Linked data
Connecting and exploiting big data

Whilst big data may represent a step forward in business intelligence and analytics, Fujitsu sees particular additional value in linking and exploiting big data for business benefit. Only once we bring together myriad data sources to provide a single reference point can we start to derive new value. Until then, we simply risk creating new data silos.

Big data is a term that has risen to prominence describing data that exceeds the processing capacity of conventional database systems. With many solutions entering the marketplace it's all too easy to focus on the technology that allows big data processing for real-time business analytics and yet to lose sight of the long-term goal of integrating many data sources to provide even potential.

This paper has been written for Chief Technology Officers and senior architects in order to set out some of the challenges that big data solutions may bring and to suggest one approach that Fujitsu feels has the potential to provide particular value – the use of linked data to integrate data sources and better enable the exploitation of data to meet the challenges of the business, namely an expectation of near real-time business intelligence, free from the technology limitations that are imposed by a particular database structure.

The paper explains the concept of big data before examining the various approaches taken to data management in an enterprise IT context, explaining why each approach presents its own challenges. Some readers may wish to skip this section of the report and move straight to page 8, which examines the functions that a big data solution must perform, before the paper considers the related concept of linked data and its potential to act as a unifying technology, along with the associated architectural patterns. We then examine the integration of big data and linked data within an organisation, including the challenges that need to be addressed before providing a view of the future of data, summary and conclusions.

Throughout the paper, extensive reference is made to a variety of third party and Fujitsu resources that have been used.

Table of contents

Table of figures

Introduction

As work started on writing this paper in the closing weeks of 2011, big data was one of the IT industry's hottest topics. Unfortunately, just as cloud computing was previously hyped (before it became accepted as a business model) big data is in danger of being over-hyped to the point that it becomes an annoyance for some and confusing for many. But there is, undoubtedly, some value to be derived from analysing ever-increasing volumes of data that were previously inaccessible, or just too difficult to process – looking for patterns and information that were only previously available to huge corporations because of the costs involved and then presenting that to the business in a way that can be used to drive new or developing business areas. Now, through a combination of commodity hardware, cloud computing scale and open source software, big data processing and effective presentation is becoming accessible for even small start-ups [1].

What's so big about big data?

Right now we are generating more data than at any point in our history. In just the four years (up to 2010) the volume of data increased by a factor of six to an estimated 988 exabytes [2]. The "digital universe" was expected to pass 1.8 zettabytes in 2011 [3], is expected to reach 2.7 zettabytes in 2012 and may approach 8 zettabytes by 2015 [4].

To get some example of just how significant this explosion in data is, think of a single megabyte of information being equivalent to a square metre of land mass. In 2010 we would have covered the world with data. By contrast, in 1920 we would have covered an area the size of Madagascar, and in 2020 we will need 1700 globes to represent the volume of data we will have generated.[5]. Or, to visualise this volume in another way, the 988 exabytes of data in 2010 is roughly equivalent to a stack of novels from the Sun to Pluto and back [2]. This data explosion shows no sign of abatement and is likely to accelerate with new data types (in both senses: record structure; and social as well as transactional) together with greater access to networked devices (such as smart meters and smartphones with geopositioning data).

That data is amassed from a variety of sources but there are two in particular that are driving this explosion, combined with decreasing storage costs:

- The "Internet of things" with a variety of sensors collating information on our activities and our environment (the number of connected devices globally is expected to rise 11-fold from 4.5 billion in 2010 to 50 billion in 2020 [2]).
- The social web of networks sharing information about our activities, interests, location, likes and dislikes. In addition to those consider the private data stores created on our financial transactions, 'phone calls [6], health records [7], CCTV [8], etc. together with other online activities generating text, audio, video, click-streams, log files and more.

The McKinsey Global Institute describes big data as "the next frontier for innovation, competition and

Key terms

The following terms are used in this paper:

- Big data – in commonly-used term to describe data that exceeds the processing capacity of conventional database systems.
- Linked data – an approach taken to linking data such that it becomes more useful/accessible than it would be in isolation.
- Structured data – data stored in accordance with a strict schema for database management purposes.
- Unstructured data – data with no schema, or with a loose, implied schema (e.g. social media updates, log files, etc.)
- Business intelligence – a term used to describe systems used to analyse business data for the purpose of making informed decisions.
- Terabyte (TB) – a unit of data storage equal to 10^{12} bytes.
- Petabyte (PB) – a unit of data storage equal to 10^{15} bytes.
- Exabyte (EB) – a unit of data storage equal to 10^{18} bytes.
- Zettabyte (ZB) – a unit of data storage equal to 10^{21} bytes.
- OLTP (OnLine Transaction Processing) – an approach taken to data storage to support transactional systems, using a highly structured database.
- OLAP (OnLine Analytical Processing) – a form or structured database engine particularly suited to the analytical modelling used in data warehouses.
- SQL (Structured Query Language) – a commonly-used language for querying OLTP and OLAP databases.
- NoSQL (not only SQL) – an alternative approach to data storage, used for unstructured and semi-structured data.
- Hadoop – a framework for development of open-source software for reliable, scalable distributed computing. The Hadoop stack consists of a highly distributed, fault tolerant, file system (HDFS) and the MapReduce framework for writing and executing distributed, fault tolerant, algorithms. Built on top of that are query languages (live Hive and Pig). Hadoop can be considered as "NoSQL data warehousing" and is particularly suited to storing and analysing massive data sets.
- KVS (Key value stores) – in a key value store, data is stored by key and the value is just a blob (i.e. the data store is not concerned about the structure of the data). Key value stores are easy to build and they scale well. Examples include MongoDB, Amazon Dynamo and Windows Azure Table Storage and they can also be thought of as "NoSQL OLTP".
- RDF (Resource Description Framework) – a standard for data interchange on the web.
- SKOS (Simple Knowledge Organisation System) – based on RDF, but used to express hierarchical data.
- OWL (Web Ontology Language) – an extension to the RDF framework, providing an ontology to define and classify data and the relationships between data.
- REST (REpresentational State Transfer) – a style of software architecture that has gained widespread acceptance as a simpler alternative to SOAP and WSDL [50]. RESTful web services are implemented using REST principles and HTTP.
- SPARQL (SPARQL protocol and RDF Query Language) – a protocol and language for querying linked data that uses the RDF format.
- SQOOP (SQL to Hadoop) – a term used to describe bridges/gateways between SQL and NoSQL databases based on the Hadoop framework.

productivity" [9] but, put simply, it's about analysing masses of unstructured (or semi-structured) data which, until recently, was considered too difficult, too time consuming or too expensive to do anything with. Not only is some of that data now required to be kept for regulatory or compliance purposes (e.g. compliance with Safe Harbor in the United States or with data discovery laws in European Union nations) but there is also insight to be gained from looking for patterns in the way that we interact. For example: carbon information may be required in countries where sustainability legislation is being introduced; information about financial transactions and bank liquidity may be required to comply with financial regulations; scientific breakthroughs create new information sources (either new discoveries or new sensors); location information can help to optimise the physical position of moving assets (human or inanimate); and social graph information can be used to help companies better understand the ways in which they work to improve knowledge-worker productivity (as time and motion studies did for manufacturing) [10].

Not all of the data we produce is "big data". Forrester identified four main attributes to classify a data source as being big data[11]. This was later adopted by IDC [3], Gartner and others as follows:

- Volume – big data is massive, typically 100s of terabytes or petabytes; however the "big" is a relative term depending on the type of data. Storing this volume of data in a data warehouse is extremely expensive. But having more data upon which to run analysis can increase the number of factors to take into account and therefore improve the results [1].
- Velocity – big data may arrive quickly, in real time and it can be difficult to make timely decisions as a result; we need more insight, not more data and the challenge is to provide the right information at the right time, with the right degree of accuracy [3]. This leads to the need to stream data processing, either because it arrives too quickly and so some data must be discarded or where an immediate response is required [1], perhaps followed later by a more accurate analysis.
- Variety – big data has different structures and shapes, making it difficult to analyse with traditional database technologies (because of their rigid schemas); we may need to create application mash-ups to fully realise the potential of combining these data sources [3] whilst avoiding the traditional costs of integration – and we may find that other data stores are simpler and more efficient (e.g. a NoSQL database, a dedicated XML store, or a graph database) [1].
- Value[1] – most big data is low value until rolled up and analysed, at which point it becomes valuable; in addition, the availability of low-cost open-source solutions and commodity infrastructure (including cloud computing) is making systems that were previously only available to government agencies and large corporations cost-effective for a broader market.

