H., & Waheed, M. (2019). Two Decades of Research on Business Intelligence System Adoption, Utilization and Success – A systematic literature review, *Decision Support Systems*, vol. 125, p.113113, https://doi.org/10.1016/j.dss.2019.113113

Al-Daraba, K., Al-shami, S. A., Rashid, N., & Qureshi, M. I. (2025). Systematic Review of Factors Influencing Adoption of Business Intelligence Systems, *Discover Sustainability*, vol. 6, no. 1, p.936, https://doi.org/10.1007/s43621-025-01876-5

Angeles, M. D. P., & Castro, V. G. (2013). V+H: Hybrid Architecture for DSS and OLTP, *International Journal of Information Management*, vol. 33, no. 6, pp.940–947, https://doi.org/10.1016/j.ijinfomgt.2013.08.006

Ariyachandra, T., & Watson, H. (2010). Key Organizational Factors in Data Warehouse Architecture Selection, *Decision Support Systems*, vol. 49, no. 2, pp.200–212, https://doi.org/10.1016/j.dss.2010.02.006

Bell, E., Bryman, A., & Harley, B. (2019). Business Research Methods, 5th edn, Oxford: Oxford University Press

Bog, A., Sachs, K., Zeier, A., & Plattner, H. (2012). Normalization in a Mixed OLTP and OLAP Workload Scenario, in R. Nambiar & M. Poess (eds), Topics in Performance Evaluation, Measurement and Characterization, vol. 7144, [e-book] Berlin, Heidelberg: Springer Berlin Heidelberg, pp.67–82

Božič, K., & Dimovski, V. (2019). Business Intelligence and Analytics for Value Creation: The role of absorptive capacity, *International Journal of Information Management*, vol. 46, pp.93–103, https://doi.org/10.1016/j.ijinfomgt.2018.11.020

Caldeira, M. M., & Ward, J. M. (2002). Understanding the Successful Adoption and Use of IS/IT in SMEs: An explanation from Portuguese manufacturing industries, *Information Systems Journal*, vol. 12, no. 2, pp.121–152, https://doi.org/10.1046/j.1365-2575.2002.00119.x

Champa, S. S., & Segall, R. S. (2026). Integrated Data Analytics, Business Intelligence, and Machine Learning Architecture for SMEs: Framework for small and medium enterprises, *International Journal of Business Analytics*, vol. 13, no. 1, pp.1–38, https://doi.org/10.4018/IJBAN.409371

Chen, D. Q., Preston, D. S., & Swink, M. (2015). How the Use of Big Data Analytics Affects Value Creation in Supply Chain Management, *Journal of Management Information Systems*, vol. 32, no. 4, pp.4–39, https://doi.org/10.1080/07421222.2015.1138364

Chen, H., Chiang, R. H. L., & Storey, V. C. (2012). Business Intelligence and Analytics: From big data to big impact, *MIS Quarterly*, vol. 36, no. 4, pp.1165–1188, https://doi.org/10.2307/41703503

Chen, Y., & Lin, Z. (2021). Business Intelligence Capabilities and Firm Performance: A study in China, *International Journal of Information Management*, vol. 57, p.102232, https://doi.org/10.1016/j.ijinfomgt.2020.102232

Codd, E. F. (1970). A Relational Model of Data for Large Shared Data Banks, *Communications of the ACM*, vol. 13, no. 6, pp.377–387, https://doi.org/10.1145/362384.362685

Coleman, S., Göb, R., Manco, G., Pievatolo, A., Tort-Martorell, X., & Reis, M. S. (2016). How Can SMEs Benefit from Big Data? Challenges and a Path Forward, *Quality and Reliability Engineering International*, vol. 32, no. 6, pp.2151–2164, https://doi.org/10.1002/qre.2008

Côrte-Real, N., Oliveira, T., & Ruivo, P. (2017). Assessing Business Value of Big Data Analytics in European Firms, *Journal of Business Research*, vol. 70, pp.379–390, https://doi.org/10.1016/j.jbusres.2016.08.011

Cosic, R., Shanks, G., & Maynard, S. B. (2015). A Business Analytics Capability Framework, *Australasian Journal of Information Systems*, vol. 19, https://doi.org/10.3127/ajis.v19i0.1150

