published in the *International Journal of Information Retrieval Research,* Vol. 6 Issue 1, pages 52-64, IGI Global, 2016.

The Data Swarm – A Next Step for Distributed Data Analytics

Jeffrey Smith *and* Manjeet Rege
Graduate Programs in Software
University of St. Thomas
St. Paul, MN 55105

Background

First, we must understand the evolving nature of the current data analytic environment and what's driving the need for new approaches to analytics, as well as innovative applications of those approaches. The need to find ways to decentralize the analytic discipline wherever and whenever possible continues to grow.

The modern enterprise is inundated with data, as data volumes across the enterprise grow by 35%-50% every year. Gartner analysis shows that Corporate America will process more than 60 terabytes of data each year, which is around a thousand times more than a decade ago (Beath C., Becerra-Fernandez I., Ross J., & Short J. , 2012). Because of this dramatic increase in data, substantial investments are being made in technologies that service this data such as integration of data, business intelligence tools, and data warehousing infrastructure, as well as in the people needed to develop and support their implementation.

Business intelligence and analytics are at the heart of the user experience in data warehousing, and Gartner estimates spending on it was around $14.1 Billion in 2013 and that it will continue to grow at roughly 7% annually over the next few years (Sallam R. L., Tapadinhas J., Parenteau J., Yuen D., & Hostmann B., 2014). Gartner also estimates that the growing field of advanced analytics is currently around a $2 Billion market and that it will continue to increase rapidly as the tools become more user-friendly and the data more accessible (Herschel G., Linden A., & Kart L., 2014).

These investments have driven large-scale business improvements by supporting the decision making process and streamlining data architectures. However significant innovations in technology – such as data discovery and interactive visualizations, predictive analytics, in-memory computing, and big data – combined with a much greater variety of users and use cases, are increasing the need for a wholesale transformation of performance, people, process and platforms. The number of use cases that business analytics must support will continue to multiply, requiring multiple tools and approaches (Chandler, N., 2015).

Gartner estimates that well over 80% of the market is continuing to follow traditional data warehouse approaches as seen in Figure 1 (Edjlali R., & Beyer M. A., 2014). However, with the fast paced technological transformation and introduction of unstructured and nontraditional data, as well as new types of advanced analytics, it is also predicting that traditional data warehouse practices will be outdated by the end of 2018 (Beyer M. A., & Edjlali R., 2014).

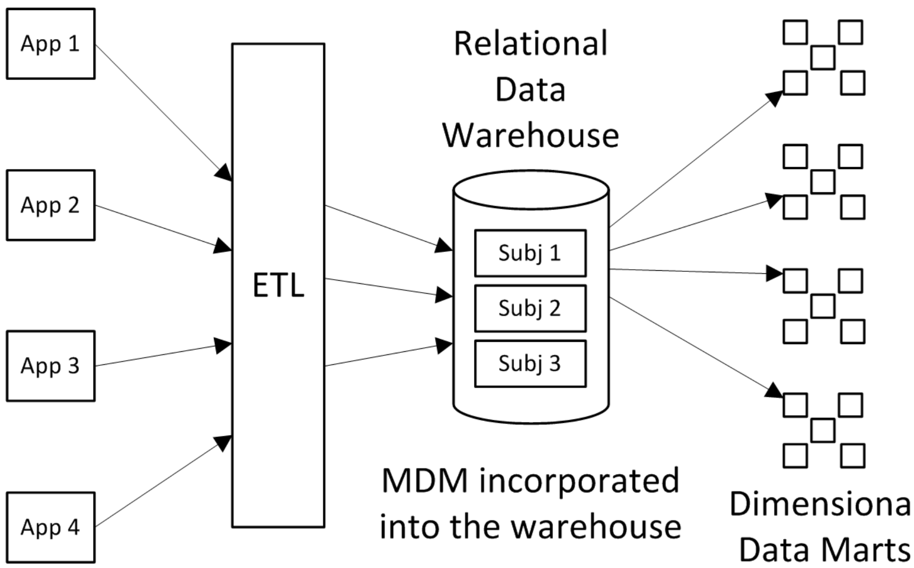

Figure 1 – A traditional data warehouse following the combined Inmon/Kimball models. (Adapted from Inmon W.H., 2015).

The death of the traditional data warehouse is imminent because, not only will data warehouses be expected to continue supporting reporting and traditional style business intelligence, they will also have to support integrated information for nearly incompatible service-level expectations in different analytic use cases, data provisioning for analytics embedded in operational applications, and hybrid transaction and analytical processing.

For the past five years Gartner has been pushing a concept, which they refer to as the Logical Data Warehouse (LDW), to address this analytic evolution. The LDW is focused on the concept that a data warehouse is principally an integrated data management platform that facilitates data asset consolidation and supports time variance in using the data – and is not specifically a physically centralized repository. Over the past five years the concept of using different technologies for these very different SLAs for data management for analytics has gain rapid acceptance in the industry and for many has even become preferable (Beyer M. A., 2014).

Figure 2 shows an example of the LDW in a typical corporate setting. Users throughout the enterprise can run their analytics from multiple places within the LDW such as in the operational marts, the ODS's, the data marts or even the cloud.

