

**\*Revised Manuscript with source files (Word document)**

[Click here to download Revised Manuscript with source files \(Word document\): document.docx](#) [Click here to view linked References](#)

# A Service-Oriented Framework for Collating Retail Intelligence

Mark Anderson , Joseph Bolton

Department of Computing, Edge Hill University, St. Helens Road, Ormskirk, Lancashire,  
L39 4QP, United Kingdom

---

## Abstract

This paper presents a case study that explores the impact of a software development project on a Small to Medium Enterprise in the United Kingdom as a means of delivering improved understanding of data in the retail sector. In this paper, the link between the actions undertaken by management in retail and the relationship with the environment provided by IT systems is considered. Many retailers in the United Kingdom make use of sensor devices to understand the behaviour of their customers. As retail outlets grow over a period of time, the diversity of sensor devices may change as new devices are installed. Equally, outlets that are operated within retail groups will collect and store data locally. As a consequence, management within the retail sector face a number of challenges to understand the operation of individual outlets and the holistic performance of retail chains. As a result, both the IT systems and also the working practices employed to complete the day to day tasks essential to meet the needs of a retailer's customers rapidly become un t for purpose. The case study considered in this paper reviews the requisite practices adopted by a service provider in the business intelligence sector, and the positive impact that the company realized through the re-engineering of both IT systems and business work ows. This paper demonstrates the e cacy of applying current software engineering methods to the redesign of IT-based business practices as opposed to more traditional approaches.

---

## 1. Introduction

The retail environment is increasingly becoming reliant on a broad range of sensors to derive retail intelligence; understanding the behaviours of customers and the impact that can be driven through store layout and management. The sensor devices that are deployed are capable of gathering data around many areas of the shopper experience. Examples of the devices may include people counters, motion detectors, and so on. These can be coupled with additional

---

Corresponding author

Preprint submitted to Elsevier

April 25, 2017

data capture devices, such as in-store cameras and point of sales (POS) systems to provide a rich data set that can inform decision-making by a retailer. The operational decisions may determine how many stations to deploy at given locations on a retail site, whereas the strategic decisions may assist in locating goods within a site to achieve high levels of sales.

A typical installation within a retail outlet consists of a number of data capture devices that are linked to an in-store PC. The PC would then have the manufacturer software for the devices installed on it, and all data recorded by the devices is uploaded to that PC. Some systems, such as in-store camera systems, rely upon bespoke software solutions to analyse data (the video stream in this case) to produce output. Often this output is stored in a non-standard format.

There are a number of issues that arise for retailers when using these monitoring systems. In the first instance, the retailer may not be able to integrate existing devices that they own prior to the purchase and installation of the in-store system. Equally, should new devices become available then these may not be readily integrated into the software system. Finally, should the supplier of the in-store system cease trading then the retailer would be left without a means of migrating their systems. However, aside from the technical implications that exist there are also management complexities. Should a retailer own multiple outlets that are geographically disparate then collecting the data to understand the performance of the outlets becomes challenging. Equally, mapping sources from multiple sensor devices to construct a multivariate data model for richer interpretation may also be a concern or, at the very least, a labour-intensive task.

This paper reports upon a research project that is seeking to design and implement a centralized framework for device data collection and reporting (Figure 1). Section 2 considers the existing approaches that are typically adopted within the retail sector to gather and analyse data, whilst section 3 expands upon this to explore the challenges that are faced when extracting data from capture devices for further analysis. Section 4 then explores the reporting processes that have been encountered and discusses the optimisation of these. The framework supports the extraction of data from a device, to be retrieved through a data collector software component. The framework is also designed to support the mining of data captured across multiple outlets for a single retailer to facilitate the interpretation of the data stored. This is supported through the correlation of multiple recorded events by the in-store sensors using web-service technology. Sections 5 and 6 discuss the implementation of the framework, and section 7 offers an evaluation of the work to date. It is noted that the limitation of this work to date is that the approach has only been applied to two retail clients; albeit one retailer with a small number of outlets dealing in high-value goods and the other with a national chain of outlets dealing in low-value goods.

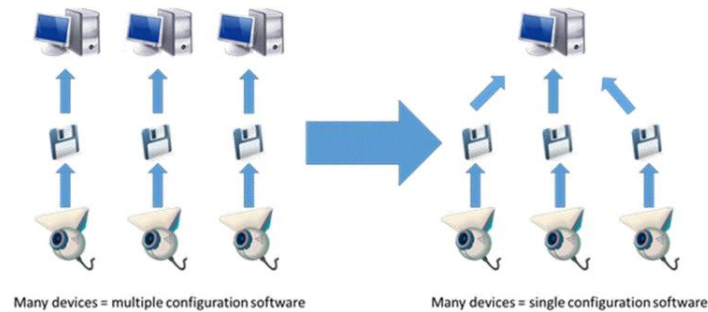


Figure 1: Distributed vs Centralised Data Storage

## 2. Existing Approaches to Retail Analysis

Within the retail sector, the collection of data to understand customer experience and behaviour is critical to the successful operation of outlets. The data is used to drive marketing campaigns, comprehend customer behaviour, determine the success of product lines, and plan ordering and stocking requirements for a store [14][17]. The data captured within a single store may extend to include customer numbers linked to time of day, unattended customers for retailers with large or high value goods (such as cars), customer movement or loitering within a store environment, demographics and facial recognition. Leveraging this rich data set can help retailers to operate in a cost-effective and efficient manner whilst customising the environment for the consumers concerned [32].

However, the issues arise as retailers own outlets in multiple geographic locations. As each outlet may record events from capture devices, for a retailer to understand the operation of each store they must collate data from the different locations and then collate this data in such a manner that it can be interpreted and compared. For some devices, such as people counters, this might be a straightforward, although time-consuming, task. The data is invariably stored as a time-date stamp index numeric value and comparison between sites is a process of collating the captured data and undertaking a comparison. However, should the data capture involve multiple streams, as indicated above, then the analysis of captured data becomes significantly more difficult.

To achieve the task of collating the data and generate reports pertaining to the operation of an outlet, a retailer would need to undertake a number of tasks (Figure 2). The challenges that face the retailers, and especially those who manage a chain of outlets, are the training requirements for store managers to be able to access captured data and filter the data according to requested date ranges through multiple software packages, and the potential for erroneous data to be collated as there is significant need for human action within this sequence of events [32]. Additional features of this process are that there may be large amounts of data to be processed [32], as retailers collate data on an hourly basis, and there may be a significant time delay between the data being captured and

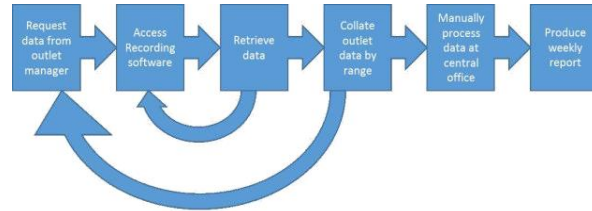


Figure 2: Typical Work ow for Retail Reporting

the data being processed. This work addresses the issues outlined by proposing an alternative means of reviewing the processes such that an effective distributed system can be designed to re-engineer the work ow.

### 3. Additional Challenges

Alongside the difficulties faced when gathering data from geographically spaced locations, there also exists the challenge of triangulating results from different sensors that may exist within a single location. This is a key step as a wide range of sensor devices are typically deployed within retail outlets to capture data for the purpose of generating retail intelligence; information to aid the understanding of customer behaviour used to improve the retail experience. The use of sensors can relate directly to sales or to security and fraud prevention [29]. The devices deployed for retail analytics can be generalised to exist within a number of key areas, these being

#### People Counting

Video Analytics (utilising demographics or trigger zones)  
Location Mapping (e.g. using WiFi hotspots)

#### Point of Sales Systems

Many of the sensors are based on camera technology, and it is the supporting software that is used to interpret video streams and generate retail intelligence. Through the triangulation of multiple streams, it is possible to build richer interpretations of the data gathered and build holistic understanding of customer behavior [11]. However, even if the hardware technology used in some sensors might be similar, there is no consistent mechanism through which the devices capture and return the data gathered. Manufacturers of the devices have developed bespoke solutions to reading the input from the sensors. These implementations have utilised file-based (using text, XML or CSV) or database implementations. In part, these implementation decisions may have been driven by the nature of data being returned from the devices. For example, a people counter may only return a timestamp and an integer representing the number of

individuals detected whereas a video analytics system will capture video clips, still images and a range of alerts that may have been configured coupled with a timestamp.

When taking these factors into account, there is significant complexity when collating data from multiple devices in order to understand emerging patterns of behaviours exhibited by consumers. As the software packages to support the sensors may generate data in widely differing formats then the collation and interpretation of those data sources becomes difficult. This complexity can increase should a retailer decide to integrate new devices into their outlets, thereby increasing the range of data formats that must be interpreted by their store managers.

