OPERATIONAL INTELLIGENCE PERFORMANCE INFERENCING ACROSS BUSINESS PROCESSES

Rajeev Kaula

Department of Information Technology and Cybersecurity, College of Business, Missouri State University, Springfield, MO, USA

ABSTRACT

Business process intelligence improves operational efficiency that is essential for achieving business objectives, besides facilitating competitive advantage. As an organization is a collection of business processes, operations in one business process do influence or have relationship with other business processes. Consequently, from an operational intelligence standpoint, insights from one business process may have their genesis or implications in the performance of some other business process. This paper outlines a framework to sequence insights in the form of performance inferencing across multiple business processes. The framework logically sequences insights across business processes in the form of business rules. The paper illustrates the concepts through a prototype that is implemented in Oracle's PL/SQL language.

KEYWORDS

Business Intelligence, Process Intelligence, Business Process, Oracle, PL/SQL

1. INTRODUCTION

Business intelligence (BI) is a set of techniques that transform data into information to generate insights on business operations and competitive environment [11,12,35,41]. While the role of BI in discovering new business opportunities has gained a lot of attention [10,34,48], the utilization of its concepts to enhance business process insights through operational intelligence is evolving [5,15,16,20,22,24,27,28,29,31,32,38,39,42]. As organizations operate through inter-connected business processes, insights into their process performance through operational intelligence is essential to achieve business objectives, besides facilitating competitive advantage.

The traditional approach in business intelligence is to first model data in a data warehouse in the form of multi-dimensional models through an analysis of business operations involving business activities or business processes [36]. Thereafter, BI generates insights through online analytical processing (OLAP) analytics with multi-dimensional models in the form of star schema or its variants [1,25,26,43,30,49]. Such analytics provide information on what combination of dimension factors are associated with various measure values or its aggregations. Even though OLAP analytics are important as it allows an organization to make sense of data by providing insights into business process operations, such insights are essentially a snapshot on some aspect of these operations.

As an organization is a collection of business processes, operations in one business process do impact or have relationship with other business processes. Consequently, from an operational intelligence standpoint, insights from one business process may have their genesis or implications in the performance of some other business process. This can become evident through a logical sequencing of individual insights across business processes.

DOI : 10.5121/ijdms.2020.12101

For example, consider three business processes sales, customer service, and shipping that often exist in many businesses. Let's say the sales business process analytics generates an insight that indicates that during the third quarter, sales units are below the success metric in the eastern region primarily because sales of product Z dropped in the eastern region. Separately, the customer service business process analytics generates an insight that customer complaints have increased beyond a minimum threshold for product Z also in the third quarter. Separately too, the shipping business process analytics generates an insight that during the third quarter late deliveries went up for product Z. By itself these individual insights have limited scope. So, if shipping recognizes late deliveries of product Z, they may not fully be aware that customers are complaining about product Z due to late deliveries. Besides, customer complaints on one product may create a negative impression about other company product or services. Similarly, if customer service notices increased product Z complaints it may pass it on to sales or other business process, but its impact on overall company sales may not be obvious till sales analytic insights emerges. But if these individual business process insights are chained or sequenced the scope of the problem affecting business performance becomes much clearer.

One way to logically sequence individual insights across business processes is to (i) identify and standardize on dimension names across business processes, (ii) facilitate linking of similar dimensions across business processes during analysis, and (iii) develop a framework to logically sequence or chain the insights across individual business processes for deeper inferencing.
Identification of dimensions within a business process can be facilitated through a dimension flow model [23] which aligns business process activities to dimensional information. This facilitates closer mapping of analytics and its inferencing to a business process. Moreover, as business process activities are flow-oriented, modeling of dimensional information in a way that reconciles with the fluidity in process operations is essential.

Linking of similar dimensions across business processes can be facilitated through an enterprise wide dimension dictionary. Such dictionary can then be utilized by individual business process analytics to invoke other business processes analytics dealing with similar dimensions.

Logical sequencing of insights can be expressed through the business rules concept [18,37]. Business rules are typically expressed declaratively in condition-action terminology represented as IF condition THEN action format. A condition is some constraint, while the action clause reflects the decision or advice. From a business intelligence perspective business rules can also be utilized to express meaningful insights from OLAP analytics like specification of purposeful key performance indicators (KPIs), or suggest problem remedies [6,21,22,24]. Such business intelligence based business rules are referred in the paper as analytic business rules. Below is an example of an analytic business rule that describes a set of dimension factors that influence Win probability success factor in a Lead to Forecast business process.

```
IF      Party Type = Organization AND
        Sales Channel = Indirect AND
        Contact Role = Functional User AND
        Product Category = Desktop
THEN Win Probability > 70
```

There have been attempts at operational intelligence in the form of process monitoring, process analysis, process discovery, conformance checking, prediction and optimizations [9,17]. Besides, utilization of business rules for business process intelligence has also been explored [4,13,22,33]. However, these approaches either tie business rules to measures that are defined a priori through existing policies without much emphasis on database analysis or outline business rules for specific performance metrics. Technically OLAP analytics through constellation schema can pool in dimensions across one or more business areas. But when constellation schema is utilized it may be difficult to know which business process or its activity is involved. Constellation schema in general is business process agnostic.

This paper in nutshell will outline a framework to accomplish performance inferencing across multiple business processes. The framework logically sequences insights in the form of business rules. The paper illustrates the concepts through a prototype that is implemented in Oracle's PL/SQL language. Relevant operational intelligence research and dimension flow modeling is reviewed next. This is followed by an explanation of the framework structure and components. The paper concludes with an Oracle based prototype that illustrates the implementation of the framework.

2. RELATED WORK

Operational intelligence analyzes business processes to ensure that operational workflow is efficient and consistent with their stated objectives. The goal is to optimize such processes for successful performance. There have been four approaches towards the utilization of BI concepts for business process based operational analytics. The first approach occurs in three variations in the form of (i) using BI concepts for dynamic process performance evaluation [8,22,24,40,44,45,47], (ii) analyze event logs to improve the quality of business processes [2,3,14], and (iii) monitor process instances to inform users about unusual or undesired situations [17]. These variations are either short on implementation or apply BI analytics to individual business processes for discrete performance assessment associated with business process activities; but there is no emphasis on concepts like performance inferencing across business processes.

The second approach emphasizes analytics on selected business process activities within the modeling process [7]. It shows reference to analytic information during business process modeling as a way to incorporate BI. The approach emphasizes use case scenarios but is short on implementation details on how to evaluate performance.

The third approach focuses on utilizing BI to reduce redundant specifications of recurrent business functions when modeling business processes [46]. It fosters reuse of business function specifications and helps to improve the quality and comparability of business process models. This approach is focused on the modeling for individual business process only.

