Kimmo Viitala

Master's Thesis

Integration Architecture Development for Pori Energia

Degree Programme in ICT

2017

INTEGRATION ARCHITECTURE DEVELOPMENT FOR PORI ENERGIA

Viitala, Kimmo
Satakunta University of Applied Sciences
Degree Programme in Information Technology

ABSTRACT

Modern information technology systems today are nearly always divided into several enterprise applications and singe enterprise application is typically carried out by one software vendor with one architecture to implement one relatively limited role. Function of system integration is to enable software development that connects multiple single sector software into one application that can monitor and manage the execution of business processes.

The aim of this study is to describe system integration and its requirements and objectives. This study also aims to demonstrate different integration technologies and architecture. Final goal is to analyse Pori Energia's current integration system and construct new integration architecture and platform.

As a conclusion to the thesis can be stated that system integration is on the surface of digitalised world. Organizations are packed with different systems and software whose interoperability is business critical requirement. When well implemented, integration enables agility react to business changes and enhance daily efficiency fluency.

CONTENTS

AIM	Active Information Management,
BPEL	Business Process Execution Language
CSV	Comma-separated values
EAI	Enterprise application integration
ESB	Enterprise service bus
ICT	Information and Communication Technology
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
PESV	Pori Energia Sähköverkot Oy
PLV	Porin Lämpövoima Oy
STEP	Suomen Teollisuuden Energiapalvelut Oy
PDA	Personal digital assistant
SQL	Structured Query Language

1 INTRODUCTION

Pori Energia's current Integration solution are mostly build to from fully technical or expense saving aspect. However, integration solution should not be built only technical terms, integration is significantly more beneficial investment when architecture design is based on business model. Integration solution controls company's interfaces to contractor, customers and partners, and as company's business model changes efficient process optimization can be done through real time process monitoring. In company's business model information technology should simply be a tool and because integration is part of company's IT infrastructure, it needs to have a strategic role to improve business model.

1.1 Aims

The aim of this study is to describe system integration and its requirements and objectives. This study also aims to demonstrate different integration technologies and architecture. Final goal is to analyse Pori Energia's current integration system and construct new integration architecture and platform.

1.2 Structure of the thesis

The study is divided in to two parts first part being theory and second part being the Pori Energia Integration case study. At first study introduces system integration in general description and focuses on requirements and objective in system integration. Next chapter describes integration theory by introducing the intermediate software, integration models and design patterns.

After the theory study moves to case study where Pori Energia's current integration platform is analysed and project to develop the platform is constructed.

2 SYSTEM INTEGRATION

2.1 What is system integration

System integration is repertory of technologies and methods that enables incompatible applications to automatically communicate with each other. Integration can be equated to distributed application development, which aims to build diversified software that solves limited problems. Application can consist of the actual application logic, store used information to database and provide user interface which can be used over a network. In a working multicomponent information system different components need to have continuing connections to each other. Singular component without connection to other components is inefficient. Application can't handle information without a link to the database that stores the information and user interface is useless if it can't access the application.

Business processes in company today are nearly always divided into several enterprise applications and singe enterprise application is typically carried out by one software vendor with one architecture to implement one relatively limited role.

Function of system integration is to enable software development that connects multiple single sector software into one application that can monitor and manage the execution of business processes.