Dreseler, M., Boissier, M., Rabl, T., & Uflacker, M. (2020). Quantifying TPC-H Choke Points and Their Optimizations, *Proceedings of the VLDB Endowment*, vol. 13, no. 8, pp.1206–1220, https://doi.org/10.14778/3389133.3389138

Fotache, M., Cluci, M.-I., Taipalus, T., & Talaba, G. (2026). The Effects of Database Normalization on Decision Support System Performance, *Information Systems*, vol. 136, p.102636, https://doi.org/10.1016/j.is.2025.102636

Grover, V., Chiang, R. H. L., Liang, T.-P., & Zhang, D. (2018). Creating Strategic Business Value from Big Data Analytics: A research framework, *Journal of Management Information Systems*, vol. 35, no. 2, pp.388–423, https://doi.org/10.1080/07421222.2018.1451951

Gupta, M., & George, J. F. (2016). Toward the Development of a Big Data Analytics Capability, *Information & Management*, vol. 53, no. 8, pp.1049–1064, https://doi.org/10.1016/j.im.2016.07.004

Huynh, M.-T., Nippa, M., & Aichner, T. (2023). Big Data Analytics Capabilities: Patchwork or progress? A systematic review of the status quo and implications for future research, *Technological Forecasting and Social Change*, vol. 197, p.122884, https://doi.org/10.1016/j.techfore.2023.122884

Iranmanesh, M., Lim, K. H., Foroughi, B., Hong, M. C., & Ghobakhloo, M. (2023). Determinants of Intention to Adopt Big Data and Outsourcing among SMEs: Organisational and technological factors as moderators, *Management Decision*, vol. 61, no. 1, pp.201–222, https://doi.org/10.1108/MD-08-2021-1059

Jukic, N., Jukic, B., Sharma, A., Nestorov, S., & Korallus Arnold, B. (2017). Expediting Analytical Databases with Columnar Approach, *Decision Support Systems*, vol. 95, pp.61–81, https://doi.org/10.1016/j.dss.2016.12.002

Justy, T., Pellegrin-Boucher, E., Lescop, D., Granata, J., & Gupta, S. (2023). On the Edge of Big Data: Drivers and barriers to data analytics adoption in SMEs, *Technovation*, vol. 127, p.102850, https://doi.org/10.1016/j.technovation.2023.102850

Kasiri, N., Cirino, C., & Narimanian, C. (2024). The Patterns of Business Analytics Adoption in US SMEs: A qualitative approach, *Small Business Institute Journal*, vol. 20, no. 1, https://doi.org/10.53703/001c.115381

Kgakatsi, M., Galeboe, O. P., Molelekwa, K. K., & Thango, B. A. (2024). The Impact of Big Data on SME Performance: A systematic review, *Businesses*, vol. 4, no. 4, pp.632–695, https://doi.org/10.3390/businesses4040038

Kimball, R., & Ross, M. (2013). The Data Warehouse Toolkit: The definitive guide to dimensional modeling, 3rd edn, John Wiley & Sons.

Larson, P.-Å., Clinciu, C., Hanson, E. N., Oks, A., Price, S. L., Rangarajan, S., Surna, A., & Zhou, Q. (2011). SQL Server Column Store Indexes, in *Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data*, 12 June 2011, Athens Greece: ACM, pp.1177–1184

Liu, Y., Soroka, A., Han, L., Jian, J., & Tang, M. (2020). Cloud-Based Big Data Analytics for Customer Insight-Driven Design Innovation in SMEs, *International Journal of Information Management*, vol. 51, p.102034, https://doi.org/10.1016/j.ijinfomgt.2019.11.002

March, S. T., & Hevner, A. R. (2007). Integrated Decision Support Systems: A data warehousing perspective, *Decision Support Systems*, vol. 43, no. 3, pp.1031–1043, https://doi.org/10.1016/j.dss.2005.05.029

Maroufkhani, P., Wan Ismail, W. K., & Ghobakhloo, M. (2020). Big Data Analytics Adoption Model for Small and Medium Enterprises, *Journal of Science and Technology Policy Management*, vol. 11, no. 4, pp.483–513, https://doi.org/10.1108/JSTPM-02-2020-0018

Mikalef, P., Pappas, I. O., Krogstie, J., & Giannakos, M. (2018). Big Data Analytics Capabilities: A systematic literature review and research agenda, *Information Systems and e-Business Management*, vol. 16, no. 3, pp.547–578, https://doi.org/10.1007/s10257-017-0362-y