The fourth approach [19] outlines a framework to reengineer business processes structure through data analytics on external data. The approach lacks implementation details and does not consider concepts like performance inferencing across business processes.

3. DIMENSION FLOW MODEL

Dimension flow model [23] is a graphical conceptual method to identify dimensional (analytic) information that can be considered as relevant for analyzing business process activities. Dimension flow modeling is based on information flow modeling concepts [22,30] and is valuable because it provides a basis for separating information from transactional processing for analytical processing. Development of dimension flow can be beneficial for (i) understanding the nature of analytic information without the complexities of data storage, and (ii) comprehending how business process activities are affected by the such information. Figure 1 shows the generic outline of a dimension flow model.

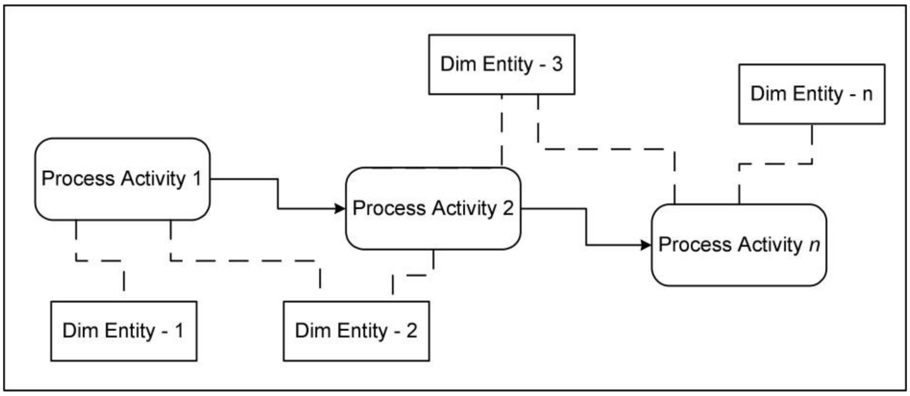

Figure 1. Dimension Flow Model

In Figure 1, the business process model consists of various activities labeled as Process Activity 1, Process Activity 2, and so on. Each process activity's utilization of dimensional information is represented through various dimensional entity types like Dim Entity - 1, Dim Entity - 2, and so on. It is possible that the same dimensional entity type may be utilized by other process activities, like Dim Entity - 2 interacts with Process Activity 1 and Process Activity 2, while Dim Entity - 3 interacts with Process Activity - 2 and Process Activity - n.

The dimensional entity types of the dimension flow model are derived from the transactional entity relationship model (ERD) of the business process. Each dimensional entity type structure may include some or all the attributes of the associated transactional entity type that are essential for the purpose of analysis. Unlike a transactional ERD data model the dimensional entity types are standalone entity types which can later be transformed as dimensions in associated multi-dimensional models.

Figure 2 shows an example of a dimension flow model adapted from Oracle's Order to Pay business process. The diagram is a simplification of a similar business process as outlined by Oracle E-Business Suite (ERP) software. It can be categorized into five stages: (i) configure sales order, (ii) plan and prepare for order shipment, (iii) ship order and logistics, (iv) invoice customer on the order shipment, and (v) process order payment.

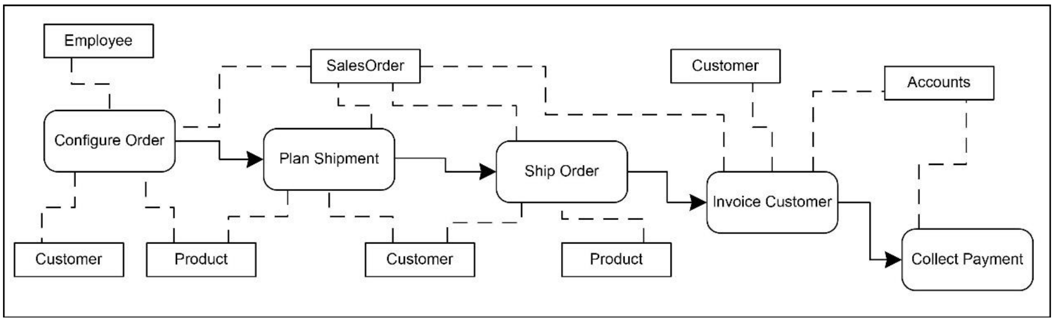

Figure 2. Order to Pay Dimension Flow Model

4. PERFORMANCE INFERENCING FRAMEWORK

The performance inferencing framework as shown in Figure 3 spans multiple business process. Each business process will have four intrinsic components referred as Business Process Analytics, Analytic Business Rules, Analytic Analyzer, and Generate Rationale. Two additional components Dimension Dictionary and Business Process Insights will be shared among business processes. The framework components are explained now.

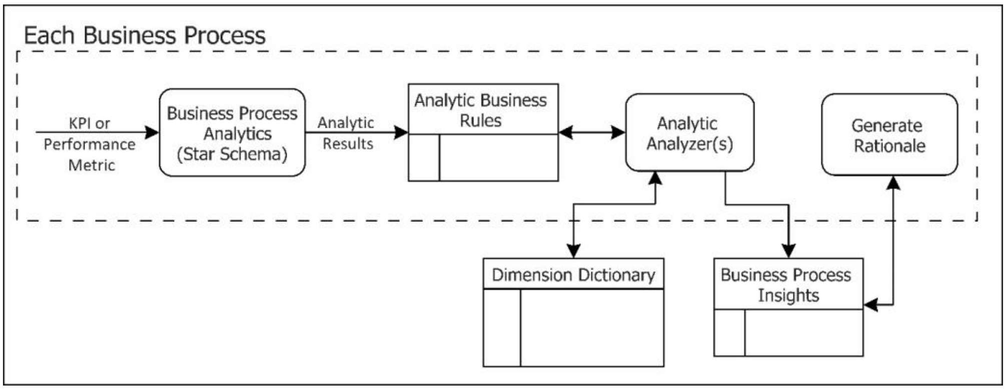

Figure 3. Performance Inferencing Framework

Business Process Analytic component would be star schema analytics to derive information on the performance of the business process based on some success metric or measure. The results of the star schema analysis would be stored as business rules in the next component Analytic Business Rules as a database table. Essentially, each business process will have its star schema analytics to derive information on the performance of their respective business process based on some success metric, and then have their analytic results stored as business rules in their respective analytic business rules dictionary.

The analytic business rules component database table structure is shown in Table 1. The table attributes are as follows: ID is the primary key. IF Dimension1, IF Dimension2, and so on are dimension attributes. THEN Measure1, THEN Measure2, and so on are star schema fact measures. The THEN Flag is the status of the business rule with respect to the success metric.

