BUSINESS ACTIVITY MONITORING FOR LOGISTICS MANAGEMENT SYSTEMS

DİLEK TİTİZ

İzmir Institute of Techonolgy
July, 2013

BUSINESS ACTIVITY MONITORING FOR LOGISTICS MANAGEMENT SYSTEMS

A Thesis Submitted to
the Graduate School of Engineering and Sciences of
İzmir Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of

MASTER OF SCIENCE

in Computer Software

by
Dilek TİTİZ

July 2013

İZMİR

We approve the thesis of **Dilek TİTİZ**

Examining Committee Members:

_____

Asst. Prof. Dr. Tolga AYAV
Department of Computer Engineering, Izmir Institute of Technology

_____

Asst. Prof. Dr. Tuğkan TUĞLULAR
Department of Computer Engineering, Izmir Institute of Technology

_____

Asst. Prof. Dr. Derya BİRANT
Department of Computer Engineering, Dokuz Eylül University

11 July 2013

_____

Asst. Prof. Dr. Tolga AYAV
Supervisor, Department of Computer Engineering
Izmir Institute of Technology

_____ _____

Prof. Dr. İ. Sıtkı AYTAÇ
Head of Department of Computer Engineering

Prof. Dr. R. Tuğrul SENGER
Dean of Graduate School of Engineering and Sciences

ACKNOWLEDGEMENTS

This thesis would not been finished without the help, contribution and support of the people around me, I can only mention about some of them here.

I would like to thank to my thesis advisor Asst. Prof. Dr. Tolga AYAV for his insight on this study. I also thank to Asst. Prof. Dr. Tuğkan TUĞLULAR who encourage me studying on this topic. Besides encouragement, his guidance and suggestions during the long process of the thesis I also appreciate.

This thesis is projected as San-Tez which is a program that ruled by Ministry of Science, Industry and Technology, with the enterprise partner; Bimar Inc. I also thank them for the support and collaboration.

I especially thank my parents, they always support me at all, their love and support bring me here.

And finally, most important support is from my husband, Osman TİTİZ, his patience and support during this thesis work make possible to finish this work as successfully.

ABSTRACT

BUSINESS ACTIVITY MONITORING FOR LOGISTICS MANAGEMENT SYSTEMS

Enterprises use different software for different purposes, such as accounting management, customer relationships, so that information technologies become an integral part of business processes. A business process turns out to be a combination of these different software, besides the received information/data by one of software tools that comes from another one. Different software systems cannot have been aware of each other directly.

When the domain under consideration is container logistics, in addition to the different software, there are many relationships with other companies and so there are many different systems, variable types and file types that are being used. There are different transportation modes like highway, marine or airway, and by the expansion of intermodal transportation, tracking data is getting harder.

For container logistics being a complicated business, tracking data is a necessity. The necessity is to track and to control the data that comes from outer systems and inner systems to an interoperable platform. This necessity enables trackable business processes and thus increases the business performance.

In the scope of this thesis, a business activity monitoring environment is created in case of ARKAS Holding, which works on container logistics domain and Bimar Inc., which is the software provider of it. As it was mentioned above, this tracking needs cause to creation of the proposed and implemented monitoring environment for such a case of ARKAS Holding that does intermodal transportation, which has many endpoints in its business processes and there are too many data to be tracked.

There are many different endpoints, such as marine, roadway and depot, in container logistics; also there are many different data types that are coming from different systems. Thus, the ‘Business Activity Monitoring’ environment is created as this thesis’ subject to make these different systems talk to each other, also to enable gathering data directly and automatically to achieve real-time monitoring.

ÖZET

LOJİSTİK YÖNETİM SİSTEMLERİ İÇİN İŞ ETKİNLİĞİ İZLEME

İş süreçlerinin ayrılmaz parçası haline gelen bilgi teknolojileriyle birlikte şirketlerde değişik amaçlara yönelik farklı yazılımlar kullanılmaya başlamıştır; muhasebe yönetimi için ayrı, müşteri ilişkileri için ayrı, hatta şirket içi ilişkiler için bile kullanılan birbirinden farklı yazılım araçları bulunmaktadır. Bir iş süreci, bu yazılımların bir araya gelmesi ve birinden alınan bilginin/verinin diğeri tarafından bilinmesi yoluyla oluşmaktadır. Ancak birbirinden farklı olan bu yazılımların birbirinden haberdar olabilmesi direkt olarak mümkün değildir.

İş sektörünü lojistik özelinde ele alırsak, kullanılan farklı yazılımların yanında, dış şirketlerle olan çalışmalarda da birbirinden farklı sistemler, veri tipleri ve dosya tipleri olabilmektedir. Kombine taşımacılığın yaygınlaşmasıyla birlikte, çalışılan taşıma modlarındaki artış da veri takibini zorlaştırmıştır.

Lojistik sektörü gibi karmaşıklığı yüksek iş alanları söz konusu olduğunda veri takibi ihtiyaca, hatta zorunluluğa dönüşmektedir. Bu zorunluluk, dış şirketlerden ve iç sistemden gelen verinin birlikte çalışabilme ortamında takip ve kontrol edilmesidir. Bu takip, kullanılan yazılımlardan ve çalışılan sistemlerden elde edilen veriler ile iş süreçlerini izlenebilir kılmak ve bu sayede hem şirketlerin performansını artırmak hem de müşteri memnuniyeti için gereklidir.

Bu tezin kapsamında, konteyner taşımacılığı sektöründe çalışan ARKAS Holding ve yazılım grubu olan Bimar A.Ş. örneğinde bir iş etkinliği izleme ortamı oluşturuldu. Önceki paragrafta bahsedilen takip ihtiyacından kaynaklanan bu ortamı oluşturma fikri, kombine taşımacılık yapan ARKAS Holding örneğindeki gibi uç noktaların çok fazla olması ve takip edilmesi gereken bilginin fazlalığı nedeniyle doğdu.

Konteyner taşımacılığındaki farklı uç noktalardan - denizyolu, karayolu, depo gibi, alınması gereken farklı formatlardaki veriyi toplayıp, bu toplanan veriyi (İş Zekasından - Business Intelligence farklı olarak), gerçek-zamanlı olarak kullanıcıya sunmak için 'İş Etkinliği İzleme' ortamı bu tezin konusu olarak oluşturulmuştur.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

Local or global, any enterprise needs to know what is going on in its workflows and have to inform its customers about products and services. Gathering the information and displaying it is a task for an enterprise and it must be handled through information systems.

> "Quick and accurate information from different systems combined at one platform could prove to be vital for the survival of an enterprise. Easy accessibility of information by customers and information flow within businesses is a must for every enterprise. Information is gathered from disparate systems within an enterprise, which results in the enterprise not being confined to one technology based system" [2].

As mentioned, gathering data from different systems and present it accurately is a necessity for enterprises and to handle this process, there must be a system, tool or briefly a platform for it.

For a case study, this thesis is worked as a San-Tez project with Bimar Inc. which is the information technologies branch of ARKAS Holding that is one of the biggest players of container logistics area. San-Tez is a program that encourages universities and enterprises to make collaborations to create projects that are useful and usable for the enterprises, besides contribution to scientific works and to commercialize the scientific research works. So this San-Tez project is collaboration work of Bimar Inc and İzmir Institute of Technology. Project started on June 2011 and went on until December 2012 under the name of "An Integration and Monitoring Framework for Logistics Business Processes" with the id 0093.STZ.2011-1.

Officially; the aim of the LESBAM project is; to develop "An Integration and Monitoring Framework for Logistics Business Processes" to allow interoperability of enterprise business softwares of the corporations that work on container logistics area and make traceable the business data that gathered from these business softwares. [20].

There are two academic theses that are derived from this SAN-Tez project, one of them is "Service-Oriented Integration of Information Systems for Logistics

Management" which is handled using ESB tool of Microsoft and it is in writing phase. And the second one is this thesis, "Business Activity Monitoring for Logistics Management Systems" which is handled using BAM tool.

A whole container logistics, which is the domain area of this thesis, is a heterogeneous process that consist road transport, railway transport, marine transport and airway transport. These all types may be included in one transportation or some of them may be included. Either one or more transport type is included in one, there will be more than one points to be tracked. So this gathering and presenting information from different systems case is examined and finding solution for this case becomes the subject of this thesis.

In summary, the aim of this thesis project is to;

- use BAM as a method to monitor business processes,
- optimize BAM to be able to match the domain (– logistics management)
- make additions to BAM to use it as effectively as possible.

CHAPTER 2

BACKGROUND

The goal of this thesis project is to find a solution for gathering and presenting important data about monitoring containers in case of Bimar Inc. In this chapter, literature review, the business domain of Bimar Inc. and BAM the monitoring tool that is chosen to improve it for the research problem will be explained.

2.1. Review of Literature

By developing technology, interdependent organizations are increased, because working together over distances is too easy today thanks to technology. So, to handle the over-distance working there is a need to intermediary to make communication of organizations beside information get/send.

In [39], they say; better performance and continuous improvement of organizations are possible by real-time analysis and measurement. BAM provides this real-time access. Business process improvement is crucial for an organization to maximize business profitability. Continuous real-time monitoring of business processes, so business performance, are for effective performance management. BAM is a critical element by gathering KPIs to meet the business goals. The objective of this paper [39] is to propose a framework that is for real-time business performance management by using BAM. Proposed BAM design procedure is defining monitoring objects as real-time, design for dashboard, defining monitoring events and defining gathering and presenting rules.

The paper [40] can be the continuation of [39]. Real-time measurement and analysis of processes is crucial for the performance improvement of enterprises. BAM (Business Activity Monitoring) provides real-time access to business process and makes enable to measure the performance of business by gathering important data. BAM is a core element of successful BPM (Business Process Management). Performance measurement is based on business processes so the BAM works on business processes.

Enterprises wants to improve their performance and this can be possible by improvement of business processes. So business processes and performance measurement are related. And this relation is provided by BAM which is conducted in execution phase of BPM. BAM makes enable to relate processes and measure their important values after that monitors them by presenting them to user. The aim of BAM usage is get results as real-time. Because, there are needs to clear away the delays, bottlenecks and insufficient use of resources. To clear away these, they must be monitored as real-time to concern and resolve the situation at that exact time it happens. This paper [40] mentions generally all about in [39], but as a new, it proposes a BAM framework that based on processes, process-based performance measurement model (PPMM) that monitors KPIs form business processes and combines the all KPIs that comes from all levels of processes. This proposed method is for continuous process improvement.

Also in [42], there is another proposal like the proposed method in [40], for interdependent organizations. Interdependent organizations cooperate much more than the before. A widespread used technology; SOA (service oriented architecture) provides working of organizations through distances with each other by the helping of services. The improvements on information technologies help organizations to work with each other easily but this working increases the complexity of business processes. Business processes show how the organizations work internally also work with other organizations. Increasing number of business processes form a complexity, while they are also complicated in themselves. To resolve this complexity, to get purer business processes and to understand them, a method is introduced in this paper [42], SN (service networks) are for managing of business process by simplifying them and monitoring of performance measures, KPI (Key Performance Indicator). To increase the performance of organization in itself also by other organizations, there must be structure and this paper [42] proposes this SN method.

In paper [43], they improve their proposed method SN and start to value calculations of SNs (Service Networks), that is proposed method in [42] to reveal the cooperation of interdependent organizations, are based on KPI. KPIs show the performance of business processes that are included in SNs. There are many partners in SNs and they have to share the information between them. This sharing also provides the more precise KPIs because KPIs are calculated by overall information from each

partner in SN. To ensure that contribution to KPI creation, there must be monitoring agreements that include all partners of SN, their events and these events' returns for KPIs. In order to measure KPIs, the business processes must be monitored. By this monitoring, the values are gathered to form KPIs. Business activity monitoring approaches are related in this gathering KPIs step of SNs method. When the processes are running, the values are gathered by monitoring as real-time. As summary of this paper [43], processes have to be monitored and the results are formed as KPIs. This monitoring work must be performed on every partner in SN and the overall results must be processes as resulted KPIs.

In paper [42], the monitoring issue is dealt for container logistics domain. In case of container logistics, there are various enterprises that work together. These enterprises increase with intermodal transportation. And because of various enterprises that are included in business processes, there are integrations between these enterprises. Because of the complexity of domain, monitoring of business processes becomes crucial. There is no environment for monitoring of business processes in case of container logistics. So, there is a lack of control to track the information that comes and goes between partners of transportation processes. To meet this need, there must be a monitoring environment. There are different processes in different business units, and so there must be appropriate gathering structure to these different systems. To handle this issue, "need to know" principle is followed on [41], this principle means, which information is necessary for which user/usage. In other words, what is needed to discover in a business process is revealed. There are integrations and so integration points between partners and these points are used for gathering data. What data to monitor and where to get these data are included in the context of this paper [41]. There is ACMDP design patterns that audit, control and monitor's abbreviation. This patterns are used for determine the view that will be examined as part of monitoring issue. So, by ACMDP and BAM (Business Activity Monitoring tool of Microsoft) a monitoring environment for container logistics can be designed.

These papers show the need of monitoring that is an increasing necessity for organizations especially that works over distances. So, the aim of this thesis, become to handle this need in case of container logistics.

2.2. Logistics Management Systems

There are some terms need to be explained before talking about logistics management systems:

2.2.1. Logistics

"Logistics is defined as a business planning framework for the management of material, service, information and capital flows. It includes the increasingly complex information, communication and control systems required in today's business environment." [4]. This quote defines logistics, besides the other many definitions of it. Logistics is a detailed subject and its definition changes according to point of view. Basically it can be defined like in the [7]; "*... the set of activities whose objective is to move items between origins and destinations (usually from production to consumption)* ..."

2.2.2. Container Logistics

"The arrangement of shipping containers within a vessel or transport mode.A logistics plan may include the order in which containers must be loaded, along with the cargo handling and freight securing instructions"[5]. Container (Figure 2.1) is a box that is very large and is used for transport freight from one vessel to another or to destination. It is useful, because they are configured for all types of transportation modes, so the freight is started to its shipment in a container moving by a truck, and then moving by a ship, after that by a train.

Among many types of logistics, the container logistics is the concern of this thesis. Our case study Bimar and ARKAS Inc. works in the container logistics domain. So the 'logistics' word means 'container logistics' hereafter in this thesis document.

Logistics business is basically transportation of some thing from somewhere to another in optimum conditions. Transportation's definition and its types are defined in below:

Figure 2.1. A container [8]

2.2.3. Transportation

"*Transportation is the movement of people, animals and goods from one location to another.*"[6]. Moving something/somebody from A to B point is transportation business. Doing this transport something in an optimum way, with maximum benefit and minimum effort is the logistics business.

There are modes of transportation; that are handled by ARKAS Holding also:

2.2.3.1. Road Transportation

Road transportation is the most common and known transportation types for years. Trucks are used for carrying containers. Road transportation is usually the first stage of the transportation; also the last step is performed by road. Get freight from the production area and transfer it to airway or ship to be sent to destination is generally begins by road transportation.

2.2.3.2. Railway Transportation

Railway transportation is carrying freight by trains along railways. It can carry more freight than a truck once and it is cheaper than road transportation but the railways

are limited and stable, so railway transportation cannot be used anywhere, there must be railway.

2.2.3.3. Marine Transportation

Transportation is performed by ships, containers are carried by ships. For the cross-country transportation, carrying by ships is better. My case study ARKAS Holding does its carryings usually by ships.

2.2.3.4. Air Transportation

Transportation that is performed by planes which is more expensive than other transportation modes but the fastest one. But this transportation mode is not our concern, ARKAS does not transport by airways.

Logistics management is; "*the process of planning, implementing and controlling the efficient, effective flow and storage of goods, services and related information form point of origin to point of consumption for the purpose of conforming the customer requirements*" [9]. Providing this process in an efficient manner with software implementation is called the 'Logistics management system'.

After explaining the container logistics, logistics management system; the aim of developing a monitoring tool for this logistics system can be understood. Logistics business is a complicated process; there are many people who work in this process, as well as various software systems as software. All points of this process have different input/output systems like one of them may be using Excel whereas other one may be using CSV file (comma-separated value). One of them may be using MS SQL Server as database management system whereas other one may be using MySQL. Where the MS BizTalk Server begins here by taking part is the 'make them talking each other'. The 'them' that BizTalk make taking are the all points that are in the process of whole transportation. A road transport order or a railway transport order is one of points in this whole process. As an official definition of BizTalk Server; "*BizTalk Server is Microsoft's Integration and connectivity server solution. BizTalk Server provides a solution that allows organizations to more easily connect disparate systems.*"[10]. Besides BizTalk Server makes the transportation points talking; Microsoft Business

Activity Monitoring (BAM) tracks the wanted data. The end user or software specialist or whoever the user of this system can decide which data will be monitored. And then the necessary arrangements are done by BAM (which will be explained later in detailed how these arrangements are done.). Finally the required data is ready to be monitored and displayed.

The aim of developing a monitoring system is important. That means; a logistics business process is complicated and there are many points to be monitored from different systems. So, here BAM takes over the role, to handle the monitoring work MS BAM was chosen and development of monitoring environment for logistics management for the case of Bimar Inc. is started.

2.3. Microsoft Business Activity Monitoring

This part is related to what is 'business activity monitoring (BAM)', reasons to choose it as monitoring tool and its benefits, as officially. And after that, its basic usage will be explained.

"*The term 'business activity monitoring' was originally coined by analysts Gartner, Inc. and refers to the aggregation, analysis, and presentation of real-time information about activities inside organizations and involving customers and partners.*"[11]. BAM is exactly what its name – it is a way and tool to 'monitor' the 'business activities' where the business activities are the parts of business processes that are wanted to be monitored as a required shape and in a desired time. As an official term; 'business activity' is "*what data is interesting for an item in the business process.*" [12].

The term "business activity monitoring" is used now as tool and method name more than just a term. When say "BAM", it is thought as "Microsoft BizTalk Server Business Activity Monitoring", even though there are many tools / softwares in name of "business activity monitoring" from various brands, like 'Oracle Business Activity Monitoring', 'IBM Business Monitor', "Adobe LiveCycle Business Activity Monitoring". Basically, they look like similar, but they differentiate in detailed features.

In this project Microsoft BizTalk Server's Business Activity Monitoring tool is chosen because in case of Bimar Inc, they work with BizTalk Server besides other Microsoft technologies, so choosing MS BAM occurred naturally.

Tracking, aggregating and presenting the required data and doing these as real time (or near real time) is a must for monitoring processes. These are must but when thinking where this data come from is the main point and the main problem of gathering data. Because business processes are heterogeneous and data can come from any system in this process. Gathering data from different systems by the time is like in the [13];

> "Quick and accurate information from different systems combined at one platform could prove to be vital for the survival of an enterprise. Easy accessibility of information by customers and information flow within businesses is a must for every enterprise. Information is gathered from disparate systems within an enterprise, which results in the enterprise not being confined to one technology based system." [13]

As mentioned, data is gathered from disparate systems and this task must be done as automatically and meaningfully. Here the BAM takes over the job; BAM's main duty is gathering data as automatically, after preparing its activities and views which will be explained after in detailed.

BizTalk Server, which is explained above, connects and enables talking the different parties (that may be agency, depot or port, in case of container logistics) that uses different systems from each other and also uses and accepts different kinds of input-output documents, like XML, EDIFACT standards. And BAM's job is gathering the required data from these different systems, documents easily while the data flows through BizTalk Server.

In Figure 2.2, BizTalk Server's components are showed figuratively and its possible relations to other systems. BAM take a part in this BizTalk structure. It can be used with BizTalk Server or not, it is not a mandatory part but it can be used and when it is used it provides benefits.

Using BAM with BizTalk Server is an advantage because they know each other but BAM could be used within other .NET based technologies as well as non-.NET technologies. It is made for monitoring data and it is a customizable environment.

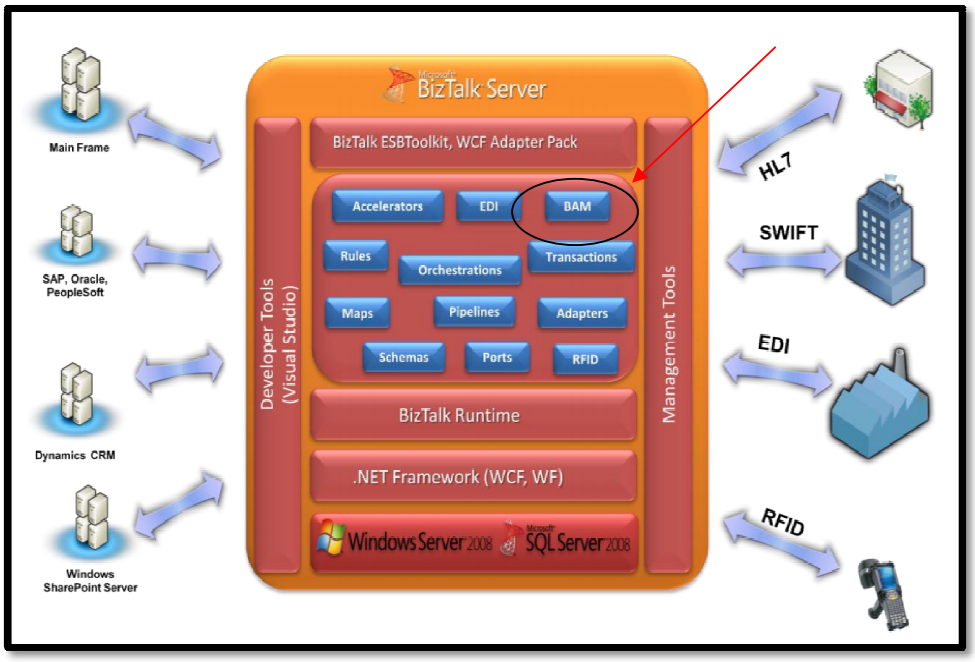


Figure 2.2. BizTalk Server overview [13]

The benefits of BAM are summarized in [16] as following:

- *BAM automates data capture and aggregation.*
- *BAM provides real-time visibility into business processes.*
- *BAM can alert users.*
- *BAM can capture data asynchronously*

2.3.1. BAM Users

There are three types of BAM users (see Figure 2.3) which are business analyst, software developer and business user (or end user).

- **Business Analyst:**

Analyst defines the important and required data and makes the aggregations. While doing this, he/she asks questions such as; "how much time after the order received, the container should be delivered to destination and is it happened in this time?", "the transportation order of this container is received when and when it is

delivered?". According to the answers that he/she gets his questions, he/she determines to form of the data to be gathered.

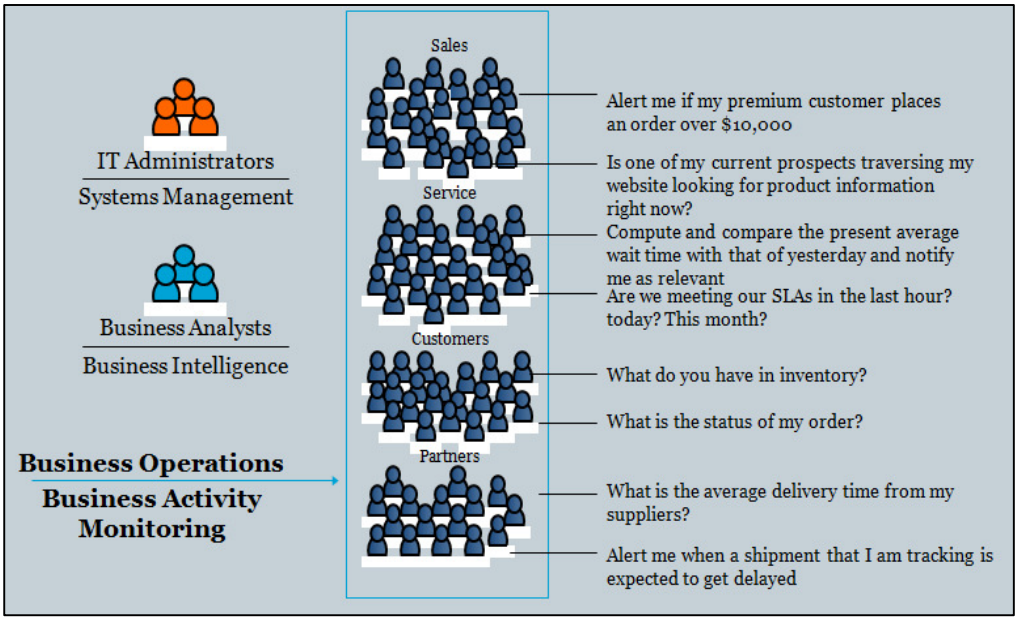


Figure 2.3. BAM Users with questions [15]

- **Software Developer:**

Software developer prepares the data model that is given by business analyst by using BAM. He/she uses BAM as a tool and forms the model of data to be able to get it.

- **End-User:**

This is a person who wants the answers of the questions that asked by analyst. End-user can see the results – the gathered data as real-time or he/she can monitor it at any time later. This user can be a customer or can be a software worker. It is up to the gathered data.

2.3.2. Features of BAM

Some terms are mentioned before parts of this thesis like activity, view. These are the given features of MS BAM that are needed to generate a BAM model.

- **Activity:** "*BAM activities are units of work which are used to gather business data*" [13]. When analyst decides to which information to be tracked, developer creates BAM activities through Microsoft Excel (which will be explained later.) according to analyst's chosen data. In fact, an activity is a data model that helps to gather data and is a database table that data is filled in it.

- **View:** A BAM view is actually a SQL view. It is created by using Excel and provides meaningful information from raw gathered data. "*BAM views helps in organizing and displaying information collected through BAM activities*" [13]. There can be more than one views for just one activity, according to need of data consumer; like one user must see some data from an activity in a way while other one must see all activity.

- **Aggregation and Filtering Data:** This feature is used and the result is seen on BAM views. When decided which data to be tracked for an activity, aggregations can be make on this data. BAM makes easy processing the raw data that before presenting it.

2.4. Platform of Microsoft Business Activity Monitoring

MS BAM has 4 main parts in itself as in Figure 2.4; tools, presentation, processing and database are these main parts:

2.4.1. Tools

- **Microsoft Excel:** "*The BAM Add-In for Excel provides a user interface that guides business analysts during the creation of Activities and Views. Excel serves as both a design tool for business analysts and a data consumption tool for business users*" [17]. As mentioned in BAM official web page, Excel is the starting point to develop a BAM environment. By BizTalk Server installation with BAM tool, an add-in comes to Excel (Figure 2.5) that is BAM short-cuts to create BAM activities and views.