Forrester's analysis highlights that the common theme is cost: it's not that existing technologies cannot meet the needs of big data, but that they cannot be cost-justified based on the benefits and risks involved. Forrester defines big data as "techniques and technologies that make handling data at extreme scale affordable" and, often, big data solutions trade off consistency and integrity for speed and flexibility [11]. Big data relates not just to new information sources: it's equally applicable for gaining new insights from data that was previously inaccessible and to accelerating and easing existing analytical processes[12].

Big data is about our ability to exploit information. In fact, it might help if we think of it as small data – nanodata – because that's what we're really talking about: huge numbers of very small transactions in a variety of formats. Whilst there are many new and evolving techniques and products aimed at managing big data, many of which need to become proven in their abilities to support business, simply applying big data solutions to unstructured and semi-structured data creates a new problem – a problem with its roots in the very earliest transactional database systems; and a problem being exacerbated by the explosion in data volumes we are seeing today.

The problem with big data

In solving one problem, our new-found ability to understand and then exploit big data has created another. Over many decades we have established models for data processing based on efficient, structured, databases and the new world of unstructured data does not fit. Consequently we run the risk of creating silos of data, each with their own limitations and benefits.

This paper puts forward a view that the key to extracting value from big data lies in a related concept; the concept of linked data exploitation. Linked data offers the potential to create massive opportunity from myriad data sources, both open and closed. And, with linked data as a broker, we have the ability to extract new data from old, creating insights that were previously unavailable, and facilitating exciting new scenarios for data processing.

[1] The original Forrester analysis used variability – and its impact on complexity – in place of value.

Approaches to data management

Databases have been key to our data processing systems (the forerunner of modern ICT) for over fifty years. Over time we've seen a gradual evolution in the way that we process and store this data and, not surprisingly, there have been significant advancements and investments along the way.

Transactional data

Transactional systems (evolutionary stage 1 in Figure 1) are used to manage something through a number of stages, for example an order is taken and goes through various stages to completion, possibly including manufacturing processes and payment processes. Other examples include financial transactions (withdrawing cash from an ATM, making a payment in a retail store, transferring money between accounts, etc.) and resource management processes (human resources, etc.).

OnLine Transaction Processing (OLTP) systems are used for data entry and retrieval within transaction-oriented applications. Characterised by short, online, transactions (insert, update, delete) the emphasis is on fast query processing and guaranteed data integrity.

Transactional systems provide simple reporting functionality, for example using Structured Query Language (SQL) queries to report on data contents

```
SELECT * FROM Employees WHERE Office=London
```

The SQL statement above is a simple query to return a list of all employees based in London and such queries can be built upon to generate simple reports.

Whilst they undoubtedly have advantages, transactional systems have their limitations too:

- It can be difficult to change the structure of an OLTP database (the schema) and locked records can cause difficulties where data is being acted on by several parties. In addition, the generation of reports requires knowledge of query languages such as SQL, increasing complexity and hence the time taken to create a report
- Increasing performance with a transactional system typically involves scaling up – providing more storage, more computing power, and more network connectivity.

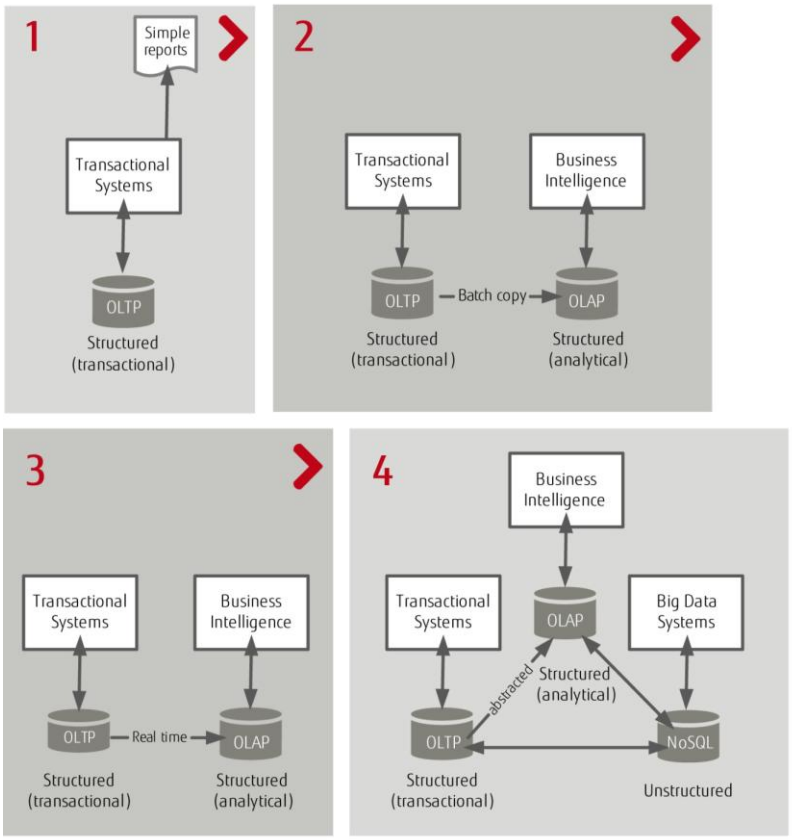

Figure 1: Evolution of the database

Analytical data

As reporting requirements became more complex, new analytical systems were developed (evolutionary stage 2 in Figure 1), storing information in OnLine Analytical Processing (OLAP) "cubes". Such systems are typically focused on business intelligence – spotting and predicting patterns, reporting on business performance, providing statistics, etc.

At some defined point (for example at the end of a day's trading), historical transaction data is moved from an OLTP database into an OLAP database. Often the history is aggregated – i.e. instead of storing an entire history of transactions, an abstracted or calculated view is provided. Analytical systems generally have a lower number of transactions but use complex queries based on consolidation, drill-down or slicing and dicing to derive meaning from data (data mining). Such systems aid organisations with making decisions (for example, based on sales data) and are often known as data warehouses. For many companies, the data warehouse is a significant investment and is the basis of business intelligence activities.

The down-side of analytical systems is that they tend to work on historical data – yesterday's, last week's or last month's transactions.

Some organisations have managed to transition to a point where they have analytical systems working on real-time (or near-time) data, maintaining a smaller OLTP database consisting just of in-process transactions and extracting, transforming and loading completed transactions into the OLAP database, possibly even using other types of databases, such as the column database management system used by Dunnhumby

to process data for Tesco's Clubcard loyalty scheme [13]. This approach (evolutionary stage 3 in Figure 1) provides significant advantages in terms of up-to-date reporting and, potentially, speed of response with alerts when certain conditions are met, whether that is making offers more relevant to consumers, improving response rates, or providing other forms of actionable data [14].

Unstructured data

Regardless of whether an organisation operates on historical or real-time analytical data, we still have two silos of structured data: OLTP and OLAP. And, as organisations are looking to create a new class of systems working on big data, we're seeing the rise of a third class of unstructured (or sometimes semi-structured) data (evolutionary stage 4 in Figure 1). Indeed, this data can be further subdivided into:

- Document-centric data where the document is the boundary of the record. Examples include standard business documents (Word, Excel, PowerPoint, etc.) in a document management system but also email messages, web pages (HTML documents), audio and video recordings, images - stored as blobs of data inside a traditional database.
- "Big data" such as social media updates, data streams from machine to machine communications, clickstreams from search queries, etc.

This "big data" is usually processed into NoSQL databases. Commonly misinterpreted as an instruction (no to SQL), it really means *not only* SQL – i.e. there are some types of data that are not worth storing in a structured database, or for which another type of database (such as a graph database) may be more suitable[2].

Rather than following the traditional database model of extract, transform and load (ETL), with a NoSQL system the data arrives and the application knows how to interpret the data, providing a much faster time from data acquisition to insight.