One of the challenges facing the project surrounded the implementation of workflows to support the access to the toolsets being developed. Whilst the development of effective tools can support collaborative working patterns [9], in this case for management teams within retail chains, the workflow to support the end users had to be designed such that it was efficient. In doing so then the adoption of the tools would be assisted, and the new tools would not be perceived as an added layer of complexity for daily tasks.

One of the outcomes that can result from workers not being able to work within business workflows is that the team members can develop a level of mistrust, leading to misinterpretation of data within retail management teams [3]. Clearly this would represent a significant risk to relationships between the teams in the retail business.

The importance of understanding the role and contribution of one outlet towards the successful operation of an entire retail business cannot be underestimated [30]. At a more holistic level for the business, then this would pose a risk to the evolution of the business, as the broader impact of an outlet, and its employees, contribution towards the outcomes of the business could not be judged in comparison to other outlets [30].

To facilitate the management of outlets in the business, both the assets and mechanisms of management would need to be considered [33]. In this context, factors may be considered to be either virtual or tangible. For the business then virtual management might be implemented through setting targets using the software system whereas tangible management may be through facilitating training to overcome team issues. A virtual asset may be identified as the localised retail knowledge that exists within a team, whereas a tangible asset may be considered to be the collated data and reports from retail outlets. To ensure that management techniques could be fully supported, a taxonomy developed by [33] to capture the styles and assets requiring support in the new software system has been adopted.

### 3.1. Architectural Challenges

Whilst the discussion above focuses upon the challenges faced when considering the required features to support the activities of the end users, there are additional system-design challenges that the project sought to address. Specifically this centres on the architectural design for such a system. By its very

Table 1: Assets and management mechanisms [33]

	Assets	
Management	Virtual	Tangible
Virtual	Use of integrated online training tools by Contract Managers and analysts for continual personal development within the business.	Team collaboration using the system to collate data and generate effective reports for senior management and clients
Tangible	Team training sessions to understand regulatory updates or changing client requirements. Outcome of training will be implemented through daily practices that meet requirements	Team meetings to identify best practice and training to implement quality assurance work flows, resulting in published work flows that are fit for practice

nature, a retail intelligence system is highly distributed in design. At the most simple level a single store may have one or two sensor devices. In this situation, the sensors (cameras) can act as individual web servers that can be accessed to draw data from. As a distributed system, these components then display the desirable qualities of high cohesion and low coupling [7]; each device can act independently as a self-contained data server.

However, as in-store complexity increases then the relationship between the components changes. Should an outlet have a number of sensors that provide data streams to an in-store data store and report generation system for more integrated reporting then the level of cohesion falls and a level of data coupling [22] is introduced. This arises as the store data store may be used for more holistic reporting based on the outputs from sensors, but the functionality is reliant upon the sensors as data sources to deliver the data to be processed.

Clearly this is undesirable. In cases where the retailer owns a number of outlets then the situation becomes worse. Typically each individual store would still operate as described in the preceding paragraph. However, the retailer operates a separate data collation and processing function. In this situation there is low cohesion, as processing is taking place at multiple points throughout the reporting system, and content coupling [22], as the central data store is reliant upon correct function by other components (Figure 3).

It should be noted that the processes and components described here may be automated or manual. It could be that the data stores are databases that are regularly updated using code-based solutions, or are spreadsheets that are manually input. The same is also possible in larger retailers with a number of stores. Irrespective of this, the challenge existed to design a system that deliver components with high cohesion and low coupling regardless of the scale of the

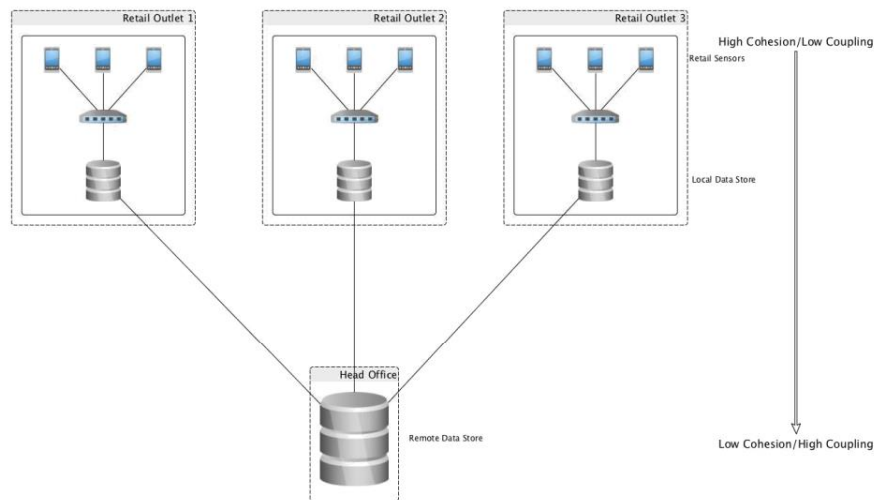


Figure 3: Changing complexity with architectural extensions

retail operation.

#### 4. Refactoring Reporting Processes

The output from the above analysis undertaken with a number of retailers in the United Kingdom demonstrated a need to redesign the work ow adopted for reporting. The aim of the rationalization of the work ow was twofold. Firstly, the process should be simplified, requiring fewer stages supported by automatic correlation of the data gathered from either individual or chains of retail outlets. Secondly, the design of the work ow should enhance efficiency and facilitate the gathering and processing of data in far shorter timeframes. The purpose of this latter aim would be to enable retailers to become more responsive to changing patterns of consumer behavior. Traditional approaches to the appraisal and re-engineering of work ows, such as Design Structure Matrix (DSM) [35][10] were evaluated. However, there existed two factors that were determined to bear a negative impact on the adoption of such techniques; the strong sequential interdependence between the tasks within the work ow, and the short timescales in which the refactoring process needed to be completed. Allied to this, the benefits of adopting a traditional work ow analysis approach were negated, as the nature of business' clients required systems that were independent of each other to protect commercially sensitive data (Figure 4).

In addressing the issues that exist within a typical reporting framework in retail as defined earlier, the decision was made to adopt a solution that would redesign the reporting work ow. Existing work ows within the retail sector often result from the retailers evolving their businesses over extended periods of time. For example, as a retailer grows, they might acquire additional premises.

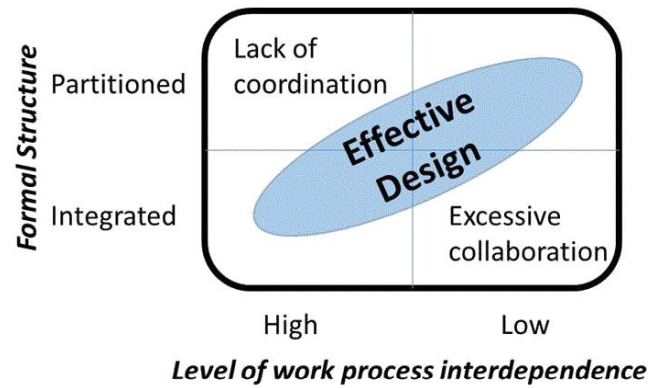


Figure 4: Relationship between work structure and process interdependence

However, the underlying practices tend to remain the same. Therefore, there exists a clear need to redesign work flows to meet evolving retailer needs, and to also link these to existing business practices to ensure that retail managers would follow the re-engineered work flows [6]. Indeed, the alignment between re-engineering the technical and organizational processes was critical in gaining a successful outcome for the project [27]. The automation of business work flows and alignment with prevailing technology, such as cloud computing as adopted in this project, is an acknowledged and beneficial process [28]. Equally, the use of knowledge management as a means of transferring knowledge within an organization has been recognized as an effective means of cognitive collaboration [23].

To better understand the impact that the framework design process would have upon retail intelligence reporting, and also to determine the potential improvements that could be achieved, the refactoring project adopted an agile approach to the project. The Agile methodology is a software development methodology [4] that has risen to become popular in the design and implementation of both intranet and Internet applications. A particular strength of this methodology is that it facilitates the decomposition of a system design and development to support an incremental approach. This leads to an approach that is inherently flexible as component design and implementation can be evaluated at regular short time periods, and is also evolutionary as system design can be adapted should the business requirements change during the period of the project.

The methodology that was chosen for the design and development of the framework, one that adopts an agile approach, facilitated an iterative approach to the project. The Agile methodology is a software development methodology that was developed in 2001 [4] and has risen to become popular in the design and implementation of both intranet and Internet applications. A particular strength of this methodology is that it facilitates the decomposition of a system



design and development to support an incremental approach. This leads to an approach that is inherently exible as component design and implementation can be evaluated at regular short time periods, and is also evolutionary as system design can be adapted should the business requirements change during the period of the project.

The bene ts to this were twofold. Firstly, the approach allowed the project development to focus on those elements of the development that became important. By using this methodology, incomplete sections are prevented from holding up other sections of the build, as they can be revisited and re ned. The methodology also aligned perfectly with the chosen design, which is modular, and service-oriented. The second bene t is that the approach is user-centric. For a system to be successfully deployed in the retail sector the approach to engage end users throughout the project was adopted; such the end users main-tained a level of ownership of the development. Whilst there is a school of thought that suggests client involvement in the full lifecycle can lead to client dissolution, a more iterative approach can be adopted when the end users are not direct clients, and therefore provides justi cation for this approach in the project [4].