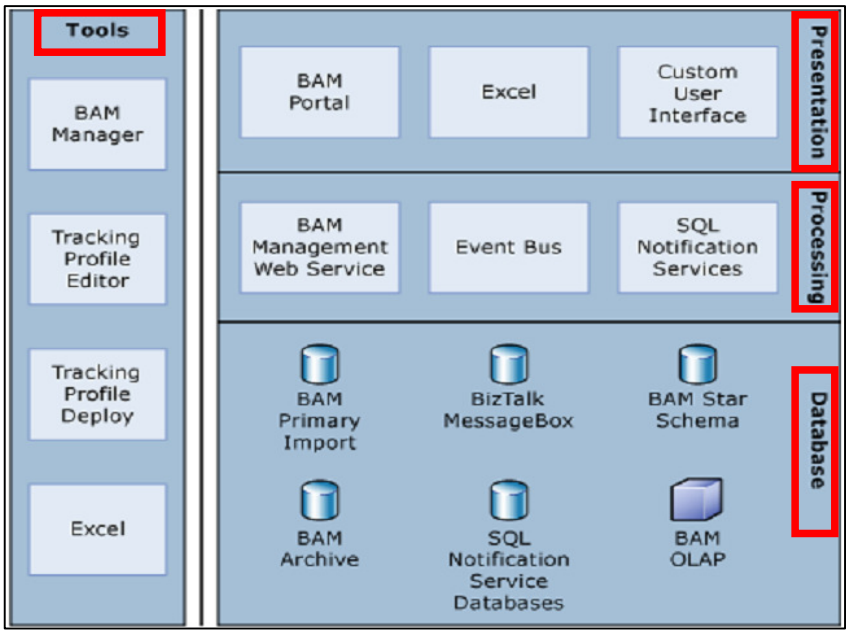


Figure 2.4. Architecture of MS BAM [17]

- **BAM Manager (BAM Management Utility):** After creating BAM activities and views in Excel, BAM Manager make possible to deploy these into system. "*The BAM Management Utility creates the necessary SQL Server databases, Analysis Services cubes, SQL Notification Services databases, and DTS or SSIS task*" [17]. There are some commands to deploy BAM definitions. They can be used through command prompt.

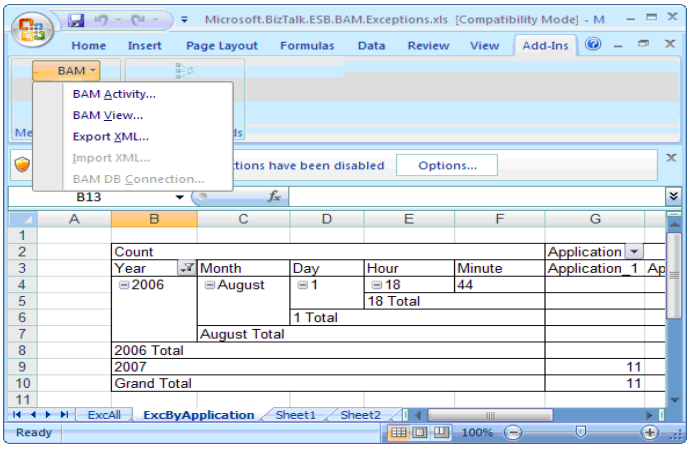


Figure 2.5. BAM add-in from Excel

- **Tracking Profile Editor (TPE):** TPE (see Figure 2.6) is an editor to map the BAM activities to flowing data through BizTalk Server. The source of data can be BizTalk Server orchestrations or messages which are input or output files.

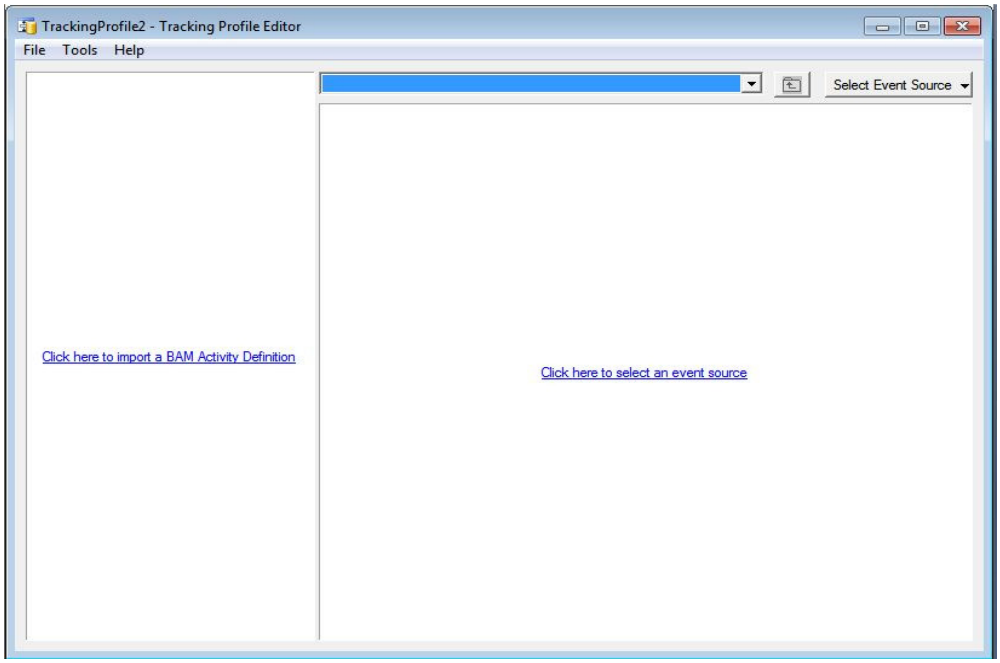


Figure 2.6. An empty Tracking Profile Editor

- **Tracking Profile Editor Deployment Utility:** Deployment unit of TPEs. This deployment utility's commands are used to deploy or remove a tracking profile.

2.4.2. Presentation

- **BAM Portal:** BAM Portal is a suggestion as in the Figure 2.7 from Microsoft for how to present the gathered data. It can be used as user interface but it is not enough for every system usually and there is a need to develop custom interface for presenting gathered data as meaningfully. As a matter of fact, Microsoft says that BAM Portal is just a suggestion, also a sample for presentation. The interface must be developed according to necessities.

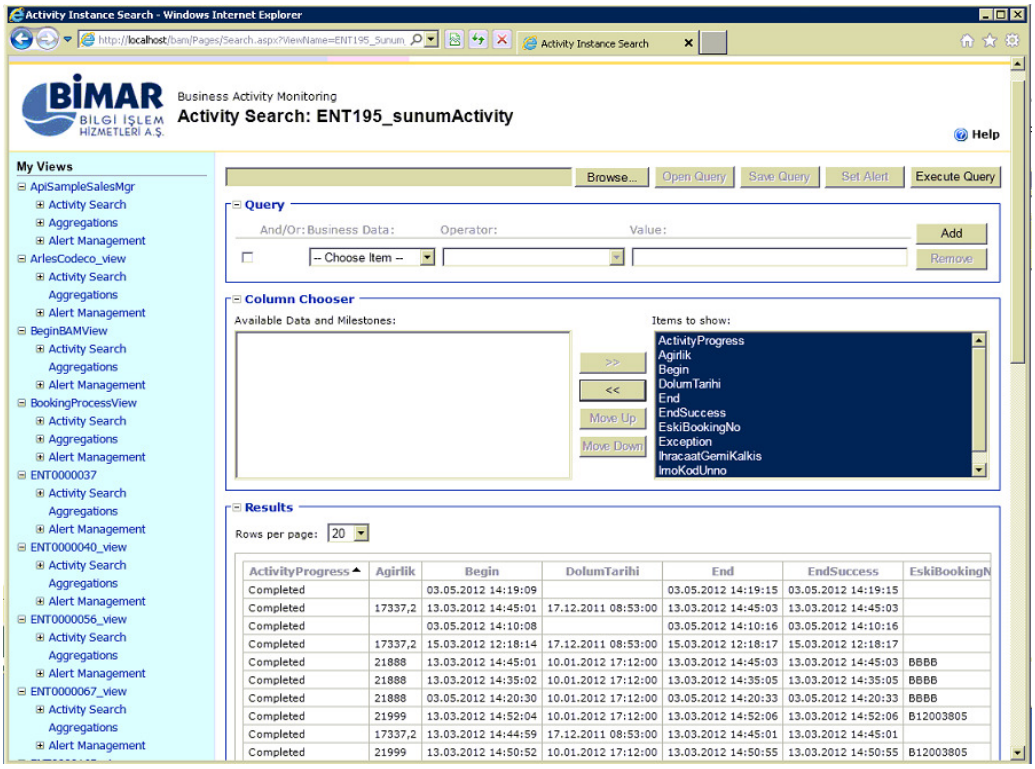


Figure 2.7. A screen shot from BAM Portal

- **Microsoft Excel:** If aggregations created by BAM views, there is one more option to see the gathered data; via Excel. Excel can show the views as a table and again real-time in its related BAM instance.

- **Custom Interface:** As mentioned in BAM Portal topic, custom interface can be prepared for the data that BAM gathers. This is the most useful option, because developers can prepare any interface according to needs.

2.4.3. Processing

- **BAM Management Web Service:** This web service is used for creating relationships between BAM databases and interfaces. BAM Portal uses this web service but it can be used in custom interface to get data from database.

- **Event Bus:** "*The BAM Event Bus Service processes tracking data (streams) stored in a source database and persists that data in a query table format in the destination database."* [17].
- **SQL Notification Services:** "*SQL Notification Services evaluates the Instance and Aggregate BAM alerts that are defined by the business user.*" [17]. These notification services are mandatory for working of BAM. They are installed during the installation BAM within BizTalk Server.

2.4.4. Databases

Databases are created as default during the installation of BAM within BizTalk Server. There are 5 direct MS BAM databases; BAMAlerts Application, BAMAlertsNSMain, BAMArchive, BAMPrimaryImport, BAMStarSchema besides the databases that are helper like SQLNotificationService database or BizTalk MessageBox database. The gathered data are hold in BAMPrimaryImport database within its activity table.

2.5. Example Use Scenario of Microsoft Business Activity Monitoring

There are ways of using BAM but in case of Bimar Inc., BizTalk Server orchestrations and input/output messages are used as for the main data sources for BAM activities. So, to describe usage of BAM, starting point is to assume that there is an orchestration that is already prepared. And according required data, analyst sets a data model from this orchestration and its input/output messages. Then this data model is implemented through Excel and deployed. After deploying BAM definitions, correlate BAM data model to data source by Tracking Profile Editor. Now, BAM is ready to gather data from data source. This is summary of usage BAM like in the Figure 2.8, now turn is to explain it in detailed.

To show usage of BAM, here a CODECO sample is selected. CODECO is a file type which is; "*A message by which a terminal, depot, etc. confirms that the containers specified have been delivered or picked up by the inland carrier (road, rail or barge).*" [18]. And there is an integration on BizTalk Server which is between ARKAS' own

depot and agency. According to ingoing and outgoing data, the key values in Table 2.1 are selected to be gathered and tracked by BAM.

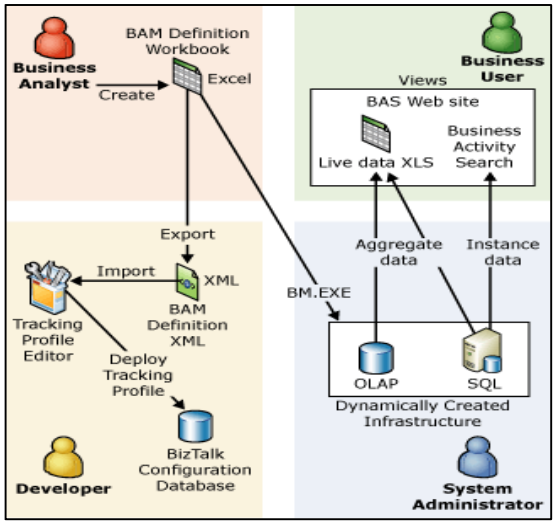


Figure 2.8. BAM usage steps [14]

Table 2.1: Data that will be captured from a CODECO Integration

	Information	**Type**
1	BookingNo	string
2	EquipmentPrefix	string
3	EquipmentNumber	string
4	MovementHistoryId	long
5	MovementLocation	string
6	ActualYear	int
7	ActualMonth	int
8	ActualDay	int
9	ActualDateHour	string
10	ActualDateMinute	string
11	ActualDateSecond	string
12	integrationCode	string
13	Begin	milestone
14	End	milestone
15	integrationDuration	bam

Information on this table is selected from this CODECO integration's orchestration (Figure 2.9):

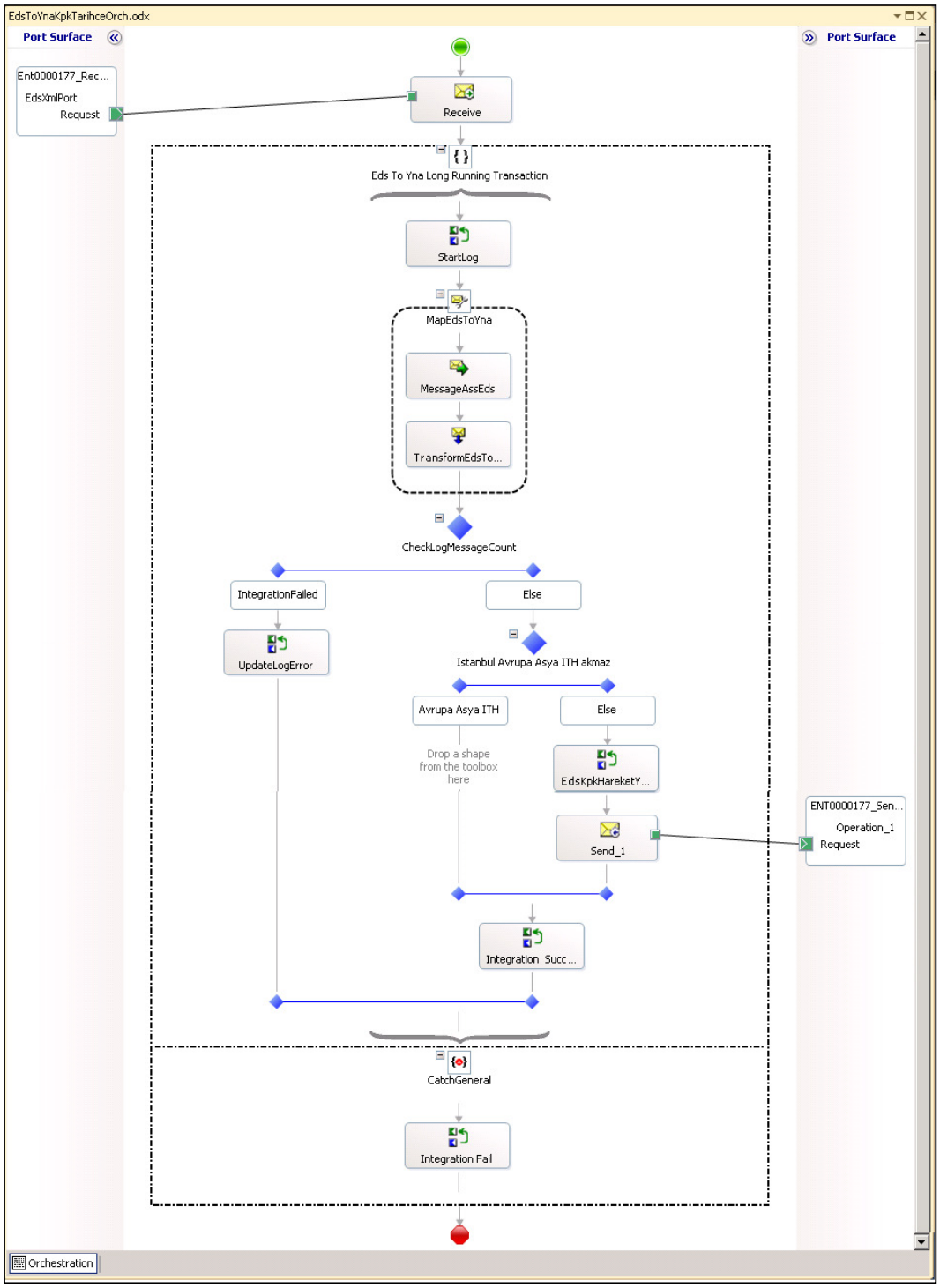


Figure 2.9. An orchestration sample from Bimar Inc.

When data model is ready, after that this model should be implemented by Excel; after that BAM development starts. As mentioned before, by installation of BAM within BizTalk Server, BAM add-in (see Figure 2.10) is added to Excel automatically. By clicking to this BAM add-in, BAM definitions can be created:

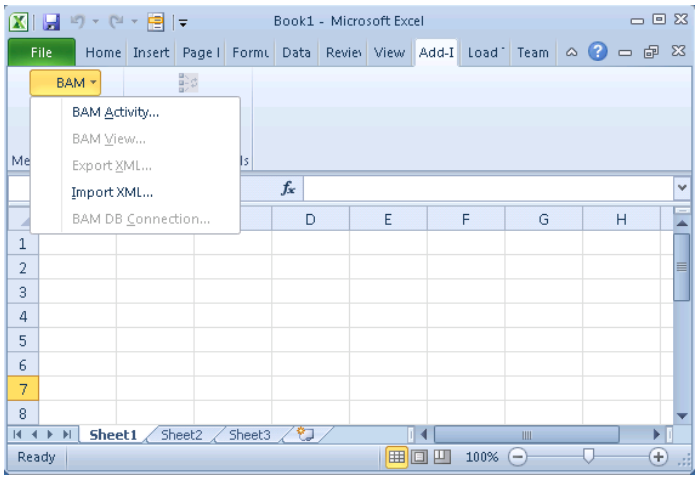


Figure 2.10. BAM add-in of Excel

Here, select 'BAM Activity...', then from coming screen, select 'New Activity' (Figure 2.11):

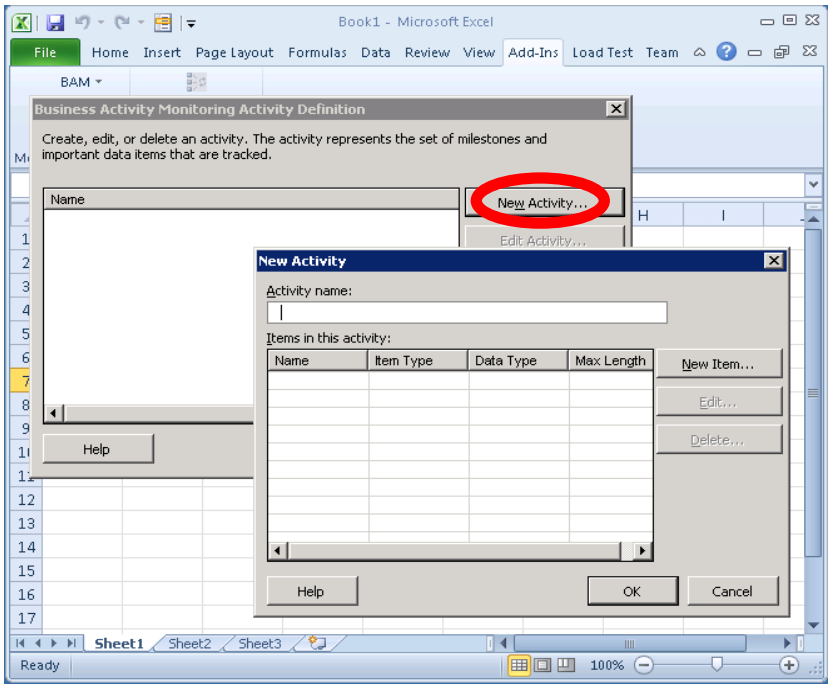


Figure 2.11. Create new activity

After naming the activity, by clicking the ‘New Item…’ button, information from data model is implemented one by one into activity.

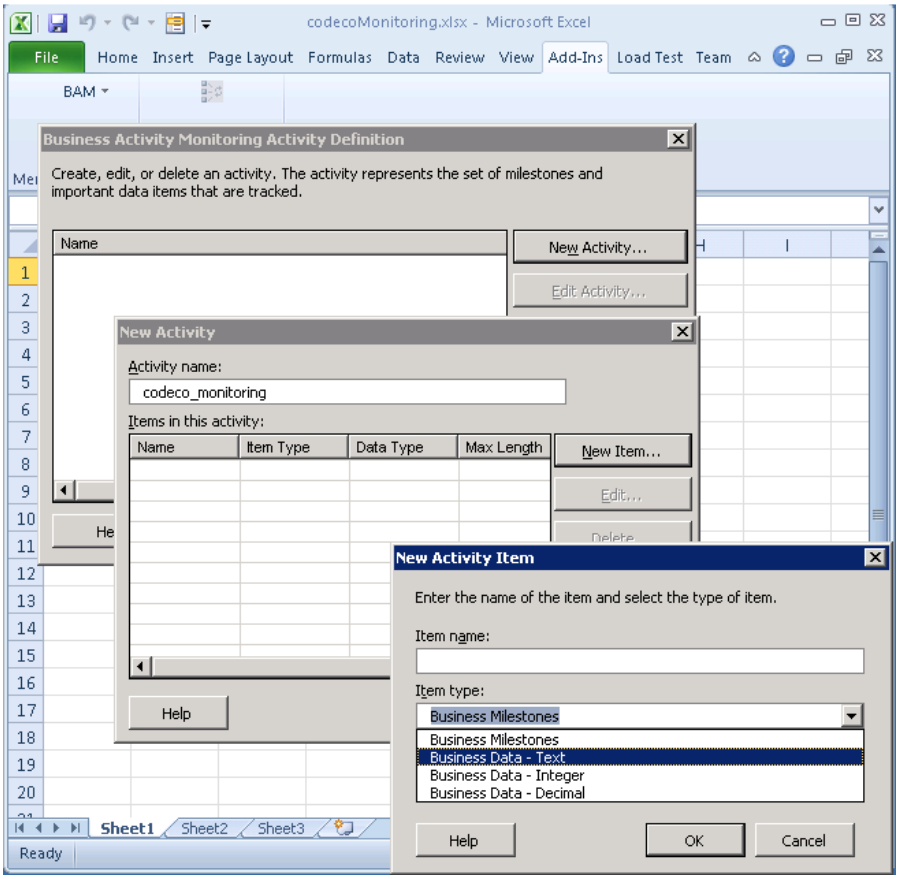


Figure 2.12. Create activity item

Here, there are four basic variable types for activity items, which can be seen in Figure 2.12. A **milestone** is date-time type to hold the information which is the time of occurrence. This time may be starting time of integration, or coming time of a ship, etc. **Text** type is for string variable where **Integer** is for integer numbers directly and finally **Decimal** type is for floating numbers like 3.221.

Data model’s activity implemented version is like in Figure 2.13. After all items are created, by clicking OK button in activity creation screen (Figure 2.13), view creation screen comes automatically. This screen (Figure 2.14) has three options; first one is creating a new view, second one is editing an existing view and the last one is deleting an existing view.

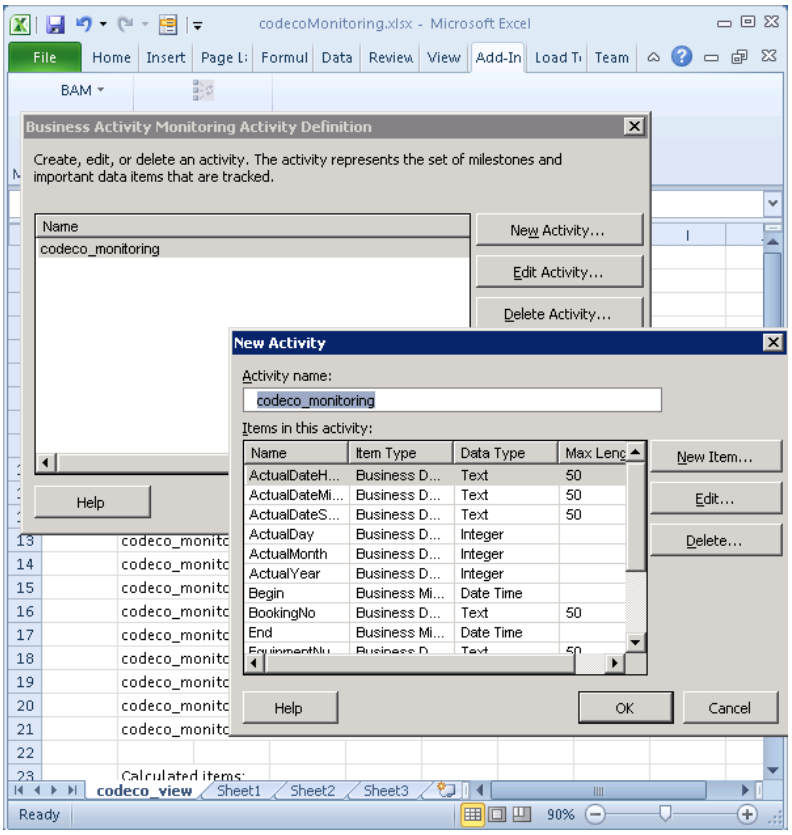


Figure 2.13. Created activity – 'codeco_izleme_TEST'

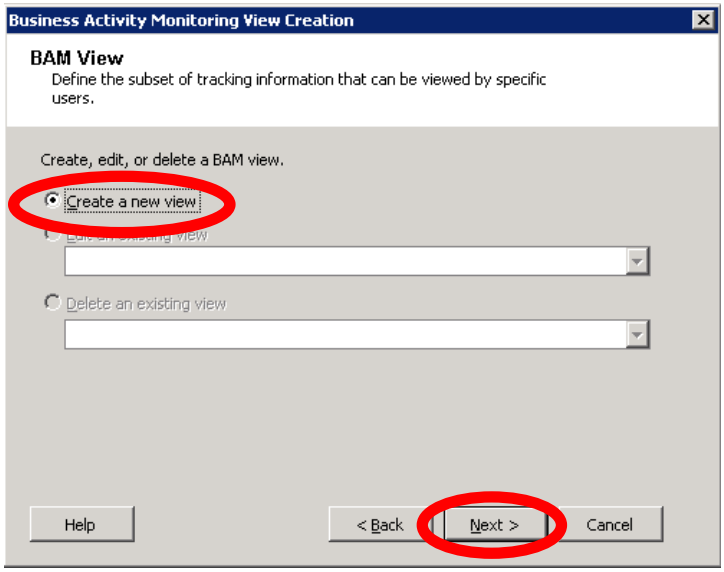


Figure 2.14. View creation

For the created activity ‘codeco_izleme_TEST’, a new view will be created. On this screen (Figure 2.14), click ‘Next >’ whereas ‘Created a new view’ is selected. After this, naming view and selecting activities for this view options will be on screen as in Figure 2.15:

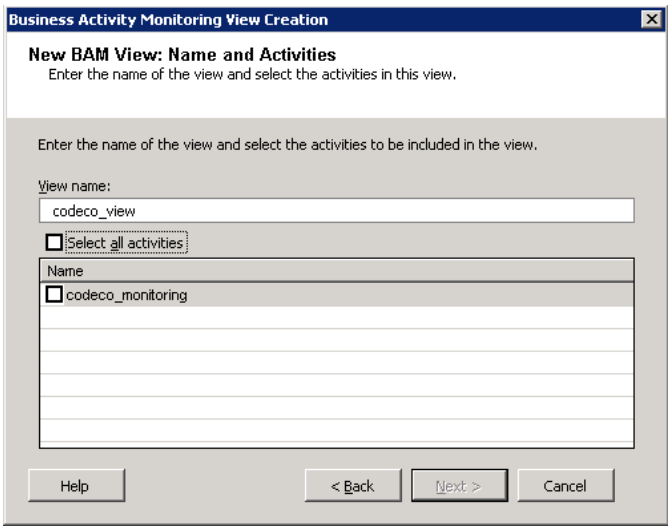


Figure 2.15. View creation – naming

A view can be formed one or more activities according to needs of end-users. Here, one activity that is created before (‘codeco_izleme_TEST’) is selected for the new view. After naming and activity selection, next screen is for selecting items that will be in view. As mentioned before, BAM view is like SQL view. It is formed as according to needs, there can be more than one view for just one activity.