System integration is not a single product or technology, it's collection of design patterns. It is a practice in which multiple single applications are constructed in a way that benefit a bigger entity. (Tähtinen, 2005)
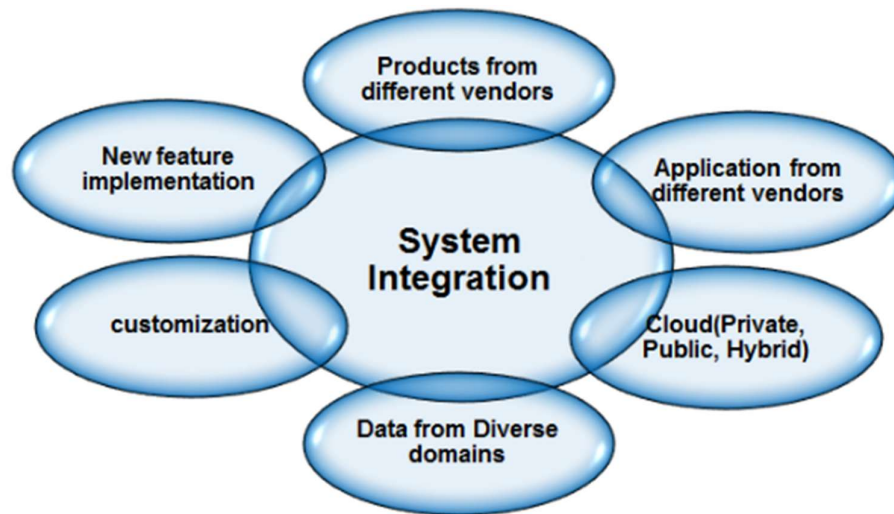
Figure 2.1 System integration (Athenaitsol, 2017)

2.1 Integration solutions

System integration is defined as operating models and technics that allow unattached information systems to share information with each other. Information sharing is done in a way that it can be controlled and monitored from a single point.

Company's information systems can be market software or customized for the company, systems can operate on various types of hardware and operating system platforms and be executed with multiple different programming languages. For efficient management and optimization of these important and multiform business processes centralized control of the data flow is crucial. (Tähtinen 2005)

For the most integration solutions are built in a scope of cost-effectiveness, but when integration is considered as a strategic solution for companies' data management and business strategy is included in integration architecture, integrations can used to create adjustable business processes and models. In this model business processes can be monitored and optimized in the changing business environment. (Tähtinen 2005)

2.2 Benefits of Integration

Most typical reason for system integration deployment is a need to optimize business processes, pursuit efficiency and savings as well as improve flexibility, reporting and monitoring.

A well-functioning business processes allow for the efficient operation of the company. When information delivery between business applications is automated and users no longer need to transfer files or mails manually processes are faster and errors decrease. In well integrated information technology environment, when information is once entered into system it integrates to other systems and users do not have to enter the information manually to several systems. Reducing user errors and working hour engagement with system integration, allows recourses to be focused on more important business functions such as planning and data utilization.

When system integration is used on all organization levels not only on individual departments, but also with its stakeholder's integration will clearly improve organizations service efficiency. Service efficiency and customer satisfaction both benefit from improved data correctness and consistency.

In business applications where information is transferred within the reach of an integration solution managing and monitoring, the integration solution can be used indirectly in the management of business processes. When data transformation between systems is timestamped and documented can this be used for reports that give important information on business flows improving processes, achieving competitive advantage and cost savings. When integrated well in to business processes system integration application can be used as a decision making tool for the management.

In addition to many of the important benefits integration solution also simplifies the organisation information system architecture. When processes and systems that don't add value are cut system maintenance gets simpler and resources can be directed to more productive activity. (Tähtinen, 2005)

3 TECHNOLOGY

3.1 Middleware models

Middleware is a mechanism that allows one entity to communicate with another entity. Intermediate software is therefore the unifying factor, which we can use to convert the program to use database from another system. Middleware software is also able to hide operating system and network technology from system developer allowing them to focus on APIs and network protocols, simplifying organizational structure of the system. There are logical and physical middleware models. Logical model presents how data is conceptually moved and physical model presents both the method of data movement and the technology used. Physical model consists of several messaging models that can be used singe model or combinations. (Linthicum, 2001)

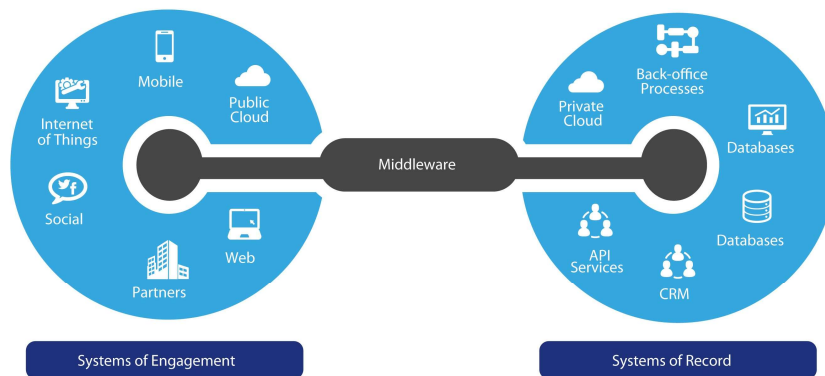
Figure 3.1 Middleware (6is-ta, 2017)