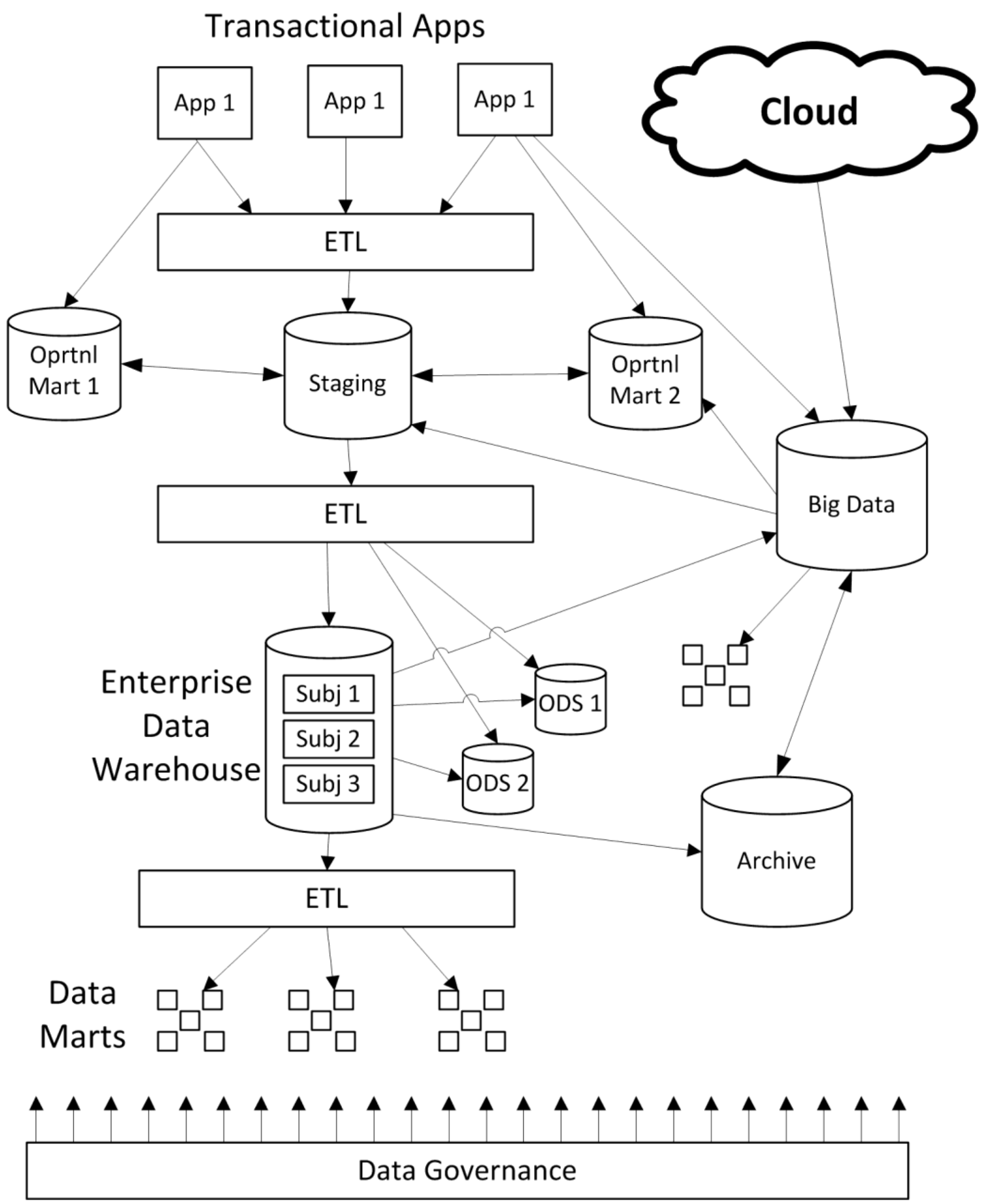

Figure 2 – A typical Logical Data Warehouse (LDW) enterprise model. Users can interact with it in multiple places such as the operation marts, ODS, data marts or even the cloud, to derive their analytics.

This need to take a more logical approach to data management and analytics can be addressed from two different directions. One approach focuses on the organization and the other approach focuses on the technology.

The organizational approach attempts to find the right balance between centralized and decentralized business intelligence and analytics teams. An overly centralized team can't deliver the domain expertise and responsiveness most organizations require. While the centralized team does a good job in creating consistency and governance across certain key subject areas, it creates a bottleneck where most users are waiting too long to have their requirements met. On the other hand, an overly decentralized team delivers plenty of domain expertise, agility, and responsiveness, but struggles to deliver consistency across its information sources and analytic models and best practices. A two-tiered organizational model, with a single centralized team working collaboratively with a collection of decentralized teams distributed throughout the enterprise, is the ideal model (Schlegel K., Milbury O., Buytendijk F., & Sommer D., 2014).

The centralized analytics team's greatest value is to provide the necessary data governance, best practices, and oversight to ensure that the company is efficiently and consistently performing its analytics. For example, there should be a standard way of calculating Cost of Goods Sold (COGS) or Gross Margin for anyone using those metrics throughout the company. Additionally, if there is a large amount of low level data being stored, the centralized analytics team would ensure that it's stored in just one place, rather than that same data being stored as multiple copies in multiple places. It would also keep track of what the true system of record is for any particular piece of data.

The decentralized analysts provide the greatest value by quickly identifying and driving business context and value to the data. This close proximity with the business and their analytic needs helps ensure the analysis is relevant and that it's rapidly adopted into business processes and activities. This also helps ensure knowledge and expertise are maintained as close as possible to the where they're actually used (Duncan A.D., 2015).

The second approach to more logical data management and analytics is to leverage advances in technology. With the rise of Hybrid Transaction Analytic Processing (HTAP) architectures, applications can finally analyze "live" data while it's being created or updated. HTAP could potentially redefine the way some business processes are executed, as real-time advanced analytics become an integral part of the process itself, rather than a separate activity performed after the fact. It has become a key enabling architecture for intelligent business operations (Pezzini M., Feinberg D., Rayner N., & Edjlali R., 2014).

To date, one of the primary technological enablers of HTAP has been In-Memory Computing (IMC). IMC is a computing style where applications depend on all the data necessary for processing to be located in the main memory of their computing environment, which is enabled by the ever-dropping memory cost and by specific application infrastructure software and cloud services. This means that there's negligible data access latency which in turn can allow processing of very large volumes of in-memory data in seconds or even sub-seconds. This enables organizations to gather a much deeper and much more responsive understanding of what's going on in their business compared with using conventional architectures (Biscotti F., Schulte W.R., Feinberg D., Schlegel K., Edjlali R., Guttridge K., Pezzini M., & Heudecker N., 2014).

However these new technological approaches can be quite complex and are still rather early in their life cycle, so there continues to be opportunities to find novel ways of addressing the problem. This paper presents a simpler approach distributed analytics by utilizing another technological concept – swarm intelligence – to create what we like to refer to as a "data swarm". While this concept of distributed analytics may not be new, this simplistic application of it in the retail setting is unique.