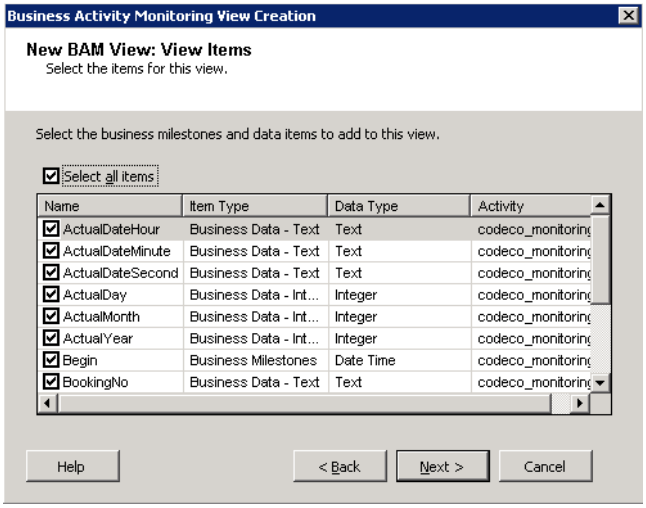


Figure 2.16. Items selected for creating view

There are 14 items in this activity (Figure 2.16), and there can be many views from this activity by some of these items can be included in view. Here, in this view, all items will be included and there will be **duration** value. Where **duration** is time between two milestones in an activity, it is stated as in Figure 2.17.

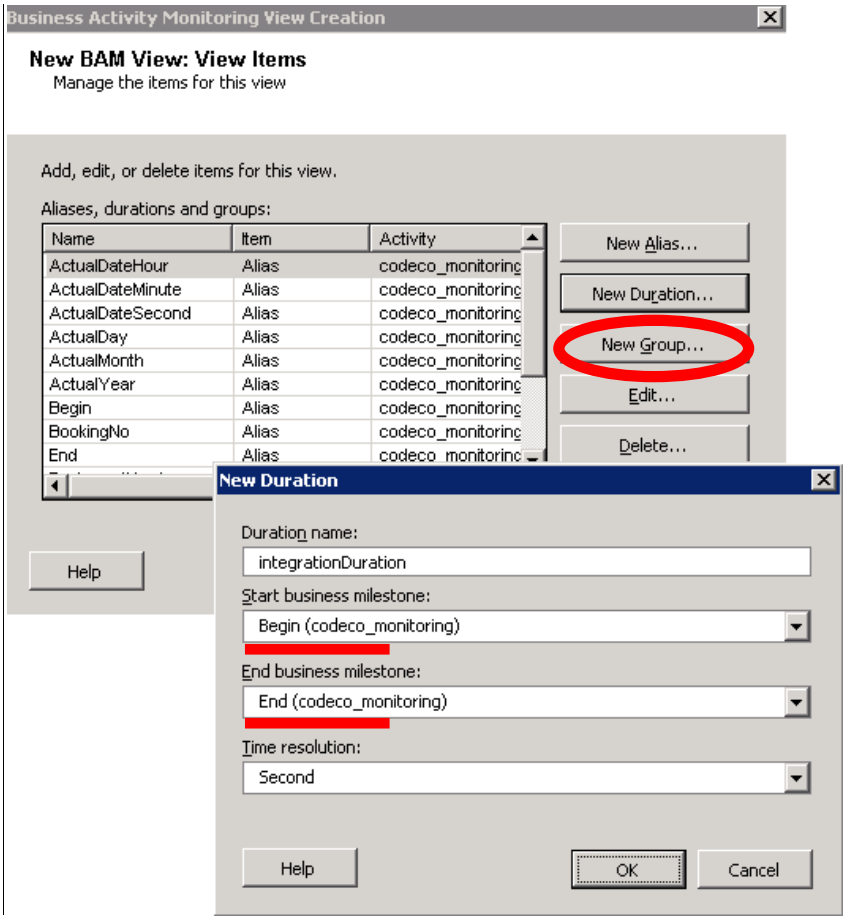


Figure 2.17. Create a duration item

In this sample, 'entSuresi' duration value is the difference between integration starting time and integration finish time.

There are other options in "Edit BAM View: View Items" screen. 'New Alias' option is for editing the name of any item, 'New Duration' option is explained above. 'New Group' option creates a group between two or more milestones to use them as just one milestone. A group can be used in some cases like; there are 2 possible ends for a process, it is finished as successfully or it fails and it has to finish. But either two ends, there is a finish time. So, to get this finish time, make a group from this to end time

either successful end or failed end (see Figure 2.18). After making this as group by the 'New group' option, this new group variable can be used as end time for process.

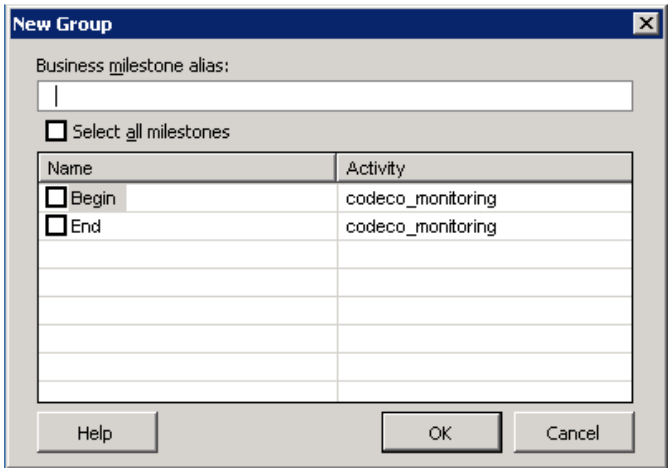


Figure 2.18. Create group from milestones

After creating necessary groups and selecting activity items, view creation is over. View and activity items can be seen on Excel like in Figure 2.20.

Next turn is to deploy this activity and view to make them working. BAM deployment is handled by console. There are BAM commands, by using them activity and view is deployed, also they can be updated by commands. To use these commands, console's domain must be changed to BAM's folder (see Figure 2.19). That is generally "C:\Program Files (x86)\Microsoft BizTalk Server 2010\Tracking", if the folder information is not changed during installation. If it is changed, the new folder's path must be used for the commands' path in console to be able to use BAM commands.

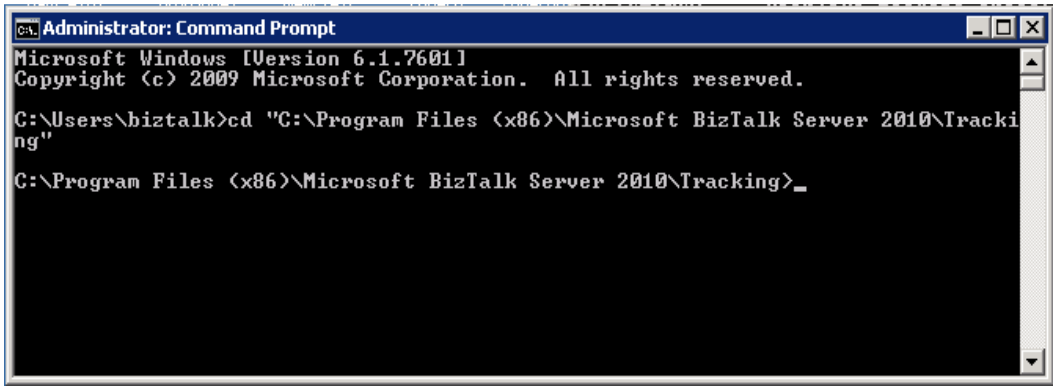


Figure 2.19. Console to use BAM commands from its installation folder

Before deploying activity and view, what are all BAM commands, that can be seen by the command ‘bm help’ like in Figure 2.21.

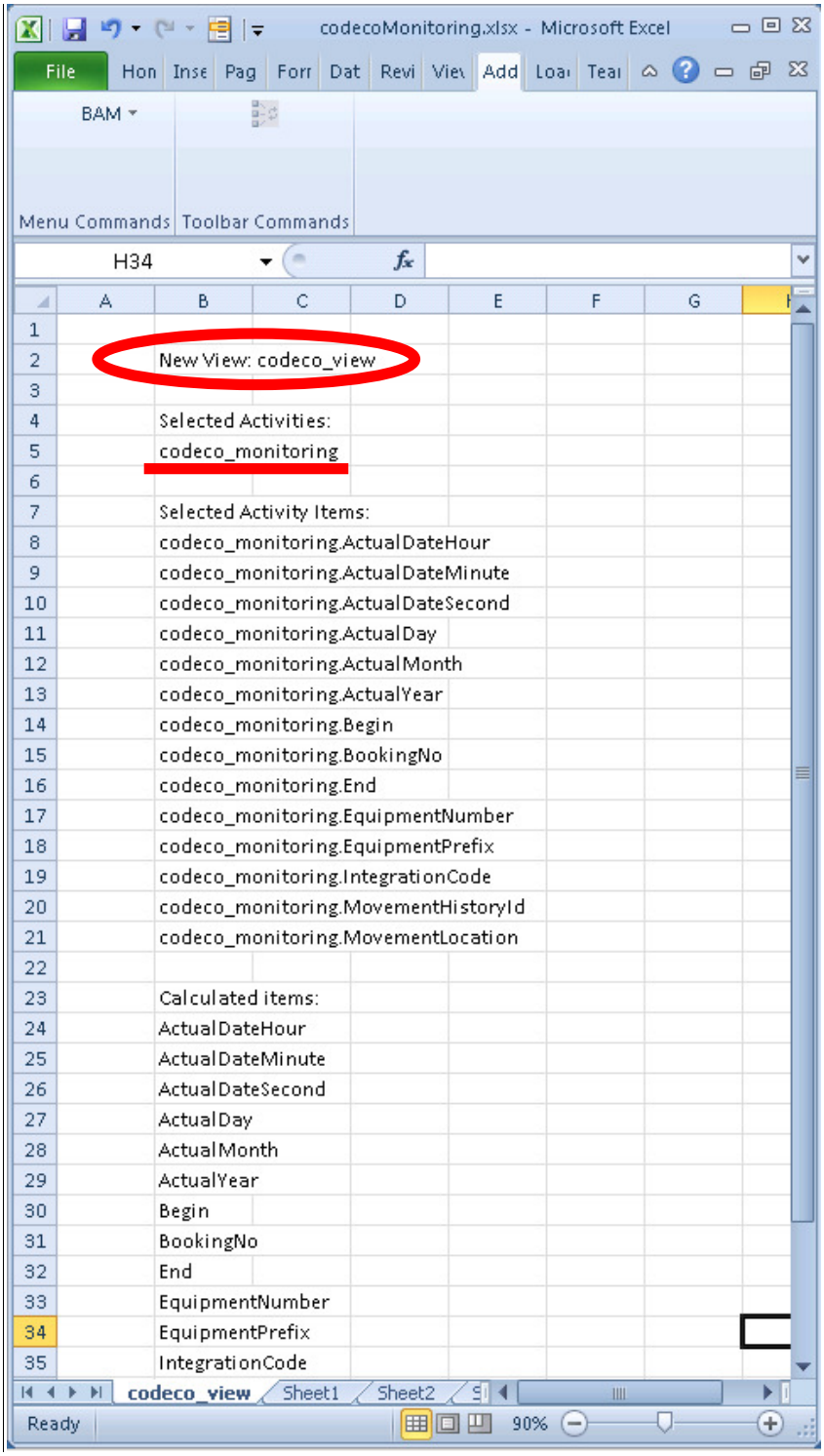


Figure 2.20. Activity and view after created

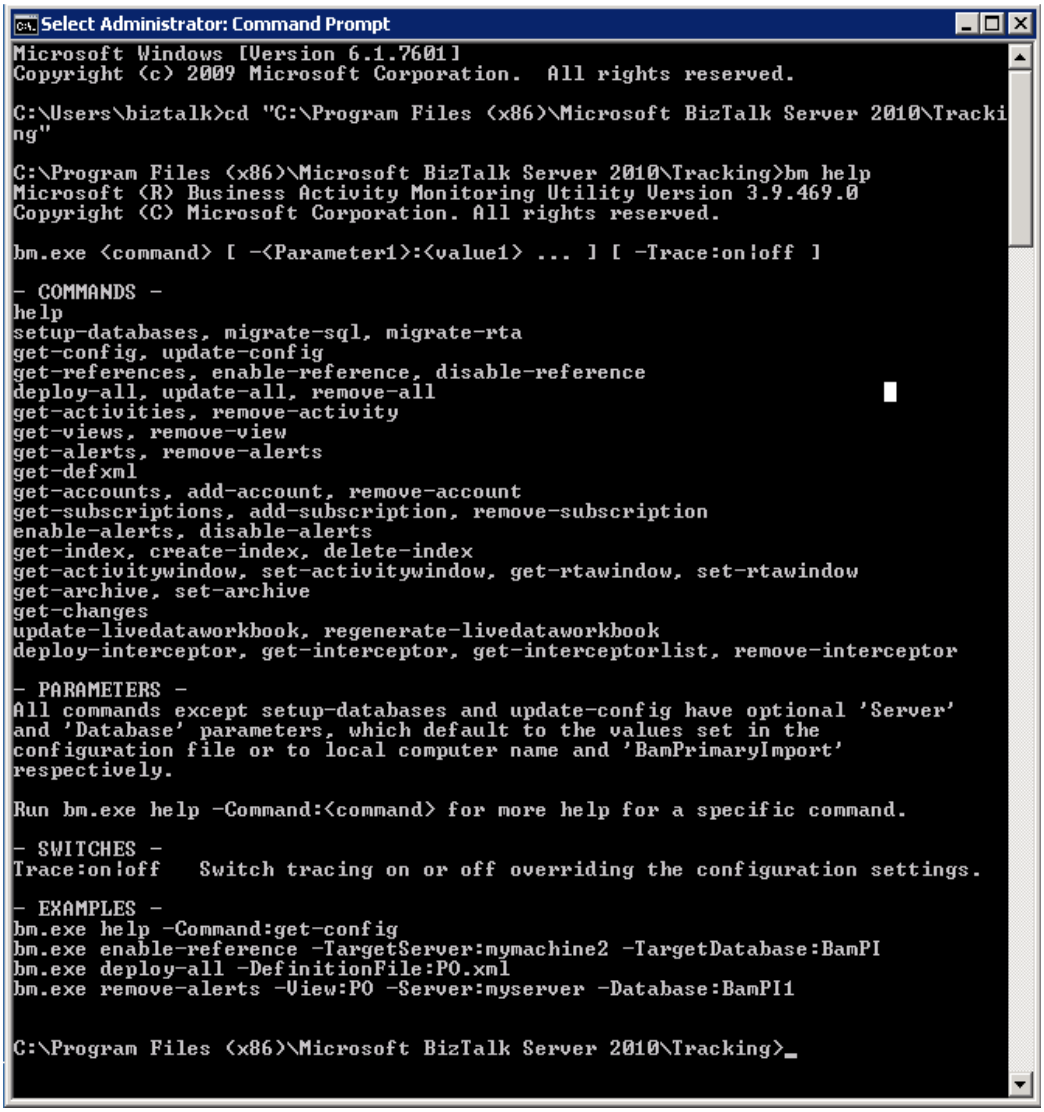


Figure 2.21. BAM commands

There are many commands of BAM to handle activity and views. For example, 'deploy-all' command is used for deploy a new activity and view. All commands begin with 'bm' word in console, such as;

'*bm deploy-all –DefinitionFile:" C:\folder..\activityA.xlsx*"'.

Or if an existing activity needs to be changed, 'update-all' command will be used. 'bm get-activities', to list all views; 'bm get-views' commands are used to list all activities. These are all the most common commands that are in use.

Now, to deploy the created activity and view; like in the above paragraph write ‘deploy-all’ command and they will be deployed into system like in the Figure 2.22.

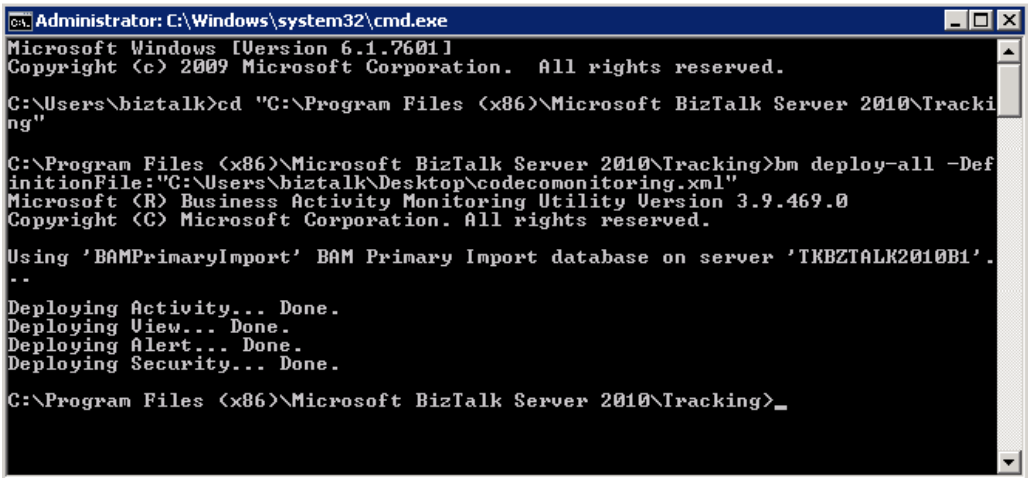


Figure 2.22. Deploy activity and view

Now, activity and view is deployed into system. The tables are created in BAMPrimaryImport database. For one activity, 5 tables are created in database. These are;

<activityname>_Active, <activityname>_ActiveRelationships, <activityname>_Completed, <activityname>_CompletedRelationships <activityname>_Continuations tables.

Most important ones are <activityname>_Active and <activityname>_Completed tables. If an instance is in <activityname>_Active table, that means its process is not over yet. After process is over, the instance is moved to <activityname>_Completed table. This is the final table; gathered data is collected here permanently.To present data, developer looks here this <activityname>_Completed table.

Where the data is gathered into is ready, but this data where comes from. By Tracking Profile Editor, as mentioned before parts, connecting the flowing data and the BAM activity is done. In figure 2.6, there is an empty TPE editor. From left side of it, the activity that will be filled is selected (Figure 2.23). The orchestration (or message) that is the data source is selected (Figure 2.24) from right side of TPE. And finally, the TPE is ready to match up data and activity (Figure 2.25).

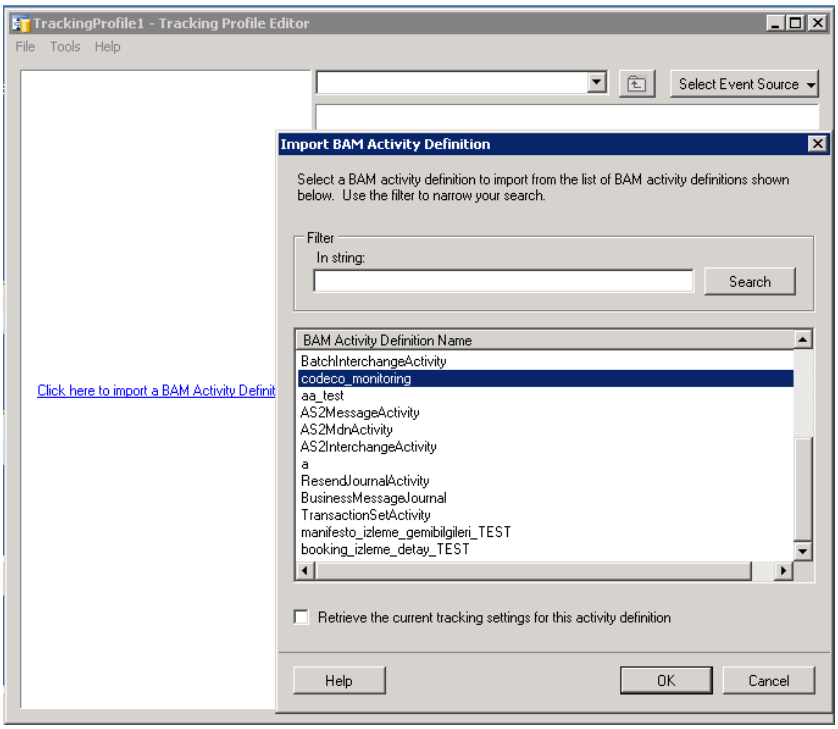


Figure 2.23. Activity import into TPE

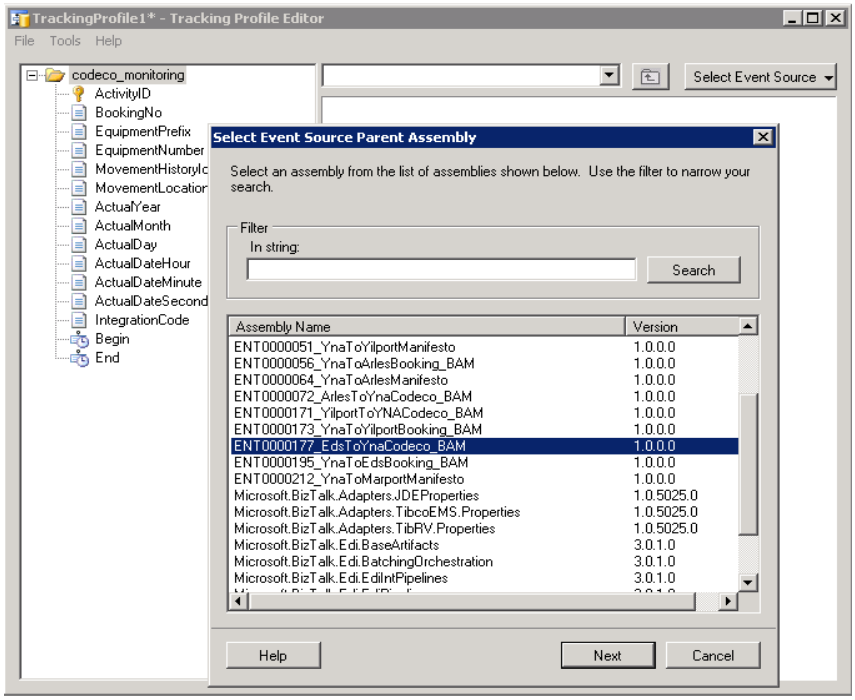


Figure 2.24. Orchestration import into TPE

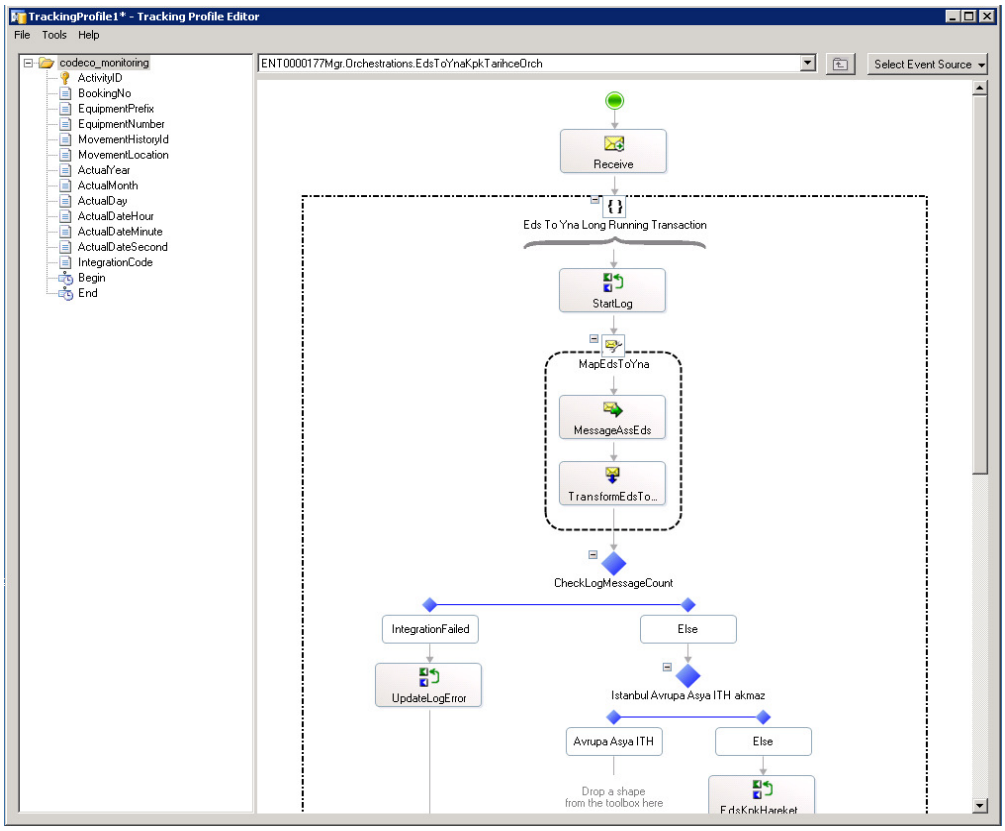


Figure 2.25. A TPE to match data and activity

The expression shapes in orchestration can be drag into items in activity in TPE. In that way, the connection is prepared. That means, if ‘receive’ shape is dragged into ‘begin’ item on the left, the receive time of the input file onto orchestration is gathered into activity. The required data must not be just from orchestration, it can be from messages that input or output messages. So, to get data from a message, click the button on right top of TPE ‘Select Event Source’, or right click on any shape of orchestration and select the necessary item. In this CODECO sample, data that is gathered will be getting from the map shape like in the Figure 2.26. After right click to map shape, ‘Message Payload Schemas’ selection is chosen then the schemas are listed like in the Figure 2.27. In this screen, after selecting the necessary schema (message), the schema listed on screen to match up with activity (Figure 2.28).