External (open) data sources

Increasingly, we also need to consume and federate data from outside our organisations – not just the sort of unstructured data that comes from social networks, sensors, etc. but also public data sets and private databases offered up on a subscription basis (for example, credit records, etc.), which may be structured or unstructured.

In addition to established sources such as data.gov in the United States of America and data.gov.uk and opendata.ie in the United Kingdom and Ireland respectively, the European Commission has launched an open data strategy to release public authority data sets. The theory is that by giving away data that has already been created using public money, an economic stimulus is provided from companies building new products and services around the data [15].

NoSQL

There are two main types of NoSQL database [49]:

- Key value stores – in a key value store, data is stored by key and the value is just a blob (i.e. the data store is not concerned about the structure of the data). Key value stores are easy to build and they scale well. Examples include MongoDB, Amazon Dynamo and Windows Azure Table Storage and they can also be thought of as "NoSQL OLTP".
- Apache Hadoop – a framework for development of open-source software for reliable, scalable distributed computing. The Hadoop stack consists of a highly distributed, fault tolerant, file system (HDFS) and the MapReduce framework for writing and executing distributed, fault tolerant, algorithms. Built on top of that are query languages (like Hive and Pig). Hadoop is more like "NoSQL data warehousing" and is particularly suited to storing and analysing massive data sets.

From a data management perspective NoSQL data might as well be random – it is unstructured, has no schema (although some data may have an implied schema) and is characterised by the "4 Vs" of big data, described previously [11] [3].

A scale out model is used to improve performance with scheduling software (such as Hadoop's Map/Reduce framework) subdividing and processing queries over multiple compute nodes and each node is typically small, so we actually increase performance by using many, small, computers – exactly the opposite of a transactional system. Whilst transactional data history gets loaded into data warehouses, the results of big data analysis in the NoSQL world are often discarded. Queries are run and then thrown away. They can be loaded into OLAP systems but that requires some form of bridging solution (like SQOOP, for connecting SQL and Hadoop) and the bridge is generally inefficient. This creates a major issue with data management that needs to be resolved.

In the UK, the government is also launching an Open Data Institute to support businesses in the exploitation of open data, based in the East London Tech City (Silicon Roundabout) and led by Professors Sir Tim Berners-Lee and Nigel Shadbolt [16].

Typically the Government-owned data sites provide metadata to describe the datasets, information about the datasets and tools for access to the datasets[17]. In the UK, new licensing constructs have been created to allow data to be used freely and flexibly [18].

[2] A breakdown of NoSQL database types is available at http://nosql-database.org/

The UK's plans include [19]:

- Linked data services to track healthcare impacts and improve medical practice.
- Enabling citizens to access their personal medical records.
- Real-time and planned information on trains, buses and road networks for more efficient transportation and logistics.
- Allowing third parties to develop applications for businesses and consumers using data sets such as weather data and house price information.

Met Office data has already been made available, data from the Land Registry will be released in March 2012, Department for Transport data will be made available in April 2012 and the NHS will release its data in September 2012 [15].

External data sources offer massive potential for commercial use but it is worth noting that they are outside the control of the organisation consuming them and their quality may be unknown. Feedback mechanisms and standards will be created for improving the quality of open data but, for now, they will be viewed with caution by some organisations. Even so, there is significant interest in open data and national media organisations have already made extensive use of public data to report on the news through a new medium of data-driven journalism, with examples including the Guardian Data Store[3]. By locating data, filtering/interrogating, combining/mashing-up, visualising and reporting stories based on data, this new model for journalists is just one example of the massive social implications of big data [20] [21].

Characteristics of the various data models

Unfortunately, as each data model has been refined, limitations have emerged that result in the creation of a silos, each suited to a particular type of system and with its own benefits and restrictions.

Data Model	Attributes	Flexibility	Age of data	Quality of data
Transactional	Known	Fixed schema	Short lived/current	High
Analytical	Aggregated	Fixed schema	Longer life/historical	(Typically) High
Unstructured	Unknown	Implied	Random	Low
External	Variable	Variable	Variable	Variable

Figure 2: Data model characteristics

Each data type has its own management requirements, and its own strengths and weaknesses. No one data model is likely to consume the others so we need a solution that acts as broker access to all type of data, minimising the overhead of maintaining several disparate data sources.

[3] The Guardian data store is available at http://www.guardian.co.uk/data.

A functional view on big data

One, simple, description of big data is that of a "firehose" that requires filtering and plumbing to ensure that the right data is received at the right time [22].

Whilst that provides an (extremely) high level view, there are certainly a number of core functions that any big data solution should provide.

Data acquisition, collection and detection

By applying rules to data streaming from sensors and other sources, an analysis of the current state can be performed to extract context and identify any triggers (i.e. something happened) or temporal analysis (for example this person said something more times than is normal over a particular period). Example triggers may include that a stock price has dropped by 20% whilst temporal analysis might show that there have been 3000 mentions of a brand on a social networking platform in the last five minutes. The resulting alerts provide notification so that decisions and actions may be taken in real time.

Data management and integration

Because of the diversity of the data extracted it's necessary to automatically categorise it for archival, regardless of whether it is real-time data from sensors and other inputs, or data from external systems (for example other databases within the enterprise).

Data analysis

In a world where speed of response is more important than accuracy, a cluster of processing nodes can be working on data to return results in near real-time whilst other tasks improve the quality of the data. Analysis should be available for both real-time and historical data, as well as providing the capability to make predictions or running simulations.

There's more to real-time analysis than just processing data streams. More complex scenarios can also be built, for example highlighting that an influential customer has contacted the customer service team with a problem: their business may not in itself trigger a VIP response but their extended network and peer group influence might warrant an alternative approach being taken to retain their goodwill.

Drill-down and aggregation are normal business intelligence activities in data warehousing (e.g. examining sales data in context with previous periods) but there is also potential to use these techniques in the world of big data. For example, as sensors become increasingly integrated into our society (in agriculture, healthcare, transportation, etc.) it will be important to know when they are likely to fail. Big data might provide the answers in terms of the average number of records a sensor will return before it dies, telling us when a sensor is approaching its half life and so plans need to be put in place to replace it.

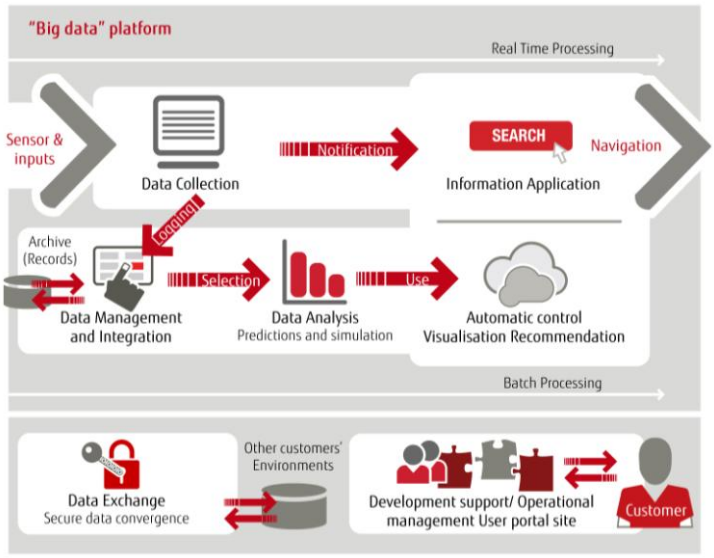

Figure 3: A functional view of a big data platform

Considerations for big data management

When selecting tools for big data analysis there are a number of considerations to take into account:

- Where will the data be processed? Using locally-hosted software, on a dedicated appliance, or in the cloud? [1] Early big data usage is likely to focus on business analytics and the need to turn capacity on/off will lead enterprises towards the cloud [12] rather than implementing private solutions.
- From where does the data originate? How will that data be transported? Often it's easier to move the application than it is to move the data (e.g. for large data sets that are already hosted with a particular cloud provider). If the data is rapidly updated then the application needs to run close to the data in order to reduce the impact of computing on the time to response [1].
- How clean is the data? Big data's variety means that it needs a lot of cleaning – and that costs (time and money). It may be more effective to use a data marketplace and, although the quality may vary, there will generally be a feedback mechanism in place [1].
- What is your organisational culture? Do you have teams with the necessary skills to analyse the data, which may include mathematics, programming and scientific instinct [1]? Complicated mathematical formulae (algorithms) are increasingly controlling our lives [26] and it may be necessary to create data science teams with technical expertise, curiosity, the ability to communicate, and creative ability to look at problems in different ways [47] [51].
- What do you want to do with the data? Having some idea about the outcomes of the analysis may help to identify patterns and discover clues in the data [1].