3.2 Point to point

In point to point model middleware provides direct connection from one application to other. Intermediate Software may be small access programs, or they may include the applications. Point to point model is a simple and fast way to connect two applications to each other. When compared to other middleware models point to point model is limited to only link two application to each other. When amount of applications are Increased quantity of connection is exponential which leads to a very challenging management. (Linthicum, 2001)

Sender   Listener

Message

Figure 3.2 Point to point (Linthicum, 2001)

3.3 Many to many

In many to many model applications are linked to many other applications with a separate intermediate software. The connection between the two applications on this model is not as fast as point to point mode, but many to many model has the advantage of good scalability, as one of the new application requires only one new connection. Because of scalability possibilities many to many model is the most used middleware model in system integration. (Linthicum, 2001)
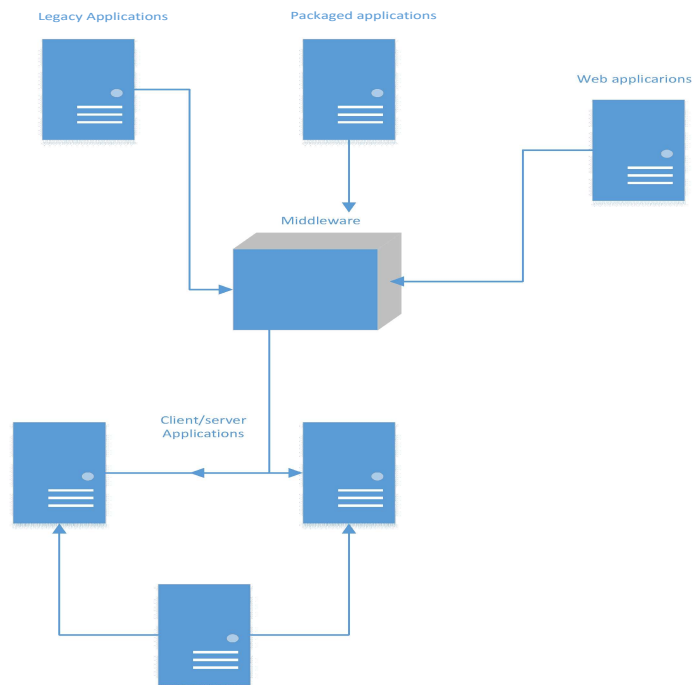
Figure 3.3 Many to many (Linthicum, 2001)

3.4 Synchronous and asynchronous

Middleware has two types of communication mechanisms. Synchronous model always has predetermined sequence and applications are dependent on middleware's and remote applications process sequence. Calling remote application stops processing on local application until remote application replies. Since the model is easily understood, it is also easy to maintain and creates simple integration design.

Disadvantages of synchronous model is coupling the application to middleware and remote application. On synchronous model problems with network or remote server stops all processing. (Linthicum, 2001)

The asynchronous model does not wait for a reply from the other party but moves the message to queue. While the answer to a task to complete the call is not faster than in synchronous model advantage in asynchronous model is that during this waiting time other tasks can be processes. Disadvantages of asynchronous model is the increasing complexity of planning and implementation. In asynchronous model attention must be paid to the state of the application when the answer arrives. When a query is sent to

several applications at the same time answers may come in a different order than they were sent. (Linthicum, 2001)

3.5   Connection-oriented and connectionless

In connection-oriented communication model every data transformation a semi-permanent connection is established before any useful data can be transferred.
The connection is terminated only when data transfer is completed. Design is generally carried out using synchronous calls. (Linthicum, 2001)

In Connectionless data model calling program does not create connection with target process and data is only forwarded to another application. Information can be sent in both directions simultaneously. Because the arrival of the received information is not guaranteed in this model acknowledgements are often used. (Linthicum 2001)