A New Approach

First, we'll provide a brief summary of the concept of swarm intelligence. Inspired by biology, it approaches decentralized systems of basic, simple agents and leverages an approach of self-organization to solve complex problems either directly or indirectly (Bonabeau E., & Theraulaz G., 2000). This happens because each individual agent follows a very basic set of rules and patterns, that when viewed collectively, allows them to accomplish some very big things.

For example, each ant in a colony would seem to have its own agenda, however at the same time the group functions as if it's organized. Each ant seamlessly integrates its own activities without any supervision. This self-organized cooperation within the colony has been verified multiple times by scientists studying this amazing behavior: in very simple terms the coordination arises from individual actions. Although simple in practice (one ant following the pheromone trail of a second ant), these individual interactions help them solve seemingly difficult challenges (such as figuring out the most efficient route to possible food sources). Thus, the organized behavior that comes from a group of ants, termites or other types of these insects is what we refer to as "swarm intelligence" (Bonabeau E., & Theraulaz G., 2000).

Figure 3 shows an illustration of how this typically works in a colony of ants. When ants set out in search of food they leave a trail of pheromones behind. An ant taking the shortest route to the food returns the most quickly and then immediately sets back out on the same short route to get a second helping of food. This now adds even more pheromone to the trail which in turn causes other ants to be drawn to it, who also add their own pheromones to the mix. As the pheromone concentration rapidly increases, it becomes so powerful that all other trails are quickly abandoned for this one.

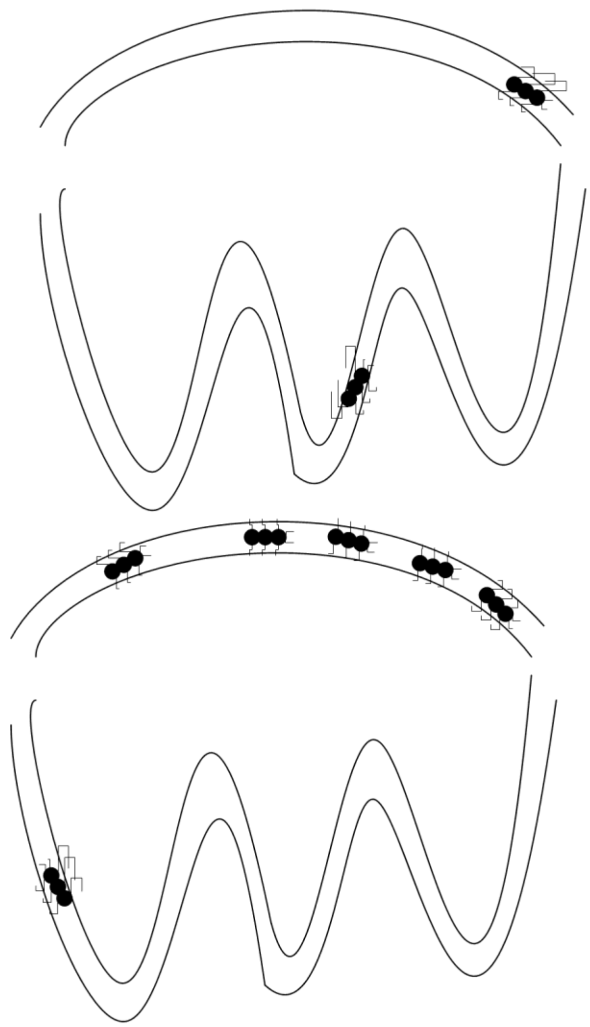

Figure 3 – In the top image the shorter path allows more ants to complete the trip while leaving a trail of pheromones. Since the shorter path quickly builds up a much stronger amount of pheromones it becomes the preferred path for all the ants as seen in the bottom image. (Adapted from Krink, T., 2001).

Scientists first began researching the field of swarm intelligence in the early 1950s. They were fascinated by the way these animals and insects coordinated together and communicated so intelligently and they wanted to understand how they did it. One of the first researchers in this area was French biologist Pierre-Paul Grassé. He focused on the organized approaches used by termites to build their nests. (Grasse P., 1984). What he found was that even though each termite had a very limited neural capacity, when they each followed their basic set of very simple and limited tasks, they collectively created colonies of hundreds of thousands of termites that were self-sustaining and able to construct extremely elaborate structures.

A very practical application of swarm intelligence to a modern day problem resulted in millions of dollars of savings for the United Parcel Service (UPS) Corporation. By gathering the accumulated experience of hundreds of drivers and the process they used to learn from each other in a similar way to ant colony routing, they realized that more right turns would really save time. The rationale behind this was fairly obvious. Nearly every left turn involves having to deal with an oncoming lane of traffic. This means that when you're making a left turn you're often guaranteed to have to wait for oncoming traffic to clear. Additionally, your risk of being involved in an accident also increases. When UPS started applying the concept of more right turns across all their routes they saved three million gallons of gas within the first year (Fisher L., 2009). This gas savings was accomplished by simply training each driver to attempt to make more right turns whenever possible.

These same simple concepts of swarm intelligence can equally be applied to the world of data analytics. Wherever you have a large amount of data collectors, all organized and structured the same way, there's a certain amount of analytics that can be performed locally at each data collector to help improve its business processes, without first having to come to a centralized location to be processed. Since each data collector is organized and structured the same way, the analytic code and tools needed to do this, can be easily replicated and maintained at all locations with minimal cost. If the business processes at each data collector can get just a little more effective by leveraging these local analytics, the aggregate value of the data collector's efficiencies can quickly become quite large.

The other advantage to this approach is that we've minimized data latency in the model, so that the local data collector is able to analyze and understand its information more quickly, rather than waiting for the data to be sent to a centralized location, analyzed there, and then the results eventually returned to it. Additionally, if the remote data collector were to lose network connectivity entirely, or become isolated in some other way, it would still be able to function autonomously, and continue to provide its collective value, based on its own internally generated data and local analytics – it removes the centralized dependency entirely.

The Retail Problem

We've chosen to use a large "bricks and mortar" style retail model to illustrate these concepts. In this example the retailer has around 1,500 stores spread across a similar geographical region. Each store has a Store Manager and an average of eight sales employees working at any given time who are able to sell products to customers through six different Point of Sale (POS) systems. The POS systems are connected to a store application server and store database server, which is then connected through the cloud to a central corporate headquarters. You can see a layout of this in Figure 4.