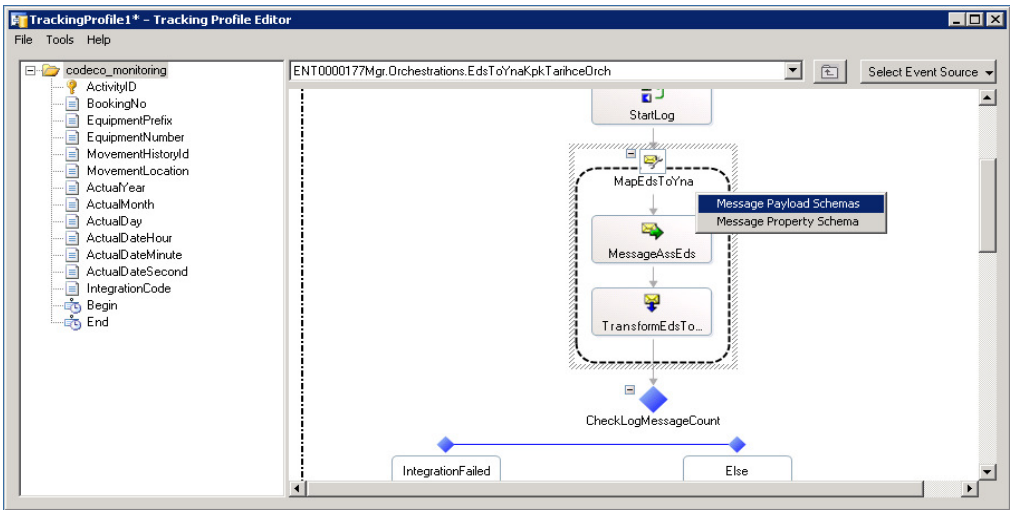


Figure 2.26. Get data from message inside an orchestration

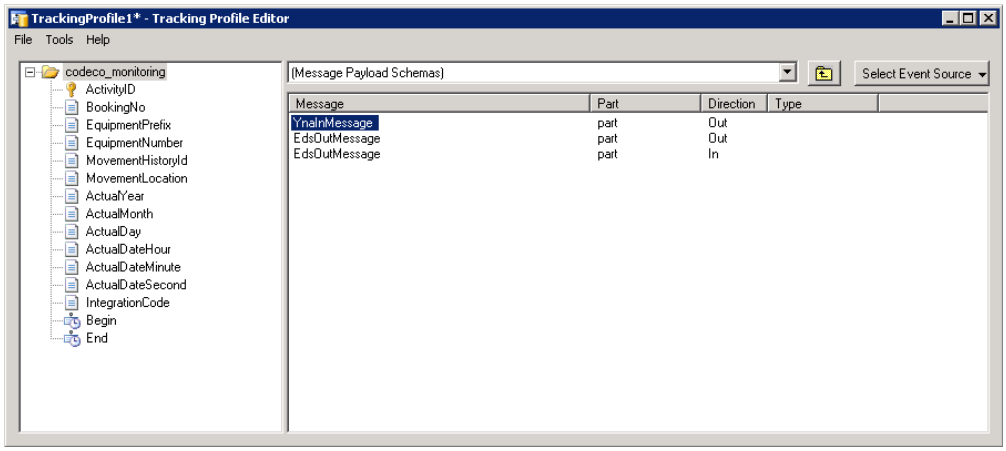


Figure 2.27. Listed schemas

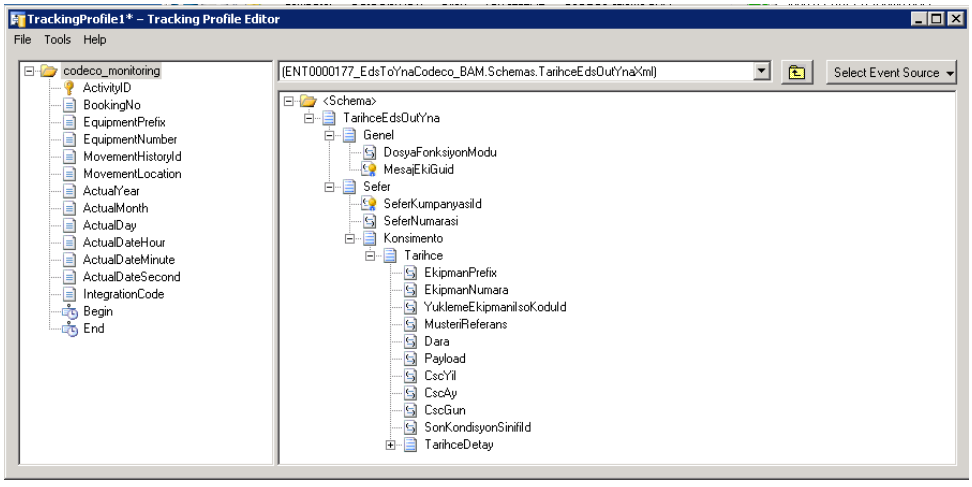


Figure 2.28. Selected schema's details

Like in the orchestration, in this schema sample, the schemas elements can be drag and drop into activity's items. After matching up the message's elements to activity's items, TPE looks like in Figure 2.29, the message elements are under activity items:

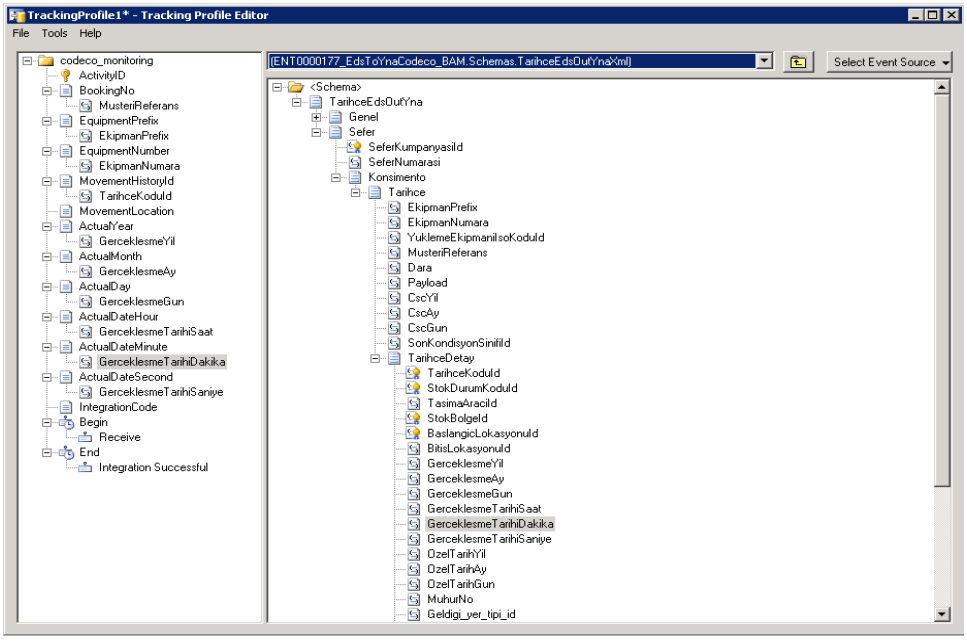


Figure 2.29. Matched up activity and integration

After matching up is finished and the TPE saved, then by applying the TPE, BAM is ready to gather data. From 'Tools' menu at right top corner of TPE, select 'Apply Tracking Profile' (see Figure 2.30). Now the TPE is applied. That means TPE is on duty now.

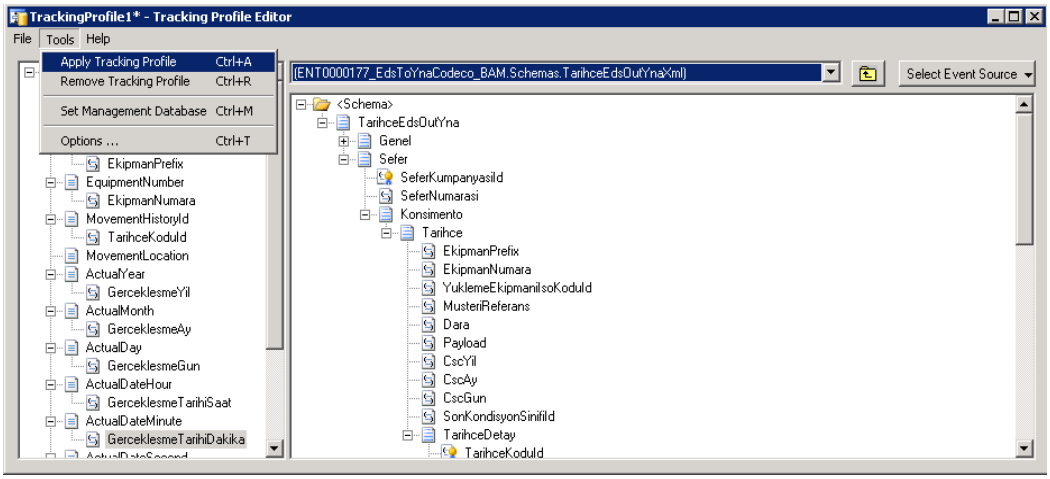


Figure 2.30. Apply TPE

Activity and view is prepared and deployed, and then TPE is prepared and deployed. Now, data is gathered into activity table (see Figure 2.31) each time integration works.

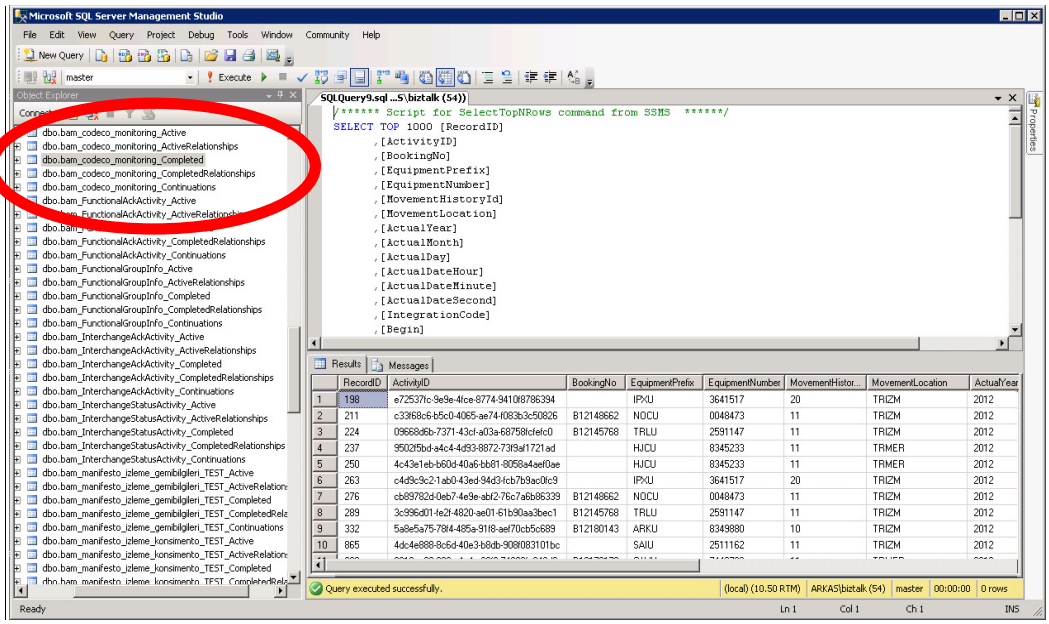


Figure 2.31. A db screenshot: activity tables and gathered data on activity

Data is gathered on activity tables, and now can be displayed on BAM Portal. BAM Portal, as it mentioned in part 2.3.2, is the presentation offering of Microsoft to display the gathered BAM data. Figure 2.32 shows the results of gathered data in BAM Portal.

In this Figure 2.32, columns that are underlined as red are the columns of activity table, so they are the items of activity that created in Excel. BAM Portal just gets the whole activity table from database without any interpretation on data. So there is a need for presentation tool from this point. And a custom portal is created in this thesis' scope which will be explained in Chapter 5 – Bimar Case Study.

In this chapter, the domain business area of thesis is explained besides the selected monitoring tool 'Microsoft Business Activity Monitoring' and its usage is explained with a sample scenario.

As addition to this background information, the methodology for finding the important values to be monitored will be explained.

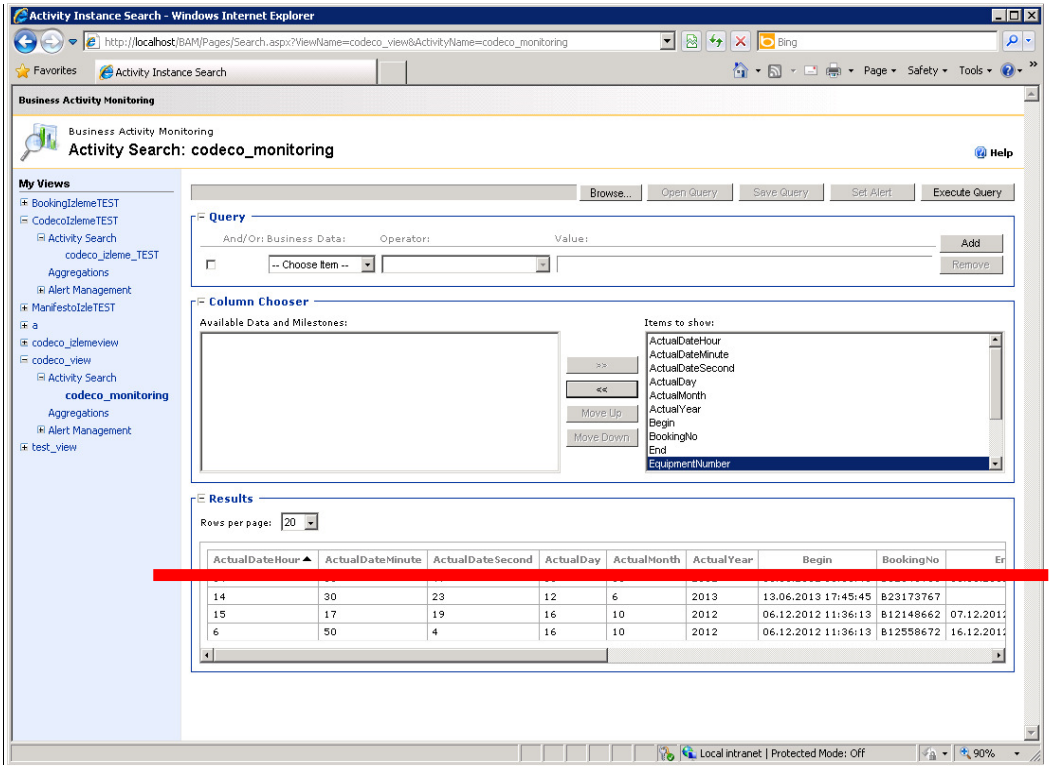


Figure 2.32. BAM Portal

CHAPTER 3

METHODOLOGY

In this chapter, the methodology for creating business processes and defining the important data to be able to monitor later by the solution business activity monitoring will be explained.

Container logistics is complicated domain with many business processes and workflows and there are many points that are needed to be monitored. As can be seen in Figure 3.1, the transportation processes can be very complex because of the plenty of points exchanging data in this process.

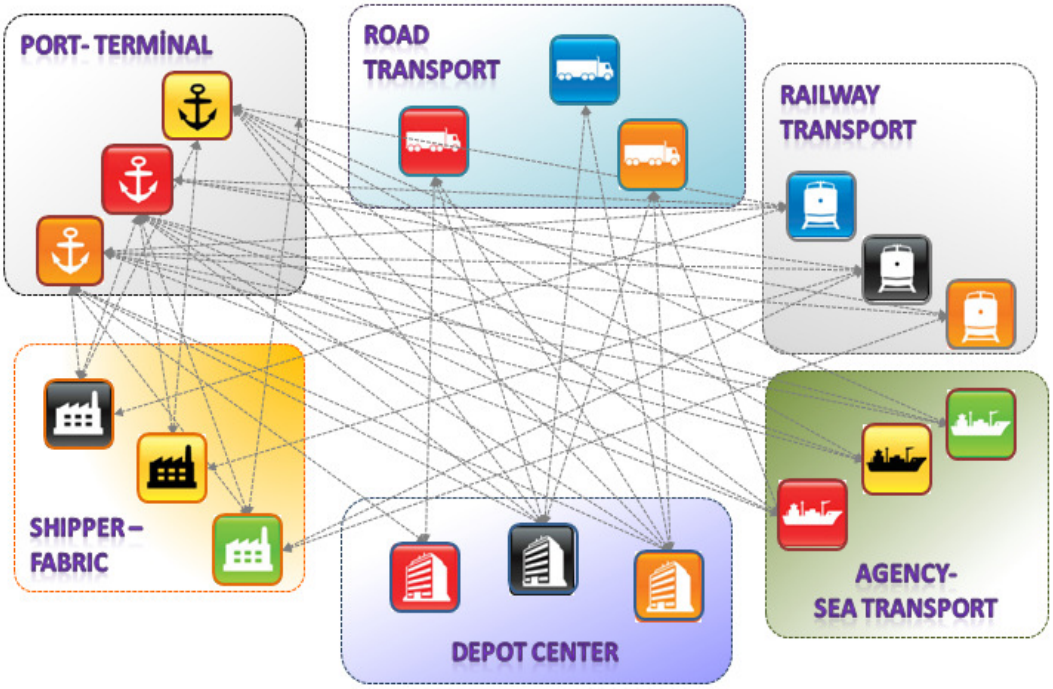


Figure 3.1. Intermodal transportation complexity

In Figure 5.2 in Chapter 5, there is a sample scenario to show the intermodal transportation sample process. There are different systems whereas different data types. The complexity can be incremented according to any other conditions. In such a

complicated sequence of processes, there must be a scheme to handle the all points in the whole transportation by not compromising any issue.

To form this scheme, there are some terms to define the situation and the technology. Starting with 'business process' is; "… *a sequence of steps that consumes inputs and generates output with added value.*" [21]. The scenario in Figure 5.2 can be sample to this business process definition but it is actually the sequence of 'business processes', that means there are more than one business process in this one scenario. So defining the business processes according to domain is important. For example, depot business process is different from terminal business process, but they both come together and create a different business process which consists of these processes. If there are different processes, which means there are different data is needed to be monitored according to the process. If the process is manifest business process, the 'loading time' is an important value whereas in CODECO integrations, 'historyID' is important and needed to be displayed. But by getting together these values, the main process' all important data and their details become revealed.

The business process whose definition is given above paragraph from [21], is a sequence of steps where these steps are different little business parts. So there are different data on every step. And there are different points to get data in different types. There are some definitions to clarify what the important data is:

- **Measure:** "*A measure is an agreed upon concept of quantification.*" [25]. Measure is defining the quantification of something. For example; distance in km, area in m^2.
- **Metric:** "*Term metric has more of a goal or performance nuance to it, as opposed to the fundamental or unit specific term measure.*" [25]. The difference to the measure is; measure is the value at the measured time but metric is the limit of that measure, for example when the house area is more than 150 m^2, its price is getting expensive.
- **KPI (Key Performance Indicator):** "*KPIs are measurable industry, department or task relevant performance metrics that are evaluated over a specified time period, and compared against acceptable norms, past performance or targets."*[25]. KPIs are the important data that is mentioned above.

Before a value is measured, which the value can be anything according to the domain of business and the processes, a metric definition is agreed and after the value is measured, its interpretation is through by the KPI; like the house's area is 200 m^2, and

according to the defined metric – 150 m^2, is passed out very much. And that the comment is the house is bigger than the metric, so it must be very expensive.

Akçay [24] says that "*If you can measure, then you can manage*". That clarifies the importance of using the KPIs.

> "In such a today's dynamic business environment, maximizing and optimizing business performance is a critical requirement for maximizing business profitability and returning shareholder value. For better business performance or continuous process improvement of an enterprise, effective measurement and analysis of the performance of managerial activities is essential" [22].

This quote explains the importance of measuring and managing a business.

The follower quote is from the application form of San-Tez project where this thesis is derived from this San-Tez project; Business events form the basis of the "Business Activity Monitoring" infrastructure. A business activity refers the set of data that is desired to deduct from business processes. Control points of Business activity monitoring can be considered as the key performance metrics or the definition of the data that is deducted from the ongoing business processes [20]. In this quote, it is indicated that controlling the data provides controlling the business.

As mentioned in previous paragraphs, the logistics sector is more complicated than the many business areas; therefore controlling the all systems in one screen is a need. To handle this, there must be a scale structure to show the performance of the system. This structure is measuring the values and interpreting the system according to gathered KPIs. So, first step is the KPI definition. Also the business processes are needed to be defined.

In Bimar case, the business processes are used for years without documentation. In this San-Tez project context, the documentation of the business processes of Bimar is done by interviewing the some employees who know the whole operation in their work. (See Appendix 1 for the business processes).

After the business processes are created, and then the next step is KPI definition. Some terms are given previous paragraphs, but which is which does not be known. There is a sentence that exists in every KPI or measurement of performance related documents, where the sentence is; "*all KPIs are metrics but not all metrics are KPIs*" [34]. So there may be metrics but the aim is to inference useful KPIs from them as it is said in [34]; "*While metrics can be a measure of just about anything, KPIs are the*

measures that matter most". Anything can be measured but which of them can cause to change something for sake of business. KPIs are the measured metrics but according to them, there could be need to change in process or data or anything that will affect the future of the process.

Transition from measure to metric is to give meaning to the measured value such as measuring the apples gives the result that there are three kilograms apple. Deciding to buy three kilogram from apples is a metric, which is determined according to the necessities. But metrics are not enough to help improve performance of the business. So there must be a transition from metric to KPI that that will help to improvement of performance besides the pleasure of customers.

To do this transition, there must be some criteria and decisions according to the needs, to provide the right improvement and getting meaningful results. "*Move from metric to KPI by first establishing a target from which we can calculate achievement*" [32]. These decisions must be made for performance of business, so it must be decided with the employees also managers.

There is a quote from the application form of San-Tez project that is the project this thesis is derived from it;

> "Business events form the basis of the "Business Activity Monitoring" infrastructure. A business activity refers the set of data that is desired to deduct from business processes. Control points of Business activity monitoring can be considered as the key performance indicators or the definition of the data that is deducted from the ongoing business processes." [20],

to show the KPIs are the control points that are decided where to put according to the monitoring needs. So defining which data will be KPI values and where to get these KPIs as a control point is the concern of this thesis to form the data set and then monitor these data.

There are steps for creating KPIs:

1. Divide the domain into parts such as; container logistics → terminal transportation
2. According to the focuses of all parts, define the important details
3. Work with employees also managers to get valuable results besides teaching them working according to KPIs
4. By starting to measure metrics, gather data

5. Track all metrics' effects on employees and according to the result, decide which will be KPIs.

To get powerful KPIs, these steps can be repeated until evaluation of settled KPIs. In Bimar case, the KPI finding process is performed with the related employees, also with managers. Which KPIs will be gathered is decided in such a process but where they can be got is in next paragraphs:

After business processes are documented and examined, the possible KPI values are started to detect by interviewing the employees of Bimar. To decide which values will be the KPIs and monitored by BAM, a general business processes document is prepared as in Figure 3.3. This business process document contains all types of movements of data in Bimar. And it consists also all the integration types in generally.

In this figurative process in Figure 3.3, between two shapes there is a detailed business process according to the integration type for example booking, manifest, and billing. And by this general vision on these integrations and their categories, the KPI deduction is started. After the integration categories are selected and listed as booking, CODECO and manifest integrations, which integrations will be monitored by defining their KPIs are decided and listed. Finally, the input and output files are revealed and the KPI definition process begins.

As it mentioned before, KPIs changes from domain to domain, there is not always a 'starting time' as a KPI, or a temperature value is not valuable information for many systems. So, after limiting the domain as container logistics, the next limit is integration categories (see Figure 3.3). There must be three stacks of KPIs because of the selected three categories of integrations to be implemented in this thesis' context.

After defining the integration categories, the integrations are decided to be monitored in this system. The integrations' list is given in Chapter 5 as Table 5.1. The interviewing with employees of Bimar for KPI definition process is started after integration list is formed. The business processes are passed through and the point to get KPI values are decided. The main points to get values are the input files that coming through the system. Because information comes by the input files and then interpreted in BizTalk Server and after processed, it is sent to the necessary output points. The coming data is important for the system because it can be sent as lack of information, or it can be sent with wrong data or wrong data type. Besides the possible deficiencies in

input data, the defined metrics are important for the control. This metrics can be; 'the going time of the ship cannot be before than the loading time of ship' or 'the container has dangerous goods inside and its control is performed or not in the loading?'

To get this integration number value, the input file is useless, so there must be a control point in integration's inside. The integration number is obtained from the Bimar's own created framework by custom code.

The integration's time variables can be control point for the system, for example the beginning time of the integration which is the input file's entrance time to system, or the failure time of the integration is valuable information. And again, they cannot be got from the input file. These can be got from the BizTalk Server's system information from the background.

To summarize the control points, they can be put in the input file (or it can be output file according to the system or the flowing data), the integration's system and BizTalk Server's system information.

After the control points are decided, the final list of KPIs is decided by working employees. So the values that are going to be gathered are ready. These KPIs lists are given in Chapter 5, Table 5.2, Table 5.3 and Table 5.4 as categorized in the integration types booking, CODECO and manifest integrations.

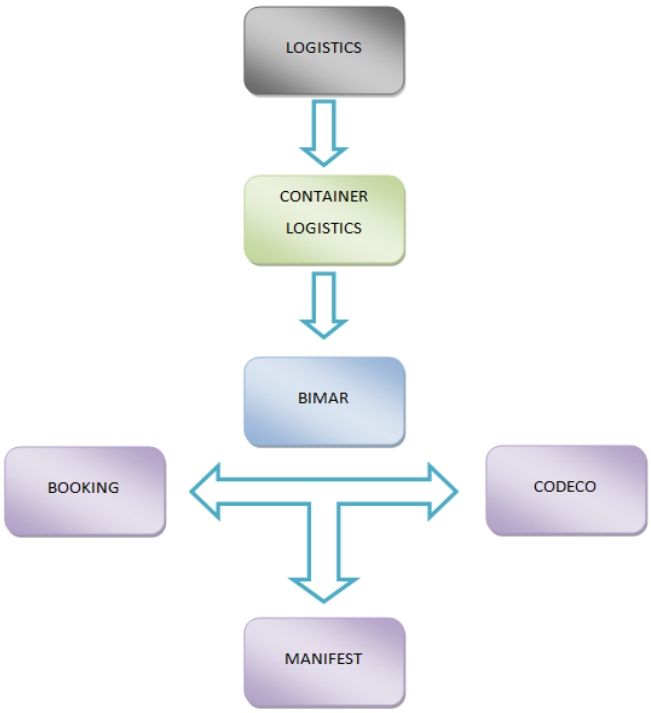


Figure 3.2. Steps of limiting domain to form the KPIs

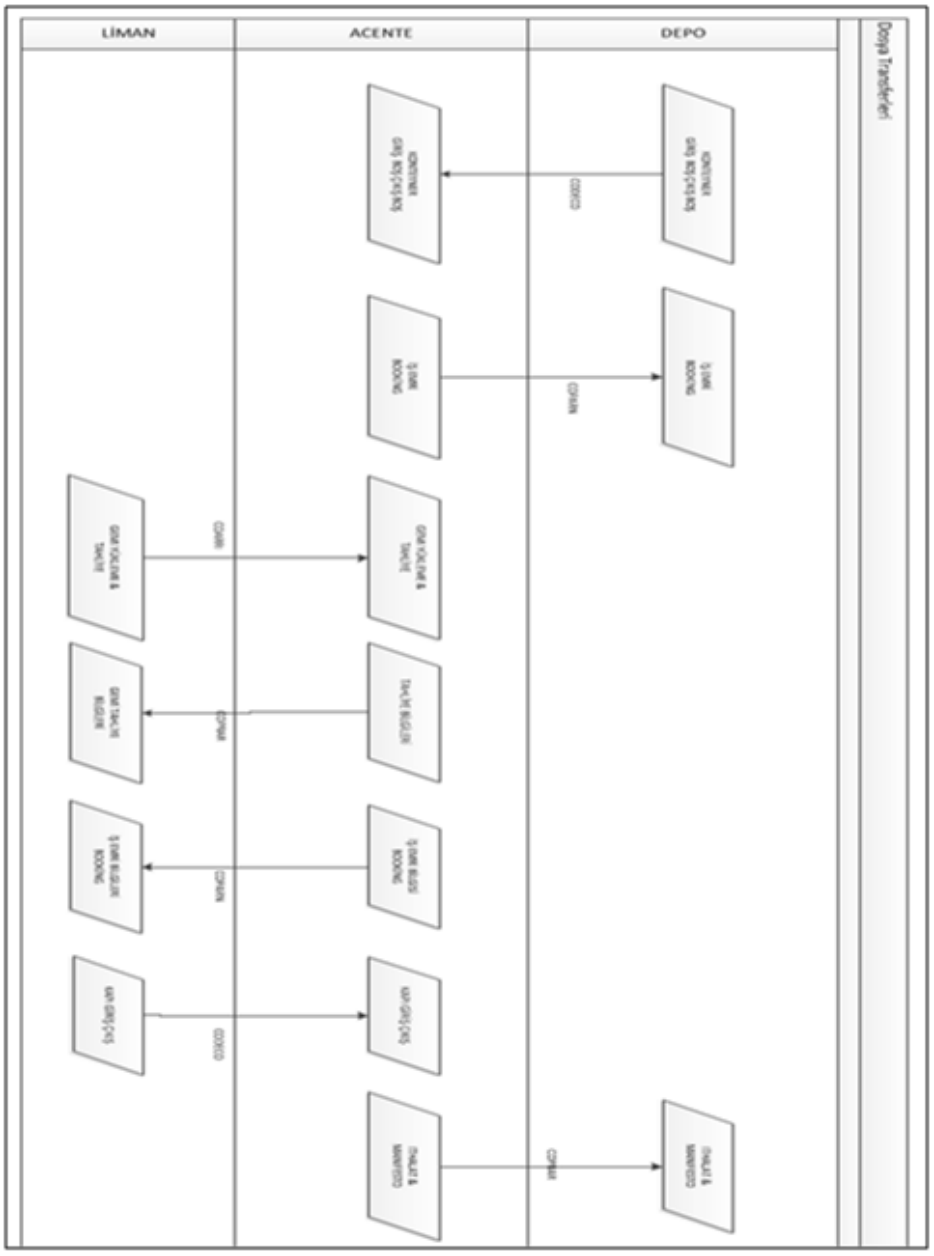


Figure 3.3. File transfers business process

This chapter explains the methodology of finding KPIs on business processes and the way of defining these KPIs. Next chapter will explain this thesis' monitoring solution after the information that is given as the monitoring tool MS BAM that is explained in Chapter 2 and the methodology that is explained in this chapter, Chapter 3.