The analysis and design of the reporting work ows required a process of decomposition to ensure that an e ective solution was reached. A bottom-up approach was taken with each task that was identi ed in the work ow process. However, one element that had not been incorporated into the work ows was the necessary training functionality that would enable store managers to update their skills in alignment with the retail environment reporting requirements. Therefore there existed challenges to both realign the technical and organiza-tional approaches [27] and to integrate training features to facilitate cognitive collaboration [23] as critical developments to ensure success of the project.

An Agile approach o ers useful techniques to assist with the development of software products. However, additional methods and tools can be applied to understand the requirements and progress of a project throughout its lifecycle. These are both important factors; particularly so in the case where the project is highly user-centric, as was the situation in this project. As it was the retailers who would ultimately drive the requirements of the system, there was a need to ensure that their needs were captured e ectively. To achieve this, an acceptance test driven development (ATDD) approach was adopted. In following ATDD, the so-called "three amigo" meetings could take place between a developer, a tester and a retail representative to determine business case scenarios [24]. The focus of these scenarios centred on the business work ows that the retailers had adopted. These then led to test cases being developed for the developer to design and deliver code that would pass the tests. To aid in the tracking of progress throughout the project, Kanban was adopted as a lean knowledge management method [12].

A key indicator of the success of the project, and the approach taken, has been the improved understanding of customer trends by the retailers in the trial phases of the software evaluation. This has been achieved through the visualisation of the collected data that has been realized through the interface redesign.

Table 2: Improving Retailer Understanding

Benefit	Achieved by
Presentation of data	Interface redesign to encompass data visualization
Interpretation of data	Triangulation of multiple data streams
Interactivity/exploration of data	Functionality such as drill-down

Whilst the system collects broadly the same data that was previously available to retailers, the triangulation and visualisation of this data has offered valuable insights that had not been previously detected. The data presentation layer offers additional functionality to retailers and provides a richer environment in which to explore trends. For example, the retailers can drill down into the data sets that are available, and have the option to make comparisons between their stores should they own a chain of outlets. A summary of these outputs can be found in Table 2.

## 5. Architectural Design

It is important at this stage to consider coupling and cohesion within the system. The design philosophy that was applied to the architecture of the system necessitated a highly modular approach such that individual features could be offered to each retailer dependent upon their needs. In this context, the concepts from microservices have been adopted. In particular, the feature of the services are that they are focussed on performing one task well and that they are autonomous [21]. The latter is a key feature as this would deliver low coupling and high cohesion. In relation to the manual processing of this data within the existing workflows, there are ultimately two elements being performed that were extracted through the Agile approach. The first would be to collate the data from the sensors (in a manual process this might be accessing each sensor through its provided software interface and extracting the data for the required time period). The second process would be to process this data for analysis and long term storage (in a manual process would involve transcribing the data into spreadsheets). There is an additional feature that the Agile approach also uncovered: the system needed to be asynchronous. In existing workflows there is no concept that the collation or processing of data would cause delay within the system. To this end, the system design required the implementation of a queuing mechanism to support asynchronous communication between the services [21].

In addressing these requirements, it would seem a clear option to explore a Service Oriented Architecture (SOA) approach; developing the server-side using Web Services. Indeed, as the project would be utilising Amazon S3 services, then the choice of RESTful Web Services would align with the storage architecture [1]. As an architectural choice, RESTful Web Services offer a number of benefits for the implementation of such a system:

1. A uniform interface based upon Web representations [25].
2. Support for CRUD [16].

### 3. Growing popularity as an architecture of choice [2].

However, the design of a framework has a requirement to support the integration of a number of sensors for data gathering. Therefore it is also dependent upon key features of the environment in which the sensors are deployed. Previous examples of sensor integration have considered the integration of devices on manufacturing sites, such as those projects that examined Field Device Integration (FDI). However, the potential exists in this scenario for the sensors to be deployed across a much larger geographical area as retailers may manage a number of outlets. However, each site would be connected to the Internet and would have a server installed to manage the devices and capture data locally. Implementing a framework in this manner offers flexibility for retailers. For example, some retailers will not connect in-store systems to the internet or may require protection from internet connections failing. These cases are provided for within the designed architecture as the data collector continues to store sensor output should the internet connection fail and will push the collected data on re-establishment of the link. Equally, in-store servers required to operate the sensors can remain isolated from the internet, with only the data collector maintaining an active link.

The key design decisions to be considered for the framework reside within three core features of the architecture implementation

1. Whether the data is stored locally or centrally within a cloud-based solution
2. Whether the data is processed on site or in a central cloud-based system
3. Whether the data is accessed on demand (pull) or at regular intervals (push)

Whilst the design of the framework is determined to a greater extent by these decisions, there are a number of external factors that have provided influential considerations in the design of the framework. The management and maintenance of a software system is greatly simplified should the software be installed on a central server and accessed by remote clients, rather than deploying the application on a number of clients. Indeed, the rationale for centralising the processing is even more compelling as the client sites are geographically disparate and would lead to a complex maintenance mechanism. Equally, the storage of data for the framework may be managed such that the data is stored locally on client site, or remotely. A similar rationale for centralised storage exists, as maintaining data storage at client sites would require technical understanding at each site to ensure appropriate maintenance and backup procedures are undertaken. Taking these factors into account, the design of the architecture relates directly to the shift in delivery of cloud services towards a model whereby services are aligned to a vertical industry sector [15] (see Figure 5); in this case, the retail sector. Adopting this approach ensures that the services that are delivered are relevant and directed towards the needs of the sector in question.

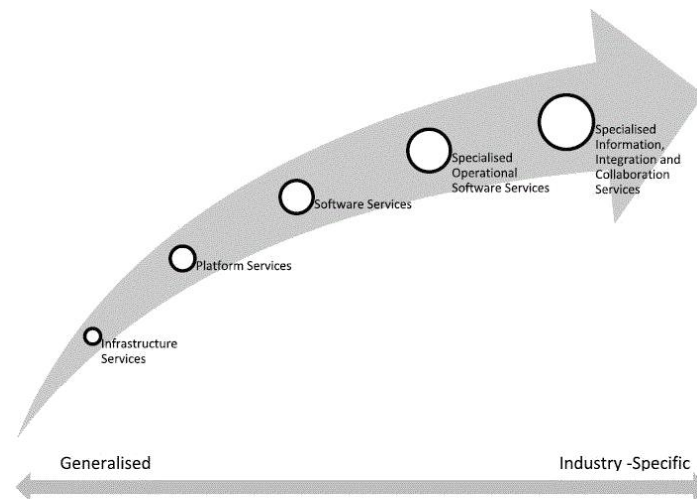


Figure 5: Evolving delivery of cloud services[15]

One component of the architecture that is more challenging to determine hosting requirements is the conversion module. As identified earlier, sensor devices generate data in a number of different, often incompatible, formats. A software module is required to process the data being captured by the devices and convert this into a format that can be synthesized with other data streams to provide a rich source of retail intelligence. This conversion may use distributed processing at the point of source following data capture, or may be performed once data has been transmitted to the central data store. An additional requirement for the system must be taken into consideration to determine an effective solution; the integration of new devices, or device types, as older devices for data collection. The nature of the sector results in a situation whereby existing users of the framework may acquire new, or updated, devices to improve retail intelligence. Equally, new users of the system may possess legacy systems that may not be supported at the point of adoption. As has been previously determined, the manufacturers of the older devices to be integrated into the framework do not conform to data format standards leading to a situation whereby new devices may require the development of bespoke connector software to process and reformat the data streams from their devices. This has led to the design of a platform that utilizes a multiple ontology approach [8]. The importance of focussing on this element of the implementation is central to developing software that facilitates the integration of data from heterogeneous data sources [20]. The architecture has therefore been designed (Figure 6) such that the conversion of data streams would be performed centrally on a cloud-based platform. This would ensure that the maintenance and deployment of connectors is simplified, and that the entire user base of the architecture may benefit should one user require the implementation of a new connector.

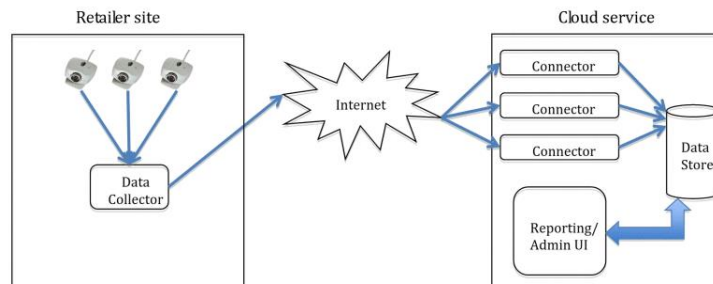


Figure 6: Architectural Model for Distributed Data Collection

## 6. Implementation Strategy

To fulfil the potential of the agile approach that had been adopted, the implementation of the software was designed to use the Model-View-Controller (MVC) pattern and deployed on an Amazon AWS instance. The rationale for this decision was that the MVC model would best support an iterative design process [13], and a cloud-based system would simplify deployment of additional or updated features. This decision aligned with the requirement to align the process remodeling with asset remodeling to embody knowledge management [14][33]. Clearly, however, the need to address the central driver of the project (the need for improved team working and co-ordination) led to a focus on the reporting framework as a core process [18].