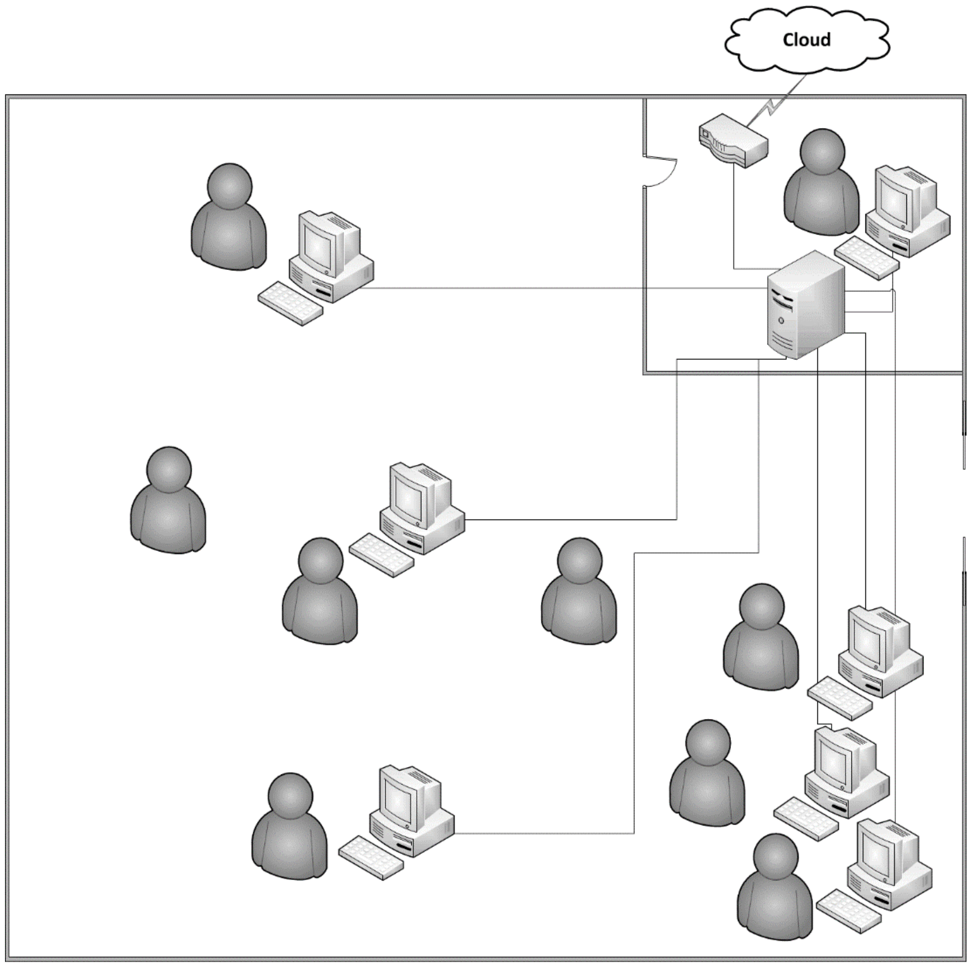

Figure 4 – Store Layout of our example. Note the different POS systems and employees spread out throughout the store all connected to a central server in an office area.

Typically, each POS transaction is sent real time to the store's application server at the conclusion of each transaction. A job on the server consolidates all the transactions and then sends a batch of them up to the corporate headquarters data center every fifteen to thirty minutes for processing.

The Store Manager needs to know how much each employee is selling and how the store is performing overall throughout the day. The Store Manager uses that analysis to provide continuous, real time feedback and coaching to the employees, as well as to make adjustments to store selling strategy. He/she typically needs this information no less than once an hour throughout the sales day.

In the traditional data warehouse model this analysis would be accomplished by having each of the 1,500 stores send their transaction details in batches every fifteen to thirty minutes to a server in the corporate data center where it would be staged in a landing zone, and initial processing done on the data to make it ready for further use. Next the data would be moved into either an operational data store or a data warehouse for general storage and analytical use. A portion of that data would then typically be moved into a dimensional model in a retail data mart that supports analysis of the data at various data hierarchy levels needed by the retail organization. Finally, this retail data mart would be made available to all the consumers, including all 1,500 Store Managers, through some sort of web client or dashboard that they would typically access through the cloud. You can see an example of this flow in Figure 5.