CHAPTER 4

PROPOSED BUSINESS ACTIVITY MONITORING's ANALYSIS, DESIGN AND ARCHITECTURE

In this chapter, the proposed business activity monitoring solution and its architecture is explained.

Measure, metric and KPIs are defined and explained in Chapter 3. Here, how and where they can be gathered is explained. There must be structure in the system to be able to control it in anyway. And if a business system cannot be measurable, then it cannot be improved. There is a need to measure the performance of the system, and they will be by KPIs as it explained in Chapter 3.

There are metrics that are in every system from any domain. The important and should be considered issue is how this system will be improved continuously and how error handling are performed automatically. So, there is a need to monitor the system and so there must be a decided structure to create this monitoring system.

The metrics comes with the input files and the KPIs are already decided which of them will be gathered as KPIs. But what is the sequence of these steps, what is the structure of monitoring is considered like in Figure 4.1.

To handle this structure, the method 'Business Process Improvement (BPI)' is used, where the BPI is "*a systematic approach to help an organization optimize its underlying processes to achieve more efficient results*" [35]. As mentioned in this quote, BPI is a systematic approach. The structure of monitoring environment is created as based on this approach because the aim of the monitoring is improvement of the performance of system, and thesis' aim is handle this performance improvement by monitoring the whole system. So the monitoring environment is created according to BPI as adapted to container logistics.

BPI is creating business processes according to the performance improvement. BPI approach, "*focuses on achieving customer pleasure as signified by reducing costs, reducing waiting time, reduce complaints and increase turnaround time*" [38]. So the aim is customers' pleasure and the works are performed for this.

Figure 4.1.
Monitoring Flowchart

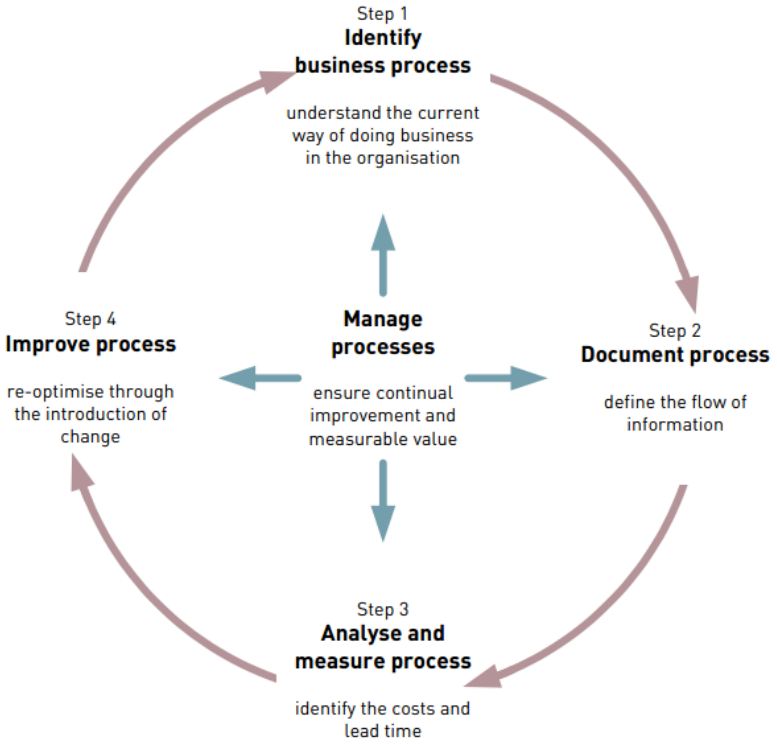


Figure 4.2. BPI cycle [36]

BPI works by:

- *Defining the organization's strategic goals and purposes*
- *Determining the organization's customers*
- *Aligning the business processes to realize the organization's goals* [38]

Business process improvement (BPI) implementation steps are listed as below [38]:

1. Develop the Process Inventory of enterprise

2. Establish the foundation/scope of improvement

3. Draw the Process Map of selected business processes

4. Estimate Time and Cost of selected business processes

5. Verify the Process Map - review the step 3

6. Apply Improvement Techniques on selected business processes

7. Create Internal Controls, Tools, and Metrics for business processes to minimize potential errors.

8. Test and Rework for seeing the step 7 works as planned

9. Implement the Change when needed on business process

10. Drive Continuous Improvement - according to above 9 steps, continue to improvement on business processes

These steps and aim are implemented to proposed monitoring environment as followings (Figure 4.2):

MS BAM tool is a useful starting point for monitoring problems. But it has deficiencies. It is not a whole package to be able to monitor a whole business system. It is a tool but it is not like install it then it runs properly and easily forever because of not being a plug-and-play tool. There are many arrangements to be able to implement it into the system but again there is lack of control points.

Because the monitoring issue differs from system to system and MS BAM implementation also differs from system to system. Besides its implementation and other deficiencies, the presentation tool that is offered by MS BAM is not enough to use efficiently. Monitoring issue is not just gathering the important information, without presenting it meaningfully and logically to the user, the gathered data means nothing.

After these deficiencies of MS BAM, this thesis' subject is cleared; by based on the MS BAM tool, the aim of this thesis' project became to create a monitoring

environment for container logistics area that will be useful, easy to use, understandable results and automated working area.

Before starting to implementation, the MS BAM's analysis is performed and explained in Chapter 2 and the domain analysis with the important value examination is performed and explained in Chapter 3. After the implementation preparations are performed in Chapter 2 and Chapter 3, the next step is to explain the proposed monitoring solution's architecture and design.

By provided deficiencies of MS BAM and the performed domain analysis, first step is to decide implementation steps and the architecture of this monitoring solution. To cover the MS BAM's deficiencies, some features are needed to be added to monitoring environment.

One of these features is creating solution to gather data from the looping structures as in the Figure 5.8. Direct data gathering is not possible from the looping structures by MS BAM. Because MS BAM takes any tag as a data and there is no explanation for that situation; if there is more than one same tag in the data source, which one is going to be gathered into BAM's system? The example for this looped situation is given in Figure 5.8, there are more than one 'Loading' tags in this file and when it is matched up to BAM activity in TPE, which 'Loading's value is gathered is ambiguous. As in many sectors, logistics' data is not a simple format, not always may be, but usually the data comes by more than one information like in 'Loading' tag sample. So there is a need to create an environment to get these looped data easily and without any data loss.

To handle this looped data problem, a solution is created. By using the MS BAM API, custom code is written to be gathered all data from looped structures by embedding this code into BizTalk Server's integrations. This looping structures problem is explained in Chapter 5 Part 5.1, in detailed.

Microsoft's monitoring presentation offer also is not useful, especially in logistics area. Because there are many KPIs and they are needed to be interpreted and then be presented to user. But Microsoft BAM Portal do not have an interpretation structure, it directly shows the all rows from BAM tables. So to handle this problem, a presentation tool; Bimar BAM Portal is created in this thesis context as an end to project which is explained at the end of Chapter 5.

By these deficiency detections and solutions for these deficiencies are added to BAM environment. There was a monitoring solution that offered by Microsoft and now there is a proposed monitoring solution in context of this thesis. The Microsoft's solution became a starting point for this thesis' solution. So, by taking Microsoft's offer as a base, some features that are explained previous paragraphs are added to this solution.

This thesis' proposed solution is created as in Figure 4.3. Except for 'Bimar Monitoring Portal' and 'Custom Code' parts, other parts are from the original Microsoft BAM tool, these two parts are the added features by concept of this thesis project.

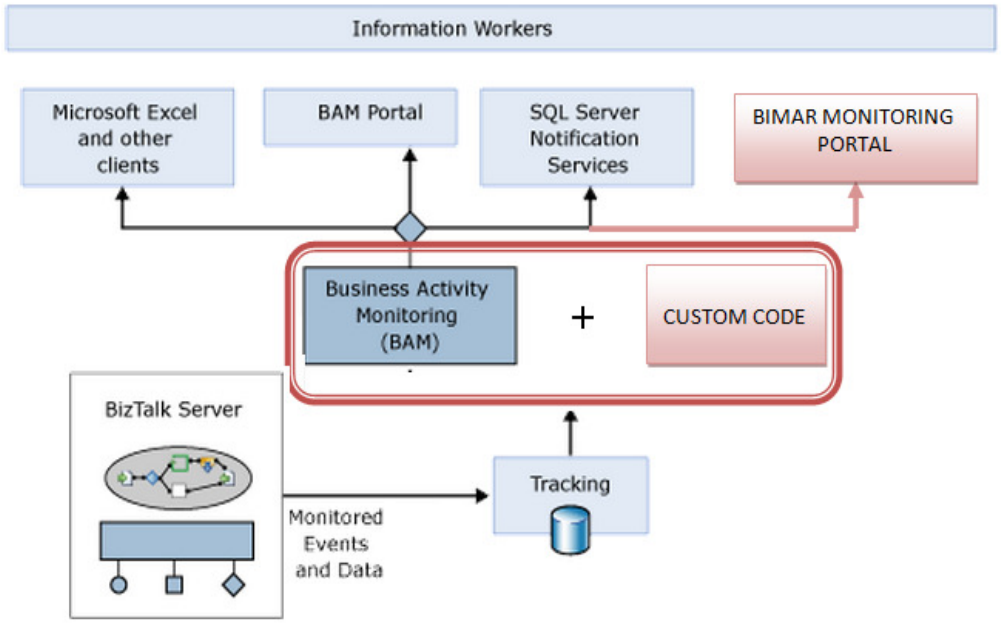


Figure 4.3. Proposed BAM Architecture [27]

This Figure 4.3 is superficial seem of structure. The architecture of the proposed solution is created according to SOA which is abbreviation of 'Service Oriented Architecture'.

> "Analysts have predicted, pundits have professed, professors have lectured, companies have scurried to sell what they had, as SOA products – often missing the point that SOA is not a product. It's about bridging the gap between business and IT through a set of business-aligned IT services using a set of design principles, patterns, and techniques." [21, 23].

As mentioned in this quote, SOA is not a product. It is the method for developing your business according to some paradigm like in said [13],

> "Service Oriented Architecture (SOA) can be thought of as a paradigm or an architectural style. It is not a framework or product that can be bought and used. It is more of a thinking approach which helps in designing interoperable software architectures according to enterprise business needs."[13].

This is actually an approach to design the business architecture. As it mentioned in [21], software is thought as different components that need to communicate each other. And like in the [13], it helps to creating interoperable architectures. The structure is that; divide the software into independent parts, and make them talking to each other. By providing this, the fault in any part of this software causes no fatal results in software in general because of the divided structure.

These independent parts are 'services' which are "*A Component capable of performing a task*" [28]. Whereas service oriented architecture is;

> "The policies, practices, frameworks that enable application functionality to be provided and consumed as sets of services published at a granularity relevant to the service consumer. Services can be invoked, published and discovered, and are abstracted away from the implementation using a single, standards-based form of interface" [28].

The benefits of implementing the system according to SOA;

- *The promotion of reuse*
- *The ability to combine services into new, composite applications*
- *The use of loosely-coupled services through a standard interface* [29]

Dividing the structure into services provides the above benefits, besides these, the implementation of the project gets easier because implementing the small parts is easier than implementation of the whole system. By starting small steps, the project's future improves more and more. And by combining the finished services, the systems appear as modules.

MS BAM itself is a web services combination actually. On a BAM installed server, if it is typed on browser as:

“http://localhost/BAM/BAMManagementService/BamManagementService.asmx”

Then the result page (Figure 4.4) shows the web services that run at background of BAM. Its architecture already in a SOA structure, so implementation of monitoring environment as SOA is easy and logical.

After the required changes are performed, the necessary parts are added to system, the sequence diagram for the monitoring process becomes as in Figure 4.6. Also, there is a figure (Figure 4.5) from BAM book [16] that shows data flow in MS BAM. The sequence diagram of proposed monitoring solution and these two figures are overlap as logically.

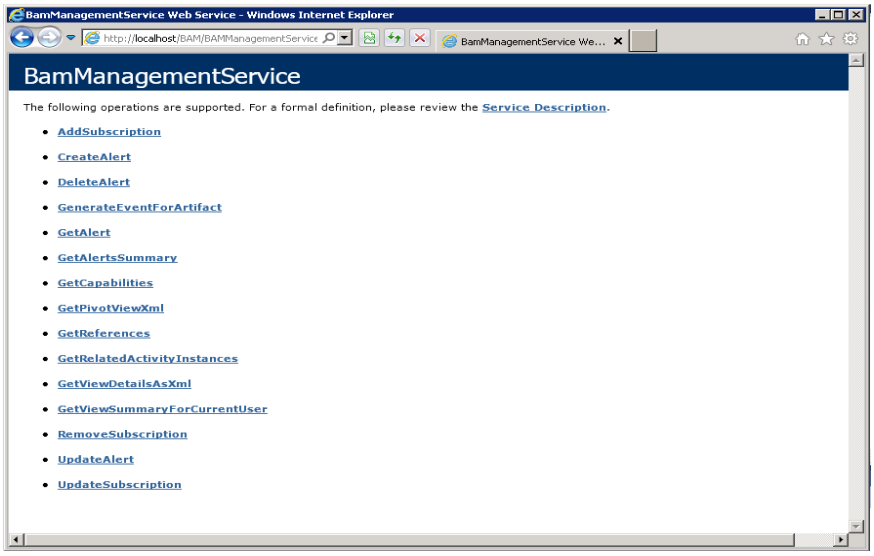


Figure 4.4.BAM Management Services

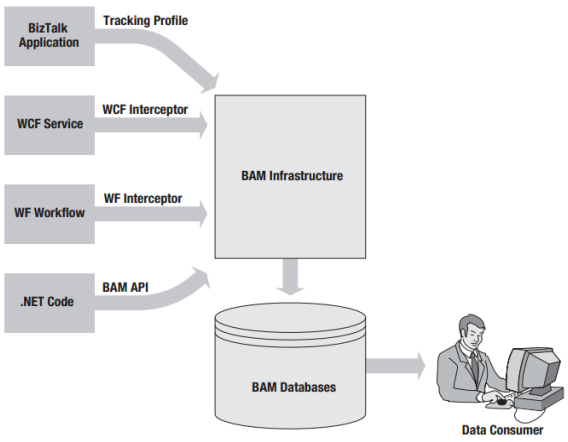


Figure 4.5. MS BAM data flow [16]

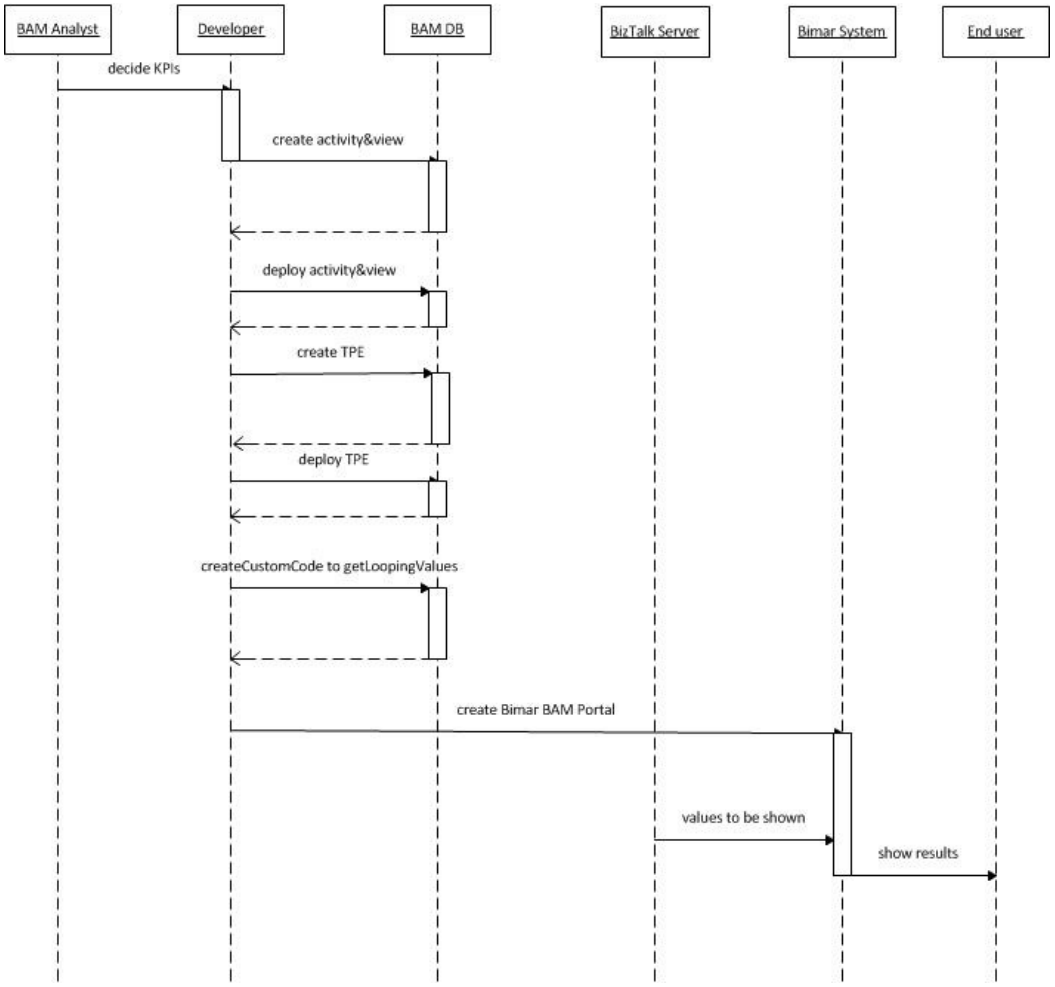


Figure 4.6. Monitoring process sequence diagram

This sequence diagram is working on business processes that are created in this thesis context. These processes are Bimar's business processes that are in use and documented according to Workflow method. Workflow is rules combination that defines the steps of events which happen connected to each other. The business processes of Bimar also events that comes after the related event is performed. As an example, there is depot business process in Appendix 1.3; depot process begins with integration's output to depot system that starts the process. According to this data file, freight is unloaded to depot. Then unloaded freight containers are examined and are got a grade according to its situation, and goes like that.

This method is chosen because of its benefits:

- *Improvement in the business process efficiency because of the less labor power.*
- *Standardization of business processes.*
- *Preventing the data loss because of using no physical document.*
- *Automating the business process tracking* [30]

Because of these benefits in [30], business processes are created according to workflow structure.

In this chapter, the proposed monitoring solution for the container logistics area is explained. The proposed business activity monitoring solution is based on the MS BAM tool and the methodology that is explained in Chapter 3. The analysis, design and the architecture of the proposed solution's implementation will be explained in next chapter, Chapter 5.

CHAPTER 5

BIMAR CASE STUDY

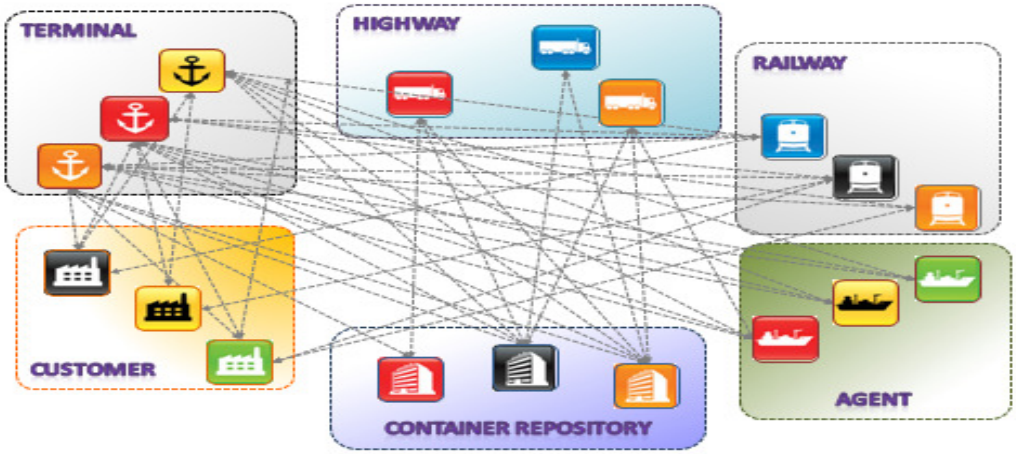


Figure 5.1. Intermodal transportation complexity [20]

In this chapter, the implementation of BAM for Bimar Inc is explained. As it can be seen from the Figure 5.1, logistics can be very complicated, especially intermodal transportation is more complicated than other types of logistics. To handle this complexity, the first step of the project was analyzing the Bimar structure. So, this thesis project is started by evaluating the business processes like in the Figure 5.1. In case of Bimar, business processes that Bimar uses were examined and these business processes documented as Microsoft Office Visio document. There are five types of business processes with this analysis work, which are agency-exportation business process (Appendix 1.1), agency-importation business process (Appendix 1.2), railway business process (Appendix 1.3), depot business process (Appendix 1.4), terminal-container entrance business processes (Appendix 1.5), terminal-loading/unloading business process (Appendix 1.6) and highway business process (Appendix 1.7).

After documenting the business processes, the integrations are revealed by examining the every business processes. These individual integrations are stated with their input/output files. The integrations that are monitored by BAM are collected in three categories: First one is the booking integrations that are shipping orders. Second

one is CODECO integrations that are the answers to booking integrations like the loading to ship is done, or evacuation from depot is done. Finally, the manifest integrations that are the shipping orders which is for a whole ship. In these three categories, 14 integrations are selected to be run with BAM:

Table 5.1. Integrations to work with BAM

Booking
ENT0000037_YnaToCatlogicBooking
ENT0000050_YnaToMarportBooking
ENT0000056_YnaToArlesBooking
ENT0000173_YnaToYilportBooking
ENT0000195_YnaToEdsBooking
CODECO
ENT0000040_MarportToYnaCodeco
ENT0000072_ArlesToYnaCodeco
ENT0000171_YilportToYnaCodeco
ENT0000177_EdsToYnaCodeco
Manifest
ENT0000014_YnaToCatlogicManifesto
ENT0000051_YnaToYilportManifesto
ENT0000064_YnaToArlesManifesto
ENT0000212_YnaToMarportManifesto

The next step is preparation of the development and implementation environment for BAM framework that is the subject of this thesis. A server is started to work for BAM integrations with in this context. This server is installed as a Windows Server 2008 R2; this is the necessary operating system for BizTalk Server 2010. After that, MS Visual Studio 2010 and SQL Server 2008 are installed. The Excel is mandatory for normal BAM implementation, so it is installed. After all installations, BizTalk Server 2010 is installed with the necessary settings for BAM configuration. Finally, the server is ready to run integrations on BizTalk and monitor them by BAM.

After the monitoring environment is prepared, working on this development server is started with moving these integrations on this server without BAM at first. These are run on MS Visual Studio and tested if they work properly or not. The input and output samples are taken from Bimar and then integrations are tested by these samples.

The integrations have been running so the BAM implementation conditions are granted after all. To implement BAM, the data to be monitored is needed to be evaluated. To evaluate these data in following three categories; booking, CODECO and manifest integrations are examined and a scenario is formed that is a business process in Bimar Inc and the required data are determined in this scenario from the key points to be gathered and monitored by BAM environment.

This process begins with the containers that are sent by a shipowner from abroad and there is a manifest file to define the transportation process from shipowner to agency. With this manifest information, agency sends a booking file to terminal to unload the defined containers from ship to terminal. After unloading the containers, terminal sends a CODECO file to agency to notify it for unloaded containers. The containers are in terminal now, and they are needed to be transport to somewhere which can be the customer, or depot or any other point in this complicated process. In this scenario, customer needs to this containers' content, so agency gives ordino file to customer that is the bill for containers' transportation. Besides sending this file to customer, agency sends a booking to CATLogic system which is the highway transportation system of ARKAS Holding. So by this booking, the containers are transported to customer by highway and at the end, CATLogic sends a CODECO file to agency to notify it about the end of transportation.

This scenario is handled by BAM besides other scenarios in Bimar, and it is selected to explain BAM structure that is handled by this thesis project.