3.6   Direct and queued communications

In Direct connection model intermediate layer of software accepts the message from the calling programs and guides it to the target program. Direct connections are usually synchronous. (Linthicum, 2001)
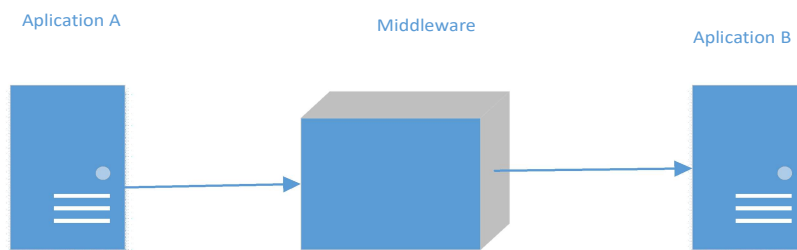

Figure 3.4 Direct communication (Linthicum, 2001)

Queued communications are generally carried out with queuing manager that places messages to queue. Remote application retrieves the message from the queue. Because the remote program does not need to be active when calling program sends message to it, communication middleware does not block either the calling program or the remote program from proceeding with processing. For this advantage over direct communication, queued communication used in most message oriented middleware. (Linthicum, 2001)
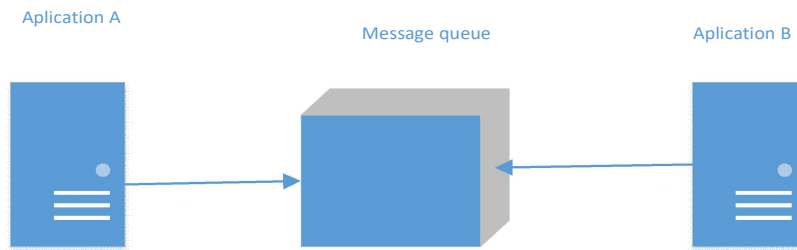
Figure 3.5 Queued communications (Linthicum, 2001)

3.7   Publish and subscribe

In publish and subscribe model, the publisher of the application will send the topic from which it wants to share information for publishing middleware. The middleware send all the participants information about the topic. After this, all application interested in the topic may be subscribe it. All subscriber applications receive only the messages sent by the publisher of the application to that topic. In this model, the publisher of applications do not need to know anything about the subscriber applications. (Linthicum, 2001)
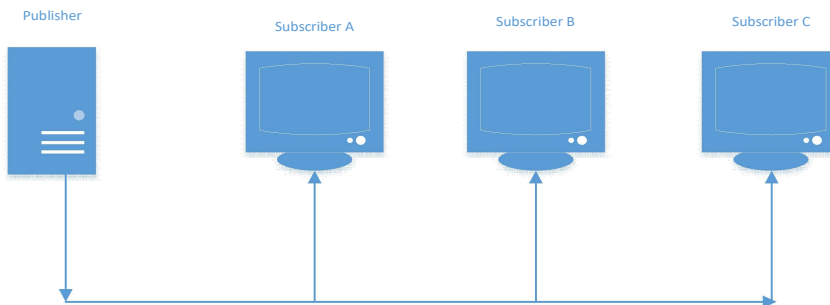
Figure 3.6 Publish and subscribe (Linthicum, 2001)

3.8 Request/response

In request/response model applications make request trough request/response middleware. Middleware transmits request to remote application and waits for the response. Response is then transmitted to source application.(Linthicum, 2001)
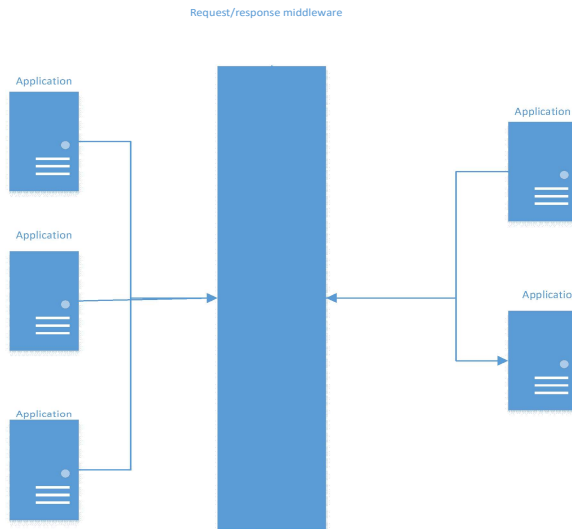
Figure 3.7 Request response (Linthicum, 2001)