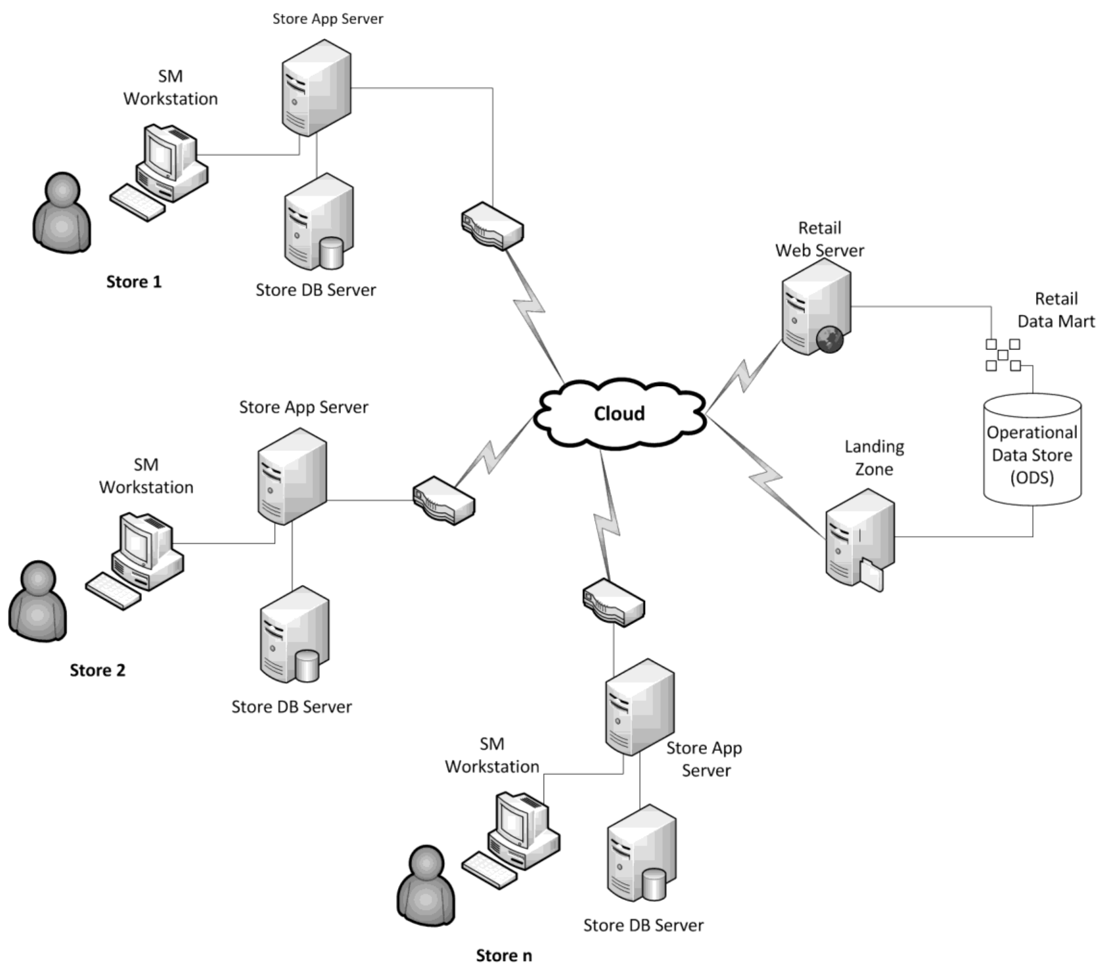

Figure 5 – Existing retail reporting architecture which sends the data all the way to a centralized location for processing and preparation before returning a set of reports back to each store through the cloud.

However inherent in this architecture are several fundamental problems. First, even though all the data that Store Managers need, to be able to assess their employee and overall store performance, exists within the walls of the store, the existing model requires that the data be shipped over the network, along with the data from 1,499 other stores, to a central place to be processed and prepared for viewing before he/she can see the results. This centralized processing requirement puts a large load on the network, and can create latency in the system, which can inhibit the end user's timely ability to view this very valuable information. Additionally, if there are any issues with the network or any device outside the walls of the store, the data would be not complete the full circle and the Store Manager would be unable to view the information at all.

The Solution

The solution to these problems can be derived from the application of the swarm intelligence concepts we talked about previously. If we consider that every store is organized exactly the same, with the same applications, and same basic IT infrastructure, then each store would be comparable to one of the ants in the swarm intelligence example. The store has a simple job to do, and that is to sell. In order for the store to do that as effectively as possible the Store Manager needs some basic metrics letting him/her know how well each employee is performing that task throughout the day. The Store Manager can do that totally independent of all the other stores, and can be self-directed in the methods and processes he/she uses to increase those sales. If each store is focusing on maximizing its individual sales, then collectively they can have a very large impact on the company similar to a colony of ants. However, the key to applying a concept like swarm intelligence, is to create something that is extremely simple, exactly the same across all stores, and low cost, with little or no additional licensing fees.

In order to accomplish this, we're suggesting that a simple data analytics function be created for each store, to enable local viewing of some basic analytics, independently and throughout the day. The approach takes a handpicked slice of the POS transaction data feed that is being sent up to corporate and retains it locally so it can be analyzed in a simple dashboard within the store. This dashboard would be available for the Store Manager and the rest of the store's employees to view throughout the day. This dashboard would also continue to be available to throughout the store, if anything were to happen to the store's connectivity, such as an offline situation, the store would be able to remain self-sufficient and operational.

The following is an example of the details of how this could work. For simplicity's sake, we are restricting this example to just sale transactions vs. return, exchange, voided or any other type of transactions. A typical POS sales transaction contains all the information related to that transaction. It would include details about every item that was purchased including SKU, product name, product category, quantity, price, and discounts (if any). It would also include transactional information about when, where, and by whom it happened, such as employee ID of the person ringing up the transaction, POS ID, time and date, a unique transaction ID. It also would include payment information, such as credit/debit card numbers, check information, coupons/discounts, tax applied and any other payment related information. Finally, it could also include customer information such as a customer ID, or perhaps a membership or loyalty number.

Some of the information in the POS transaction feed would of course be very sensitive and require extra protection, but for the types of reporting that would typically be needed within a store, any sensitive information could be left out of the solution, which would keep it simpler and less expensive. The majority of the data actually needed for this local dashboard would be at the basic transaction level of information. There would be some additional high level item information like quantity of items in the sale and pricing, as well as the total sale amount, total discounts and tax. This subset of information would provide all the data necessary to quantify individual employee and overall store performance and because none of the information would be sensitive, no extra data security would be necessary.

Next we'll illustrate the source data and a typical data model that would support this local dashboard. POS systems can pass their transaction information in a variety of ways, but for the purpose of this example we'll keep it simple, and say that the POS passes its information as XML. Figure 6 is an example of what a single transaction might look like in XML (We're intentionally keeping sensitive data out of our example).

```xml
<Sale>
  <Tran Dtls>
    <Store Nbr>231</Store Nbr>
    <POS Nbr>3</POS Nbr>
    <Tran Nbr>000123</Tran Nbr>
    <Tran DTTM>2015032610103262</Tran DTTM>
    <Emp Nbr>234657</Emp Nbr>
    <Subtotal>77.36</Subtotal>
    <Tax>5.23</Tax>
    <Total>81.59</Total>
  </Tran Dtls>
  <Basket>
    <Item>
      <SKU>23416598723</SKU>
      <Price>17.45</Price>
      <Qty>1</Qty>
    </Item>
    <Item>
      <SKU>11756883265</SKU>
      <Price>39.99</Price>
      <Qty>1</Qty>
    </Item>
    <Item>
      <SKU>85431267999</SKU>
      <Price>4.98</Price>
      <Qty>4</Qty>
    </Item>
  </Basket>
  <Payments>
    <Tender>
      <Tender Type>2</Tender Type>
      <Tender Amt>60.00</Tender Amt>
    </Tender>
    <Tender>
      <Tender Type>1</Tender Type>
      <Tender Amt>21.59</Tender Amt>
    </Tender>
    <Discounts>
      <Discount Type>1</Discount Type>
      <Discount Amt>1.00</Discount Amt>
    </Discounts>
  </Payments>
</Sale>
```


Figure 6 – Sample XML of a POS sale transaction (sensitive data excluded).