Within the operation of the software framework, the processing can be viewed in 8 key stages (Figure 7). These are

1. A sensor driver software component extracts data from a device and mar-shals this data into a common format
2. The data collector service receives streams of data from the drivers and performs validation checks to ensure that there are no inconsistent or illegal values
3. The data collection service stores data in a local cache as a backup in case the communication link to the Amazon Web Service (AWS) instance cannot be reached
4. Data is transmitted over the internet at regular xed intervals
5. Within AWS, as data is received it is stored in an S3 bucket as a temporary data store. At the same time a message is placed in the Simple Queuing Service (SQS).
6. When a threshold of messages is reached, or a pre set timeout has lapsed, the messages and data in SQS and the S3 bucket are purged and processed. This reformats the data for the presentation layer.
7. Processed data is stored in a central database for that retailer.
8. When a request for a report page is received from an end user, the web frontend and reporting framework extract relevant data from the database and process for presentation.

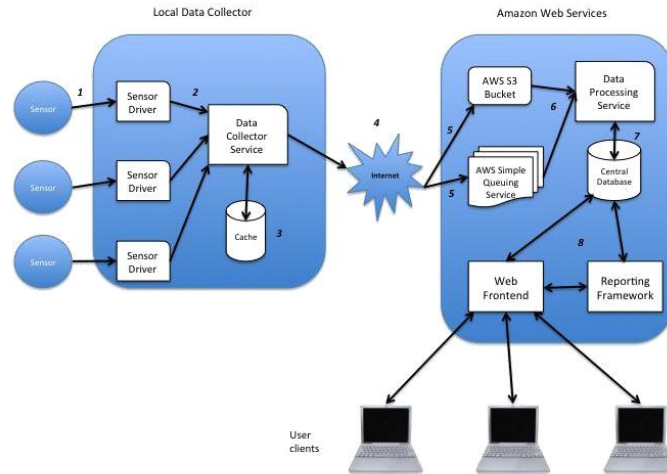


Figure 7: Detailed architectural model

Whilst the architecture that has been presented above overcomes complexities related to the inherent distributed nature of the eld devices and the processing framework, a significant concern would arise relating to multitenancy. Whilst not solely an issue related to cloud-computing implementations, multitenancy is closely associated with centralised data stores and arises when data from multiple users is stored in a single repository [34]. For Field Device Integration solutions, multitenancy would not become an issue as the frameworks are implemented to operate within a single manufacturing environment to manage and maintain sensors and actuators related to the manufacturing processes. In the retail environment, however, the sensors are deployed across multiple, potentially competing, retailers. The issue in this respect is ensuring that commercially sensitive data does not leak or become accessible to end users other than those who are authorised to access the data.

Multitenancy is often associated with cloud-based solutions because of the nature of the data stores that are often deployed. Many cloud data storage systems do not make use of relational databases, but instead is based on object storage models [19]. The mechanism for determining the ownership of the data then relies upon a unique identifier for the owner of the data being stored against each record. The rationale for this means of implementing the data stores lies with a need to implement an elastic and portable storage system; one that can grow, shrink and move to meet the needs of the end users most effectively.

Within the context of the retail environment for device integration, the relative size of the data being collected and processed is small when compared to order management systems, for example. Typically the eld devices will gener-

ate bytes, or at most kilobytes, of data related to events that may occur; these may be triggers or time-driven for example. Within this framework, data may also be replicated or stored with the data collectors at each retailer site thereby relieving or reducing the requirement for legacy data storage within the framework service. For example, video analytics software will capture short clips of video (potentially 10 or 20 seconds in length) and store these on site such that they are available on demand.

To address concerns related to multitenancy, and inherent security concerns regarding commercial sensitivity of data, the framework utilises a Platform as a Service (PaaS) model in which each end user will be allocated a separate virtual machine (VM) [26]. The hardware requirements for the individual VMs is low compared to deployments for other systems, such as e-commerce systems, and the small footprint of each VM results in an agile, yet secure, system model. The VM will host the connector software, a data store and the core components required to implement the reporting and administrative functions required to perform initial processing within the framework. Additional components may be deployed, dependent upon the requirements of the retailer.

## 7. Evaluation

The refactoring project was a twenty-four month process. To achieve a successful and effective outcome this required consultation with retailers from all market sectors. In relation to the methodological approach, the adoption of an Agile methodology complemented the need to place the end users at the centre of the re-engineering tasks. Indeed, this is a fundamental requirement of the ATDD method that was adopted. There have been distinct advantages in adopting this approach over more traditional methods for evaluating work flows, not least the benefit of gaining a clear understanding from the end users of their needs. Alternative approaches, such as DSM considered in this project, suffer from issues pertaining to data collection; namely the challenge to collate relevant data in a timely manner [5]. The use of ATDD for this purpose overcame the issue by including the retailers at all stages of the refactoring and development process. It also facilitated a deeper exploration of individual features at each stage of the workflow as the process decomposed functionality into components.

The implementation process could then build upon the designs developed as an outcome of the refactoring by adopting the concepts supporting microservice-based architectures. As the functionality was decomposed by the analysis phases, the use of a microservice approach facilitated the development of service-based components that demonstrated high cohesion and low coupling. This was an architectural feature that was identified early in the project to enhance scalability in the software product.

The architectural choice to move away from a web service-based model was key to the successful delivery of the project. Central to this decision was a need to design out two of the challenges that are faced when implementing web services: performance and reliability [31]. Had a web service implementation been chosen then performance would become an issue due to the potential for

large and heterogeneous data types that would require collation. As identified previously, the sensors deployed in retail scenarios produce their output through varied mechanisms as there are no industry-led standards for data output from such devices. This data may also be text-based or may be in a binary format. This would then lead to two potential areas for performance issues: a large, multi-faceted set of data to be processed at regular intervals alongside increased complexity that can arise from semantic issues whereby a lack of industry standards may require services to be crafted to interpret output from individual sensors that may report the same features in different ways [31]. Equally, the system would be wholly reliant upon a reliable internet connection between the retail outlets and the central hosted server. This also proved to be a technical issue as some retailers do not permit external internet connections using broadband due to security concerns; the only solution in this situation being the installation of a 3G connection specifically for the transmission of data to the server. The designed solution alleviates these issues. The processing of the data is situated within each store, thereby distributing the computation required to marshal the data into a standard format for the server. Equally, resilience has been designed into the solution. The adoption of a queuing solution allows the system to process data as it arrives at regular intervals. Should a data collector not be able to transmit to the central queue, it will store the data until it can next make a connection.

Throughout the project, technical difficulties were faced in relation to the installation of capture devices, and also to the establishment of internet connections that would enable the data collectors to transmit data through the connectors to the data store. Additional challenges related to the demarshalling of data streams generated by the sensors and pushed to the manufacturer's software. Whilst some software utilizes text-based file formats (such as XML or CSV) to store events, and others use small SQL databases, there are examples that use bespoke binary file formats. Extracting data from such formats was a lengthy process, particularly when encountering time zone differences for manufacturers in East Asia. The Agile approach assisted with addressing the concerns of the end users, and also allowed the development cycle to continue should manufacturers be slow to respond to queries. The incremental approach allowed users to evaluate reports as they were generated and thereby input into the process and perceive the benefits that could be achieved [6].

A major element of achieving the improvements in reporting workflow has been delivered through the redesign of the practices employed when collating the data required. The original workflow did not support retailers to operate in an agile manner to respond to their customers, and required users to manipulate data from a number of different sources and software systems to access and process the data required for report generation. The re-engineering of the workflow has offered the requisite data at-a-glance (Figure 8).

In order to determine the success of the re-designed workflows and the supporting software framework, the software has been deployed to a number of small to medium sized retailers who operate multiple outlets but within a fixed geographical region. As the retailers adopted the software, it became apparent



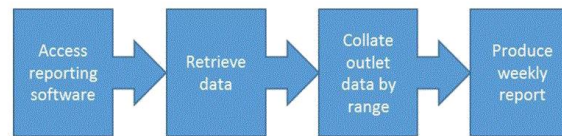


Figure 8: Redesigned Work ow for Retail Reporting

that they were benefiting not only from the timeliness of the data that they could access across all the stores they owned, but also the richness of understanding and consistency of data that they had access to. This has resulted in the test group of retailers making use of increasingly fine-grained levels of data to understand the daily operation of their outlets broken down by hour, rather than a weekly view broken down by day. This is particularly true when the retailers triangulate the data.