3.9 Send and forget

In Send and forget model, the application can broadcast a message without having to worry about who receives it. The model is designed to give applications the ability to broadcast messages to multiple recipients without response features. (Linthicum, 2001)

3.10 Types of Middleware

The change in integration architecture is merging typical features of middleware to each other. Most middleware types now perform tasks with features of other middleware type. However, particular middleware types still solve certain type of tasks.
This chapter describes the technical aspect of the middleware software by going through the various implementation of technologies.

3.11 Database-oriented middleware

Database-oriented middleware facilitates transmission between a program and the database or different databases. Database-oriented middleware is typically used to extract information from local or remote databases. Database-oriented middleware can communicate with database using either standardized call-level interface or native database middleware. Native Database-software middleware offers database special functions and are often faster than the general command interfaces, but on the downside that they work only one manufacturer databases. (Linthicum, 2001)

3.12 Message Oriented Middleware

Message Oriented Middleware transmits a particular message from one application to another without interfering on its content. Because message-based middleware uses notion of messages to communicate between applications, direct coupling between applications is not needed. The asynchronous model that message oriented middleware uses allows message to be transmitted to queue manager from witch the message is delivered to its final destination. Message queue allows every application to function without interruption from middleware layer. (Linthicum, 2001)

3.13 Remote procedure calls

Remote procedure call can invoke function with application and have that it executed on remote application. Because remote procedure calls are synchronous, to be able to perform remote procedure call, they stop the execution of the program. When applica-

tion makes a remote call it keep the connection open until communication has completed. Remote procedure calls advantage is its simplicity, on the other hand this simplicity is gained with huge performance lost and inability to scale to big systems. (Linthicum, 2001)

3.14 Distributed objects

Distributed objects refers to a technique where a component from another application can be used as if it were the same application. Invitations can happen within the same machine or over a network. Distributed objects use standard interfaces and protocols to communicate with each other and are often implemented on top of remote call protocols. The difference between the remote call is that distributed objects bring applications developers all object-oriented software development techniques, such as encapsulation, recovery and diversity. (Linthicum, 2001)

3.15 Transaction-based middleware

The transaction-based middleware coordinates the flow of information movement and method sharing between many resources. Transaction processing is designed to maintain a system's Integrity, by ensuring that interdependent operations on the system are either all completed successfully or all cancelled. Sharing methods with transaction-based middleware is effective, but data transmission is slow compared to other possibilities. Benefits of the transaction-based middleware is scalability, fault tolerance and architecture, which centralizes application processing. (Linthicum, 2001)

Transaction Processing monitors provide mechanism for application logic and arrange communication between applications. They are based on transactional concept where a unit of works has a beginning and end. In transaction processing the transaction is either completed or transaction proses monitors restore situation to a point where it was before the start of operations. In a problem situation where databases or queues have been updated, transaction monitors roll back all changes. Transaction processing monitors provides the service that guarantees the integrity of services and resource management for processes. (Linthicum, 2001)

3.16 Message brokers

Message Brokers function is not application development, but brokers are technology that allows application to communicate without understanding anything about the other application. Message brokers are able to administrate information movement between two or more resource and they can transform schema and content of the information as it moves between applications and databases.
Message broker can change the format of messages the recipient can understand. In addition to these functions, broker is able to route the message data to the right place on the basis of its content. Transmission of messages can also be controlled on the basis of the rules laid down, for the message brokers may include a rule engine. (Linthicum, 2001)

3.17 Integration architecture models

When designing application integration, content of organizations business models and data well understand. Designer must understand how business processes will be automated and the importance of these processes.
There are basic methodologies to assist designers and provide opportunity to examine best practises to determine which processes and data needs to be integrated.