As mentioned previously, we do not need all this information, so a subset of the information is parsed out and inserted into a small data mart. A sample model could look something like the simple star schema you see in Figure 7.

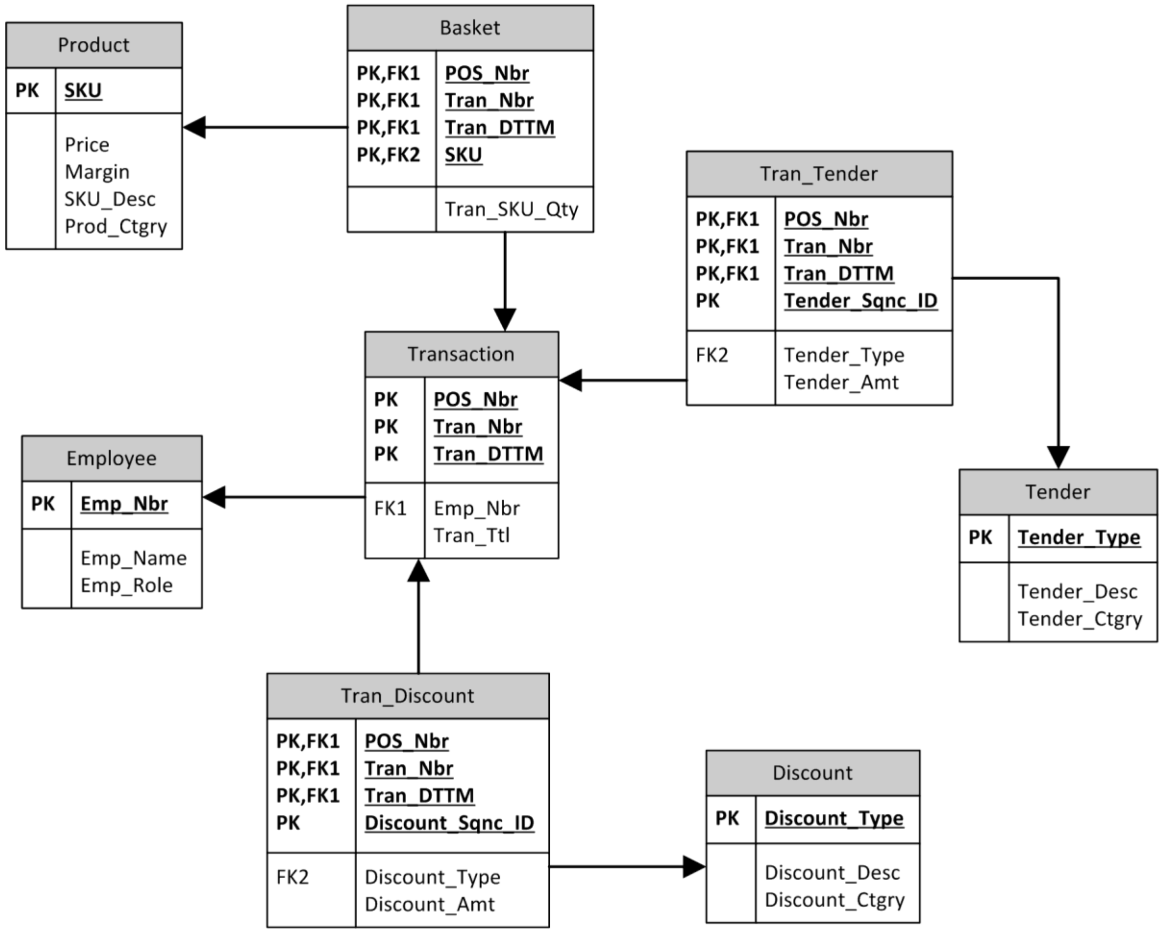

Figure 7 – Sample data model of a simple star schema data mart fed with fact data from the POS transaction feed near real time and dimensional data from the corporate data warehouse by a nightly batch process.

As you see in this schema, we have created an extremely simple data mart in a star schema. The dimensional data (Employee, Product, Tender, Discount) would most likely be updated on a nightly basis with batch jobs coming from a central place, such as a corporate data warehouse, while the fact tables would be populated near real time from the POS XML feeds.

This solution is not intended to drive long term strategic analytics but instead to drive short term tactical analytics. Thus, the amount of data retained would be limited to between two days' and a week's worth of transactions. This would allow the store to be able to do day over day comparisons or trending for the past week, while at the same time keeping the size of the database small and the queries fast. Anything greater than a week's worth of data would start to become unwieldy and break the primary rule of keeping this model very simple. The solution is meant to provide simple, tactical, and operational analytics to drive autonomous real time behavior and processes.

Once the data has been collected and is available locally, the Store Manager needs a way to interact with it. Visualization tools such as Tableau or a mainstream business intelligence client would work, but would also require training and licensing, which could be prohibitive because of the additional cost and complexity. Since all that's really needed is some simple graphs and tables, the simplest solution would be to make those available through a web browser leveraging a language such as "R" to create and present the results. It would require some initial IT coding experience to create them but, once created, the code could be implemented across all the store servers for virtually no additional cost.

Alternatively these tables and graphs could be created using common, readily available tools such as MS Excel, with the tables and graphs referencing the small data mart we described above. This has the added advantage of presenting the data in a tool that most business people are familiar with, and from which they could perform additional ad hoc analysis (if data governance rules allowed.)

We used MS Excel to produce Figure 8 and Tables 1 & 2 below. They are just a few examples of the types of analysis that could be done from a simple local data mart and are easily created and supported at very low cost by leveraging infrastructure already in place. After the model has been created in one store it can be easily replicated across the entire chain with little to no additional development. The examples would be used by both the Store Manager and the individual sales associates and are simple and easy to understand for even high school level employees.