Key to the analysis function is the ability to both search and filter – two commonly used data reduction techniques. Searching is most useful when it's known what data is required, whereas filtering assists where irrelevant data needs to be selectively eliminated in order to gain insights [23].

Information application, automation and visualisation

The key to the success of big data analytics is making it simple. In a world where data has value, a market will appear for data as a service – effectively data clouds with access sold on a subscription basis, either by transaction volumes or access to entire data sets. In such a world, visualisation is the presentation layer and we can expect gadgets (for example in HTML) to make it easy to integrate data. To some extent, this is already happening with sites like Pachube opening themselves up as brokers for data sources from the Internet of things. Using such a service it becomes trivial to integrate, for example, an energy usage monitor with a website in a home automation context. Scale this up to a regional or national grid level and there is something that has real value, particularly when combined with other sources.

Information application is about combining web services to create new services. By making it simple for business end users to create mash-ups with data, we enable a world of dashboards that allow the creation of business insight and, potentially, the creation of new business models, built around applications. The resulting recommendations present information in context to users. Whereas Facebook built our social graph and Runkeeper is establishing our health graph [24], we will see new graphs for different classes of data. We'll also see new analysis and algorithms being used with new visualisation, just as in recent years we've seen the emergence of word clouds and fractals as forms of representing information [10].

There is another angle that's often overlooked: analysis – using the resulting data to derive value – and that often involves a human element [11], spawning start-ups like Kaggle that crowdsource big data analysis [25]. Indeed, there is some suggestion that data science is increasingly important in our lives, with complex mathematical algorithms working on a variety of solutions, from the stock market to the supermarket [26].

Automation is another important element of a big data solution – if big data systems can take action on notifications and analysis then competitive advantage can be gained; however it's no silver bullet. In one example, Credit Suisse was fined $150,000 by the New York Stock Exchange for failing to supervise an algorithm that had gone wrong and, a few months later, the "Flash Crash" wiped 9% off the Dow Jones Industrial Average in a few minutes, suspected to have been triggered by two algorithms locked in battle [26] [27].

Data exchange

Federation allows data to be exchanged, securely, with other environments (customer data and external data stores) when needed, integrating data in a manner that is defined by the data owner.

Development support and operational management

Development support functions and operations management functions for the entire platform should be provided in a user portal. As a result, all development resources can be managed and operated in an integrated manner.

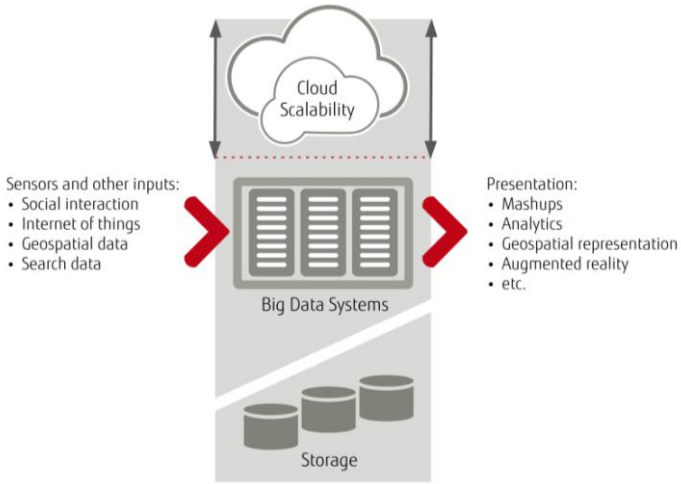

Figure 4: Big data in the cloud

Big data in the cloud

With many organisations taking a cloud first approach for new systems, cloud is an important consideration. Just as for other cloud applications, some instances will be hosted publicly and others will make use of private solutions.

Regardless of the cloud model, marrying big data solutions with cloud infrastructure, platforms and applications makes it possible to provide scalability to handle fluctuations in data inputs, or in the requirements of the presentation outputs whilst the overall volume of data stored and access grows over time (see Figure 4).

These presentation outputs will create new business models, for example based on mash-ups, analytics, geospatial representation and augmented reality but the integration between the big data systems and the databases holding the data is absolutely key, resulting in new architectural patterns for exploiting big data.

Architectural patterns for exploiting big data

As an industry, we've invested decades of experience into managing structured data and have developed established methods for improving performance and optimising queries. With the rise of big data, we are likely to create reference architectures that include a parallel technology in the enterprise data warehouse [12], dividing data into "two worlds" (structured and unstructured/semi-structured). Indeed, the problem is even worse than that – the structured data is in OLTP databases and in OLAP data warehouses, then there is big data from multiple sources (maybe even external sources) and, potentially, we end up storing the results from our analysis somewhere else.

Now, as we develop new systems to manage big data, there is an opportunity to look at the best architectural pattern to apply, considering not just the type of data being captured and stored, but the types of systems (i.e. what it's being used for) and the structure (or lack of) for that data.

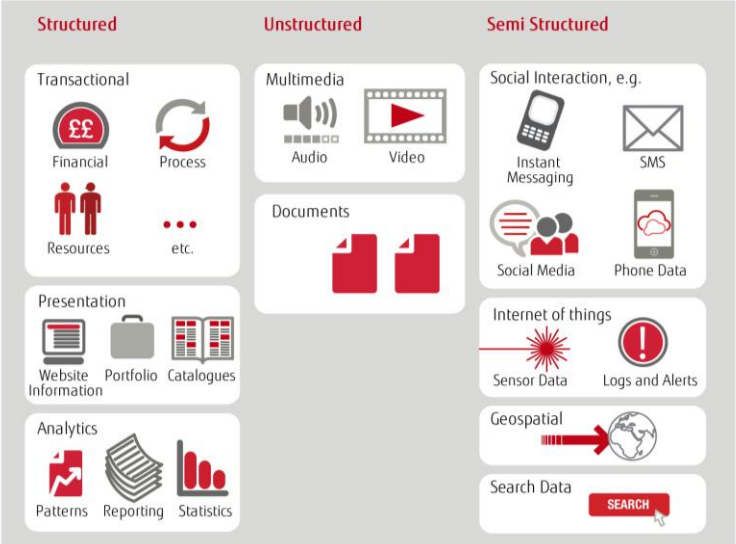

Figure 5: Architectural patterns for data

For some types of data, a traditional, structured database is still the best solution. Document-centric unstructured data might require another approach (maybe even just a file system) whilst semi-structured (NoSQL) systems remain especially useful for certain activities in relation to some of the new classes of data that are being exploited under the banner of big data.

Major database vendors like Oracle, IBM and Microsoft are concentrating on the integration of SQL and NoSQL systems within their product portfolios. This is a perfectly valid approach but it doesn't solve the issue of data management – indeed it exacerbates the issue. Fujitsu believes that, instead of thinking of big data as being synonymous with NoSQL, perhaps we should think of big data as NoHadoop (not only Hadoop). The question is not so much about which database model (structured or unstructured) is best, but about how we bring them together.

The optimal approach is to use linked data as the reference for big data systems to query against, providing analysis, search and alerting.

Linked data

Whilst there are exceptions, big data tends to be unstructured (i.e. has no formal schema) and metadata (data about data) becomes important (for example location data can help to make some sense of the data in that it provides some structure). Linked data provides some significant advantages in tying together different records to provide a view of the bigger picture.

To give an example, humans can relate different types of information on a web page (for example: a profile page about someone, with some status updates and some geo-tagged pictures) but machines can't. If, instead of linking to the container (i.e. a web page), we link to the data within that container, we can create machine-readable relationships [28].

In short, linked data sees the web as an immense database that can be mined to link to data, rather than document-based resources; however it may be helpful to look in more detail at how linked data works.