In optimising access to the data and visualising results in such a way that the data can be interpreted more easily, one of the retailers reported that a significant time saving had been achieved for their regional managers. Prior to the implementation of the system, regional managers would receive the weekly footfall figures as spreadsheets with over 300 tabs representing each store. Correlating the data would take managers at least an hour to perform. The implementation of the new reporting framework has facilitated a far more streamlined workflow process in which the managers can now access the same data in a matter of seconds. Furthermore another retailer, selling high value goods, reported that the implementation of the system, and the understanding of their customer base, had assisted them in driving new initiatives to engage with a broader sector of customers. Prior to the implementation the customer base consisted of a demographic group that predominantly were older adults (aged 60+). By using the reports generated through the system, the retailer was able to implement marketing initiatives over a 12 month period to successfully engage younger adults (aged 30+). The effectiveness of the campaigns was determined through the timely data delivery through the new reporting framework, and the possibility for correlation of multiple data streams. Retailers have now requested new functionality to enhance the real time review of outlet performance.

## 8. Conclusion

Any business that operates within the retail sector faces numerous challenges to not only operate successfully within the trading environment, but to maintain a competitive advantage within their businesses. The importance of building an enhanced retail experience for consumers has taken on greater significance with the movement towards online shopping as a preferred retail choice amongst consumers. The business that has been the focus of this case study is a Small to Medium Enterprise (SME) in the United Kingdom offering retail intelligence services. The business also operates within retail security. This places the

business in an ideal situation to assist retailers leverage maximum output from investment in sensor-based devices.

As the client base of the business increased, an opportunity to implement an integrated software platform was identified. The clients typically owned chains of retail outlets and, through a pilot project operated by the business, indicated the need for retailers to have access to, and manage, the data being recorded from multiple sensor devices located in-store. The pilot project also demonstrated the power of triangulating results from multiple sensors, with a visualization of the data sets being overlaid to facilitate deeper understanding of underlying trends.

To achieve a successful outcome to an essential refactoring project, the team employed an agile software methodology. The rationale to utilize a methodology such as this drew upon its central features of being dynamic, incremental and extensible. The strength of the approach in the context of this project aligned with changing client demands. Throughout the course of the project, all stakeholders were engaged with the development cycles, providing feedback and input to the subsequent development phases. The resultant software product is fit for purpose in supporting retailers to access and interpret data collected by in-store sensors. To date, the service-oriented software has been deployed in two retail chains and the retailers involved are indicating that significant improvement has been delivered to support their weekly reporting tasks. This has led to a step change in the reporting process that the retailers use in each of their outlets and has led to an increased demand for richer reporting tools to the business.

## 9. Acknowledgements

Our thanks to Innovate UK for their assistance and support in undertaking this project.

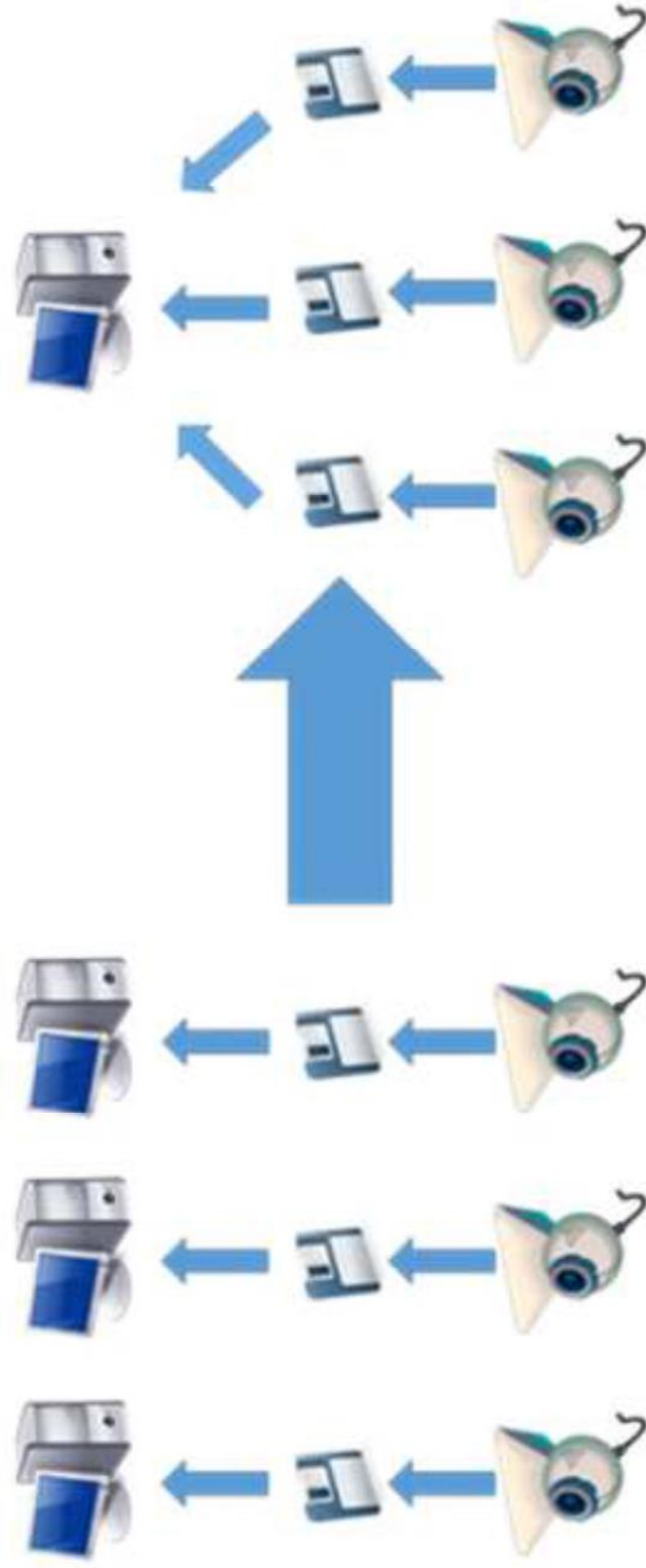
## 10. Bibliography

- [1] Amazon simple storage service api reference. Technical report, Amazon, 2011.
- [2] P. Adamczyk, P. H. Smith, R. E. Johnson, e. E. Ha z, Munawar", and C. Pautasso. REST and Web Services: In Theory and in Practice, pages 35{57. Springer New York, New York, NY, 2011.
- [3] A. Ali and F. Kohun. Dealing with some isolation to minimize doctoral attrition - a four stage framework. International Journal of Doctoral Studies, 2:33{49, 2007.
- [4] T. A. Alliance. History: The agile manifesto. Retrieved Oc-tober 2014 from Manifesto for Agile Software Development: <http://agilemanifesto.org/history.html>.

- [5] T. Browning. Applying the design structure matrix to system decomposition and integration problems: A review and new directions. *IEEE Transactions on Engineering Management*, 48(3):292{306, 2001.
- [6] P. Chan and C.Land. Implementing reengineering using information technology. *Business Process Management Journal*, 5(4):311{324, 1999.
- [7] G. Coulouris and J. Dollimore. *Distributed Systems*. Pearson Education, 2013.
- [8] I. F. Cruz and H. Xiao. The role of ontologies in data integration. *JOURNAL OF ENGINEERING INTELLIGENT SYSTEMS*, 13:245{252, 2005.
- [9] D. Duarte and N. Snyder. *Mastering Virtual Teams: strategies, tools and techniques that succeed* (2nd edition). Jossey-Bass, San Francisco, CA.
- [10] S. Eppinger. Innovation at the speed of information. *Harvard Business Review*, pages 149{158, 2001.
- [11] M. Friendly. Milestones in the history of data visualization: A case study in statistical historiography. Available: <http://www.datavis.ca/papers/gfkl.pdf>. Last accessed 07/04/2015., 2005.
- [12] M. Hammarberg and J. Sunden. *Kanban in Action*. Addison-Wesley, 2014.
- [13] D. Haughey. Waterfall v agile: How should i approach my software development project? Retrieved July 31, 2014 from Project Smart: <http://www.projectsmart.co.uk/waterfall-v-agile-how-should-i-approach-my-software-development-project.html>, 2010.
- [14] Y. P. I. Kang and Y. Ki. A framework for designing a work ow based knowledge map. *Business Process Management Journal*.
- [15] A. James and J.-Y. Chung. Business and industry specific cloud: Challenges and opportunities. *Future Generation Computer Systems*, 48:39{45, 2015.
- [16] H. Kilov. From semantic to object-oriented data modeling. In *Proceedings of the First International Conference on Systems Integration.*, pages 385{ 393, 1990.
- [17] N. Llewellyn and C. Armistead. Business process management: exploring social capital within processes. *International Journal of Service Industry Management*.
- [18] S. M. M.A. Balzarova, C.J. Bamber and J. Sharp. Key success factors in implementation of process based management: a uk housing association experience. *Business Process Management*, 10(4):387{399, 2004.