Short version of this scenario is shown in Figure 5.2:

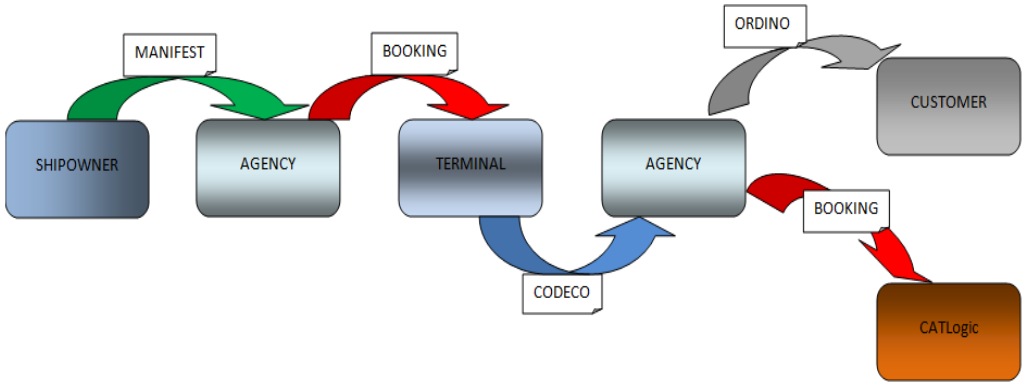


Figure 5.2. A transportation scenario

In this scenario, there are five integration points and three integration types. Where the integration points are;

1. Shipowner to Agency by manifest file
2. Agency to Terminal by booking file
3. Terminal to Agency by CODECO file
4. Agency to CATLogic by booking file

So, there are three categories for integrations; booking, CODECO and manifest. The concerned data will be extracted from these integrations. Before showing the BAM implementation on these integrations; the required data to be gathered and displayed can be explained like below parts:

5.1. Data from Booking Integrations

A booking is an order, which may have much equipment where equipment is a whole content to carry by one container. And there are goods under equipment where they will be transported. As a summary, there is a booking file which may have equipment(s) where equipment may have goods. So, booking data are examined in two sectors; first one is the common data that coming by every booking integration and these are handled as 'booking header' data. Second one is the detail of a booking integration and handled as 'booking detail'. So, think this structure as a file; there is booking's general information like 'bookingNumber', and detail of this file has equipment's information. That means there is one booking and more than one

equipment so there is a looping structure for equipments. Because of this looping structure, there are differences on implementation of BAM for looping structure against flat structure. The determined data from booking integrations like “Agency to Terminal” or “Agency to CATLogic” is:

Table 5.2. Booking data to be monitored

	Booking Header		Booking Detail:
1	IntegrationCode	**1**	ContainerBrand
2	FileStatusCode	**2**	ContainerNumber
3	TargetSystemID	**3**	EquipmentCurrentLocationID
4	BookingTypeID	**4**	LoadingLocationID
5	VesselVoyageID	**5**	LoadingDate
6	VesselExportDepartureDate	**6**	DepartureAddress
7	BookingStatusID	**7**	OldBookingNo
8	BookingCreationDate	**8**	TransportationTypeID
9	BookingNumber		
10	LineID		
11	BookingCustomerID		
12	BookingPersonInCharge		
13	BookingCustomerName		
14	BookingCustomerAddress		
15	LoadingPortID		
16	UnloadingPortID		
17	VesselName		
18	VesselVoyageNumber		
19	Begin (integrationBeginTime)		
20	End (integrationEndTime)		
21	IntegrationDuration		

The data from the List 5.2 are prepared as activities ‘booking_izleme_header_TEST’ and ‘booking_izleme_detay_TEST’. The header activity is matched up to integrations on TPE directly. There are differences with the BAM sample that explained in Chapter 2. In that sample in Chapter 2, the ‘ActivityID’ value on activity side, which is the primary key of header activity, is not edited, and it is

given automatically an activity id in database. In this real implementation, to be able to make a relation between header and detail activities, there is a need for a unique key. Because of the being primary key, 'ActivityID' information is unique for every instance in activity table. So, the determined value for this field in activity table also needs to be unique. And also it can be achieve from Tracking Profile Editor and also from integration inside because of the need for creating a relationship between detail and header information.

After some research on incoming and outgoing data of all integrations, the 'InstanceID', which is the incoming data file's instance id is selected as 'ActivityID' where that can be achievable from the TPE and from orchestration inside.

After deciding what will be the 'ActivityID', this matching up is performed on TPE. Here the header activity is ready. But there is still the detail activity needed to be implemented. But because of the looping lines, there is a need for different operations. Besides the looping lines, there is a value that is selected as required data which is 'IntegrationCode'. It is the integration id of Bimar own system and needed to be displayed if there is a need to. And this value is not coming by the input or output file. So there is a need to get this value by custom code, like handling the detail activity by custom code.

Figure 2.9 shows an orchestration of CODECO integration. As can be seen, there are shapes in the Figure 2.9, which are the expressions of orchestration. Briefly it can be said that orchestration is an expressions train; it is following the expression after expression. These expressions have many types such as 'receive' shape where it is the starting point of and orchestration that is taking the input file by the defined receive port. But there are shapes that can be written C# code inside (see Figure 5.3).

The detail activity matching up and getting the integration ID steps will be done by writing custom code and by the expressions. This custom code differs from integration type to other because their required values are different and most important thing is the looping line can be nested like in booking integrations. Besides it can be just one loop like in the manifest integrations or it can be not looped like in CODECO integrations. There is no need to implement looping lines code in CODECO integrations but there is a need for getting the integration ID value by custom in also CODECO integrations.

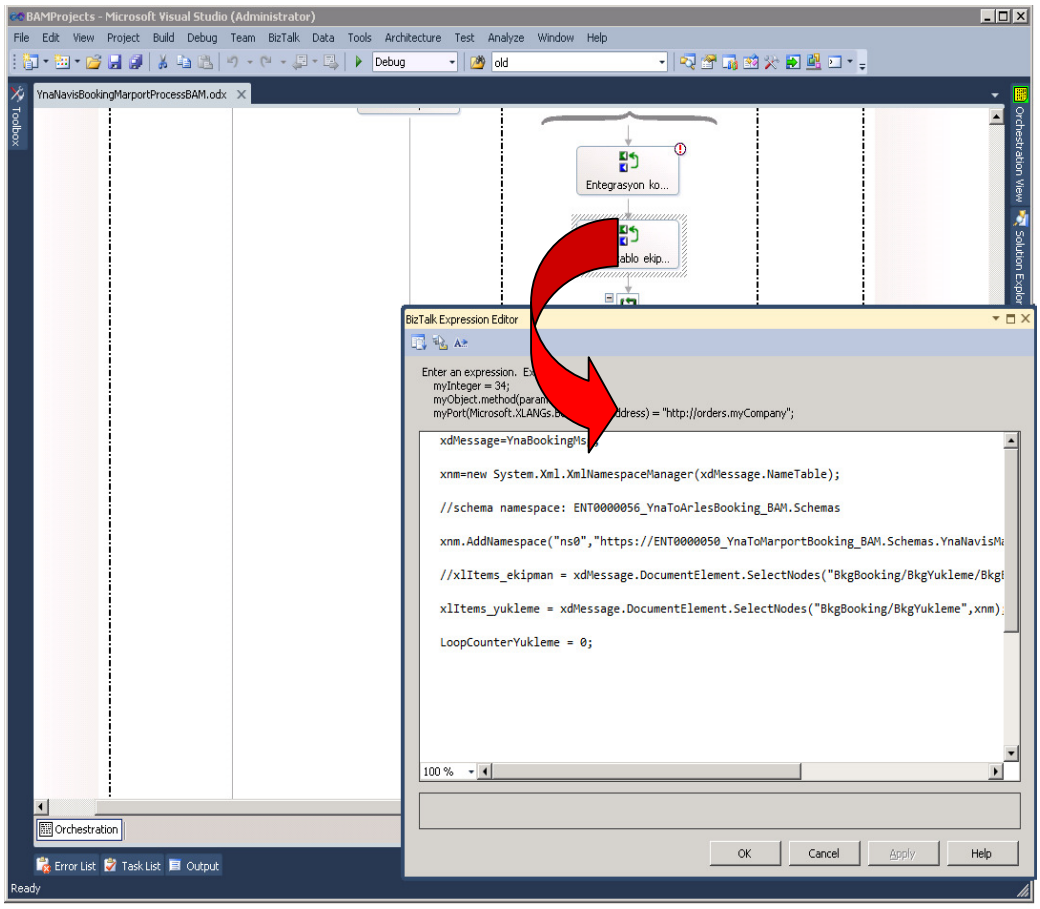


Figure 5.3. Inside of an expression

Here the looping lines solution for booking integrations; the values ‘LoadingLocationID’, ‘LoadingDate’, ‘DepartureAddress’, ‘OldBookingNo’ and ‘TransportationTypeID’ from booking detail table are listed under ‘Loading’ tab in input XML file, whereas ‘ContainerBrand’, ‘ContainerNumber’ and ‘EquipmentCurrentLocationID’ are under ‘equipment’ tab under ‘Loading’ tab. So for booking integrations, there is need for nested loops to get firstly the ‘Loading’ tab’s values, after that for every ‘Loading’ get the every ‘equipment’ information (Figure 5.4).

The circled area in Figure 5.4, which is a booking integration’s BAM implemented version, has 5 expressions besides the loop operators that are the green arrows in the Figure 5.4. Assume that the expressions are numbered as 1 to 5. Expression 1 (Figure 5.5) starts with achieving instance id from system to use it as ‘ActivityID’.

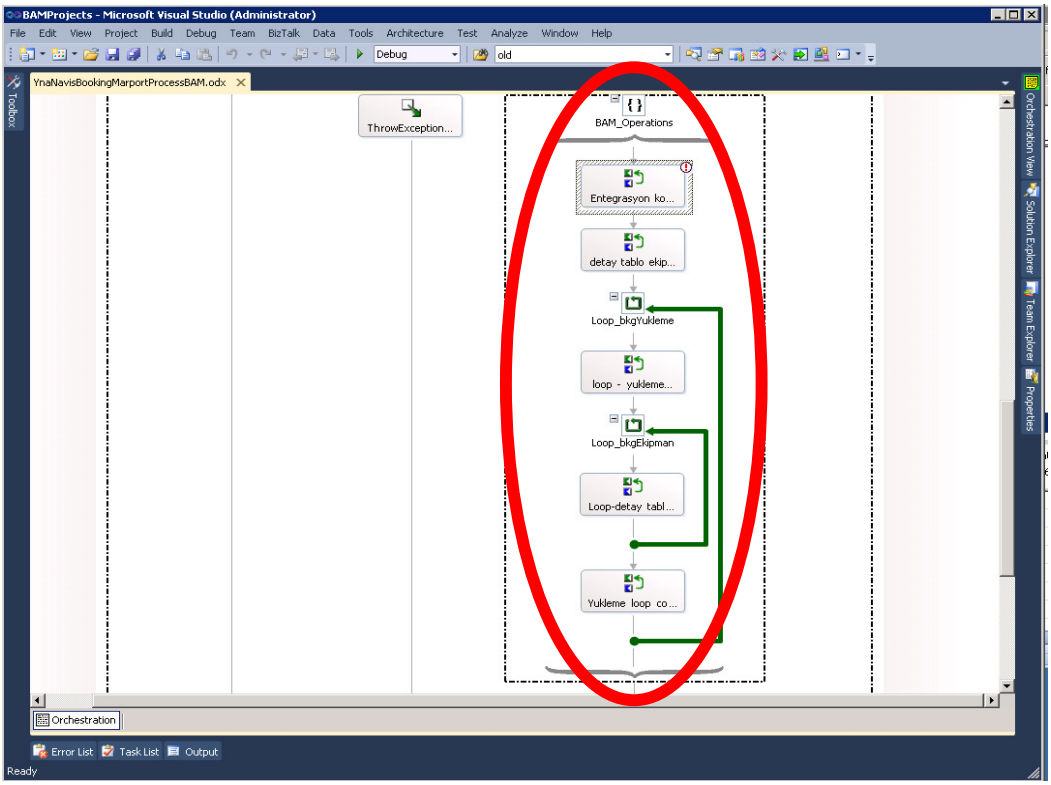


Figure 5.4. Booking integration's detail activity matching up situation

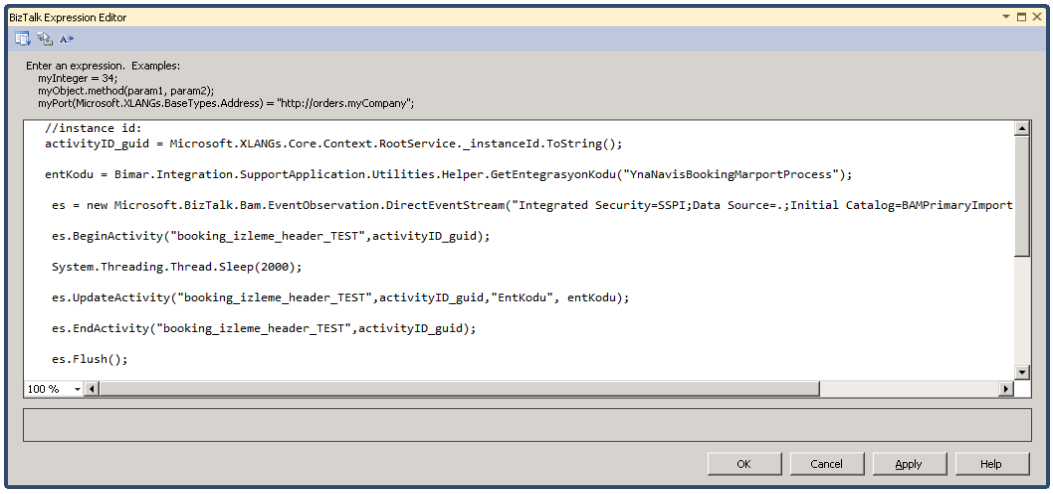


Figure 5.5. Expression1 from booking orchestration's BAM part

After getting instance id, the integration id is got from Bimar Integrations system by the name of incoming schema. Then the next lines (Figure 5.5) are the API of BAM, which supplies to connect database and starting an activity with the 'ActivityID', and

then the first value that is written on database is integration id. That is an update operation on BAM API.

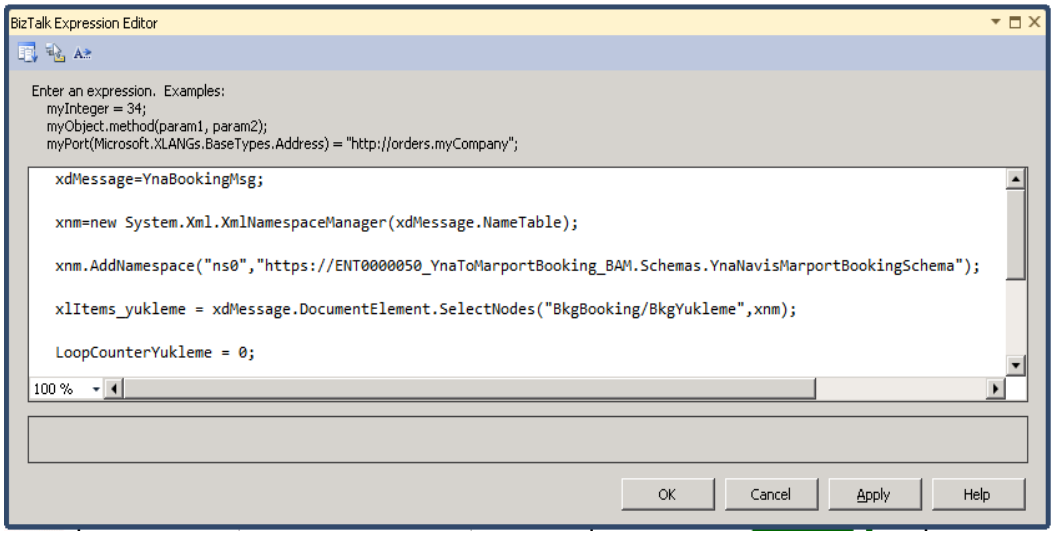


Figure 5.6. Expression2 from booking orchestration's BAM part

In Expression 2 (Figure 5.6), the message operations are done. The source message for BAM to gather data is the input file of the integration and it is defined in this shape at first line. The looping lines' first tag, which is 'Loading' here, is given to code at the 4th line and then the loop is started by the line;

```
LoopCounterYukleme < xlItems_yukleme.Count
```

This statement is for running the loop until all 'Loading' tags are traced and processed. This begins with the determined values from the input message. This can be seen in Figure 5.7. After these values are gathered, next step is to get all equipment of 'Loading'. Because there can be more than equipment under a 'Loading' tag. So after first 'Loading' is processed by getting the singular values like 'cikisYeriAdresi', the equipment's loop is coming. The 'Loading' and its all 'Equipment' can be seen in the representational structure of booking message's this part in Figure 5.8.

And to get this equipment one by one, the condition is;

```
LoopCounterEkipman < xlItems_ekipman.Count
```

Loop goes over till the all equipment is traced under one ‘Loading’. And then the next ‘Loading’ tracing starts.

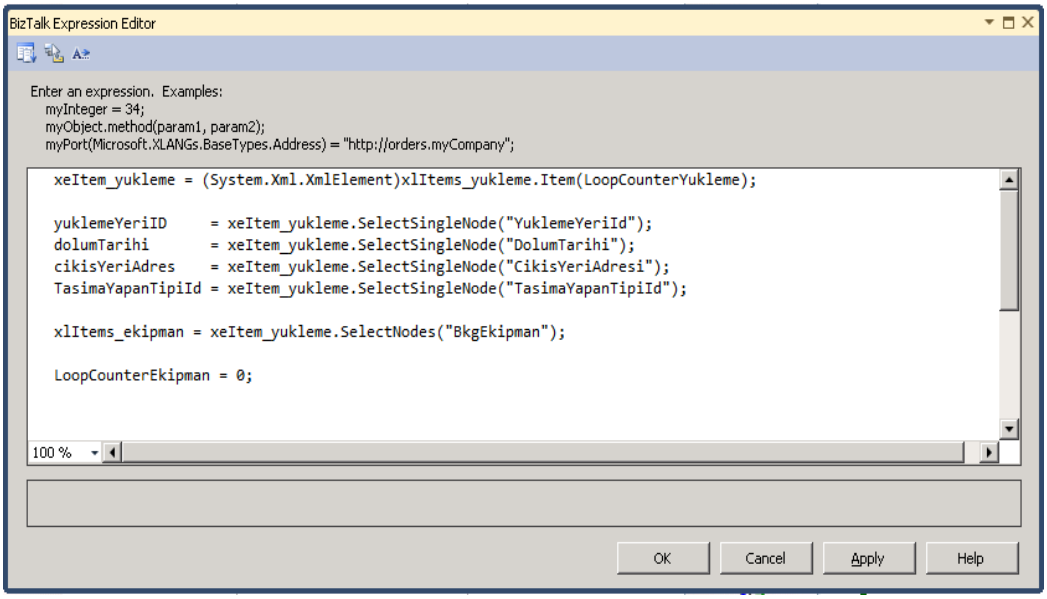


Figure 5.7. Expression3 from booking orchestration’s BAM part

Figure 5.10 shows that the equipment’s values are gathered and then the ‘Loading’s values and equipment’s values are updated to detail table. The line;

```
es.AddRelatedActivity("booking_izleme_detay_TEST",ItemID,"booking_izleme_header_T
EST",activityID_guid);
```

is the relation between header activity and detail activity, where the ‘AddRelatedActivity’ method comes with BAM API, and takes arguments as detail activity’s name and its id, header activity’s name and its id. This makes connection between header and detail activities.

The last expression shows the next ‘Loading’s starting time, if all equipment is traced for one ‘Loading’, the next ‘Loading’ is started to traced (Figure 5.9).

By these expressions, the looping lines issue is resolved. In summary, header activity matched up with input message on TPE, whereas the detail activity matched up by using BAM API inside orchestration.

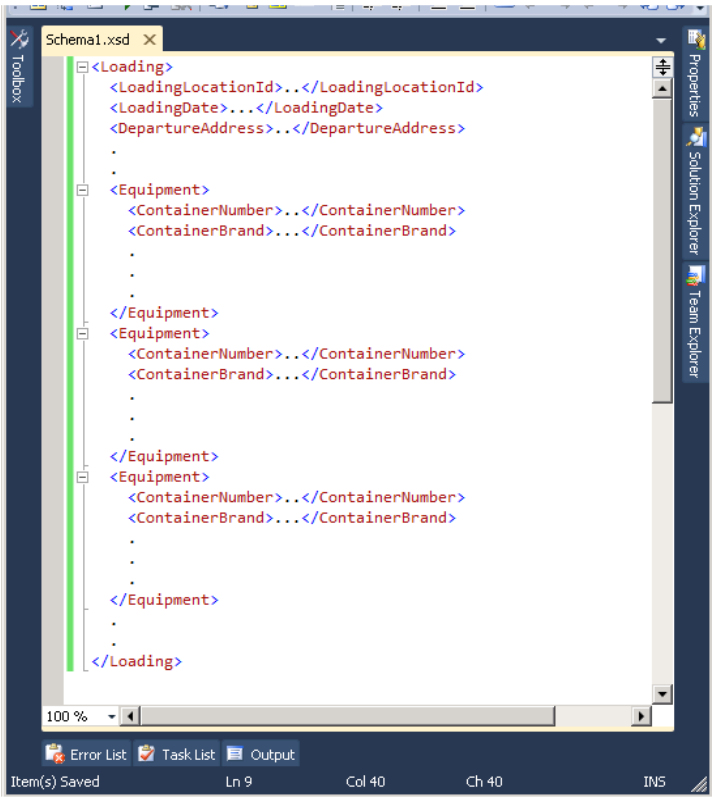


Figure 5.8. Booking input message's outline

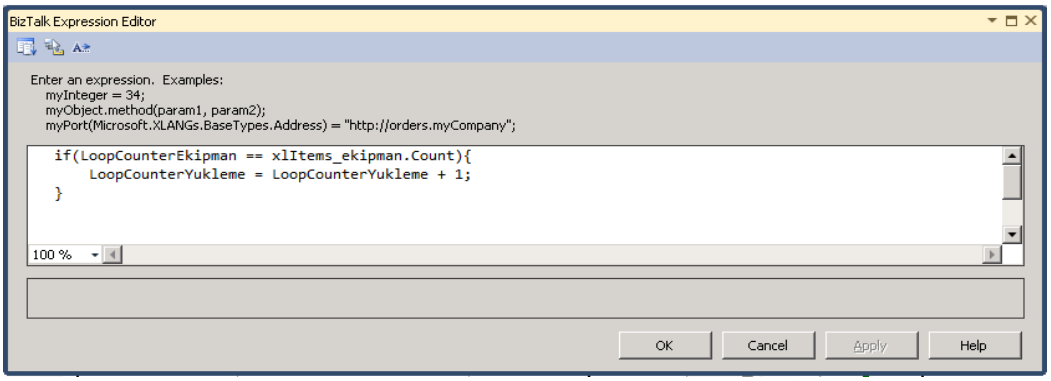


Figure 5.9. Expression 5, end of counter for 'Loading'

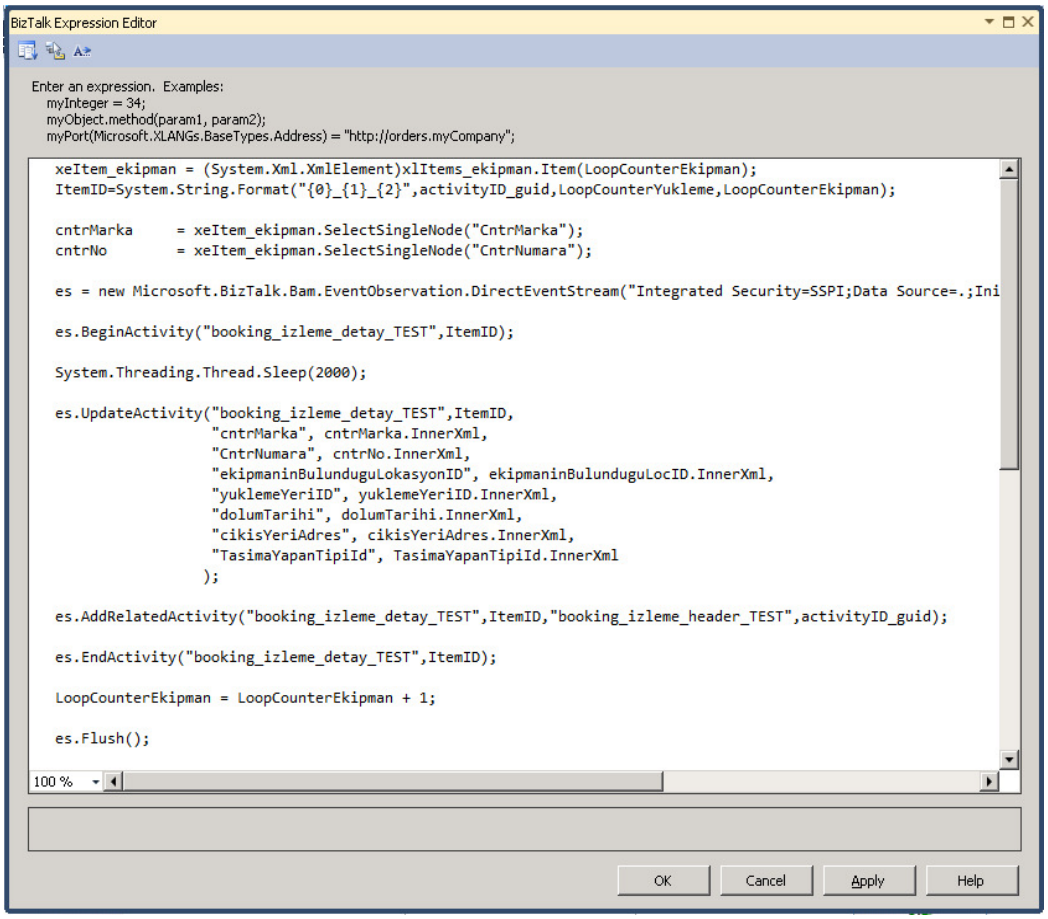


Figure 5.10. Expression 5 from booking orchestration's BAM part

There are five integrations in category of 'booking'. So all this steps are performed for all these five booking integrations. So this solution must be implemented in other booking integrations, the content of the expressions are same except for message name, so the other booking integrations' expressions will not be shown here. This is for detail activity, for matching up all booking integrations' header activity; there is a trick on TPE. The all integrations can be matched up under just one TPE (Figure 5.11). The drag from orchestration to drop on activity on the left is done by all integrations and it can be seen in Figure 5.11, there are 5 items under every activity item on the left part of TPE.

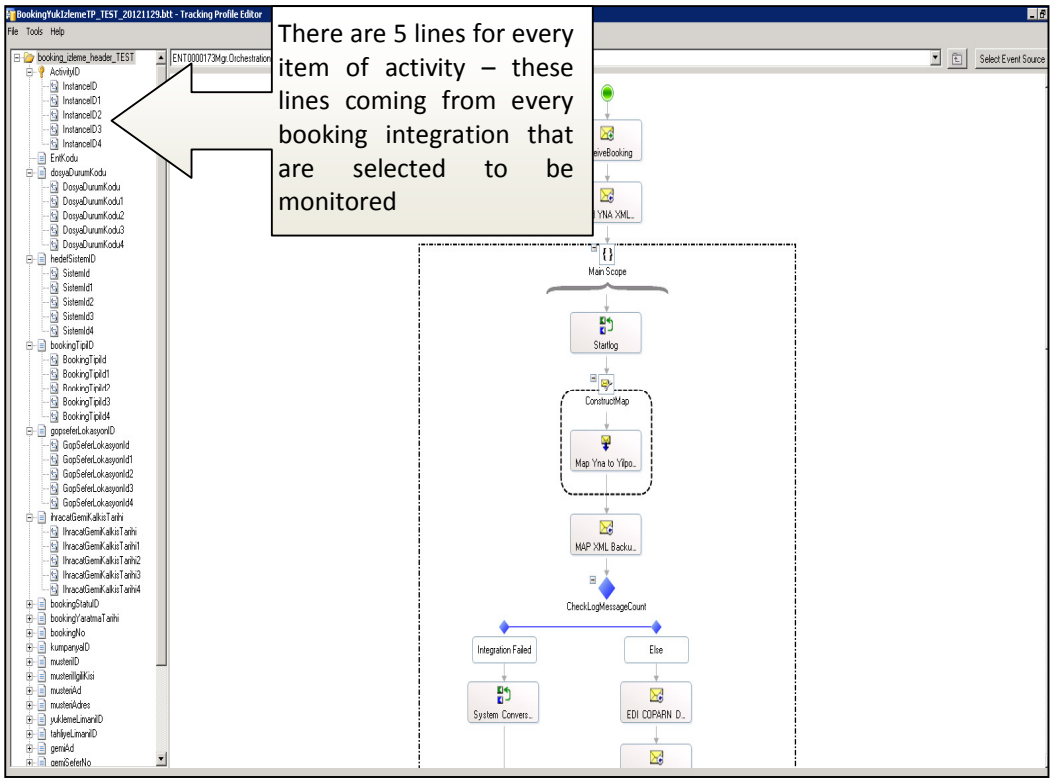


Figure 5.11. TPE of header activity for booking integrations

5.2. Data from CODECO Integrations

CODECO integrations are the gate in/gate out operations which they are the answer of the point that get the booking to handle a job over containers like unloading them to terminal or transporting from depot to somewhere. These integrations do not contain a looping structure, so these are easier than booking integrations.