How linked data works

There are three rules to linked data [29]:

1. Linked data uses HTTP URIs – not just for documents as with "traditional" websites but for the subjects of those documents – places, products, events, etc.
2. Fetching data using the URI returns data in a standard format, with useful information such as who is attending an event, where a person was born, etc.
3. When the information is retrieved, relationships are defined, for example a person was born in a particular town, that a town is in a particular country, etc. Importantly, those relationships are also expressed using URIs, so looking up a person links them to the town where they were born, and that links to a region or a country, etc.

A model is required to identify the data within a resource and a file format to encode it in – one such format is Resource Description Framework (RDF) [28], a standard for data interchange on the web[4]. RDF extends the linking structure of the Web to use Universal Resource Indicators (URIs) to name the relationship between things as well as the two ends of the link [30]. It provides a framework (structure) for describing resources (assets) [31].

Effectively, using addresses to identify assets RDF is to a web of data what HTML is to a web of documents[5]. Or, to put it another way the Internet is about connecting computers, the world wide web is about connecting documents and now we are focusing on a "giant global graph" to connecting the things that those documents are about [32] – i.e. the data, information and material information (or knowledge) that influence our decisions [10].

RDF has an elegant solution for identifying data – using a grammar of subject, predicate and object [28] as shown in the example in Figure 6:

Figure 6: Linked data triples

These relationships are known as triples and they allow us to link directly to information and compare it with other resources.

RDF is both flexible (in that data relationships can be explored from many angles) and efficient (it is not linear like a database, nor is it hierarchical like XML) but it's still just a framework – to define and classify entities and the relationships between them ontology is required. The Web Ontology Language (OWL) is based on RDF and provides an extended vocabulary to describe objects and classes[5]. Importantly, this allows for inference: the creation of new triples based on existing triples to deduce new facts based on stated facts [31]. It is specifically defined to add capabilities to the web that may be distributed across many systems, scale, remain compatible with web standards, and be open and extendable[33].

[4] The current status of the RDF standard is available at http://www.w3.org/standards/techs/rdf#w3c_all.
[5] Information about OWL is available at http://www.w3.org/2004/OWL/.

As with human communication, vocabulary can emerge according to uptake. This can be for specific domains of interest, or to introduce additional generic ways of describing relationships between information. For example, the Simple Knowledge Organisation System (SKOS) is based on RDF but is designed to express hierarchical information[6] – broad/narrow terms, preferred terms, and other thesaurus-like relationships [31].

Whilst RDF is not intended for human readership [34] we can still use triples to build a graphical representation of the relationships [28].

In Figure 7, the round nodes are resources. Resources may be typed with a class. Square nodes are classes. A class is also a resource (i.e. metadata is also data) [35].

Each resource is represented as a web resource, named with an HTTP URI (textual properties in the content, hyperlinks to related pages).

Resources may be distributed and refer to other resources with URIs.

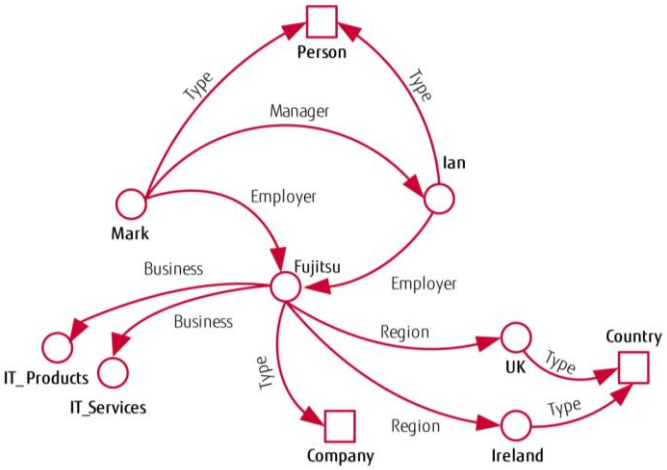

Figure 7: Linked data graph

In Figure 7, the triples are:

```
<Mark><hasEmployer><Fujitsu>
<Ian><hasEmployer><Fujitsu>
<Mark><hasManager><Ian>
<Fujitsu><hasBusiness><IT_Products>
<Fujitsu><hasBusiness><IT_Services>
<Fujitsu><inRegion><UK>
<Fujitsu><inRegion><Ireland>
```

`<Mark>` and `<Ian>` are of type `<Person>`, `<Fujitsu>` is of type `<Company>`, and `<UK>` and `<Ireland>` are of type `<Country>`.

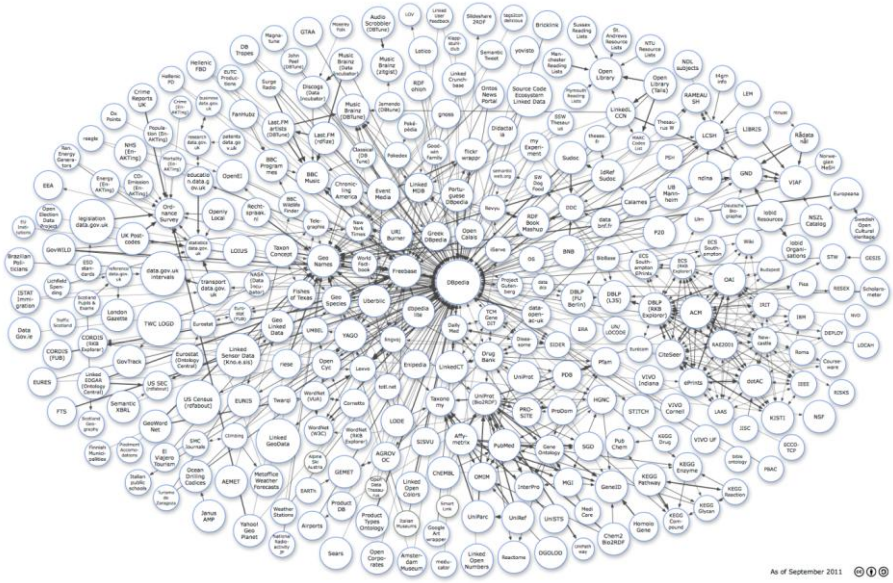

Figure 8: Linking open data

Using this structure, which is both human and machine readable, we can navigate and query data. For example, we can query based on triple matching and we can build complex queries that are then filtered.

The linked open data cloud

Using linked data based on a combination of RDF, OWL and SKOS, we can see how our data might easily be joined with other data sets. For example, extending the graph shown in Figure 7, there may be a database of companies that contains more information about Fujitsu; or a database of countries that has information on currency, GDP, government structure, etc. Quickly, many linked data sets lead to others and create a cloud of linked, open, data – the semantic web [36].

[6] Information about SKOS is available at http://www.w3.org/2004/02/skos/

In order to query linked data, a language and protocol called SPARQL has been defined[7]. SPARQL is a recursive name (SPARQL and RDF Query Language) and, whilst other RDF query languages exist, SPARQL is emerging as a de facto standard (it is also a W3C recommendation) [37].

To give one example of how linked data works in practice, the United Kingdom's data.gov.uk site releases government data to help people understand how government works, how policies are made, and to take action or make decisions based on that data [38]. The data is given URIs to allow the data to be linked. Reference data is expressed in RDF format and some of that data has a SPARQL endpoint, enabling interactive searches across the data [39].

In another example, the BBC uses linked data to collate information from across the Internet in its Wildlife Finder[8] website, as well as developing improved cross-site search functionality for its many microsites and integrating other web content into its BBC Programmes and BBC Music sites [40].

DBpedia is a website that extracts structured information from Wikipedia info-boxes[9] (to publish them as data for consumption elsewhere [41]. Now Wikipedia is providing an editable environment for data that can be used to automatically populate the info-boxes, collecting references to data to build out a database of all accessible data sources [42].

[7] Information about SPARQL is available at http://www.w3.org/2009/sparql/wiki/Main_Page.
[8] The BBC Wildlife Finder is available at http://www.bbc.co.uk/nature/wildlife; BBC Programmes is at http://www.bbc.co.uk/programmes; and BBC Music is at http://www.bbc.co.uk/music. The BBC's enhanced search service (Search+) is described at http://www.bbc.co.uk/blogs/bbcinternet/2010/01/bbc_launches_enhanced_search.html.
[9] Wikipedia info-boxes are described at http://en.wikipedia.org/wiki/Help:Infobox.