- [19] G. G. Mesnier, M. and E. Riedel. Object-based storage. *IEEE Communications Magazine*, page 8490, 2003.
- [20] C.-W. I. Mwilu, O.S. and N. Prat. Design science research contribution to business intelligence in the cloud a systematic literature review. *Future Generation Computer Systems*, 63:108{122, 2016.
- [21] S. Newman. *Building Microservices*. O'Reilly Media, Sebastopol, CA, 2015.
- [22] M. Page-Jones. *The Practical Guide to Structured Systems Design*. Prentice Hall, 1988.
- [23] J. Plass and M. Salisbury. A living system approach to the development of knowledge management systems. *Educational Technology Research and Development*, 50(1):35{57, 2002.
- [24] K. Pugh. *Lean-Agile Acceptance Test-Driven Development: Better Software Through Collaboration*. Addison-Wesley, 2011.
- [25] L. Richardson and S. Ruby. *RESTful Web Services*. O'Reilly Media, Sebastopol, CA, 2007.
- [26] C.-E. Rimal, B.P. and I. Lumb. A taxonomy and survey of cloud computing systems. In *NCM 09: Proceedings of the 2009 Fifth International Joint Conference on INC, IMS and IDC.*, pages 44{51, 2009.
- [27] M. Salisbury. Putting theory into practice to build knowledge management systems. *Journal of Knowledge Management*, 7(2):128{141, 2003.
- [28] J.-C. V. S. W. I. Schulte, S. and P. Hoenisch. Elastic business process management: State of the art and open challenges for bpm in the cloud. *Future Generation Computer Systems*, 46:36{50, 2015.
- [29] e. a. Senior, A.W. Video analytics for retail. In *Proceedings of the IEEE International Conference on Advanced Video and Signal Based Surveillance*, 2007.
- [30] H. Tajfel and J. Turner. The social identity theory of inter-group behaviour. In S. Worchel and W. Austin, editors, *Psychology of inter-group relations* (2nd edition), pages 7{24. Nelson-Hall, Chicago, 1986.
- [31] P. Vita. Challenges in the adoption and di usion of web services in nancial institutions. Technical report, Massachusetts Institute of Technology, 2004.
- [32] C. Ware. *Information Visualization: Perception for Design*. Morgan Kaufmann, San Francisco, 2004.
- [33] M. Warner and M. Witzel. *Managing in virtual organisations*. International Thomson Business Press, London, UK.

- [34] K. Wood and M. Anderson. Understanding the complexity surrounding multitenancy in cloud computing. In Proceedings of the Eighth IEEE Inter-national Conference on E-Business Engineering (ICEBE), pages 119{124, 2011.
- [35] N. Worren. Organisation Design: Re-de ning complex systems. Pearson, Harlow: England.

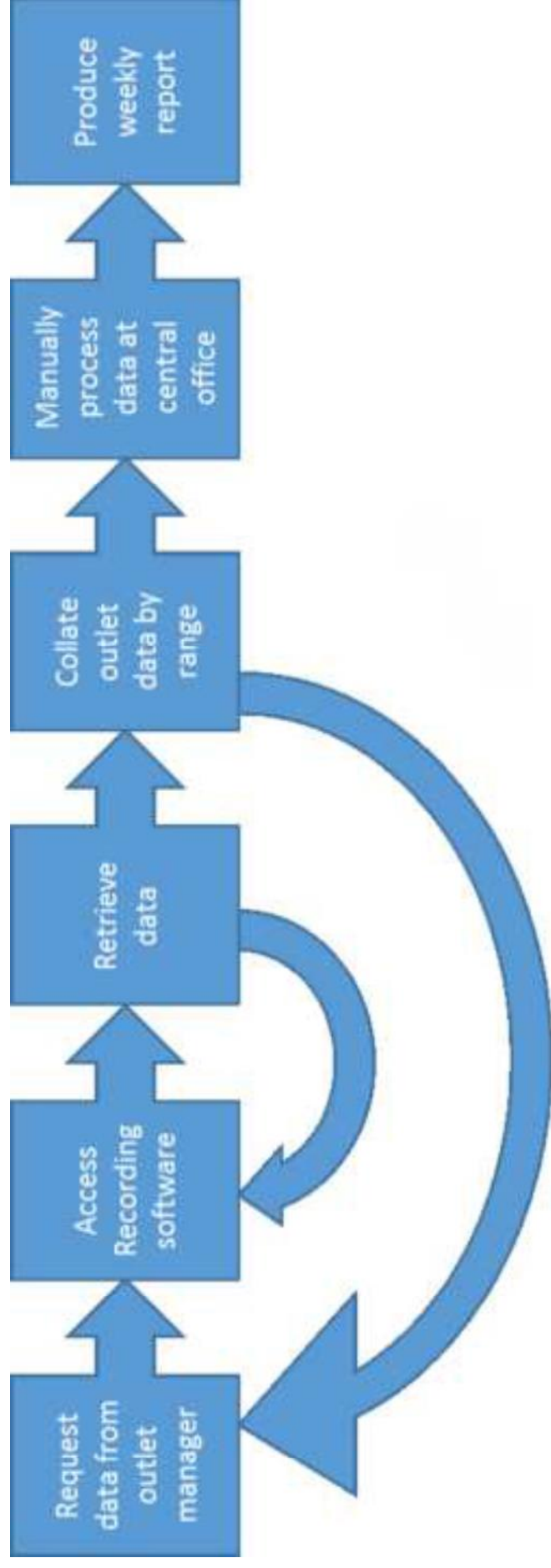


Many devices = multiple configuration software

Many devices = single configuration software

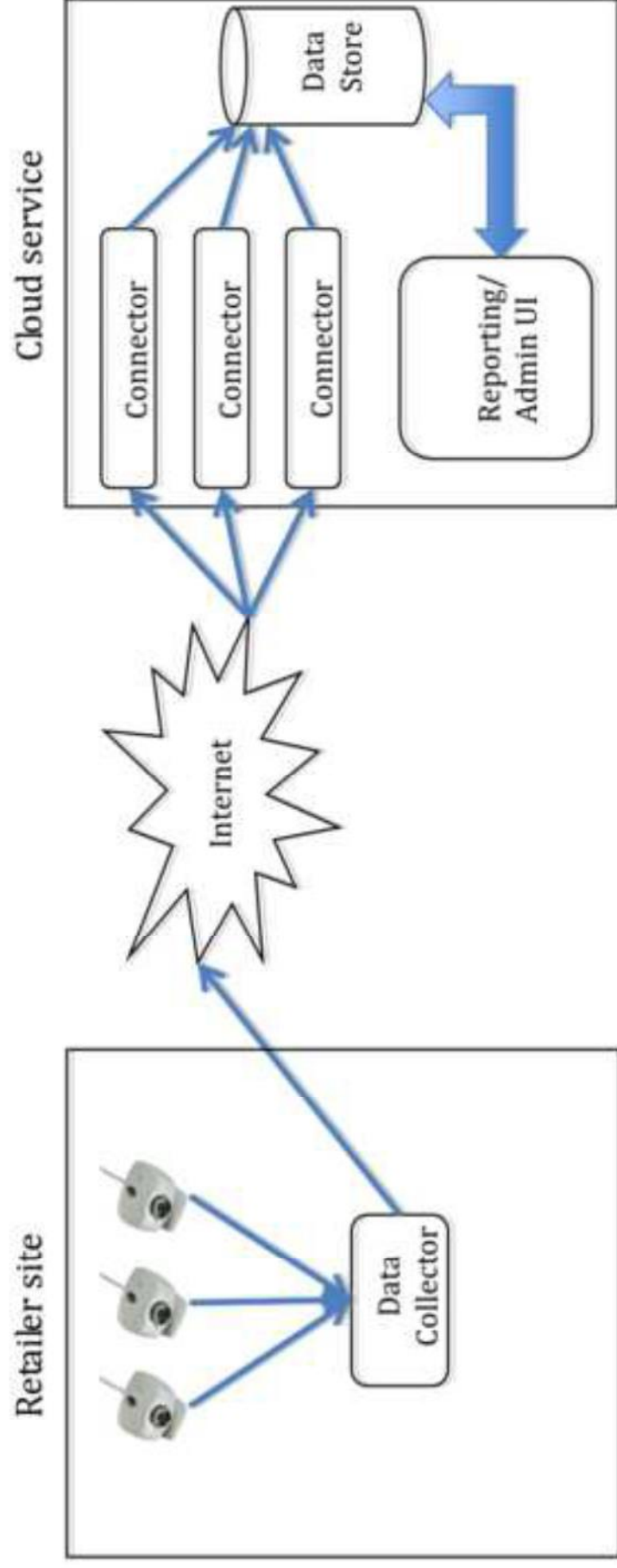
Figure

[Click here to download high resolution image](#)