Table 1. Analytic Business Rules Component

ID	**IF Dimension1**	**IF Dimension2**	**. . .**	**THEN Measure1**	**THEN Measure2**	**. . .**	**THEN Flag**
. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .

To gain deeper insight for performance inferencing each business process will also have one or more Analytic Analyzer component that will analyze the stored Analytic Business Rules database table to determine the dimension factors that are affecting the business process success and store the results in Business Process Insights component as a database table. For example, one analytic analyzer may count those dimensions that are associated more often with low or high success metric, while another analytic analyzer may look at combinations of dimension attributes that are associated more often with low or high success metric.

As business processes in organizations are inter-connected, the Analytic Analyzer will also explore the implications of its analysis with other business processes through (i) a dimension dictionary component to determine what other business processes have similar named dimensions and then (ii) call those other business processes' Analytic Analyzers for processing their respective Analytic Business Rules and store their respective insights in Business Process Insights component database table. From the perspective of the framework's operation, whichever business process analytic analyzer starts the analysis thereafter will call the analytic analyzer of other business processes utilizing the dimension dictionary.

The dimension dictionary lists dimensions associated with each business process. Similar named dimensions across other business process models may or may not have similar attributes for analysis. This requires that there should be some standardization on dimension names pertaining to various analytic entity types in the organization. The structure of dimension dictionary is shown in Table 2.

Table 2. Dimension Dictionary Component

Dimension	**Business Process**
Dimension X	Business Process 1
Dimension X	Business Process 2
Dimension Z	Business Process 1
Dimension Y	Business Process 1
Dimension Y	Business Process 2
Dimension K	Business Process 2
. . .	. . .

The business process insights component database table structure is shown in Table 3. The table attributes are as follows: ID is the primary key. BP Measure is the business process measure name. Measure status is the value of the business process measure. Dim1, Dim2, and so on are dimension names, while Dim1 Value, Dim2 Value, and so on are their associated dimension values.

Table 3. Business Process Insights Component

ID	**BP Measure**	**Measure Status**	**Dim1**	**Dim1 Value**	**Dim2**	**Dim2 Value**	**. . .**
. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .

Once all the business processes analytic analyzers associated with the initial business process analytics analyzer have completed their processing, the Generate Rationale component will interact with Business Process Insights table to output the insights in the form of performance inferencing sequence analytic business rules. The inferencing sequence insight will be based on which business process initiates the generation of insight rationale.

The nature of processing performed by Business Process Analytics component, Analytic Analyzer component, and Generate Rationale component is explained through the prototype implementation.

5. PERFORMANCE INFERENCING FRAMEWORK IMPLEMENTATION

The performance inferencing framework implementation is demonstrated through a prototype that utilizes three hypothetical business processes: Sales, Customer Service, and Shipping. Their respective star schema structure and associated tables are outlined now followed by the logic of performance inferencing framework components. The dimension structures are not hierarchical.

The prototype is implemented in Oracle through PL/SQL language. The implementation is PC based. For the sake of simplicity, the number of dimensions and their structure is limited.

5.1. Business Process Star Schema Structure

Sales business process star schema structure is shown in Figure 4. Its success metric (or the fact measure) is the number of units sold (SalesUnits). Customers (Sales_Customer), product (Sales_Product), and location (Sales_Location) are the dimensions. The table structure of the sales business process star schema dimensions and fact measure is listed after Figure 4 from Table 4 to Table 7.

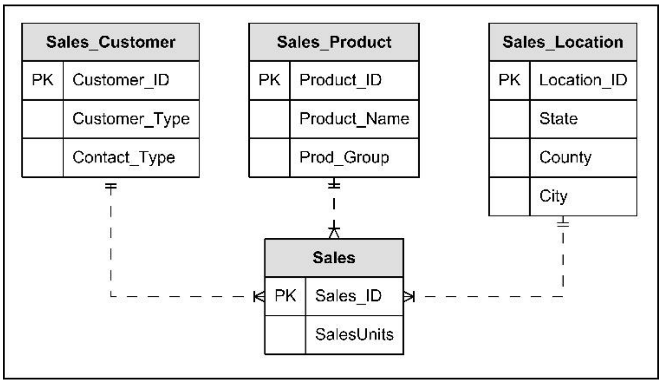

Figure 4. Sales Business Process Star Schema

Table 4. Sales_Customer

CUSTOMER_ID	CUSTOMER_TYPE	CONTACT_TYPE
101	Retail	Direct
102	Education	Indirect
103	Individual	Direct

Table 5. Sales_Product

PRODUCT_ID	PRODUCT_NAME	PROD_GROUP
1001	iPhone X	Mobile
1002	Galaxy S9	Mobile
1003	Galaxy S8	Mobile
1004	Surface Pro 6	Laptop
1005	Spectre x360	Laptop

Table 6. Sales_Location

LOCATION_ID	STATE	COUNTY	CITY
101	MO	Pulaski	Rolla
102	MO	Webster	Kansas City
103	MO	Greene	Springfield

Table 7. Sales

ALES_ID	SALESUNITS	PRODUCT_ID	LOCATION_ID	CUSTOMER_ID
1	25	1001	101	101
2	15	1002	102	102
3	10	1003	103	103
4	10	1004	101	101
5	5	1005	101	101
6	20	1001	103	103
7	20	1002	102	101
8	10	1003	103	102
9	5	1004	101	102
10	30	1002	102	101
11	15	1001	102	102
12	15	1002	103	103
13	25	1003	102	101
14	10	1004	103	102
15	5	1005	101	103

Customer service business process star schema structure is shown in Figure 5. Its success metric (or the fact measure) is the time duration of each call (Call_Length). Customers who initiate contact (Serv_Customer), product covered in the call (Serv_Product), and calls status over time (Serv_Call_Status) are the dimensions. The table structure of the customer service business process star schema dimensions and fact measure are listed after Figure 5 from Table 8 to Table 11.