Could linked data represent the next evolutionary stage for data management?

Linked data is often used in the context of open data, but it doesn't have to be that way – not all linked data is open and not all open data is linked [43] – and there is plenty of scope for using linked data within the enterprise, regardless of the approach to open data.

Through linked data, data in unstructured databases can be linked to data in traditional OLTP and OLAP stores without changing existing schemas. The linked data component maintains the relationship between databases to keep them consistent and existing applications continue to function whilst the data is exposed to other applications for consumption, including mash-ups.

Using linked data provides a highly scalable solution, based on the same principles as unstructured data (scaling out to use multiple, small, compute nodes) but by structuring the linked data around triples, as described previously, data from multiple sources can be quickly interrogated.

Effectively, linked data can be used as a broker, mapping and interconnecting, indexing and feeding real-time information from a variety of sources. We can infer relationships from big data analysis that might otherwise have been discarded and then, potentially we end up running further analysis on the linked data to derive even more insight.

It's important to remember though that even linked data is no panacea – if rigorous controls are not applied to the metamodel then it becomes yet another unstructured data source, making the problem worse, rather than better!

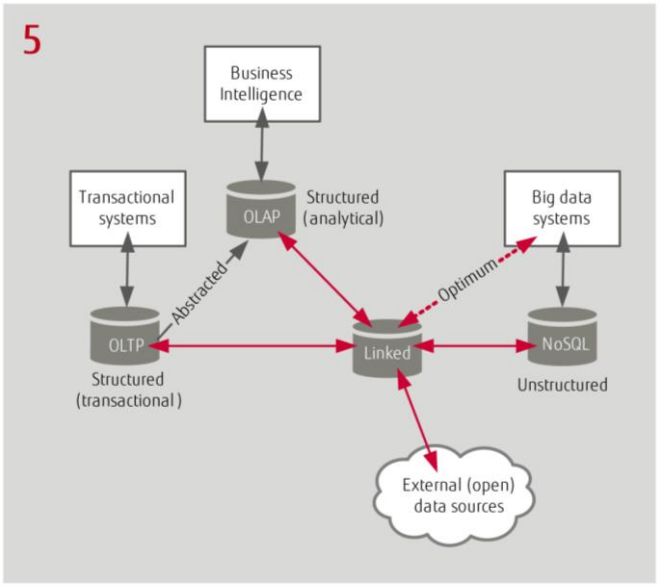

Figure 9: Linked data - the next evolution of the database?

Considerations for implementing big data and linked data within the enterprise

Most enterprise concerns for big data relate to getting the right data, at the right time, with the right quality. In order to achive this (and to trust the results), it's important to address each of the considerations in the following paragraphs.

Data integrity

Once we start to link data sources, we need to set out some rules in order to maintain the integrity of that data and avoid the creation of "data soup" contaminated by erroneous (either inaccurate or incomplete) data. As data is increasingly used to make decisions, consumers of the data services need to have confidence in the service [2].

Ontology is one consideration and, whilst the word sounds rather grand, it's really about ensuring that there is consistency in naming data. For example a *person* in one database may be a *user* or a *customer* in another system. We need to understand what is functionally and logically equivalent and what is not – in other words we need to establish a common taxonomy for all instances of data.

Integration

Over the last few years, web services have become the de facto method for integrating data sources. In the web services world, the function is fixed and we use different functional (API) calls to act upon the data. Now we're seeing a new stage of evolution as we progress towards RESTful data. In the world of linked data everything has a URI. With REST the data is fixed and the function changes, resulting in a new node with a new URI. Following the links between the nodes creates a graph and that graph defines relationships to integrate data from a variety of sources. One analogy here is that of the UNIX operating system, whereby everything is represented as a file/resource (i.e. providing all the processes to build the functionality instead of a server trying to guess the functions that may be required and exposing specific processes.

Another way to look at RESTful interfaces is as being data-oriented – i.e. data is the key asset. Process is still supported, but in the context of the data. For example the United States Navy is reported to be considering how to transform its IT operations in such a way that it is focused on the data itself, ingesting, understanding and sharing data in standard formats, allowing for better decision-making from leaders [44]. Developers consuming REST services appreciate the simplicity; above all the low barrier to entry and data transparency [45] – the RESTful approach effectively makes data discoverable for all applications, avoiding duplication [44].

As previously mentioned, consumers need to be confident that decisions are not made based on erroneous data (not to be confused with the opportunity to make fast decisions based on imprecise, or incomplete, data). The increasing use of third party data sources creates a requirement for platforms that can guarantee the provenance of data to enable safe trading of information, with appropriate checks and balances in place (just as with today's credit reference systems that are used extensively by the financial services industry) [2].

Data management (and understanding the data)

Master data management is another consideration that is really just concerned with the definitive source of data. For example, the master source of salary information is most likely to be the human resources database, details of a management structure may be stored in an ERP system, but a phone number might be user-editable in the organisational directory – or even in an external data source.

Data management is concerned with so much more than the master source though – we need to understand the data (what is its purpose?) but also be able to manage the lifecycle of any record – and big data means many more records.

In effect, data is created (arrives), maintained (exists), and then deleted (disappears) at some point in the future. That data needs to be managed such that the data held is appropriate - i.e. it's kept for the right time. But big data's potential to find new insights from old data leads to another question – do we need to consider a new approach for big data – for example keeping it forever? Or, whereas once we used to dispose of data based on its age, maybe in the world of big data we should think about value instead?

Aside from the consideration that we "don't know what we don't yet know" (i.e. there may be hidden value in data that is yet to be discovered), with linked data disposal of the data could break the chain – causing additional management headaches to maintain referential integrity – negating some of the simplicity of a linked data solution.

Data replication

Data replication is largely concerned with maintaining high availability in case one copy of the data is corrupted or unavailable. Generally, this involves storing data in multiple datacenters but the volumes of data involved in a big data solution raise challenges about achieving horizontal scalability.

Big data frameworks such as Hadoop are inherently resilient, which raises the question as to whether it's still necessary to introduce another layer of replication in the infrastructure (or simply cluster Hadoop nodes across multiple locations).

Data quality

The quality of data is another concern, including the handling of duplicate records from disparate systems. Data needs to be:

- Correct – in both its source and in its details – for example, a name must be spelt correctly.
- Complete – for example, if a given name is stored without a family name, it is incomplete. Or a record may be missing other mandatory fields such as a date of birth.
- Current – for example, containing up-to-date contact details.

In a relational database, it's fairly simple to identify missing attributes for attention but this may not be as straightforward when working with linked data taken from multiple sources in a variety of formats.

Data storage

Whilst data management is concerned with the lifecycle of data, and data replication addresses issues relating to business continuity, disaster recovery and availability, there are still issues to address regarding the medium for storing data and for archival/backup (and restoration) of data. The advent of big data solutions seems certain to finally kill off tape-based systems but consideration must also be given to the storage systems. Many platforms include a degree of redundancy and online "hot spare" disks and, as mentioned previously, frameworks such as Hadoop are inherently resilient but provision also needs to be made to manage the media upon which the data is stored. This may involve replacing failed components or marking them permanently offline, then replacing whole sections of infrastructure once a certain number of failures have occurred.

Data migration

When moving data in and out of a big data system, or migrating from one platform to another, the size of the data creates some challenges. Not only does the extract, transform and load process need to be able to deal with data in a variety of formats, but the volumes of data will often mean that it's not possible to operate on the data whilst a migration is taking place – or at the very least there needs to be a system to understand what is available, and what is currently missing (or on another system).

Not only might the structure of the data change (e.g. from a relational data source to linked data) but it will move location (impacting its URI) and there may be semantic differences, such as a "company" in one system being a "customer" in another.

Data security and access control

Finally, we need to address the important issue of who/what can access data – preventing unauthorised access to data and keeping it protected. Fine-grained access controls will need to be developed, limiting not just access to the data but even knowledge of its existence. And the world of linked data raises a new spectre in that aggregation may lead to knowledge being inferred that would otherwise be deemed a security risk. Enterprises need to pay attention to classification of data, ensuring that it's not locked away unnecessarily (limiting its potential to provide actionable insights) but equally that data does not present a security risk to any individual or company.