The determined data from CODECO integrations like “Terminal to Agency” or “Depot to Agency” is listed in Table 5.3. The data from the Table 5.3 are prepared as activity ‘codeco_izleme_TEST’. The rest of the implementation is handled same as in the BAM sample of Chapter 2, besides getting the value ‘integrationCode’ which is the integration id. The activity matched up with the CODECO integrations on TPE and to get the integration id from the system and updated it to activity; an expression is placed on every CODECO integration’s orchestration (Figure 5.12)

Table 5.3. CODECO data to be monitored

	CODECO
1	BookingNo
2	EquipmentPrefix
3	EquipmentNumber
4	MovementHistoryId
5	MovementLocation
6	ActualYear
7	ActualMonth
8	ActualDay
9	ActualDateHour
10	ActualDateMinute
11	ActualDateSecond
12	integrationCode
13	Begin
14	End
15	integrationDuration

5.3. Data from Manifest Integrations

Manifest integrations are the shipping order for a whole ship that means the ship contains many containers that belongs to different customers – agencies. The structure of manifest files is looped like booking integrations. So there is a need to divide activity into two parts. Here for manifest files, these parts will be 'ship information' and 'consignment note'. The required data are determined as in Table 5.4, as in these two parts. Manifest is a whole shipping order for a ship so there is ship information as singular that is gathered by 'gemi_bilgileri_izleme_TEST' activity and the containers to be transport by ship are belong to different customers that every customer's freight is a consignment so these information of every customer's freight are gathered by 'konsimento_bilgileri_izleme_TEST' activity as looping structure.

The looping lines solution is not different from booking integrations' solutions. Just the names of tags differ in the expressions.

The gathering data from integrations is over as explained above. Now turn is to display it in a user friendly interface, also it must be as simple as possible. So there is a

development for user interface to present gathered BAM data as meaningfully and easily usable format.

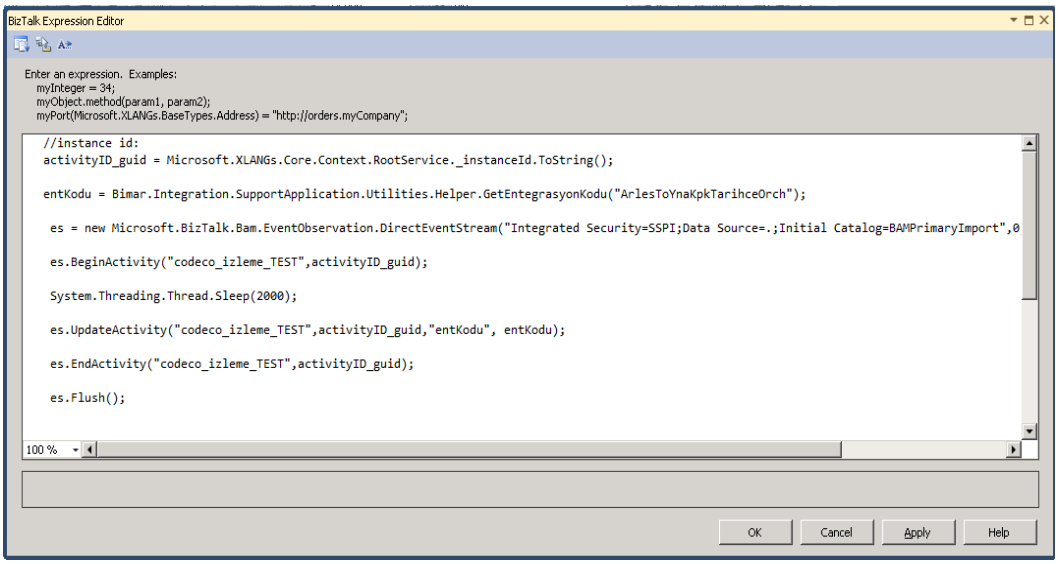


Figure 5.12. Expression from CODECO integration

Table 5.4. Data from Manifest integrations

	Vessel_Information		**BillOfLanding_Information**
1	VesselDepartureDate	**1**	BillOfLandingNumber
2	VesselExpectedArrivalDate	**2**	BillOfLandingPrintDate
3	VesselName	**3**	ContainerVesselLoadingDate
4	VesselVoyageNumber	**4**	DeparturePort
5	DepartureLocationID	**5**	ArrivalPort
6	ArrivalLocationID	**6**	ContainerPrefix
7	VesselLineID	**7**	ContainerNumber

5.4 User Interface for BAM data

To present gathered BAM data there is a need for useful tool, besides the Microsoft BAM Portal. Because this MS BAM Portal just takes the values from activity tables directly, without any interpretation on in. In case of Bimar, there is a need to show the whole business process on a portal. So this user interface is prepared as Bimar BAM Portal (Figure 5.13).

System is working according to two inputs, booking number or container number. Customer knows its containers' numbers and can trace them as he/she whenever wants. When enter booking number, the Figure 5.14 shows all containers (equipment) that under itself. By clicking any container line, its details come to screen (Figure 5.15).

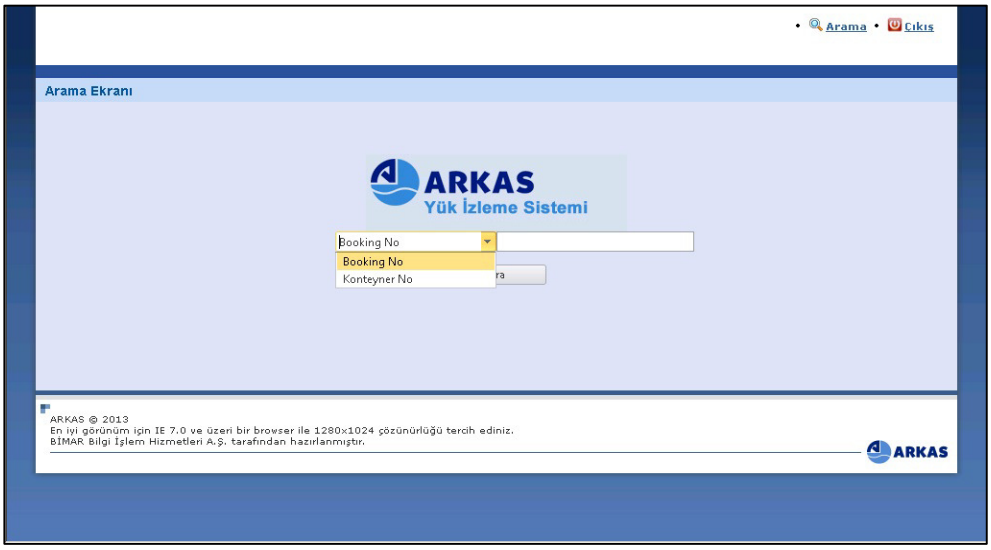


Figure 5.13. Bimar BAM Portal

Which movements are done to container, is showed in its detail (Figure 5.15) like it is entered to depot or move out from terminal.

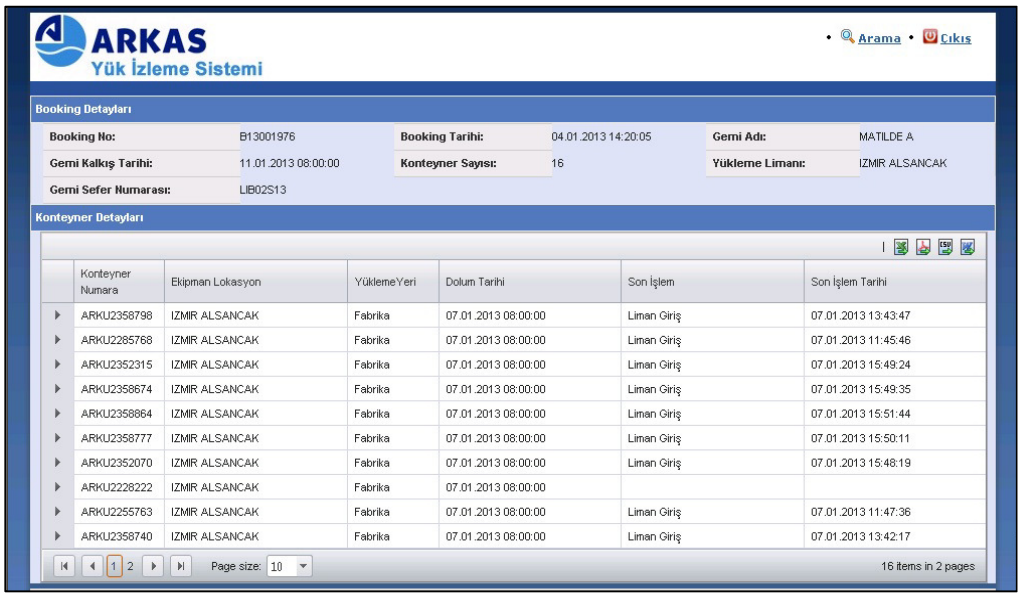


Figure 5.14. Booking's detail

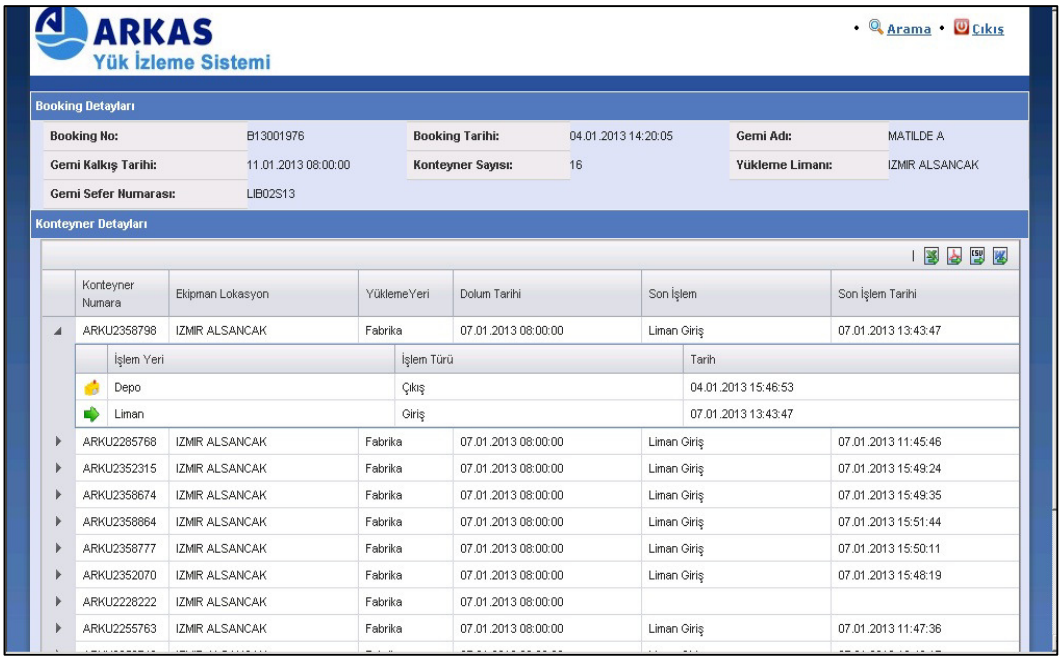


Figure 5.15. container's detail

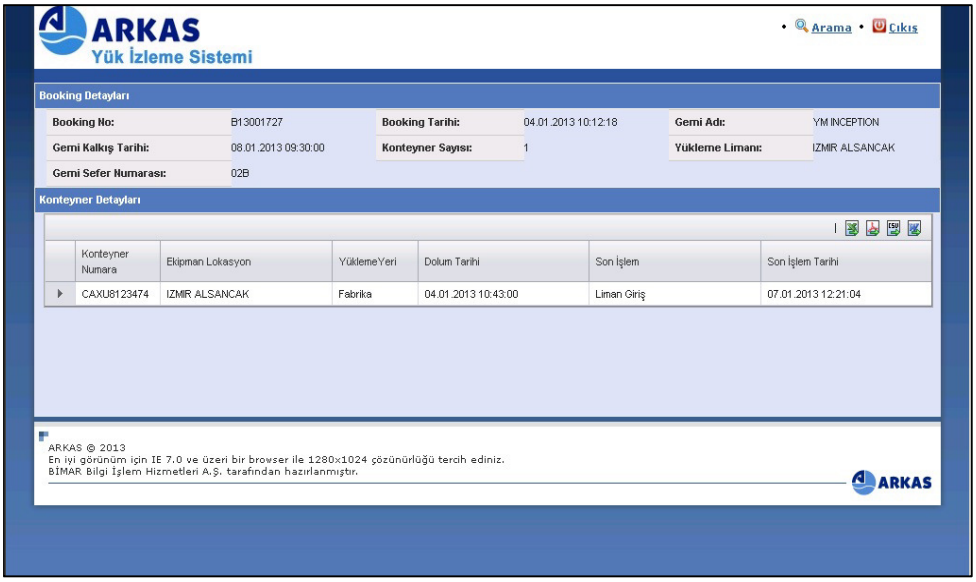


Figure 5.16. A container's information

If the search is done according to container number, the coming screen has just one container – that belongs to this container number (Figure 5.16). And again, by clicking on it, its details come to screen (Figure 5.17).

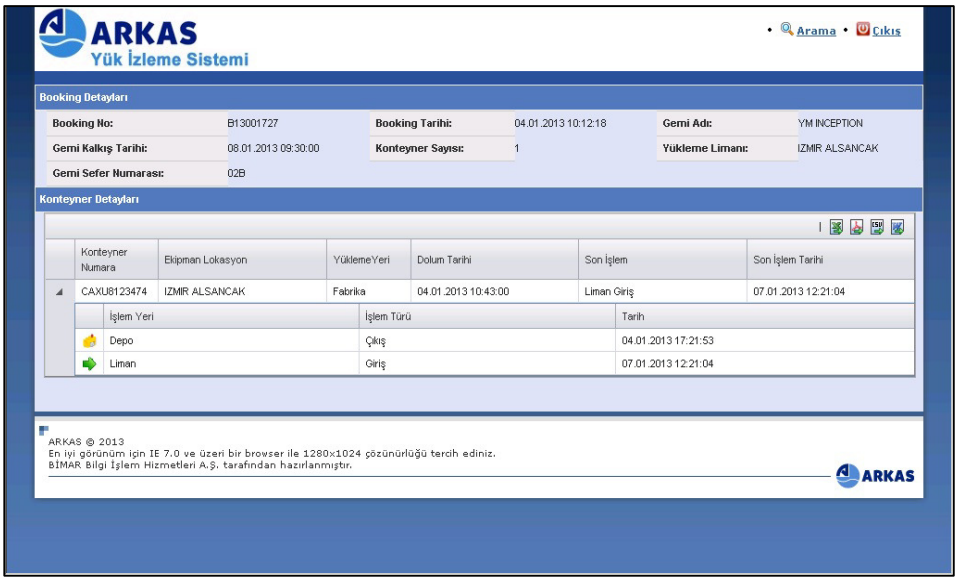


Figure 5.17. A container's detailed movements

By preparing the user interface, the BAM implementation for logistics in case of Bimar is finalized.

CHAPTER 6

CONCLUSION

The aim of this thesis is to create a monitoring environment for logistics managements systems to track all steps of the transportation. For this environment, Business Activity Monitoring (BAM) that is the monitoring tool of Microsoft BizTalk Server is taken as the base. However, MS BAM has deficiencies to provide the necessary monitoring environment. To overcome these deficiencies and fulfill the requirements of the domain, for container logistics, there is a need to implement more features besides using MS BAM.

Container logistics is a complicated business domain that has usually many processes in one transportation and to control all the points in this process, there is a need for automated, real-time processing monitoring tool. In this thesis, this problem is resolved by creating 'business activity monitoring for logistics management'.

Container logistics business domain is examined for the case of Bimar. MS BAM tool is learnt to use as the beginning point of the thesis project. After the beginning steps are performed as examining the environment, then the methodology of the business domain is investigated and the information that will be tracked by the proposed business activity monitoring is revealed by 'business process improvement' method which is an approach to find the business process to achieve the performance improvement of enterprise. This information is defined as KPIs to be monitored.

The KPIs are defined and the business processes are documented, then the monitoring implementation is started by using MS BAM. For a beginning, MS BAM is used in anywhere that it can be used. But where it becomes insufficient, then the custom implementation of BAM is started. As it mentioned in Chapter 5, custom code are written to gather the values that are could not be gathered by MS BAM directly. So the custom code are used to gather the necessary KPIs and the resulted values are formed by gathering the MS BAM plus custom code that are proposed in this thesis.

After it is started to gather KPIs, then it is necessary to display these data in a user-friendly and easy-to-use user interface. MS BAM again fall behind the necessities,

its portal offer is not useful and enough for the true results. So to make up this deficiency, a portal is created. This portal meets the need of displaying gathered information to the end user as real-time. Bimar BAM Portal is the user interface of the proposed business activity monitoring solution.

To sum up, a monitoring environment for container logistics management in case of Bimar is created by this thesis. This environment provides a whole control on the all points of transportation processes. By this thesis, a monitoring solution is provided to container logistics area in case of Bimar.

The goal of this thesis is to monitor important data and display it to end user. Next iteration could be preparing a framework for an automated platform to match integrations, systems and variables. Now, the system is mostly working by drag and drop besides little code parts. However, by code development, the manual operations can be reduced and an easy-to-use monitoring tool can be created. These improvements may be performed on creating activity and views and matching them to the selected integrations. Instead of manually selection of integration or manually creation of activities, a user-friendly interface will be useful for the beginning of monitoring process.

REFERENCES

[1] http://en.wikipedia.org/wiki/Business_activity_management

[2] Wetzstein, B., Danylevych, O., Leymann, F., Bitsaki, M., Nikolaou, C., Van den Heuvel, W.J., Papazoglou, M.: Towards Monitoring of Key Performance Indicators Across Partners in Service Networks. In A workshop held at the ServiceWave 2008 conference – Proceedings. (2008)

[3] Tom Forester (eds), "The Information Technology Revolution", MIT Press, 1985, ISBN-13:978-0-262-06095-0

[4] http://www.logisticsworld.com/logistics.htm

[5] http://www.businessdictionary.com/definition/container-logistics.html

[6] http://en.wikipedia.org/wiki/Transport

[7] Daganzo, C.F., "Logistics System Analysis", 2005, Springer, ISBN 3-540-23914-6

[8] http://www.zeyport.com.tr/english/pages/arkas_news/kasim_2011/haber10.html

[9] Council of Logistics Management (CLM) (1998), What it's all About, Oak Brook, IL, Vol. 4 No.6

[10] http://www.microsoft.com/biztalk/en/us/overview.aspx

[11] http://en.wikipedia.org/wiki/Business_activity_management

[12] http://technet.microsoft.com/en-us/library/aa995581(BTS.70).aspx

[13] Gohar, A., "Analyzing Service Oriented Architecture (SOA) in Open Source Products" (Master Science Thesis), 2010, Vattenfall

[14] http://msdn.microsoft.com/en-us/library/ee253346(v=bts.10).aspx

[15] Nieuwenhuijs, S., Yap, M., “AMIS Query: Oracle WebServices Manager (incl. ESB Sneak Preview).ppt”, Oracle Fusion Middleware Product Management Oracle Corporation, http://www.docstoc.com/docs/82098660/Oracle-ESB

[16] Sanders, J., Snowman, G., “Pro Business Activity Monitoring in BizTalk 2009”, 2009, Apress

[17] http://technet.microsoft.com/en-us/library/aa577946(BTS.70).aspx

[18] http://www.1edisource.com/learn-about-edi/transaction-sets/tset/codeco#axzz2UDCuRfBR

[19] http://msdn.microsoft.com/en-us/library/ff699598.aspx

[20] “Lojistik İş Süreçleri için Birlikte Çalışabilirlik ve İzleme Yazılım Çerçevesi Geliştirilmesi”, San-Tez Projesi, Proje Başvuru Dökümanı, 0093.STZ.2011-1, Sayfa 12.

[21] Kolář, J., “Business Activity Monitoring”, 2009, Masaryk University, Faculty of Informatics, MSc. Thesis

[22] Kang, J. G., Han, K. H., “A Business Activity Monitoring System Supporting Real-Time Business Performance Management”, 2008, Third 2008 International Conference on Convergence and Hybrid Information Technology

[23] Arsanjani, A., "Service-oriented Modeling and Architecture", IBM, 9 November 2004, http://www.ibm.com/developerworks/library/ws-soa-design1/

[24] http://www.ozelhastaneler.org.tr/kpi-key-performance-indicators--semsettin-akcay.aspx?pageID=338&nID=5359, Last Accessed 03.06.2013.

[25] http://www.klipfolio.com/blog/entry/305, last accessed 03.06.2013

[26] http://www.slideshare.net/dennis.mortensen/the-difference-between-a-kpi-and-a-metric, last accessed 03.06.2013

[27] http://technet.microsoft.com/en-us/library/aa561137(v=BTS.10).aspx, last accessed 05.06.2013

[28] http://msdn.microsoft.com/en-us/library/aa480021.aspx, last accessed 06.06.2013

[29] "The Business Case for SOA Rationalizing the Benefits of Service-Oriented Architecture", January 2005, webMethods, http://www.glintech.com/downloads/

[30] Gül, S., "İş Akış Yönetim Sistemi", 2007, Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Msc Thesis

[31] http://culturevscapital.com/stop-ignoring-what-matters-understanding-kpis/, last accessed 07.06.2013

[32]http://www.the-decisionfactor.com/performance-management/enabling-performance-management-part-three-metrics-and-kpis/, last accessed 07.06.2013

[33] Kerzner, H., "Project Management Metrics, KPIs, and Dashboards", 2011, International Institute for Learning

[34] Becher, J. D., "Mitigating Metrics Madness: How to Tell KPIs from Mere Metrics", April 2006, Volume 19, No 4, The Journal of Information Technology Management, Cutter IT Journal

[35] http://en.wikipedia.org/wiki/Business_Process_Improvement, last accessed 07.06.2013

[36] "Business Process Improvement.pdf", The Frame Group Pty Limited, http://www.framegroup. com.au/Portals/0/Documents/Capability%20Sheets/Level%202%20sheets/Frame%20Business%20Process%20Improvement.pdf, last access 07.06.2013

[37] http://www.slideshare.net/MichaelLigayo/business-process-improvement-special-report-presentation

[38] Page, S. "The Power of Business Process Improvement", 2010, AMACOM

[39] Kang, J.G., Han, K.H., "A Business Activity Monitoring System Supporting Real-Time Business Performance Management", 2008, Third 2008 International Conference on Convergence and Hybrid Information Technology

[40] Han, K.H., Choi, S. H., Kang, J.G., Lee, G., “Business Activity Monitoring System Design Framework Integrated With Process-Based Performance Measurement Model”, 2010, WSEAS Transactions on Information Science and Applications, Issue 3, Volume 7, March 2010

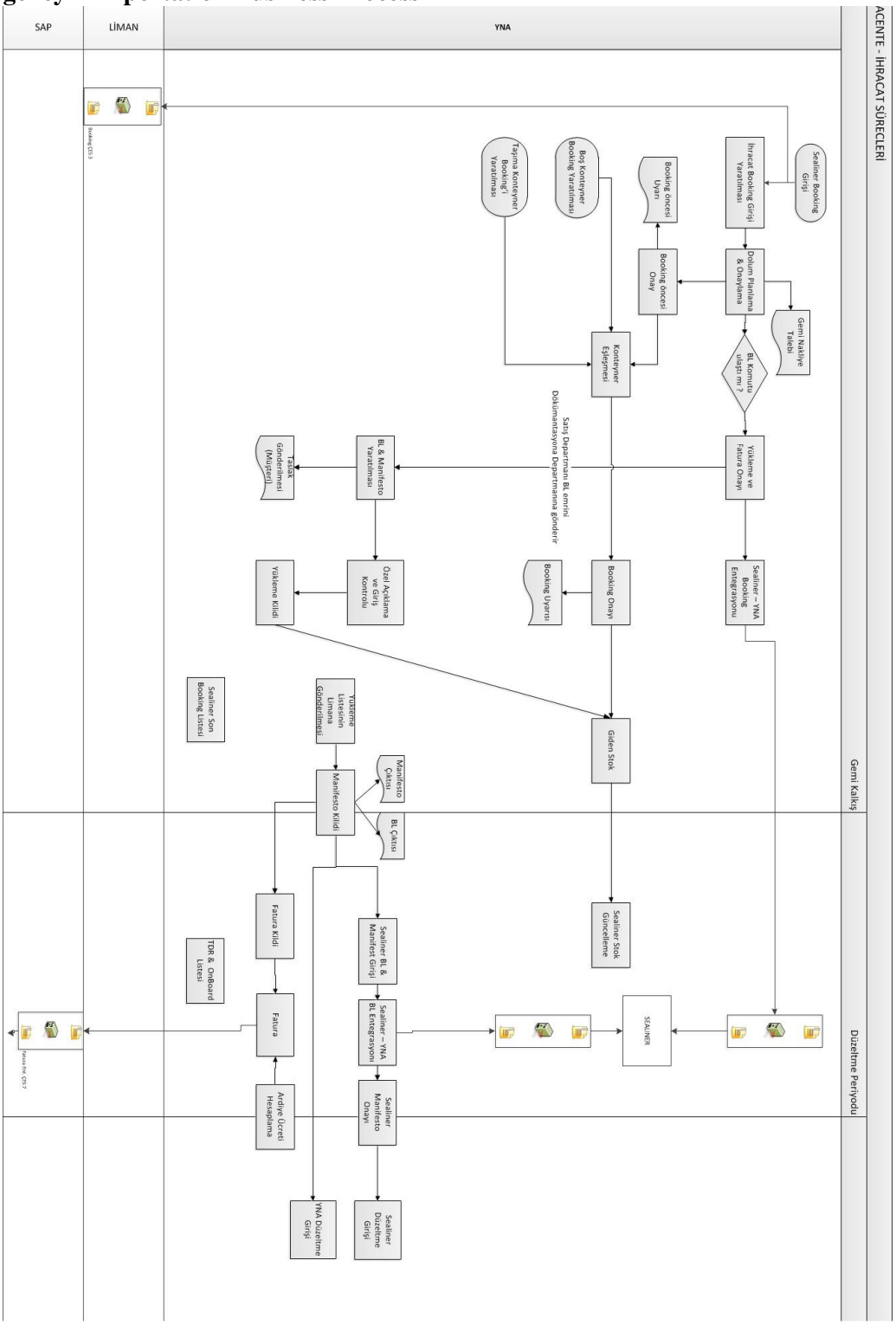
[41] Tuğlular, T., Titiz Avcı, D., Çetin, Ş., Dağhan, G., Özemre, M., Oysal, T., “An Approach to Find Integration and Monitoring Points for Container Logistics Business Processes”, 2012, The Fourth International Conferences on Advanced Service Computing, SERVICE COMPUTATION 2012, Nice, France