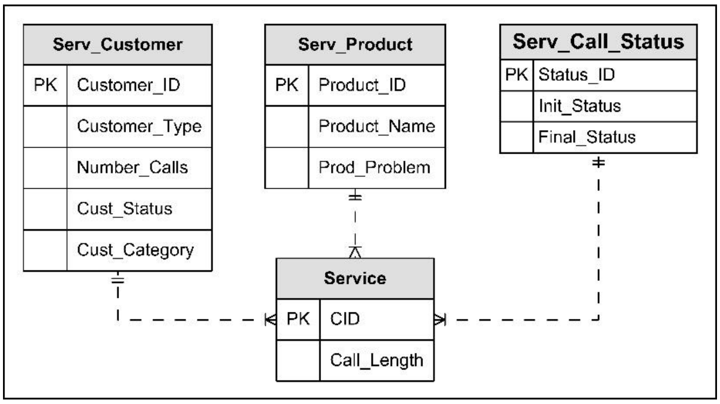

Figure 1. Customer Service Business Process Star Schema

Table 8. Serv_Customer

CUSTOMER_ID	CUSTOMER _TYPE	NUMBER_CALLS	CUST _STATUS	CUST _CATEGORY
101	Retail	10	Active	Upset
102	Education	2	Inactive	Normal
103	Individual	8	Active	Upset

Table 9. Serv_Product

PRODUCT_ID	PRODUCT_NAME	PROD_PROBLEM
1001	iPhone X	Not Working
1002	Galaxy S9	Slow
1003	Galaxy S8	Slow
1004	Surface Pro 6	Technical
1005	Spectre x360	Technical

Table 10. Serv_Call_Status

STATUS_ID	INIT_STATUS	FINAL_STATUS
1001	Completed	
1002	Elevated	Completed
1003	Elevated	Pending

Table 11. Service

CID	CALL_LENGTH	CUSTOMER_ID	PRODUCT_ID	STATUS_ID
1	5	101	1001	1001
2	2	101	1005	1003
3	10	103	1005	1003
4	4	101	1001	1001
5	2	103	1005	1002
6	5	101	1005	1002
7	5	101	1005	1003
8	10	103	1005	1002
9	8	101	1005	1002

Shipping business process star schema structure is shown in Figure 6. Its success metric (or the fact measure) is the number of units shipped (Units_Shipped) and number of units delayed (Units_Delayed). Supplier of the product (Serv_Supplier), product that is being shipped (Ship_Product), shipping location (Ship_Location), and carrier for delivery (Ship_Delivery) are the dimensions. The table structure of the shipping business process star schema dimensions and fact measures are listed after Figure 6 from Table 12 to Table 16.

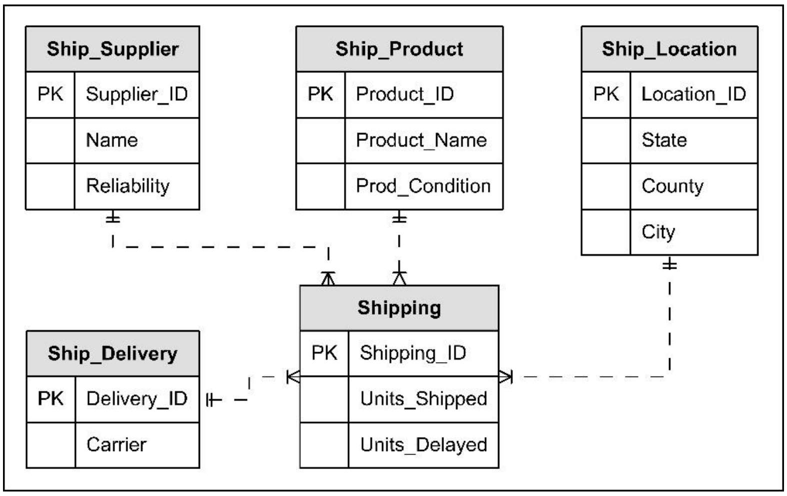

Figure 2. Shipping Business Process Star Schema

Table 12. Ship_Supplier

SUPPLIER_ID	NAME	RELIABILITY
1	Apple	Excellent
2	Samsung	Good
3	Microsoft	Excellent
4	HP	Good

Table 13. Ship_Product

PRODUCT_ID	PRODUCT_NAME	PROD_CONDITION
1001	iPhone X	Good
1002	Galaxy S9	Good
1003	Galaxy S8	Fair
1004	Surface Pro 6	Good
1005	Spectre x360	Fair

Table 14. Ship_Location

LOCATION_ID	STATE	COUNTY	CITY
101	MO	Pulaski	Rolla
102	MO	Webster	Kansas City
103	MO	Greene	Springfield

Table 15. Ship_Delivery

DELIVERY_ID	CARRIER
1	UPS
2	FedEx
3	USPS

Table 16. Shipping

SHIP _ID	UNITS _SHIPPED	UNITS _DELAYED	LOCATION _ID	PRODUCT _ID	SUPPLIER _ID	DELIVERY _ID
1	5	0	102	1002	2	1
2	4	2	101	1005	4	3
3	5	3	101	1005	4	3
4	5	1	103	1004	3	2
5	3	2	102	1005	4	1
6	10	1	103	1001	1	2
7	5	2	102	1002	2	3
8	4	3	101	1005	4	2
9	2	0	103	1003	2	1
10	2	3	101	1005	4	3

5.2. Dimension Dictionary Contents

Table 17 is the dimension dictionary associated with the three business processes OLAP schemas.

Table 17. Dimension Dictionary

Dimension	Business Process
Customer	Sales
Customer	Customer Service
Product	Sales
Product	Customer Service
Product	Shipping
Location	Sales
Location	Shipping
Supplier	Shipping
Delivery	Shipping

5.3. Performance Inferencing Logic

Performance inferencing logic is outlined now in three steps.

Step 1: Business Process Analytics component

Individual business processes will have their business process analytics component run on some routine schedule. Each business process analytics component is a database procedure. All business processes will categorize their analytics outcome with respect to their success metric as high, low, or normal.

In the prototype sales business process analytics component is based on unit sales success metric. Unit sales below 5 are considered low. Unit sales above 5 are considered normal. The results of analytics in the form of combination of dimension factors with respect to the success metric are stored in analytic business rule table categorized with status as “low” or “normal”. The following query yields low status.

```
select product_name,county,customer_type,salesunits
from sales, sales_product, sales_customer, sales_location
where sales.product_id = sales_product.product_id and
sales.customer_id = sales_customer.customer_id and
```

```
sales.location_id = sales_location.location_id and
salesunits <= (select min(salesunits) from sales);
```

The analytic business rule table for sales business process is shown in Table 18. Appendix A lists the database procedure so_analytics for sales business process analytics.