3.18 Data-oriented

Data oriented application integration is techniques and technology where information is extracted from one database processed and updated to another database.

Advantage of data oriented model is low cost and simplicity. Technology that enables information movement and reformation in databases is inexpensive if compared to other integration models. (Linthicum, 2001)

3.19 Application interface-oriented

Application interface-oriented integration uses existing interfaces to access processes and information. Applications have embedded interfaces for data sharing, but in many different formats. To integrate these systems to each other we need to access data trough interfaces, extract information and convert it to a format that target application understands and transmit the data trough target application interface. Limitations of application interface-oriented model is that it lays on specific features and functions of preconfigured application interfaces. (Linthicum, 2001)

3.20 Method-oriented

Method-oriented application integration is based on business logic sharing within enterprise. In Method-oriented model applications are able to access each other's methods without rewriting each application method every time. There are many technologies for method sharing between application like distributed objects, application servers and transactional processing. (Linthicum, 2001)

3.21 Portal-Oriented

Portal-Oriented application integration presents information from many locations, like other sites and applications in one user interface. With this method enterprises can avoid complexity and cost of several backend systems and can present information from several application with one user interface. (Linthicum, 2001)

3.22 Process Integration-Oriented

Process Integration-Oriented application integration is a management model that places Business-models layer on top of information movement mechanisms. Process integration model provides view on how information moves in the enterprise. Integrations are not made technology aspect first but by simulating business models. In process Integration-oriented model integration provides abstract process all other integration models. (Linthicum, 2001)

4 RESEARCH METHOD

The essence of a case study, the central tendency among all types of case study, is that it tries to illuminate a decision or set off decisions: why they were taken, how they were implemented ad with what result. A case study is usually selected as a research method when a single real-world case need to be studied. (YIN, 2003)

Case study can be divided in three approached technical, practical and aware. Choice between these approaches depends on what desired from the research. A technical approach is used when the researcher's is testing predetermined pattern in practical work. In functional approach, the researcher and the target community members work reciprocally and identify problems together and plan the necessary actions to solve the problems. In an informative approach, the goal is to have a practical problem and converse that to related theory (Paunonen, 1998)

Case study is cyclic proses witch is divided in to several phases. Phases of a case study are definition, projecting, implementation and evaluation phase.
Definition phase contains demand for development, preliminary development task and operating environment. At definition phase of this case study the integration needs and existing integration solutions were determined.

Projecting phase contains of production on concrete plans for change. In this phase integration architecture development project plan was conducted for Pori Energia.

Implementation phase evaluates the change process, the results achieved and the project phase consolidating change into a permanent operating model. At this point summarization of how the project has gone is done, the fulfilment of objective and experience gained is put together by the project in a final report to be used in future projects. (Paunonen, 1998)

5   CASE- PORI ENERGIA INTEGRATION DEVELOPMENT

Energy companies need to able measure and record customer's energy consumption by the hour through their information technology systems. Electric grid law regulates that electric company needs to be able to register power outage that is longer than three minutes and to be able send and receive load control function to power grid. Pori Energia has different system for customer information, energy data management and Network information system. These systems are all from different suppliers but need to change information. (Finlex, 2009)

Pori Energia

Pori Energia is Energy Company that is fully owned by City of Pori and operating primarily at the area of Pori. Pori Energia's primary products are electricity, district heading Industry energy service and contract work. Pori Energia OY started at 2006 when Pori Energia and Porin Lämpövoima emerged. Pori Energia Corporation has also subsidiaries such as Pori Energia Sähköverkot. In 2016 there were 227 employees working in Pori Energia and 19 workers in temporary employment relationship.

Permanent staff	2016	2015	2014
Pori Energia Oy,	203	215	216
Pori Energia Sähköverkot Oy	24	25	22
Energiapalvelut - STEP Oy	0	0	29

Pori Energia corporation's net sales at 2016 were 148.3 Million euros. Profit 14.6 million euros and outcome was 3.9 million euros.