The future of data management and exploitation

The explosion in data generation shows no sign of abatement. Data touches all aspects of our lives (both work and pleasure) and it's not just the aspects of this data that classify it as "big data" that we need to be concerned with – connecting data has the potential to deliver huge power. And the more we connect, the more powerful it becomes [29], creating new job roles and opportunities [26] [46] [47] but also the potential to better understand our work and make informed decisions [46].

Data drives a huge amount of what happens in our lives and it happens because *somebody* takes that data and does *something* with it [29]. By making the data machine-readable, it doesn't have to be *someone* that does the *something* – it could be *something* that does *something* with that data – perhaps only requiring human input where further investigation is required.

Unfortunately, there is a downside too – scraping, cleaning and selling big data has legal implications including copyright and terms of service violations [48] but these are unlikely to apply on data owned by the enterprise. Within the enterprise, questions will be raised about security (data classification – and the effects that aggregation has on this) but, over time, the exploitation of big data will settle down to become accepted business practice.

As data volumes continue to explode and new data sets are become open, "databots" will crawl our linked data, inferring relationships as definite or possible. This analysis is not real-time but, over time, builds up an increasingly accurate representation of our activities. Some of the relationships will require further investigation (a human element, perhaps, or maybe triggering a subsequent process). Potentially, the complexity of a query may be ascertained, together with the number of databots that are required, spawning instances and creating the appropriate cloud services to scale accordingly.

Add in some artificial intelligence and... well, the possibilities are either amazing, or frightening, depending on your point of view.

As the Internet becomes increasingly content-centric (cf. the communications-centric approach of the past), data will be stored in many separate locations owned by a multitude of organisations and individuals, spread across the globe (and even further). Some data will have been repurposed for new uses; in other cases information will be traded rather than basic data and that information repurposed to enable the extraction of new knowledge [2]. The provenance of the data will often be a factor in how trustworthy it is – and data marketplaces will increasingly be judged based on the quality of their data (as judged by user feedback) [1].

Several years ago, Sir Tim Berners-Lee wrote about what he called the "giant global graph" (in effect, linked open data) and he wrote [32]:

"In the long term vision, thinking in terms of the graph rather than the web is critical to us making best use of the mobile web, the zoo of wildy [sic.] differing devices which will give us access to the system. Then, when I book a flight it is the flight that interests me. Not the flight page on the travel site, or the flight page on the airline site, but the URI (issued by the airlines) of the flight itself. That's what I will bookmark. And whichever device I use to look up the bookmark, phone or office wall, it will access a situation-appropriate view of an integration of everything I know about that flight from different sources. The task of booking and taking the flight will involve many interactions. And all throughout them, that task and the flight will be primary things in my awareness, the websites involved will be secondary things, and the network and the devices tertiary."

In short, linked data (or information derived from it) will be at the heart of our computing interactions – no longer will we be concerned about connecting devices to access documents, but about the information that we need in order to make decisions through life.

Summary and conclusions

At the start of this paper, we drew comparisons with cloud computing in that big data is becoming massively hyped. Just as with "cloud", which is now approaching acceptance as a business model to the point that it has become the normal way to provide computing and application services, there will be a period of consolidation before we truly understand the application of "big data" solutions and "big data" becomes just "data" (i.e. business as usual). Even so, the volumes of data that we process, their variety in structure and the need to make timely decisions will lead to the creation of a new class of business systems.

These business systems include a number of capabilities that should be considered core to a "big data" solution, including:

- Data collection and detection
- Data management and integration
- Data analysis
- Information application, automation and visualisation
- Data exchange
- Development support and operational management
- Exploitation of cloud computing architectures

Because there is limited value that can be obtained from any one source, aggregation of multiple sources is key to unlocking the potential in our data; however an unfortunate side effect of the ability to analyse increasing volumes of data using commodity infrastructure and cloud services is the emergence of unstructured and semi-structured data which does not fit well in our traditional structured databases and so creates issues around data management.

With many solutions entering the marketplace it's all too easy to focus on the technology that allows processing of new data streams but simply focusing on Hadoop or other NoSQL technologies is not enough. It's important not to lose sight of the long-term goal of integrating many data sources to unlock even more potential in data – and the current technology landscape is a barrier to meeting business expectations which include:

- Greater accuracy (derived from larger data sets).
- Immediacy (near-real time data, from new data sources).
- Flexibility (not constrained by database structure).
- Better analytics (the ability to change the rules).

Linked data has the potential to provide a new architectural pattern for mapping and interconnecting, indexing and feeding real-time information from a variety of sources. Critically, it can be represented in human or machine-readable form and also allows new relationships to be inferred from existing data, creating new insights for further analysis. Linked data integrates *all* data – whether that's previously inaccessible big data that's got the IT industry in a buzz; structured data in traditional databases; or the increasing number of external (open) data stores.

For enterprises there are number of considerations (and some challenges) to address around data integrity, integration, data management, data replication, data quality, data storage and, critically, data security but none of these should be insurmountable. They do require careful planning though.

Once we've linked the myriad data stores at our disposal then we can generate new data from old, trade information, and extract new knowledge in support of more services, and even new business models, for example based around mash-ups, analytics, geospatial representation and augmented reality.

Exploiting linked data potentially offers massive opportunity in the integration of myriad data sources, both open and closed. With linked data acting as a broker, we have the ability to extract new data from old, creating insights that were previously unavailable, and facilitating exciting new scenarios for data processing.

Bibliography

1. **Dumbill, Edd.** What is big data? An introduction to the big data landscape. *O'Reilly Radar.* [Online] 11 January 2012. http://radar.oreilly.com/2012/01/what-is-big-data.html.

2. **Townsend, Eddie.** *UK Future Internet Stategy Group: Future Internet Report.* s.l. : Technology Strategy Board, 2011.

3. **Woo, Benjamin, et al.** IDC Worldwide Big Data Taxonomy. [Online] October 2011.

4. **Gens, Frank.** IDC 2012 Predictions: Competing for 2020. [Online] 2011. http://cdn.idc.com/research/Predictions12/Main/downloads/IDCTOP10Predictions2012.pdf.

5. **Sanderson, Mike.** Linked data for executives: building the business case. London : British Computer Society, 24 November 2011.

6. **Lawson, Stephen.** Nokia Siemens brings big data analytics to mobile carriers. *CIO Magazine.* [Online] 14 February 2012. http://www.cio.co.uk/news/3337309/nokia-siemens-brings-big-data-analytics-to-mobile-carriers/.

7. Everyone 'to be research patient', says David Cameron. *BBC News.* [Online] 5 December 2011. http://www.bbc.co.uk/news/uk-16026827.

8. **Best, Jo.** Big data: cheat sheet. *Silicon.com.* [Online] 19 December 2011. http://www.silicon.com/management/ceo-essentials/2011/12/19/big-data-cheat-sheet-39748353/.

9. **Manyika, James, et al.** Big data: The next frontier for innovation, competition, and productivity. *McKinsey & Company.* [Online] May 2011. http://www.mckinsey.com/Insights/MGI/Research/Technology_and_Innovation/Big_data_The_next_frontier_for_innovation.

10. *Information as Strategy.* **Raskino, Mark.** s.l. : Gartner, 2011.

11. **Hopkins, Brian and Evelson, Boris.** Forrester: Expand Your Digital Horizon with Big Data. [Online] 30 September 2011.

12. **Woodward, Alys, et al.** *IDC European Software Predictions.* s.l. : IDC, 2012.

13. SAND CDBMS 6 at dunnhumby - big data, big users, security, and analytics. *YouTube.* [Online] 15 September 2010. http://www.youtube.com/watch?v=7iN1bfqWxCk.

14. **Swabey, Pete.** Getting relevant. *Information Age.* [Online] 20 April 2007. http://www.information-age.com/channels/information-management/it-case-studies/277256/getting-relevant.thtml.

15. **Du Preez, Derek.** European Commission launches open data strategy. *Computing.* [Online] 12 December 2011. http://www.computing.co.uk/ctg/news/2131718/european-commission-launches-strategy-europe.