[42] Bitsaki, M., Danylevych, O., Van den Heuvel, W.J., Koutras, G., Leymann, F., Mancioppi, M., Nikolau, C., Papazoglou, M., “An Architecture for Managing the Lifecycle of Business Goals for Partners in a Service Network”, 2008, ServiceWave2008, p. 196-207

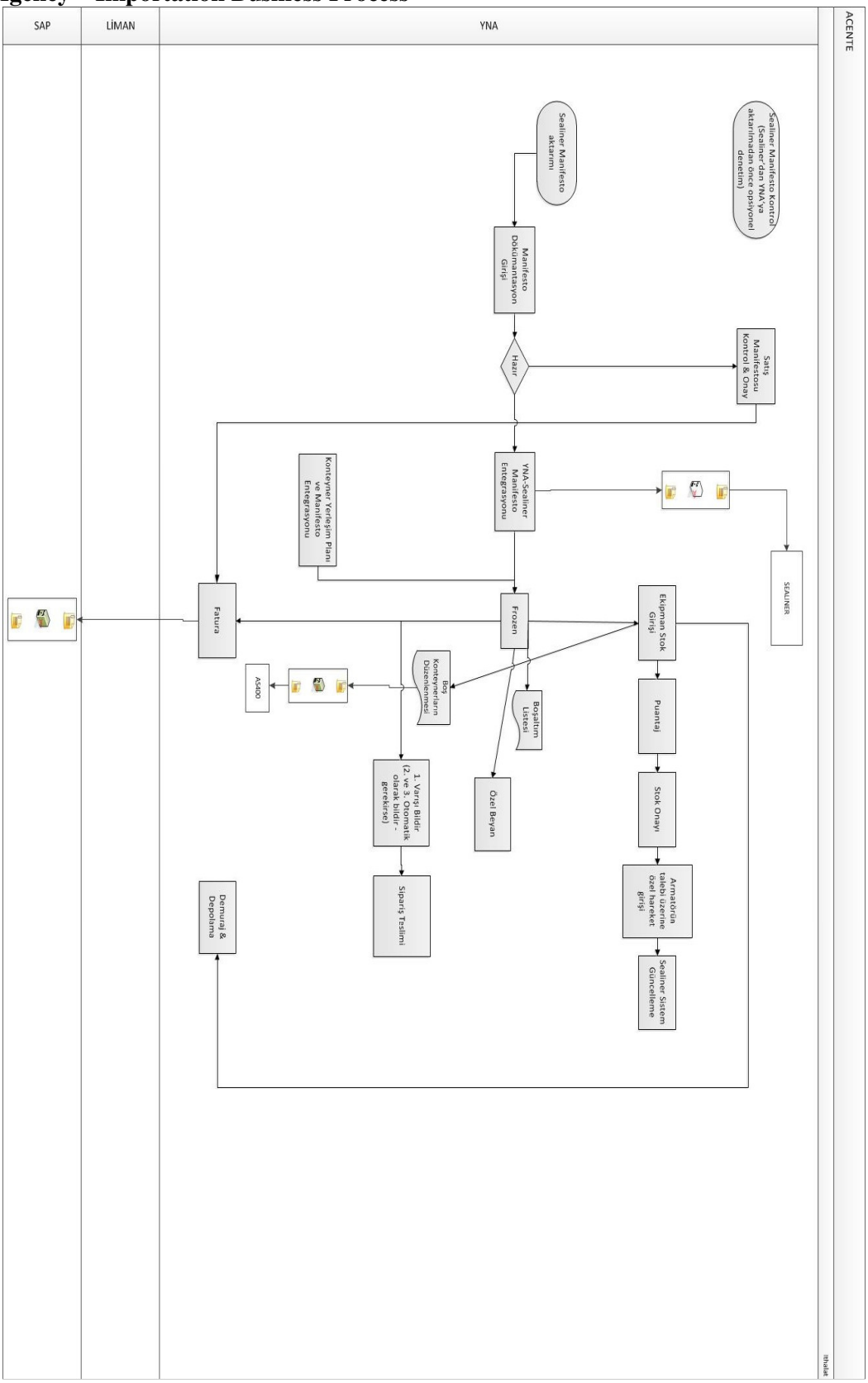
[43] Wetztein, B., Danylevych, O., Leymann, F., Bitsaki, M., Nikolau, C., Van den Heuvel, Papazoglou, M., “ Towards Monitoring of Key Performance Indicators Across Partners in Service Networks”, 2008, Proceedingss of the International Workshop on Service Monitoring, Adaptation and Beyond (MONA+ 2008)

APPENDIX A

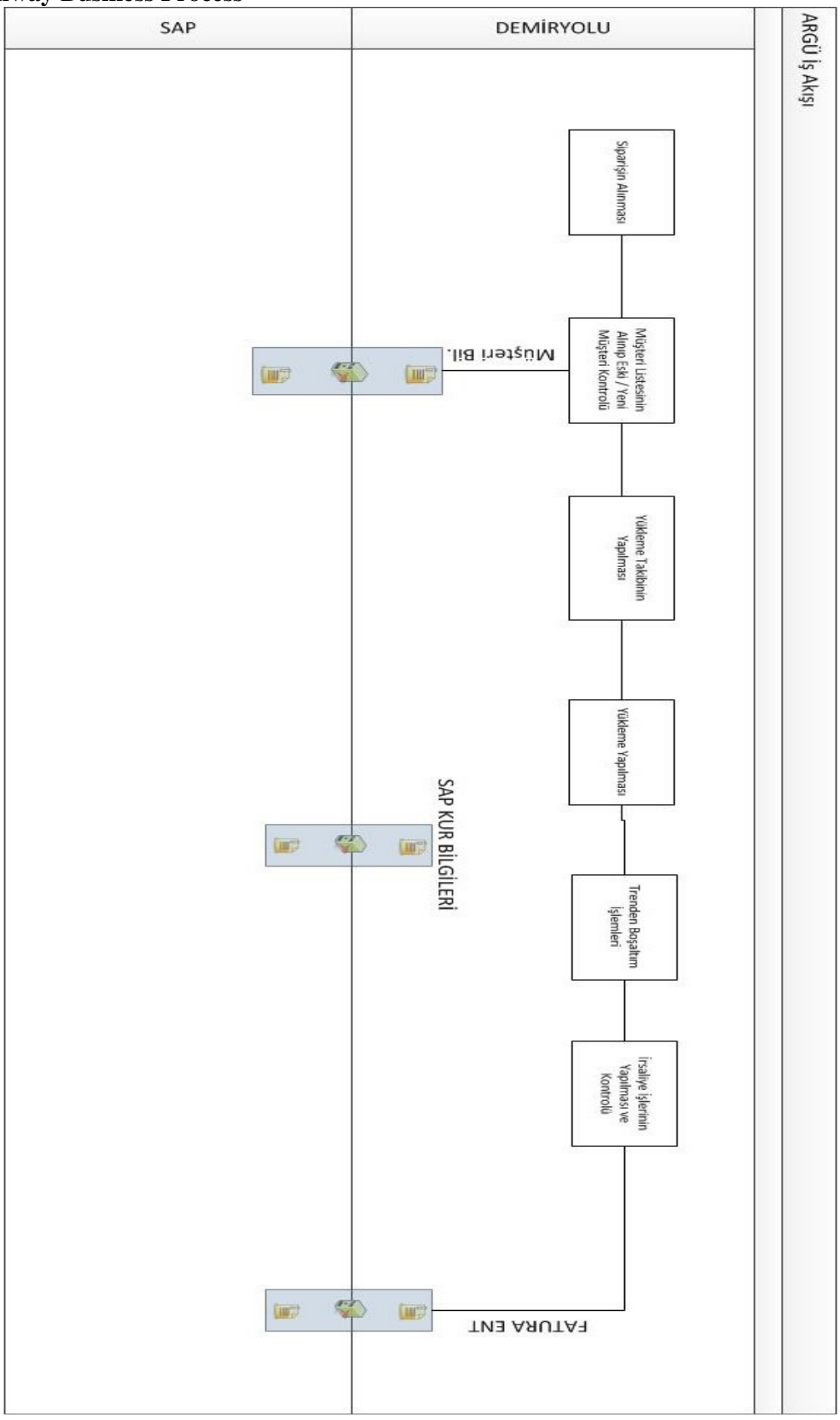
Agency – Exportation Business Process



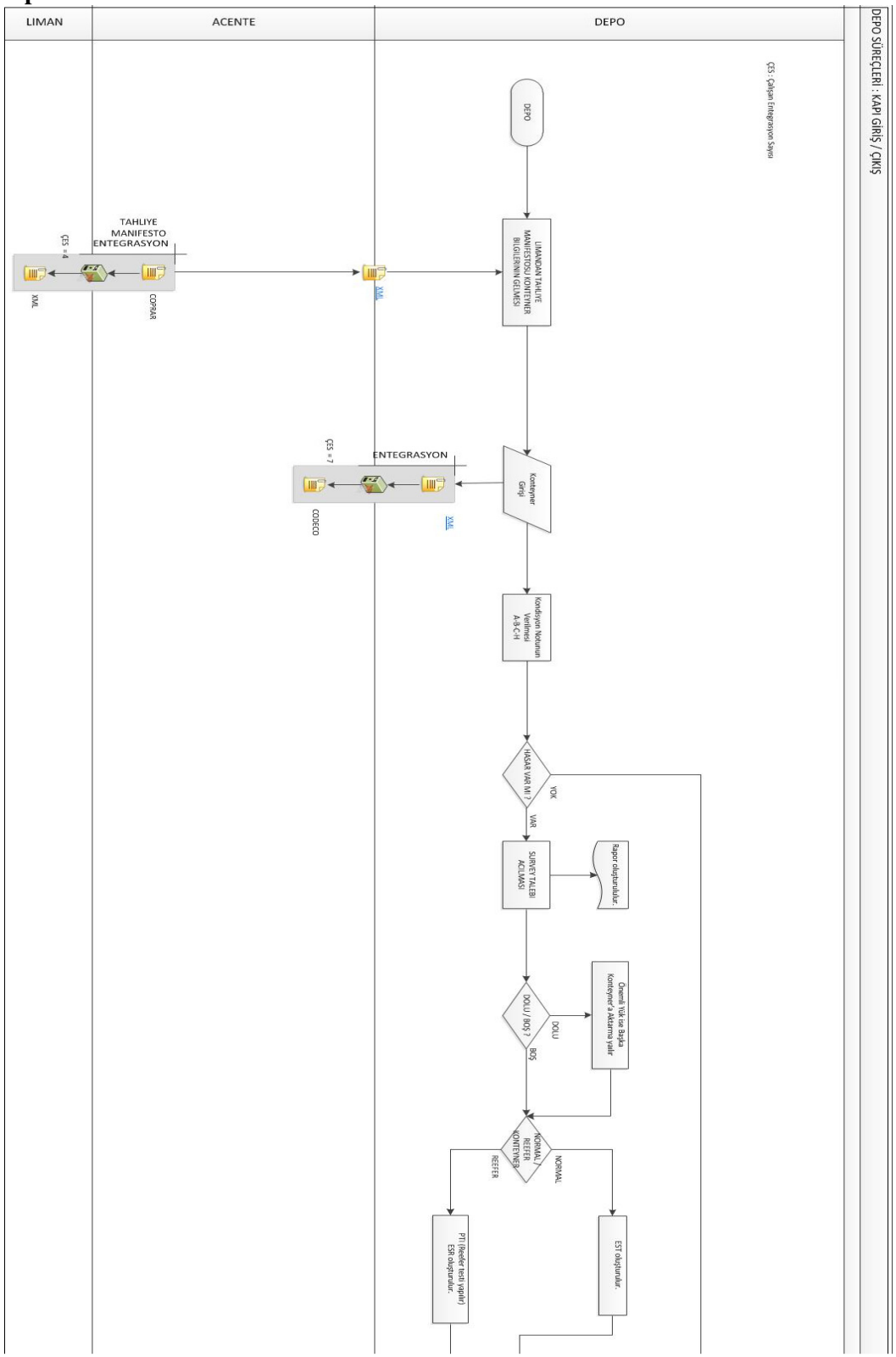
Agency – Importation Business Process



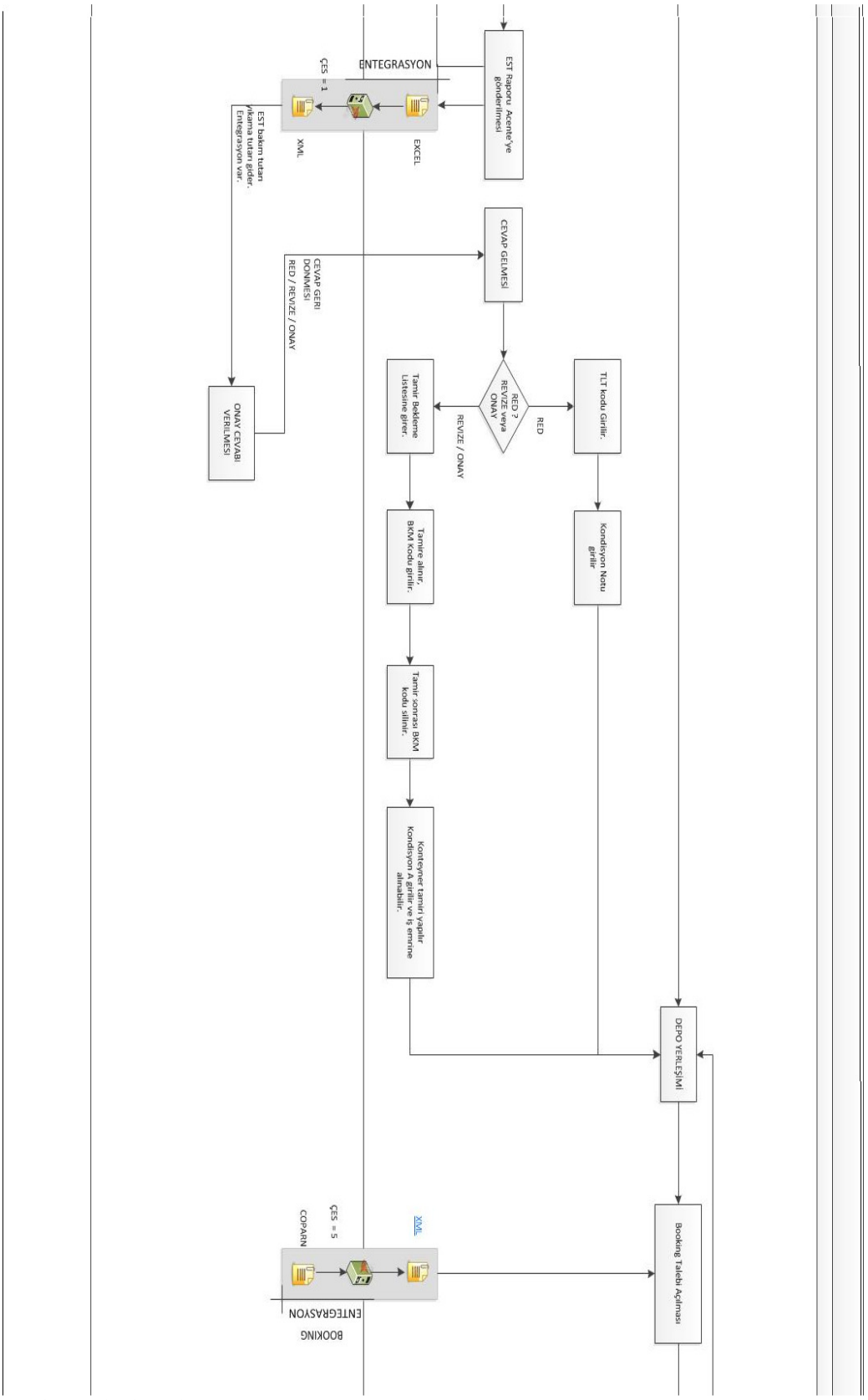
Railway Business Process



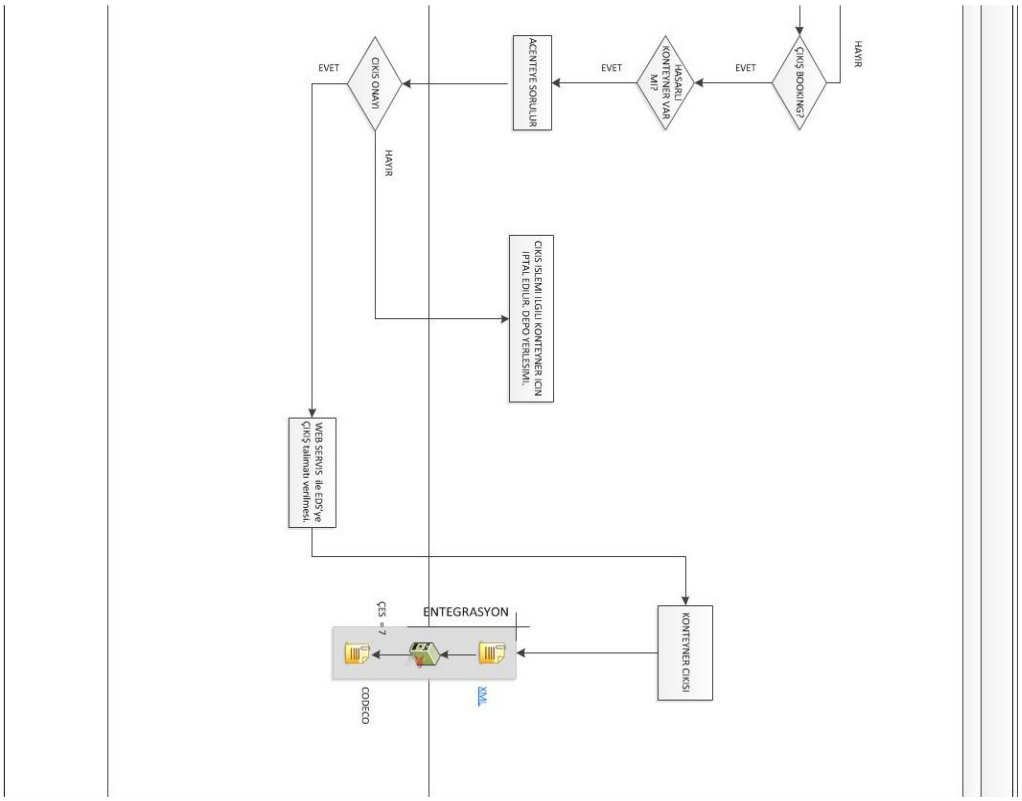
Depot Business Process



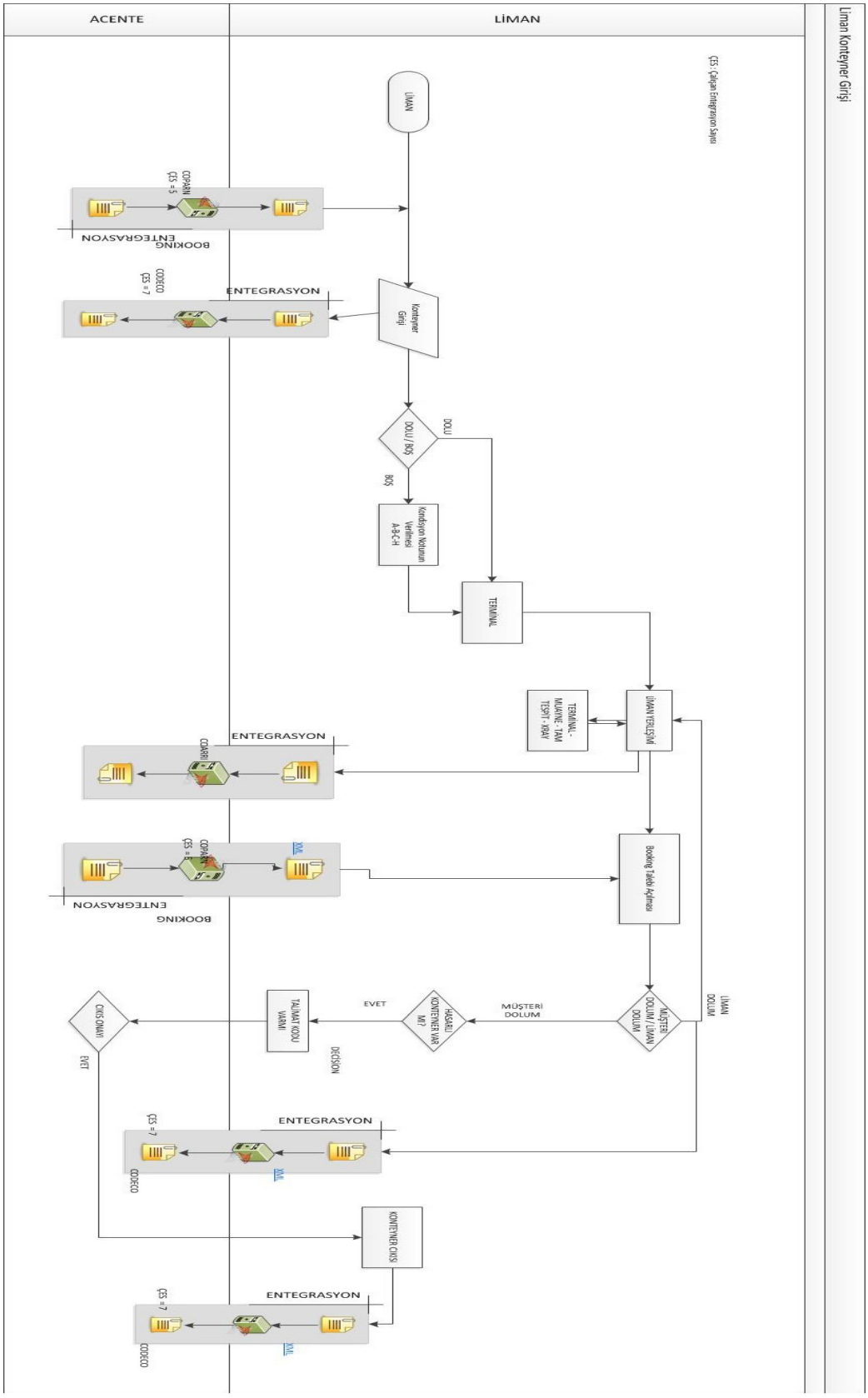
(rest of depot business is on next page…)



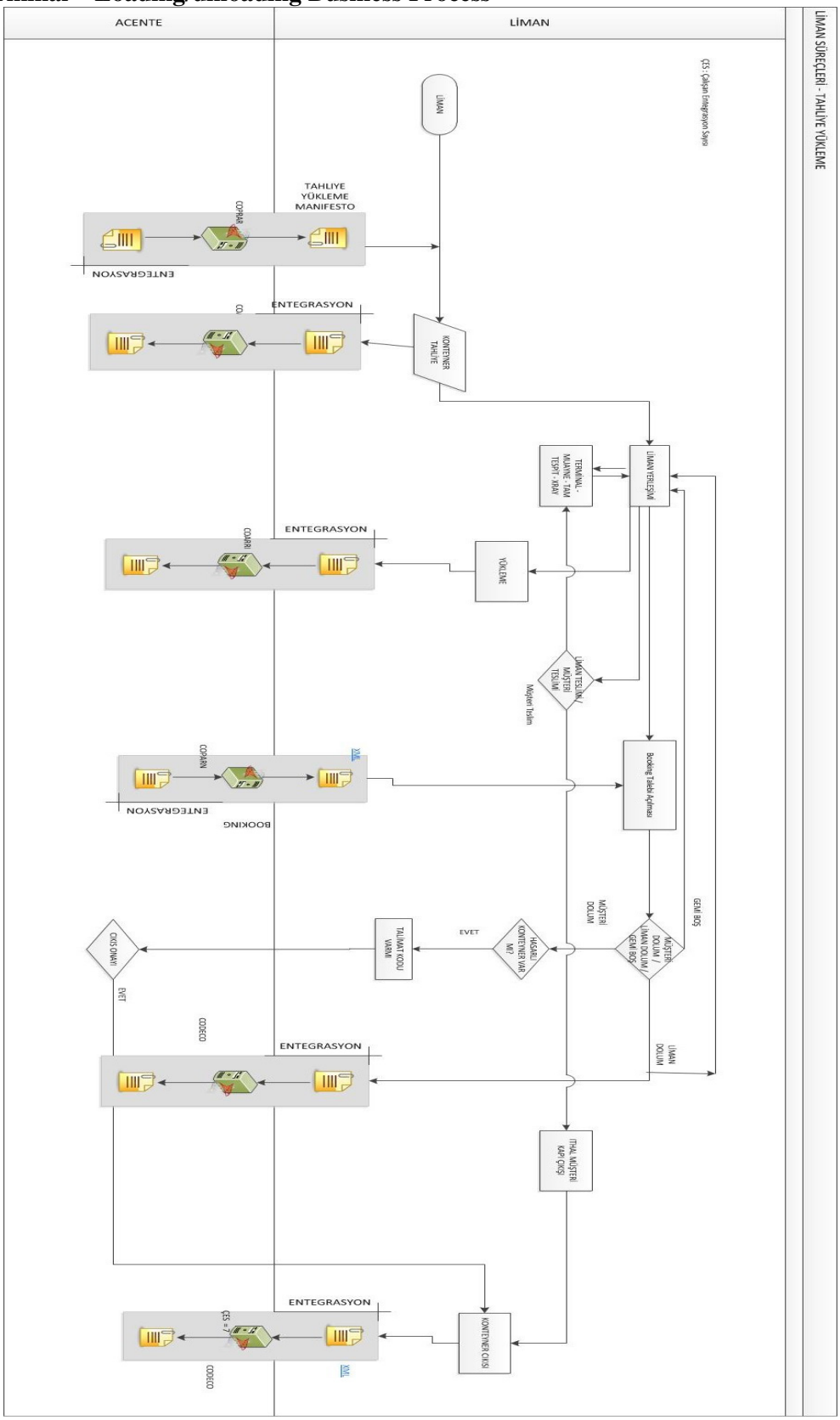
(rest of depot business is on next page…)


ÇIKIŞ BOOKING?
EVET
HAYIR
HASARLI KONTEYNER VAR MI?
EVET
ACENTEYE SORULUR
ÇIKIS ONAYI
EVET
HAYIR
ÇIKIS ISLEMI ILGILI KONTEYNER ICIN IPTAL EDILIR. DEPO YERLESIMI.
WEB SERVİS ile EDİ'ye ÇIKIŞ talimatı verilmesi.
KONTEYNER CIKISI
ENTEGRASYON
ÇIS - 7
CODECO
YMA

Terminal – Container Entrance Business Process



Terminal – Loading/unloading Business Process



Highway Business Process