	2016	2015	2014
Turnover	148,3	162	186,8
Turnover/Person (1000 €)	653,3	675	699,5
EBITDA, M€	29	28,6	32
EBITDA/Person (1000 €)	127,6	119,4	119,8
EBITDA-%	19,5	17,7	17,1
Profit, M€	14,6	12,5	15,1
Profit/person (1000 €)	64,4	52,1	56,4

Profit-%	9,9	7,7	8,1
Balance sheet total, M€	273,5	278,9	266,9
Investments, M€	21,9	18,3	33,7
Return on investment%	6	5,1	6,7
Equity ratio (%)	23,7	20,9	23,2

Electricity sales in 2016 were 1526 GWH and sales of district heating were 666 GWh. Transmission of energy at Pori Energia's area was 1156Gwh.

67.3% of total energy supply was acquired from Nord Pool energy market 32.7% of total electricity supply was produced with own production facilities.

13.6% with hydropower 14.8% with combined electricity and heat production 2.7% with condensing and 1.5% with wind power. (Pori Energia, 2016)

Used fuels

	2016 GWH	2015 GWH
Turf	505	540
Wood	713	632
Coal	42	8
Recycled fuels	50	51
Oil	24	11

5.1 System architecture

Main business systems that are integrated together are customer relation management system, network intelligence management system, and energy data management system.
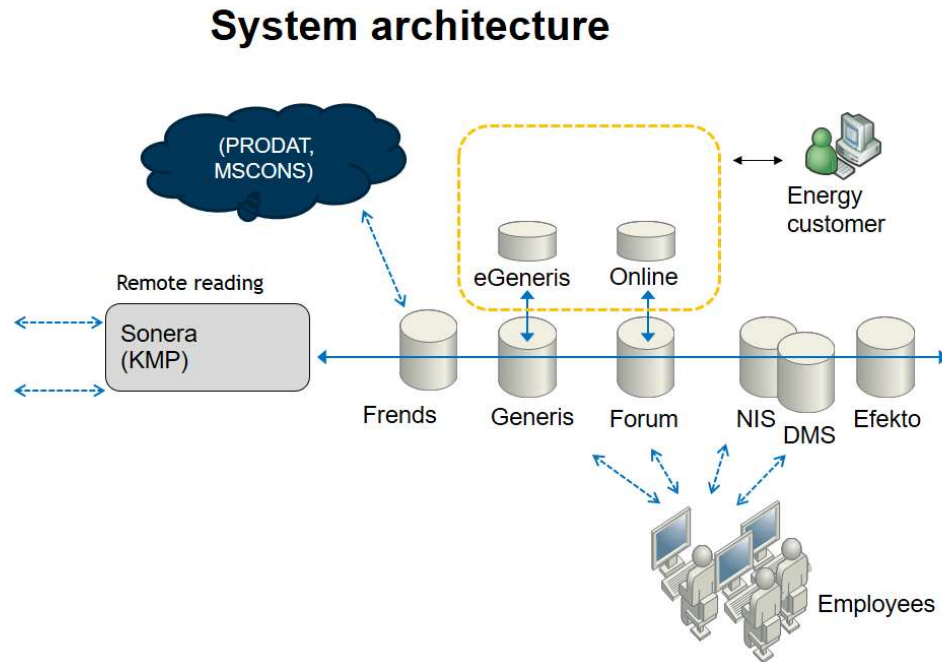
System architecture

Figure 5.1 Pori Energia system architecture (Mäkinen, Etäluennan loppuraportti, 2014)

Forum/Efekto

Customer relation management system supplied by Tieto.

CRM system maintains the customer register and handles electricity customers billing. The system has been developed for the Nordic energy market and includes functionalities that meet the requirements of the operating environment such as hourly measurement support and automated messaging between different market vendors (Tieto, 2015)

Generis

Energy Data Management software supplied by Enoro

EDM system provides tools to handle balance settlement, forecasting and pricing. Mainly Generis is used for smart meter data management, meter asset management and device control. (Enoro, 2017)

EGeneris

Customer portal for electricity consumption data.

EGeneris allows customers can monitor their own consumption at annual, monthly, daily and hourly levels. (Enoro, 2017)

NIS

Network information system software supplied by Trimble Inc.