Table 18. Sales Business Process Analytic Business Rules

SOA_ID	PRODUCT _NAME	SALES _COUNTY	CUSTOMER _TYPE	SALESUNITS	SALES _FLAG
1	Spectre x360	Pulaski	Retail	5	Low
2	Surface Pro 6	Pulaski	Education	5	Low
3	Spectre x360	Pulaski	Individual	5	Low
4	iPhone X	Pulaski	Retail	25	Normal
5	Galaxy S9	Webster	Education	15	Normal
6	iPhone X	Greene	Individual	20	Normal
7	Galaxy S9	Webster	Retail	20	Normal
8	Galaxy S9	Webster	Retail	30	Normal
9	iPhone X	Webster	Education	15	Normal
10	Galaxy S9	Greene	Individual	15	Normal
11	Galaxy S8	Webster	Retail	25	Normal

Each row in the above table is an analytic business rule. For example, the business rule pertaining to soa_id value 1 is as follows:

IF Product_Name = Spectre x360 AND
Sales_County = Pulaski AND
Customer_Type = Retail
THEN Salesunits = 5 AND
Sales_Flag = Low

Customer service business process analytics component is based on the success metric of number of calls. More than 3 calls reflect deeper concern with the product and are categorized as high. Calls less than 3 are considered low. The results of analytics in the form of combination of dimension factors with respect to the success metric are stored in analytic business rule table categorized with status as “high” or “low”. The following query yields high status.

```
select product_name,customer_type,init_status,count(*) as complaints_no
from service, serv_product, serv_customer, serv_call_status
where service.product_id = serv_product.product_id and
service.customer_id = serv_customer.customer_id and
service.status_id = serv_call_status.status_id and
init_status = 'Elevated'
group by product_name,customer_type,init_status
having count(*) >= 3;
```

The analytic business rule table for customer service business process is shown in Table 19. Database procedure for customer service business process analytics is logically similar to the sales business process analytics procedure in Appendix A.

Table 19. Customer Service Business Process Analytic Business Rules

CSA_ID	PRODUCT _NAME	CUSTOMER _TYPE	INIT _STATUS	CALLS _UNITS	CALLS _FLAG
1	Spectre x360	Retail	Elevated	4	High
2	Spectre x360	Individual	Elevated	3	High

Each row in the above table is an analytic business rule. For example, the business rule pertaining to csa_id value 1 is as follows:

IF Product_Name = Spectre x360 AND
Customer_Type = Retail AND
Init_Status = Elevated
THEN Calls_Units = 4 AND
Calls_Flag = High

Shipping business process analytics component is based on the success metric of shipping delays. If there are more than 3 delays in product shipment it reflects deeper concern with shipment and categorized as high. Delivery less than 3 are considered low. The results of analytics in the form of combination of dimension factors with respect to the success metric are stored in analytic business rule table categorized with status as “high” or “low”. The following query yields high status.

```
select carrier, county, product_name, name, sum(units_delayed) as delayed_no
from shipping, ship_delivery, ship_location, ship_product, ship_supplier
where shipping.product_id = ship_product.product_id and

shipping.delivery_id = ship_delivery.delivery_id and
shipping.location_id = ship_location.location_id and
shipping.supplier_id = ship_supplier.supplier_id
group by carrier,county, product_name,name
having sum(units_delayed) > 2;
```

The analytic business rule table for shipping business process is shown in Table 20. Database procedure for shipping business process analytics is logically similar to the sales business process analytics procedure in Appendix A.

Table 20. Shipping Business Process Analytic Business Rules

SHIP_ ANAL_ID	CARRIER	COUNTY	PRODUCT _NAME	NAME	DELAYED _NO	DELAY _FLAG
1	FedEx	Pulaski	Spectre x360	HP	3	High
2	USPS	Pulaski	Spectre x360	HP	8	High
3	FedEx	Greene	iPhone X	Apple	1	Low
4	UPS	Webster	Spectre x360	HP	2	Low
5	FedEx	Greene	Surface Pro 6	Microsoft	1	Low
6	UPS	Greene	Galaxy S8	Samsung	0	Low
7	UPS	Webster	Galaxy S9	Samsung	0	Low
8	USPS	Webster	Galaxy S9	Samsung	2	Low

Each row in the above table is an analytic business rule. For example, the business rule pertaining to ship_anal_id value 1 is as follows:

IF Carrier = FedEx AND
County = Pulaski AND
Product_Name = Spectre x360 AND
Name = HP
THEN Delayed_No = 3 AND
Delay_Flag = High

Step 2: Analytics Analyzer component

The general logic of analytics analyzer component is outlined in Figure 7. Appendix B lists the database procedure analytic_analyzer_so for sales analytics analyzer.

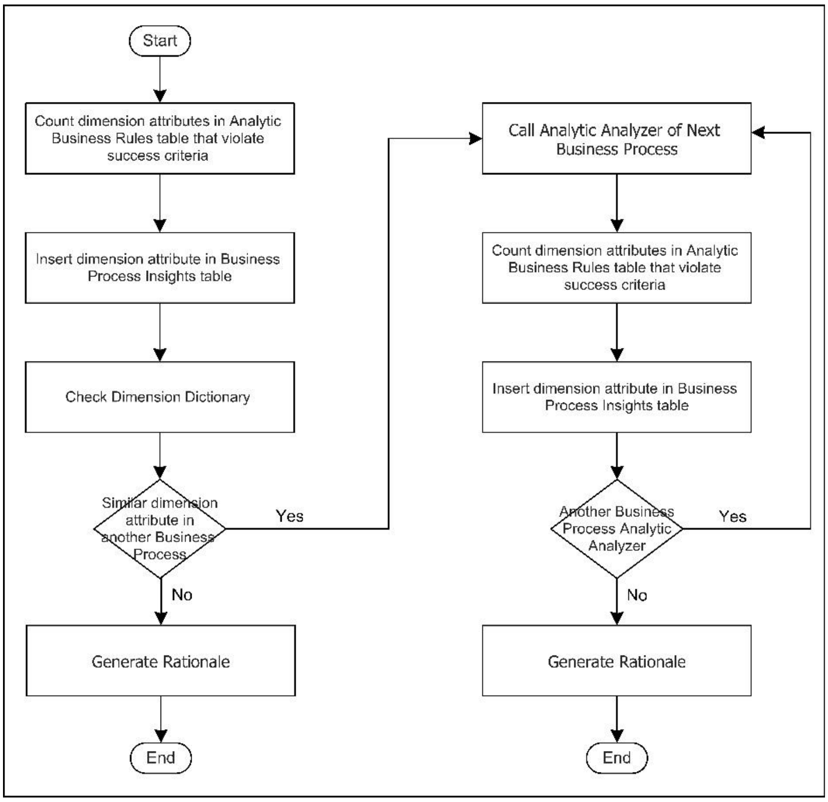

Figure 3. Analytic Analyzer Logic

Each business process will have many analytics analyser database procedures with varying input parameters. In the prototype, as the sales business process is the starting point for further analysis, inferencing sequence of business insights will commence with sales.