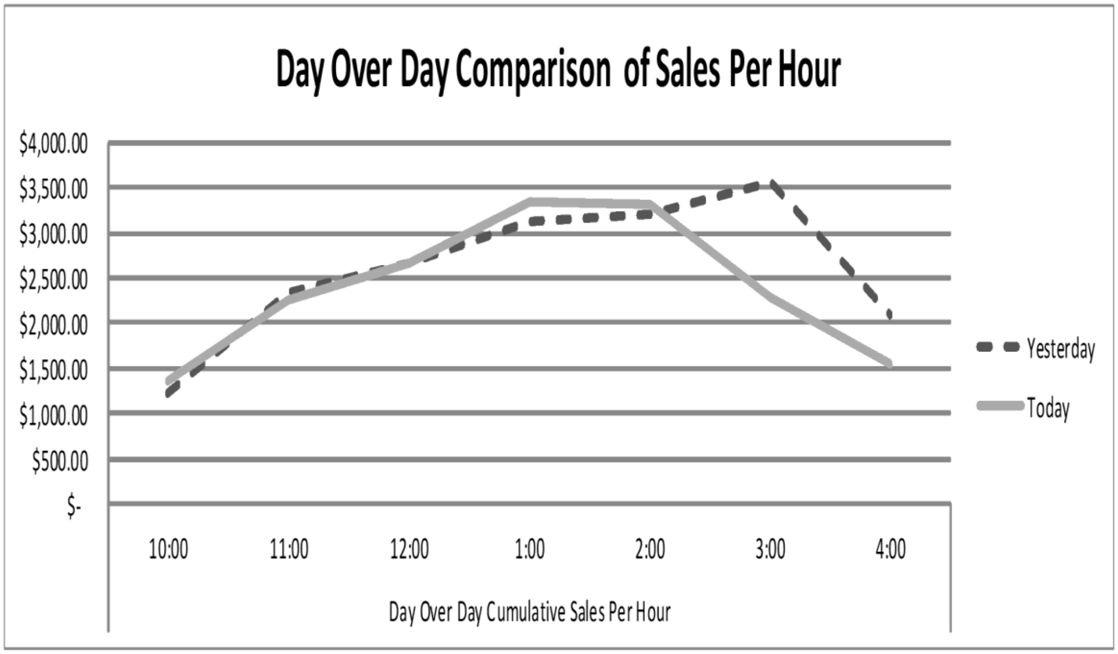

Figure 8 – Day-over-day sales comparison line graph by hour.

Figure 8 provides the store with a comparison of sales between yesterday and today. He/she can run it every hour to see how they're trending for the day. This could also be easily adapted to do a comparison against specific targets or goals that could be fed to the store data mart from corporate headquarters each night or once a week.

Table 1. Sales per hour by employee.

Today's Sales Per Hour by Employee													
Emp	10:00	11:00	12:00	1:00	2:00	3:00	4:00	5:00	6:00	7:00	8:00	9:00	Total
Kris	$367.23	$456.77	$532.12	$300.88	$311.23	$345.66	$378.66	-	-	-	-	-	$2,692.55
Joe	$321.77	$345.67	$412.45	$556.99	$234.54	$345.87	$321.77	-	-	-	-	-	$2,539.06
Mary	$334.22	$455.32	$332.41	$346.76	$465.43	$435.22	$112.31	-	-	-	-	-	$2,481.68
Sue	$99.34	$321.64	$222.54	$354.23	$678.99	$432.12	-	-	-	-	-	-	$2,108.86
Andy	$234.44	$321.45	$113.67	$453.82	$321.11	243.22	$231.56	-	-	-	-	-	$1,919.27
Carl	-	$345.66	$324.55	$455.66	$345.22	231.77	$104.55	-	-	-	-	-	$1,807.41
Sally	-	-	$456.77	$521.34	$501.23	-	$236.75	-	-	-	-	-	$1,716.09
Fred	-	-	$263.85	$354.21	$444.69	$243.61	$167.45	-	-	-	-	-	$1,473.81
Total	$1,357.00	$2,246.51	$2,658.36	$3,343.89	$3,302.44	$2,277.48	$1,553.05	-	-	-	-	-	$16,738.73

Table 1 provides the Store Manager with the sales generated by each employee on an hourly basis. This allows everyone in the store to see who's selling the most and who's selling the least, and where the Store Manager may need to provide extra coaching or other help. It can also be used as a motivational tool by posting it or circulating it in a place where all employees can see how they compare to each other and drive some friendly competition between them. They can see how they're doing from hour to hour, make adjustments, and then check again to see immediate results. It helps each employee immediately see the cause and effect of their actions.

published in the *International Journal of Information Retrieval Research,* **Vol. 6 Issue 1, pages 52-64, IGI Global, 2016**.

Table 2. Employee productivity report by hour and by product category including total sales, quantity of items, and even profit generated.