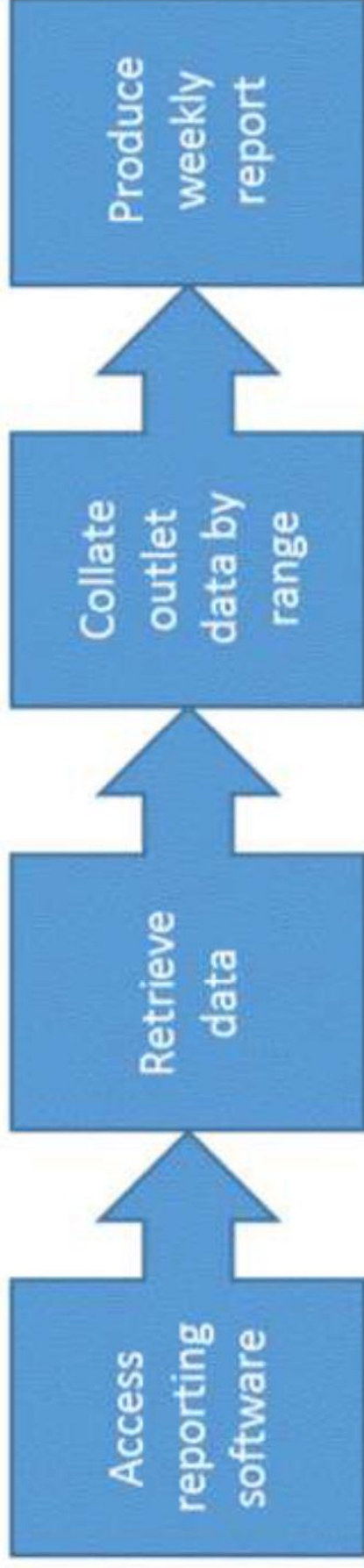


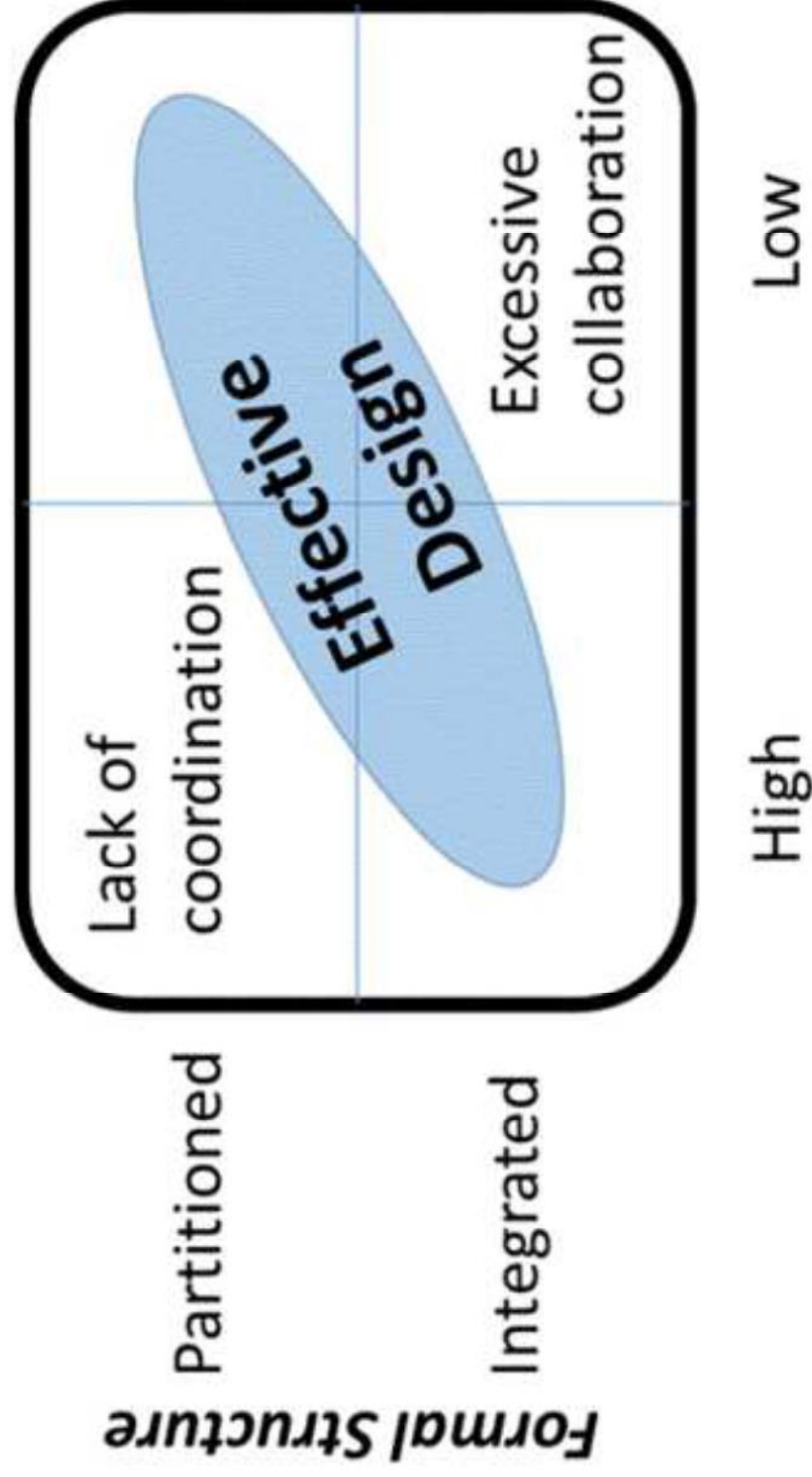
Figure

[Click here to download high resolution image](#)