The sales business process analytics analyzer logic counts how many times each dimension attribute value has appeared in the sales business process analytics table. If the dimension attribute has appeared more than once (or whatever be the threshold) then:

a. the procedure inserts the dimension attribute value into business process insights table along with all other dimension attribute values that exceed the threshold value. It is possible that an organization may decide to ignore one-time dip, but when a dimension attribute is having a dip multiple times then it may require also checking beyond the existing business process with other business processes. In the prototype, product "Spectre x360" and location "Pulaski" cross the threshold and so are inserted in the business process insight table.

b. the procedure checks with dimension dictionary for another business process with similar dimension names and individually call the analytic analyzer for those business processes. In the dimension dictionary, product dimension is part of customer service business process OLAP schema, while product and customer dimensions are part of shipping business process OLAP schemas. So, the sales order analytics analyzer procedure now calls the analytics analyzer component of customer service and shipping with product_name and cust_type as input parameter. It is possible to have many analytics analyzer procedures with different input parameters based on what combination of dimension attributes have to be checked.
c. The analytics analyzer of customer service counts how many times each product dimension attribute value has appeared in customer service calls flag value of "High". In the prototype product "Spectre x360" has high complaints, so this information is inserted in the business process insights table. Similarly, the analytics analyzer of shipping counts how many times each product and location dimension attribute values has crossed the threshold in shipping business process analytics table with delay flag value of "High". In the prototype product "Spectre x360" and location "Pulaski" have high shipping delays, so this information is inserted in the business process insights table.

Step 3: Generate Rationale

Once all the additional analytics analyzers have finished their analysis, then the analytics analyzer that started the inferencing sequence will display the insights stored in the Business Process Insights table. In the prototype the sales business process analytics analyzer calls the generate_rationale procedure to display the inferencing chain. Below is the inference sequence (or chain) generated by the Generate Rationale component.

Sales low at Location Pulaski for Product Spectre x360
Complaints high because Product Spectre x360
Shipping Delay high for Product Spectre x360 for Location Pulaski

Once the performance insights from the three business processes are outlined in the form of inferencing sequence, the dimension flow model can be utilized to focus on business process activities affected by the insight. In the prototype the inferencing sequence suggests that the sales of product Spectre x360 are low and the product has more complaints due to shipping delays. Accordingly, the shipping business process activity associated with product supplier needs to be investigated for solution and performance improvement.

6. Conclusions

This paper proposes an extension on the nature of insights provided by traditional business intelligence analytics that goes beyond an individual business process. Such deeper insights in the form of inferencing sequence across multiple business processes provides a richer assessment on the direction of business performance thereby making an organization more effective and competitive.Further research is ongoing to enhance the approach by embedding more complexity in analytic analyzer and dimension dictionary component to further improve the inferencing sequence. Another area of research is to align inferencing sequence with quantifiable business objectives with respect to multiple business processes.

REFERENCES

[1] Agrawal, R., Gupta, A. and S. Sarawagi, (1997) “Modeling Multidimensional Databases,” Proceedings of the Thirteenth International Conference on Data Engineering, pp. 232–243.

[2] Aalst, W. M. P. van der, Reijers, H. A., Weijters, A. J. M. M., Dongen, B. F. van, Alves, Song, A. K., M de Medeiros and Verbeek, H. M. W. (2007) “Business process mining: An industrial application,” Information Systems, Vol. 32, No. 1, pp. 713 - 732.

[3] Aalst, Wil van der (2018) “Spreadsheets for business process management: Using process mining to deal with “events” rather than “numbers”?”, Business Process Management Journal, Vol. 24, No. 1, pp.105-127.

[4] Arigliano, F., Ceravolo, P., Fugazza, C. and Storelli, D. (2008) “Business Metrics Discovery by Business Rules,” in M. D. Lytras, J. M. Carroll, E. Damiani (editors), Emerging Technologies and Information Systems for the Knowledge Society: Lecture Notes in Computer Science, Vol. 5288, pp. 395-402.

[5] Baars, H., Felden, C., Gluchowski, P., Hilbert, A., Kemper, H. G. and Olbrich, S. (2014) “Shaping the next incarnation of business intelligenc” Business & Information Systems Engineering, Vol. 6, No. 1, pp. 11-16.

[6] Blasum, R. (April 2007) “Business Rules and Business Intelligence,” Information Management Magazine.

[7] Bucher, T., Gericke, A. and Sigg, S. (2009) “Process-centric business intelligence,” Business Process Management Journal, Vol. 15, No. 3, pp. 408-429.

[8] Castellanos, M., Casati, F., Sayal, M. and Dayal, U. (2005) “Challenges in business process analysis and optimization,” in C. Bussler and M. C. Shan (Eds.), Technologies for E-Services, 6th International Workshop, TES 2005, Trondheim, Norway, September 2-3, 2005, Revised Selected Papers, volume 3811 of Lecture Notes in Computer Science, Springer-Verlag, pp. 1-10.

[9] Castellanos, M., Medeiros, A. Alves de, Mendling, J., Weber, B., and Weijters, A. (2009) “Business Process Intelligence,” in J. Cardoso and W. van der Aalst (Eds.) Handbook of Research on Business Process Modeling, IGI: Hershey, PA, pp. 456 - 480 .

[10] Chen, H., Chiang, H. L. Roger and Storey, Veda C. (2012) “Business Intelligence and Analytics: from Big Data to Big Impact” MIS Quarterly, Vol. 36, No. 4, pp. 1165-1188.

[11] Cody, W. F., Kreulen, J. T., Krishna, V. and Spangler, W. S. (2002) “The integration of business intelligence and knowledge management,” IBM Systems Journal, Vol. 41, No. 4, pp. 697-713.

[12] Dayal, U., Wilkinson, K., Simitsis, A. and Castellanos, M. (2009) “Business Processes Meet Operational Business Intelligence,” Bulletin of the Technical Committee on Data Engineering, Vol. 32, No. 3, pp. 35-41.

[13] Debevoise, T. (2005) Process Management with a Business Rules Approach: Implementing The Service Oriented Architecture, Business Knowledge Architects, Roanoke.

[14] Dumas, M., Rosa, M. La, Mendling, J. and Reijers, H. A. (2013) “Process intelligence,” in Fundamentals of Business Process Management, Springer: Berlin Heidelberg, pp. 353-383.

[15] Elbashira, M. Z., Collierb, P. A. and Davernb, M. J. (2008) “Measuring the effects of business intelligence systems: The relationship between business process and organizational performance,” International Journal of Accounting Information Systems, vol. 9, no. 3, pp. 135-153.

[16] Golfarelli, M., Rizzi, S. and Cella, I. (2004) “Beyond data warehousing: what's next in business intelligence?,” Proceedings of the 7th ACM international workshop on Data warehousing and OLAP, pp. 1-6.