Productivity Report for Today - Kris														
Product Category	Employee	10:00	11:00	12:00	1:00	2:00	3:00	4:00	5:00	6:00	7:00	8:00	9:00	Total
001	Ttl Sales	$122.45	$210.33	$198.34	$130.68	$99.77	$110.79	$132.66	-	-	-	-	-	$1,005.02
	Amt Sold	5	12	13	7	5	6	8	-	-	-	-	-	56
	Profit	$23.45	$67.88	$48.77	$37.61	$15.69	$20.14	$45.39	-	-	-	-	-	$258.93
002	Ttl Sales	$99.34	$67.88	$110.23	$53.34	$72.31	$89.34	$65.15	-	-	-	-	-	$557.59
	Amt Sold	2	4	5	3	6	8	4	-	-	-	-	-	32
	Profit	$45.66	$31.14	$56.70	$19.99	$24.56	$39.99	$28.82	-	-	-	-	-	$246.86
003	Ttl Sales	-	$34.56	$23.12	-	-	$14.32	$49.37	-	-	-	-	-	$121.37
	Amt Sold	-	3	3	-	-	2	6	-	-	-	-	-	14
	Profit	-	$7.65	$4.56	-	-	$2.11	$8.96	-	-	-	-	-	$23.28
004	Ttl Sales	$12.45	$29.68	$46.11	$4.53	$8.77	-	$32.14	-	-	-	-	-	$133.68
	Amt Sold	1	3	7	1	2	-	5	-	-	-	-	-	19
	Profit	$1.11	$3.91	$5.16	$0.34	$1.12	-	$4.81	-	-	-	-	-	$16.45
005	Ttl Sales	$132.99	$114.32	$154.32	$112.33	$130.38	$131.21	$99.34	-	-	-	-	-	$874.89
	Amt Sold	23	18	34	23	25	31	21	-	-	-	-	-	175
	Profit	$56.77	$46.76	$76.12	$56.76	$66.23	$67.18	$45.33	-	-	-	-	-	$415.15
Total	Ttl Sales	$376.23	$456.77	$532.12	$300.88	$311.23	$345.66	$378.66	-	-	-	-	-	$2,692.55
	Amt Sold	31	40	62	34	38	47	44	-	-	-	-	-	296
	Profit	$126.99	$157.34	$191.31	$114.70	$107.60	$129.42	$133.31	-	-	-	-	-	$960.67
	Avg Sls	$11.85	$11.42	$8.58	$8.85	$8.19	$7.35	$8.61	-	-	-	-	-	$9.10
	Avg Profit	$4.10	$3.93	$3.09	$3.37	$2.83	$2.75	$3.03	-	-	-	-	-	$3.25

Table 2 provides the Store Manager with a way to assess the detailed performance of an individual employee and to help with their coaching. They could use it to demonstrate the value in selling higher margin products or the different selling behaviors at different times of the day.

With real time data so readily available, it makes it very easy for the Store Manager to create healthy competition between the employees. It also allows them the ability to illustrate how selling a larger number of items won't necessarily have as positive an effect as selling higher margin items or at least higher profit items. Additionally, as in our previous ant pheromone example, a Store Manager can easily examine the most effective employees, identify what they're doing that works so well, and then promote those same techniques with the rest of their employees, so that their collective impact becomes even greater. With the help of this detailed reporting each sales person's strategy should improve, which in turn will lead to greater overall store profitability.

Additionally the store is able to get this set of analytics immediately as it happens, rather than waiting for it to go back to a central corporate environment to first be processed and eventually delivered. Also, if the store becomes offline due to a network cable being cut out in the parking lot or a corporate server going down, it can continue to view its analytics without issue.

Conclusion

This paper was intended to provide an illustration of how the application of swarm intelligence concepts to the world of data analytics can provide a simple low cost alternative for distributed analytics in a retail setting. The model is simple to develop and implement, it offloads local store analytic needs from the retailer's burgeoning enterprise reporting environment, and provides solutions to the latency and potential network failure points inherent in the enterprise model.

The model can be easily applied to any environment that has a large number of data collectors, which all use the same business/systems model such as a call center, in which the call center supervisors could track call center agent productivity and performance metrics near real time, without impacting their operational systems. They would just create a similar simple data mart populated from the transactional system feeds and combine it with dimensional data from the data warehouse.

As the data warehouse environment continues to evolve to a more distributed model and the Logical Data Warehouse concept continues to mature, the need to find low cost and easily managed analytic enablers also increases. This is certainly not the first time, nor will it be the last, where examples from the natural world can lead us to new and refreshing ideas on how to solve our technological challenges. The concept of swarm intelligence can be very useful in the appropriate setting. It's elegant in its simplicity, yet powerful in its impact. Our "Data Swarm" attempts to illustrate those benefits by offloading a few of the ever-increasing analytic use cases from the more complex and costly traditional retail data warehousing and analytic infrastructure.

Bibliography

Beath C., Becerra-Fernandez I., Ross J., & Short J. (2012). Finding Value in the Information Explosion. *MIT Sloan Management Review.*

Beyer M. A. (2014). Avoid a Big Data Warehouse Mistake by Evolving to the Logical Data Warehouse Now. *Gartner*.

Beyer M. A., & Edjlali R. (2014). The Data Warehouse DBMS Market's 'Big' Shift. *Gartner*.

Biscotti F., Schulte W.R., Feinberg D., Schlegel K., Edjlali R., Guttridge K., Pezzini M., & Heudecker N. (2014). Market Guide for In-Memory Computing Technologies. *Gartner*.

Bonabeau E., & Theraulaz G. (2000). Swarm Smarts. *Scientific Am*erican, Mar, 72–79.

Chandler, N. (2015). Agenda Overview for Analytics, Business Intelligence and Performance Management, 2015. *Gartner*.

Duncan A.D. (2015). Business Intelligence Teams Need to Change With the Times. *Gartner*.

Edjlali R., & Beyer M. A. (2014). Critical Capabilities for Data Warehouse Database Management Systems. *Gartner*.

Fisher L. (2009). *The Perfect Swarm: The Science of Complexity in Everyday Life*. New York, NY: Basic Books.

Grasse P. (1984). Termitologia, *Tome II, Fondation des Sociétés*.

Herschel G., Linden A., & Kart L. (2014). Magic Quadrant for Advanced Analytics Platforms. *Gartner*.

Inmon W.H. (2015). DW-2.0-Architecture.pdf. Retrieved January 10, 2015, from http://www.lltcorp.com/sites/default/files/DW-2.0-Architecture.pdf

Krink, T. (2001). "Swarm Intelligence – Introduction" EVALife Group, Dept. of Comp Science, University of Aarhus. Retrieved April 5, 2015, from http://faculty.washington.edu/paymana/swarm/krink_01.pdf

Pezzini M., Feinberg D., Rayner N., & Edjlali R. (2014). Hybrid Transaction/Analytical Processing Will Foster Opportunities for Dramatic Business Innovation. *Gartner*.

Sallam R. L., Tapadinhas J., Parenteau J., Yuen D., & Hostmann B. (2014). Magic Quadrant for Business Intelligence and Analytics Platforms. *Gartner*.

Schlegel K., Milbury O., Buytendijk F., & Sommer D. (2014). Create a Centralized and Decentralized Organizational Model for Business Intelligence. *Gartner*.