Oates, B. J., Griffiths, M., & McLean, R. (2022). Researching Information Systems and Computing, 2nd edn, Los Angeles: SAGE

Phillips-Wren, G., Daly, M., & Burstein, F. (2021). Reconciling Business Intelligence, Analytics and Decision Support Systems: More data, deeper insight, *Decision Support Systems*, vol. 146, p.113560, https://doi.org/10.1016/j.dss.2021.113560

Popovič, A., Hackney, R., Coelho, P. S., & Jaklič, J. (2012). Towards Business Intelligence Systems Success: Effects of maturity and culture on analytical decision making, *Decision Support Systems*, vol. 54, no. 1, pp.729–739, https://doi.org/10.1016/j.dss.2012.08.017

Puklavec, B., Oliveira, T., & Popovič, A. (2018). Understanding the Determinants of Business Intelligence System Adoption Stages: An empirical study of SMEs, *Industrial Management and Data Systems*, vol. 118, no. 1, pp.236–261, https://doi.org/10.1108/IMDS-05-2017-0170

Recker, J. (2021). Scientific Research in Information Systems: A beginner's guide, 2nd edn, Cham: Springer

Schneider, S., & Sunyaev, A. (2016). Determinant Factors of Cloud-Sourcing Decisions: Reflecting on the IT outsourcing literature in the era of cloud computing, *Journal of Information Technology*, vol. 31, no. 1, pp.1–31, https://doi.org/10.1057/jit.2014.25

Sharma, R., Mithas, S., & Kankanhalli, A. (2014). Transforming Decision-Making Processes: A research agenda for understanding the impact of business analytics on organisations, *European Journal of Information Systems*, vol. 23, no. 4, pp.433–441, https://doi.org/10.1057/ejis.2014.17

Song, H., Zhou, W., Cui, H., Peng, X., & Li, F. (2024). A Survey on Hybrid Transactional and Analytical Processing, *The VLDB Journal*, vol. 33, no. 5, pp.1485–1515, https://doi.org/10.1007/s00778-024-00858-9

Stjepić, A.-M., Pejić Bach, M., & Bosilj Vukšić, V. (2021). Exploring Risks in the Adoption of Business Intelligence in SMEs Using the TOE framework, *Journal of Risk and Financial Management*, vol. 14, no. 2, p.58, https://doi.org/10.3390/jrfm14020058

Transaction Processing Performance Council. (2022). TPC Benchmark™ H: (Decision Support) Standard Specification Revision 3.0.1 [PDF], https://www.tpc.org/TPC_Documents_Current_Versions/pdf/TPC-H_v3.0.1.pdf

Tsiu, S. V., Ngobeni, M., Mathabela, L., & Thango, B. (2025). Applications and Competitive Advantages of Data Mining and Business Intelligence in SMEs Performance: A systematic review, *Businesses*, vol. 5, no. 2, p.22, https://doi.org/10.3390/businesses5020022

Watson, H. J. (2002). Recent Developments in Data Warehousing, *Communications of the Association for Information Systems*, [e-journal] vol. 8, https://doi.org/10.17705/1CAIS.00801

Watson, H. J., & Wixom, B. H. (2007). The Current State of Business Intelligence, *Computer*, vol. 40, no. 9, pp.96–99, https://doi.org/10.1109/MC.2007.331

Wixom, B. H., & Watson, H. J. (2001). An Empirical Investigation of the Factors Affecting Data Warehousing Success, *MIS Quarterly*, vol. 25, no. 1, pp.17–41, https://doi.org/10.2307/3250957

Zhou, F., & Gao, Y. (2024). Performance Study on Normalization and Denormalization in MES System Databases, in *2024 International Conference on Intelligent Computing and Data Mining (ICDM),* 20 September 2024, Chaozhou, China: IEEE, pp.90–94

Özcan, F., Tian, Y., & Tözün, P. (2017). Hybrid Transactional/Analytical Processing: A survey, in *Proceedings of the 2017 ACM International Conference on Management of Data*, SIGMOD/PODS'17: International Conference on Management of Data, May 2017, Chicago Illinois USA: ACM, pp.1771–1775, https://dl.acm.org/doi/10.1145/3035918.3054784