[17] Grigori, D., Casati, F., Castellanos, M., Dayal, U. , Sayal, M. and Shan, M. (2004) “Business process intelligence,” Computers in Industry, Vol. 53, No. 3, pp. 321-343.

[18] Halle, B. V. (2002) Business Rules Applied, John Wiley & Sons, New York, NY.

[19] Jha, M., Jha, S. and O'Brien, L. (2016) “Combining big data analytics with business process using reengineering,” Proceedings of the IEEE Tenth International Conference on Research Challenges in Information Science, Grenoble, France.

[20] Kakhki, M.D. and Palvia. P. (2016) “Effect of Business Intelligence and Analytics on Business Performance,” Proceedings of the Twenty-second Americas Conference on Information Systems, San Diego, pp. 1-10.

[21] Kaula, R. (2009) “Business Rules for Data Warehouse,” World Academy of Science, Engineering and Technology, Vol. 59, pp. 951- 959.

[22] Kaula, R. (2012) “Business Intelligence Process Metrics Specification: An Information Flow Approach,” International Journal of Computers and Their Applications, Vol. 19, No. 4, pp. 262-272.

[23] Kaula, R (2017) “Operational Intelligence through Performance Trends: An Oracle Prototype,” International Journal of Business Intelligence and Systems Engineering, Vol. 1, No. 2, pp. 140-165.

[24] Kaula, R (2018) Affinity Clusters For Business Process Intelligence, International Journal of Database Management Systems (IJDMS), Vol.10, No.5, October 2018.

[25] Kimball, R. and Ross, M. (2002) The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling, John Wiley & Sons, New York.

[26] Kimball, R., Ross, M., Thornthwaite, W., Mundy, J. and Becker, B. (2008) The Data Warehouse Lifecycle Toolkit, John Wiley & Sons, New York.

[27] Larson, D. and Chang, V. (2016) “A review and future direction of agile, business intelligence, analytics and data science, International Journal of Information Management,” Vol. 36, No. 5, pp. 700-710.

[28] Linden, M., Felden, C., and Chamoni, P. (2010) “Dimensions of business process intelligence,” Proceedings of the International Conference on Business Process Management, pp. 208-213.

[29] Lönnqvist, A. and Pirttimäki, V. (2006) “The Measurement of Business Intelligence,” Information Systems Management, Vol. 23, No. 1, pp. 32-40

[30] Loshin, D. (2003) Business Intelligence: The Savvy Manager’s Guide, San Francisco: Morgan Kaufman.

[31] Marjanovic, O. (2007) “The Next Stage of Operational Business Intelligence: Creating New Challenges for Business Process Management,” Proceedings of the 40th Annual Hawaii International Conference on System Sciences, pp. 215c-215c.

[32] Marjanovic, O. (2010) “Business Value Creation through Business Processes Management and Operational Business Intelligence Integration,” 43rd Hawaii International Conference on System Sciences (HICSS), pp. 1-10.

[33] Mircea, M. and Andreescu, A. (2009) “Using Business Rules in Business Intelligence” Journal of Applied Quantitative Methods, vol. 4, no. 3, pp. 382-393.

[34] Olivia, R. (2009) Business Intelligence Success Factors: Tools for Aligning Your Business in the Global Economy, Wiley & Sons, Hoboken, N.J.

[35] Olszak, C. M. and Ziemba, E. (2007) “Approach to Building and Implementing Business Intelligence Systems,” Interdisciplinary Journal of Information, Knowledge, and Management, vol. 2, pp. 135-148.

[36] Ponniah, P. (2010) Data Warehousing Fundamentals for IT Professionals (2nd Edition), New York: John Wiley & Sons.

[37] Ross, R. (2003) Principles of the Business Rule Approach, Addison-Wesley, Boston.

[38] Richards, R., Yeoh, W., Chong, A.Y.L. and Popovič, A. (2019) “Business Intelligence Effectiveness and Corporate Performance Management: An Empirical Analysis,” Journal of Computer Information Systems, Vol. 59, No. 2, pp. 188-196.

[39] Rudometkina, M. N. and Spitsyn, V. G. (2014) “Detection of processing model basic elements in intellectual analysis of flexible processes through business process intelligence,” Proceedings of the 9th International Forum on Strategic Technology (IFOST), pp. 97-101.

[40] Sa, J. O. and Santos, M. Y. (2017) “Process-driven data analytics supported by a data warehouse model,” International Journal of Business Intelligence and Data Mining, Vol.12, No.4, pp.383-405.

[41] Sharda, R., Dalen, D. and Turban, E. (2013) Business Intelligence: A Managerial Perspective on Analytics, Prentice-Hall, Upper Saddle River.

[42] Shollo, A. and Galliers, R.D. (2016) “Towards an understanding of the role of business intelligence systems in organisational knowing,” Information Systems Journal, Vol. 26, No. 4, pp. 339-367.

[43] Sen, A. and Sinha, A. (2005) “A comparison of data warehousing methodologies,” Communications of the ACM, vol. 48, no. 3, pp. 79-84.

[44] Sohail, A. and Dominic, P. D. D. (2012) “A gap between Business Process Intelligence and redesign process,” Proceedings of the 2012 International Conference on Computer & Information Science (ICCIS), pp. 136-142.

[45] Tan, W., Shen, J.W., Xu, L., Zhou, B. and Li, L. (2008) “A Business Process Intelligence System for Enterprise Process Performance Management,” IEEE Transactions on Systems, Man, and Cybernetics, vol. 38, no. 6, pp. 745-756.

[46] Thom, L.H., Reichert, M., Chiao, C.M., Iochpe, C. and Hess, G.N. (2008) "Inventing Less, Reusing More, and Adding Intelligence to Business Process Modeling," in S.S. Bhowmick, J. Küng, R. Wagner (editors), Database and Expert Systems Applications: Book Series Title: Lecture Notes in Computer Science, vol. 5181, pp. 837-850.
[47] Torres, R., Sidorova, A. and Jones, M.C. (2018) "Enabling firm performance through business intelligence and analytics: A dynamic capabilities perspective," Information & Management, Vol.55, No. 7, pp. 822-839.
[48] Watson, H. J. and Wixom, B.H. (2007) "The Current State of Business Intelligence," Computer, vol. 40, no. 9, pp. 96-99.
[49] Wrembel, R. and Koncilia, C. (2007) Data warehouses and OLAP: concepts, architectures, and solutions, Hershey: Idea Group Inc..Lee, S.hyun. & Kim Mi Na, (2008) "This is my paper", ABC *Transactions on ECE*, Vol. 10, No. 5, pp120-122.