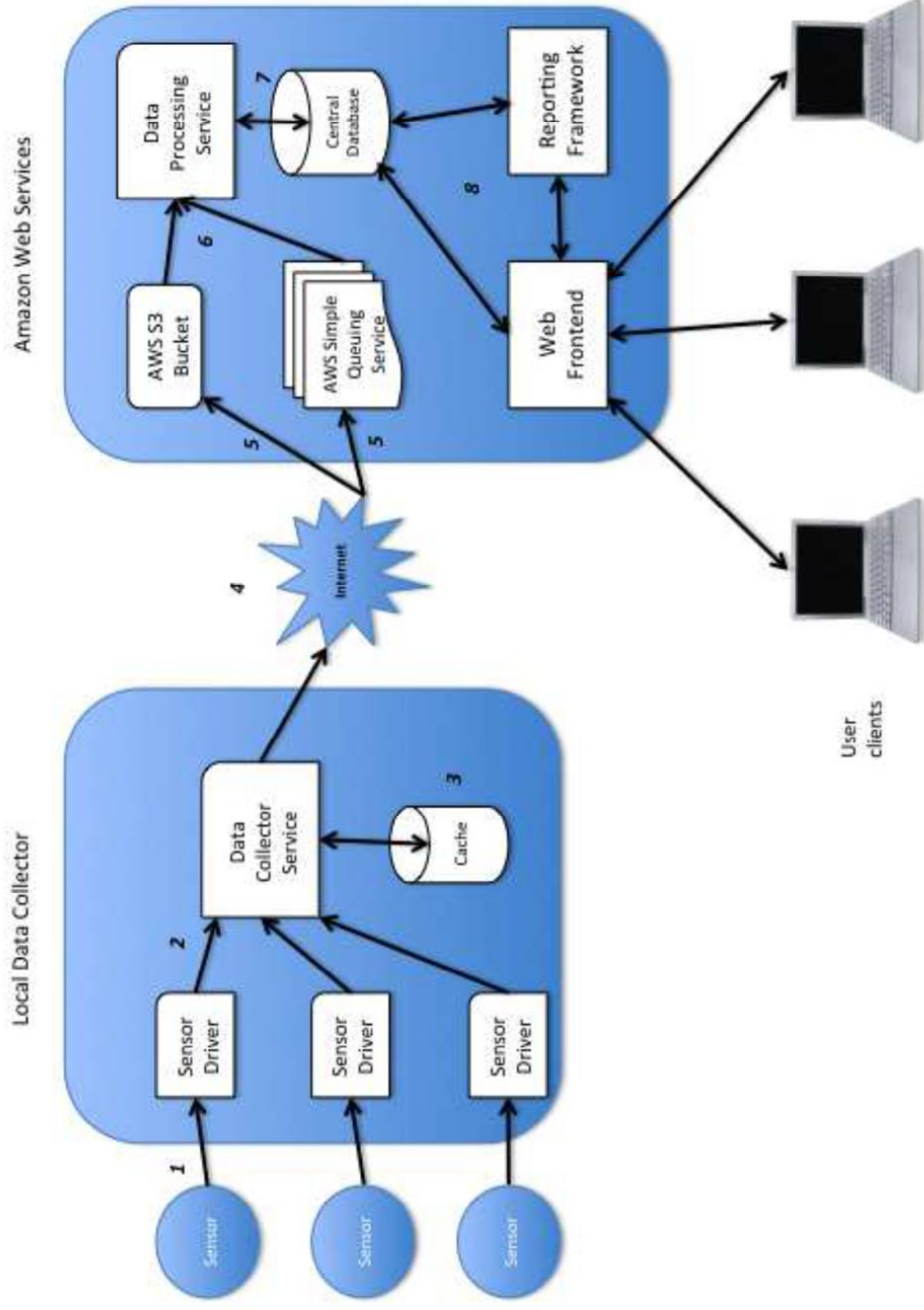




***Level of work process interdependence***

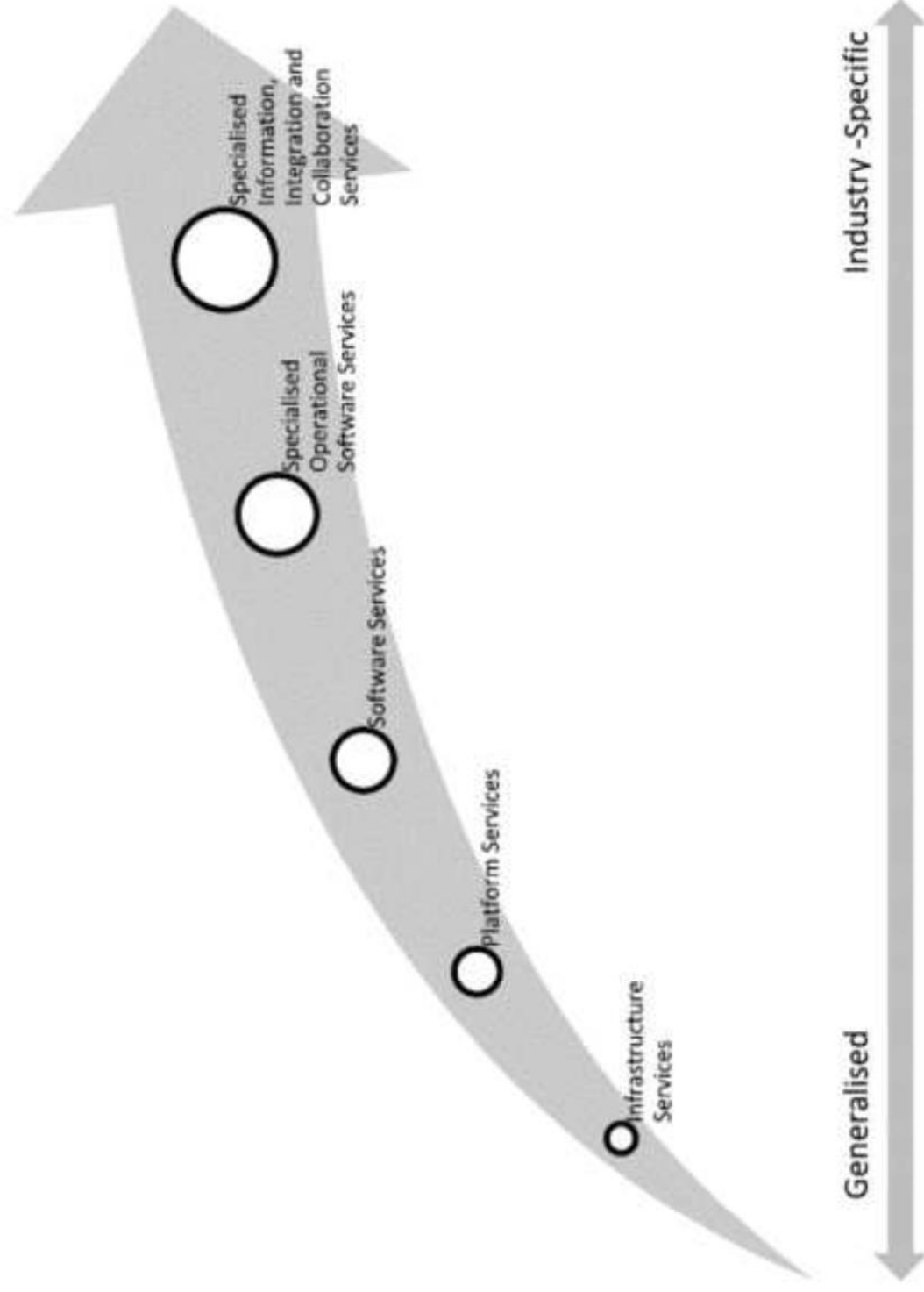
Figure

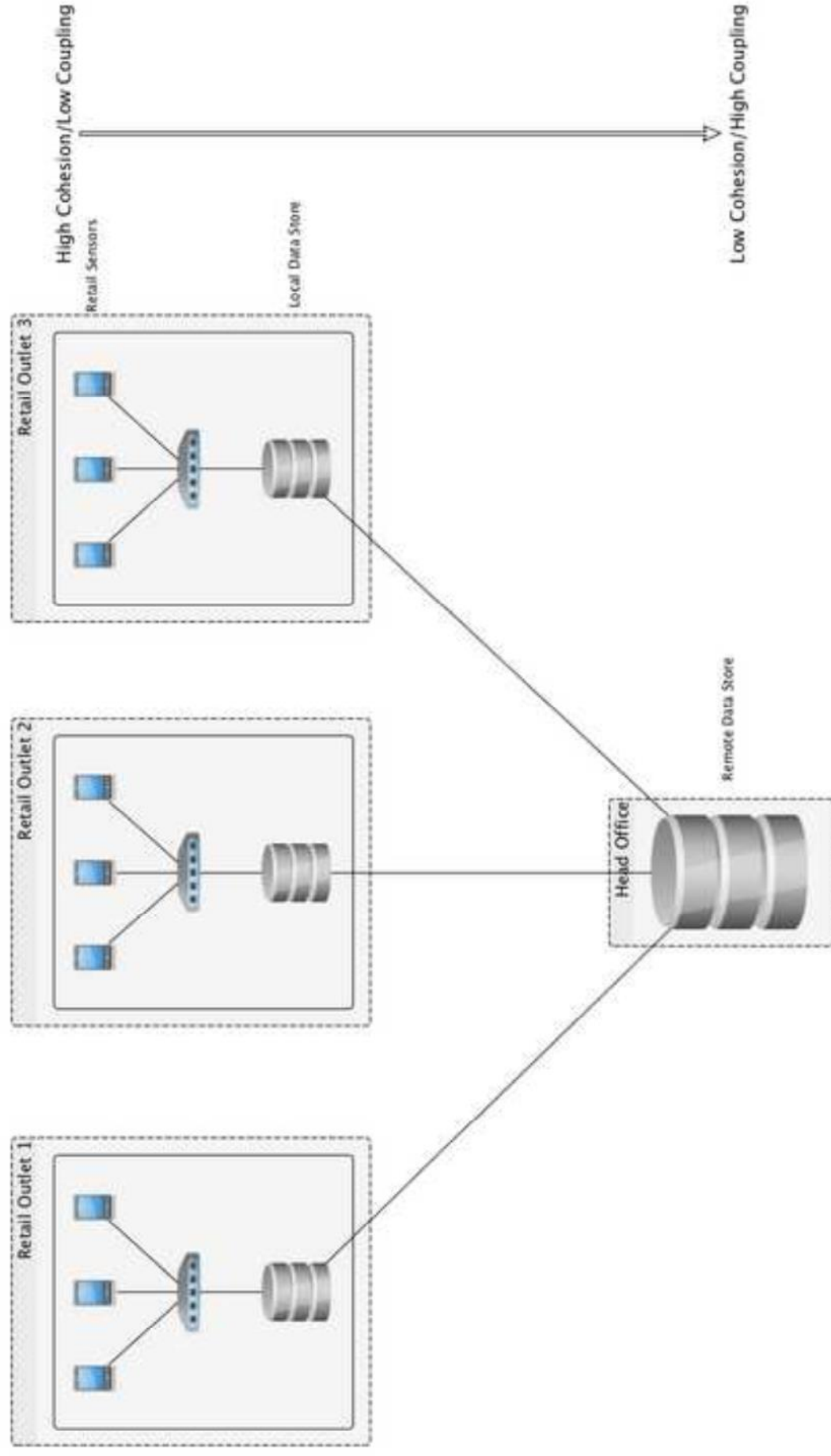
[Click here to download high resolution image](#)



Figure

[Click here to download high resolution image](#)





## AUTHOR BIOGRAPHIES

PROFESSOR ANDERSON received a BSc (Honours) degree in Computer Science from the University of Liverpool in 1993, and a PhD from the University of Liverpool in 1998. He is the Director of the Centre for Data Analysis and Representation at Edge Hill University. His teaching areas include Computer Programming, Physical Computing and Interface Programming. His research areas include Software Quality Assurance and Data Visualization. Professor Anderson may be reached at [mark.anderson@edgehill.ac.uk](mailto:mark.anderson@edgehill.ac.uk)

JOSEPH BOLTON received a BSc (Honours) in Computing (Application Development) from Edge Hill University in 2014. He is a Knowledge Transfer Partnership Associate at Edge Hill University. His work focuses on data processing and representation with a particular focus on collecting data from many discreet sources and locations, and combining it for later viewing in a central location. He may be reached at [boltonj@edgehill.ac.uk](mailto:boltonj@edgehill.ac.uk)