16. Open Data Institute to help drive innovation and growth. *Technology Strategy Board (InnovateUK).* [Online] 29 November 2011. http://www.innovateuk.org/_assets/0511/open%20data%20institute%2029nov11%20final%20(2).pdf.

17. About Data.gov. *Data.gov.* [Online] http://www.data.gov/about.

18. UK Government Licencing Framework. *The National Archives.* [Online] 2011. http://www.nationalarchives.gov.uk/information-management/uk-gov-licensing-framework.htm.

19. Open data measures in the Autumn Statement. *UK Cabinet Office.* [Online] 29 November 2011. http://www.cabinetoffice.gov.uk/news/open-data-measures-autumn-statement.

20. **Bradshaw, Paul.** How to be a data journalist. *The Guardian Data Blog.* [Online] 1 October 2010. http://www.guardian.co.uk/news/datablog/2010/oct/01/data-journalism-how-to-guide.

21. **Lorenz, Mirko.** Data-driven journalism: what is there to learn? *Slideshare.net.* [Online] June 2010. http://www.slideshare.net/mirkolorenz/datadriven-journalism-what-is-there-to-learn.

22. **Wang, R Ray.** *Twitter.* [Online] 19 January 2012. https://twitter.com/#!/rwang0/statuses/160140204873744385.

23. **Wu, Michael.** Searching and filtering big data: the 2 sides of the "relevance" coin. *Lithosphere.* [Online] February 2012. http://lithosphere.lithium.com/t5/Building-Community-the-Platform/Searching-and-Filtering-Big-Data-The-2-Sides-of-the-Relevance/ba-p/38074.

24. **Kim, Ryan.** RunKeeper builds a fitness network with Healh Graph API. *GigaOm.* [Online] 7 June 2011. http://gigaom.com/2011/06/07/runkeeper-builds-a-fitness-network-with-health-graph-api/.

25. **Taylor, Colleen.** Kaggle gets $11M to crowdsource big data jobs. *GigaOm.* [Online] 3 November 2011. http://gigaom.com/2011/11/03/kaggle-funding-max-levchin/.

26. **Barton, Robin.** Code of conduct: The relentless march of the algorithm. *The Independent.* [Online] 15 January 2012. http://www.independent.co.uk/news/business/analysis-and-features/code-of-conduct-the-relentless-march-of-the-algorithm-6288080.html.

27. **Treanor, Jill.** Ultra-fast trading blamed for 'flash crash'. *The Guardian.* [Online] 8 July 2011. http://www.guardian.co.uk/business/2011/jul/08/ultra-fast-trading-blamed-for-flash-crash.

28. **Acuna, Antonio.** Linked data for executives: building the business case. London : British Computer Society, 24 November 2011.

29. **Berners-Lee, Tim.** Talks: Tim Berners-Lee on the next web. *TED.* [Online] February 2009. http://www.ted.com/talks/tim_berners_lee_on_the_next_web.html.

30. Resource description framework. *W3C semantic web.* [Online] 2 October 2004. http://www.w3.org/RDF/.

31. **Lovinger, Rachel.** RDF and OWL: A simple overview of the building blocks of the semantic web. *Slideshare.* [Online] December 2007. http://www.slideshare.net/rlovinger/rdf-and-owl.

32. **Berners-Lee, Tim.** Giant Global Graph. *Massachusetts Institute of Technology Decentralised Information Group.* [Online] 21 November 2007. http://dig.csail.mit.edu/breadcrumbs/node/215.

33. **Herman, Ivan.** Web Ontology Language (OWL). *W3C Semantic Web.* [Online] 15 October 2007. http://www.w3.org/2004/OWL/.

34. **W3Schools.** RDF Tutorial. *W3Schools.* [Online] http://www.w3schools.com/rdf/.

35. **Menday, Roger.** A perspective on DaaS. s.l. : Fujitsu Laboratories of Europe Limited, 4 October 2011.

36. **Cyganiak, Richard and Jentzsch, Anja.** Linking Open Data cloud diagram. [Online] September 2011. http://lod-cloud.net/.

37. RDF Query Language. *Wikipedia.* [Online] http://en.wikipedia.org/wiki/RDF_query_language.

38. About data.gov.uk. *Data.gov.uk.* [Online] http://data.gov.uk/about.

39. Linked data. *Data.gov.uk.* [Online] http://data.gov.uk/linked-data.

40. **Ralmond, Yves, et al.** Case study: use of semantic web technologies on the BBC web sites. *W3C semantic web use cases and case studies.* [Online] January 2010. http://www.w3.org/2001/sw/sweo/public/UseCases/BBC/.

41. **Mendes, Pablo.** About DBpedia. *DBpedia.* [Online] 8 November 2011. http://dbpedia.org/About.

42. **Wallis, Richard.** WikiData - announcing Wikipedia's next bg thing. *Data Liberate.* [Online] 7 February 2012. http://dataliberate.com/2012/02/wikidata-announcing-wikipedias-next-big-thing/.

43. **Zaino, Jennifer.** The power is in the link. *semanticweb.com.* [Online] 6 January 2012. http://semanticweb.com/the-power-is-in-the-link_b25765.

44. **Serbu, Jared.** Navy struggles to find the way ahead on big data. *Federal News Radio.* [Online] 20 February 2012. http://www.federalnewsradio.com/?nid=412&sid=2754767.

45. **Menday, Roger, et al.** *Linked IT - the BigGraph Concept.* s.l. : Fujitsu Laboratories of Europe Limited, 2011.

46. **Lohr, Steve.** The age of big data. *The New York Times.* [Online] 11 February 2012. http://www.nytimes.com/2012/02/12/sunday-review/big-datas-impact-in-the-world.html?_r=1&pagewanted=all.

47. **Patil, DJ.** Building data science teams. *O'Reilly Radar.* [Online] 16 September 2011. http://radar.oreilly.com/2011/09/building-data-science-teams.html.

48. **Watters, Audrey.** Scraping, cleaning, and selling big data. *O'Reilly Radar.* [Online] 11 May 2011. http://radar.oreilly.com/2011/05/data-scraping-infochimps.html.

49. **DeWitt, David J.** Big data - what is the big deal? *Professional Association for SQL Server.* [Online] 14 October 2011. http://www.sqlpass.org/summit/2011/Live/LiveStreaming/LiveStreamingFriday.aspx.

50. **Rodriquez, Alex.** RESTful web services: the basics. *IBM DeveloperWorks.* [Online] 6 November 2008. https://www.ibm.com/developerworks/webservices/library/ws-restful/.

51. **Taylor, Chris.** Career of the future: data scientist. *Mashable.* [Online] 13 January 2012. http://mashable.com/2012/01/13/career-of-the-future-data-scientist-infographic/.

About the authors

Ian Mitchell, Chief Architect

As Fujitsu UK and Ireland's Chief Architect, Ian Mitchell runs the regional CTO Office where he is responsible for technology strategy and governance, innovation, thought leadership and sustainability. Ian has over 20 years in the IT industry, including roles in software development, Enterprise Architecture and as a major account level CTO. Ian's experience covers both IT and business architecture implementation, having delivered several large scale enterprise wide SOA solutions, making Ian well placed for providing cloud-related advice and guidance. Ian may be found on Twitter @ianmitchell2.

Mark Wilson, Strategy Manager

Mark Wilson is an analyst working within Fujitsu's UK and Ireland Office of the CTO, providing thought leadership both internally and to customers, shaping business and technology strategy. He has 17 years' experience of working in the IT industry, 12 of which have been with Fujitsu. Mark has a background in leading large IT infrastructure projects with customers in the UK, mainland Europe and Australia. He has a degree in Computer Studies from the University of Glamorgan. Mark may be found on Twitter @markwilsonit.

Acknowledgements

Thanks to the following who have contributed advice, knowledge and information to this document:

- Mike Dunn
- Vincent Hughes
- David Gentle
- Mark Locke
- Dr Roger Menday
- David Smith
- Martin Summerhayes
- Dr David Snelling
- Andrew Snowden
- Jon Wrennall

OFFICE of the CTO

Contact
FUJITSU
22 Baker Street, London, W1U 3BW
askfujitsu@uk.fujitsu.com
www.fujitsu.com/uk

REF: 3378

Fujitsu Services Limited. Registered in England no 96056.