APPENDIX A

```
create or replace procedure so_analytics as
cursor so_low is
select product_name,county,customer_type,salesunits
from sales, sales_product, sales_customer, sales_location
where sales.product_id = sales_product.product_id and
sales.customer_id = sales_customer.customer_id and
sales.location_id = sales_location.location_id and
salesunits <= (select min(salesunits) from sales);
so_low_row so_low%rowtype;
cursor so_high is
select product_name,county,customer_type,salesunits
from sales, sales_product, sales_customer, sales_location
where sales.product_id = sales_product.product_id and
sales.customer_id = sales_customer.customer_id and
sales.location_id = sales_location.location_id and
salesunits >= (select avg(salesunits) from sales);
so_high_row so_high%rowtype;
begin
for so_low_row in so_low loop
insert into sales_ord_analytics values
(soa_seq.nextval,so_low_row.product_name,so_low_row.county,so_low_row.customer_Type,so_low_ro
w.salesunits,'Low');
end loop;
for so_high_row in so_high loop
insert into sales_ord_analytics values
(soa_seq.nextval,so_high_row.product_name,so_high_row.county,so_high_row.customer_Type,so_high_
row.salesunits,'Normal');
end loop;
end;
```

APPENDIX B

```
create or replace procedure analytic_analyzer_so is
cursor cur1 is
select sales_county, count(*) sales_county_ctr from sales_ord_analytics where sales_flag = 'Low' group
by sales_county having count(*) > 1
order by sales_county_ctr desc;
cur1_row cur1%rowtype;

cursor cur1_ext is
select dim,bp from dim_dict where dim = 'Location' and bp <> 'Sales';
```

```
cur1_ext_row cur1_ext%rowtype;

cursor cur2 is
select product_name, count(*) product_name_ctr from sales_ord_analytics
where sales_flag = 'Low' group by product_name order by product_name_ctr desc;
cur2_row cur2%rowtype;
cursor cur2_ext is
select dim,bp from dim_dict where dim = 'Product' and bp <> 'Sales';
cur2_ext_row cur2_ext%rowtype;

cursor cur3 is
select customer_type, count(*) customer_type_ctr from sales_ord_analytics
where sales_flag = 'Low' group by customer_type having count(*) > 1
order by customer_type_ctr desc;
cur3_row cur3%rowtype;

cursor cur3_ext is
select dim,bp from dim_dict where dim = 'Customer' and bp <> 'Sales';
cur3_ext_row cur3_ext%rowtype;

tlocation varchar2(20); tproduct varchar2(20); tcustomer varchar2(20);
flag_cs_loc varchar2(3) := 'off'; flag_cs_prod varchar2(3) := 'off';
flag_cs_cust varchar2(3) := 'off'; flag_ship_loc varchar2(3) := 'off';
flag_ship_prod varchar2(3) := 'off'; flag_ship_cust varchar2(3) := 'off';

begin
open cur1;
fetch cur1 into cur1_row;
if cur1%found then
  if cur1_row.sales_county_ctr > 1 then
    tlocation := cur1_row.sales_county;
    dbms_output.put_line('tlocation '||tlocation);
    for cur1_ext_row in cur1_ext
              loop
              if cur1_ext%found then
                if cur1_ext_row.bp = 'Customer Service' then
                  flag_cs_loc := 'on';  --analytic_analyzer_cs;
                elsif cur1_ext_row.bp = 'Shipping' then
                  flag_ship_loc := 'on';  --analytic_analyzer_ship;
      end if;
              end if;
              end loop;
  end if;
end if;

open cur2;
fetch cur2 into cur2_row;
if cur2%found then
  if cur2_row.product_name_ctr > 1 then
    tproduct := cur2_row.product_name;
    dbms_output.put_line('tproduct '||tproduct);
    for cur2_ext_row in cur2_ext
              loop
              if cur2_ext%found then
                if cur2_ext_row.bp = 'Customer Service' then
                  flag_cs_prod := 'on';  --analytic_analyzer_cs;
                elsif
```

```
          cur2_ext_row.bp = 'Shipping' then
          flag_ship_prod := 'on';  --analytic_analyzer_ship;
         end if;
        end if;
        end loop;
  end if;
end if;

open cur3;
fetch cur3 into cur3_row;
if cur3%found then
  if cur3_row.customer_type_ctr > 1 then
    tcustomer := cur3_row.customer_type;
    dbms_output.put_line('tcustomer '||tcustomer);
    for cur3_ext_row in cur3_ext
        loop

        if cur3_ext%found then
         if cur3_ext_row.bp = 'Customer Service' then
          flag_cs_cust := 'on';  --analytic_analyzer_cs;
         elsif
          cur3_ext_row.bp = 'Shipping' then
          flag_ship_cust := 'on';  --analytic_analyzer_ship;
         end if;
        end if;
        end loop;
  end if;
end if;

-- insert into business process insights table

if (tlocation is not null) and (tproduct is not null) then
insert into bp_insights values(rational_seq.nextval,'Sales
','low','Location',tlocation,'Product',tproduct,null,null,null,null);
end if;
if (tlocation is not null) and (tproduct is not null) and (tcustomer is not null) then
insert into bp_insights values(rational_seq.nextval,'Sales
','low','Location',tlocation,'Product',tproduct,'Customer',tcustomer,null,null);
end if;

-- call other BP analytic analyzers
-- So there could be multiple procedures based on what dimension attribute is null

if (flag_cs_loc = 'on') and (flag_cs_prod = 'on') and (flag_cs_cust = 'on') then
--analytic_analyzer_cs('Sales',tproduct,tlocation,tcustomer);
end if;
if (flag_cs_loc = 'on') and (flag_cs_prod = 'on') and (flag_cs_cust = 'off') then
--analytic_analyzer_cs('Sales',tproduct,tlocation);
end if;
if (flag_cs_loc = 'off') and (flag_cs_prod = 'on') and (flag_cs_cust = 'off') then
analytic_analyzer_cs('Sales',tproduct);
end if;
if (flag_ship_loc = 'on') and (flag_ship_prod = 'on') and (flag_ship_cust = 'on') then
--analytic_analyzer_ship('Sales',tproduct,tlocation,tcustomer);
end if;
if (flag_ship_loc = 'on') and (flag_ship_prod = 'on') and (flag_ship_cust = 'off') then
analytic_analyzer_ship('Sales',tproduct,tlocation);
```

```
end if;
if (flag_ship_loc = 'on') and (flag_ship_prod = 'off') and (flag_ship_cust = 'off') then
--analytic_analyzer_ship('Sales',tlocation);
end if;

--display inferencing sequence
infer_seq;
end;
```