Nis is an online information system consisting of an intelligent network model and integrated geospatial features. Nis is used to check technical dimensioning of the current and planned power grid to ensure that the components of the network implement the electricity distribution requirements. (Tekla, 2015)

KMP

Consumer metering service is service supplied by Telia.

The remote reading service is based on the intelligent meter installed at the customer point and the communication between meter and the Energy Company's information systems. The central function is consumption data reading and providing information to the Energy Company's information system.

Also electricity quality data and control commands between network operation and maintenance are delivered in KMP service.

Communication between interfaces.

In standardized data transmission data is send between interfaces electronically in form where information is presented in generally accepted standard. In standardized format information is transferred is named, grouped and arranged predetermined way. Data involving in certain issue is always presented the same way. Depending

on the situation content of the data can vary but same information can always be found on the same place and with same name in the file. (Tieke, 2010)

Extensible Markup Language

XML is a standardized markup language where the context can added to the information. XML languages can be used both as a format for communication between systems and as a format to save documents. The XML language is a structured description language that helps to parse big amounts of data to more easily understandable format (w3, 2017)

Work order data flow

Pori Energia uses electric work orders to book maintenance and installation work from suppliers. Electric work order is delivered to suppliers PDA with integration platform trough consumer metering service.

When work order is created from customer management system it will converted to several forms in several system before an electrician gets the work order to his personal digital assistant. At the first stake work order is enriched with data from the Network information system and converted in format readable for energy data management system. At the second stake work order is routed to prober workflow and converted to a format readable for consumer metering service. KMP will then transport work order to electricians PDA. After work is done electrician will send ready work order to consumer metering service. Integration service retrieves the work order from consumer metering service and enrich it again with data from network information system then convert and deliver the work order to energy data management system. Last stake is converting ready work order to a format readable for customer management system. (Mäkinen, Etäluennan tietojärjestelmät, 2014)
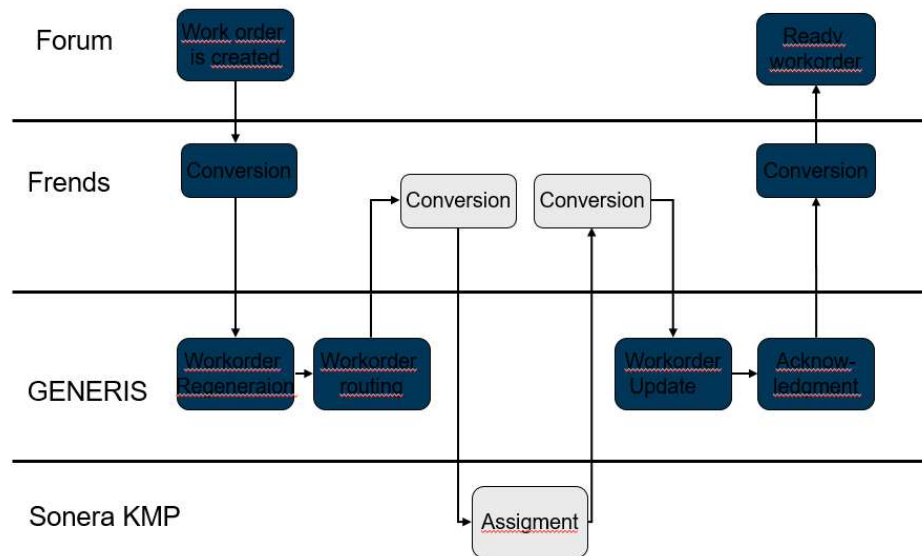
Figure 5.2 Work order data flow

5.2 Existing Integration Platform

As an integration platform Pori Energia is using Frends Iron 3.3 which is a system integration and proses management product from Frends Technology providing enhanced integration options. It is primarily meant for enabling enterprise batch processing to systems with multiple passive endpoints that need to send or receive data.

Frends Iron 3.3 is developed for enterprise application integration tasks, providing possibility to communicate with multiple different applications. Iron can be used for message based brokering capabilities and for batch-based transfers.

Frends Iron 3.3 consists of three functional parts. Iron service for executing parameterized batch processes according to pre terminated schedules. Log service for logging processes and messages from Frends or Biztalk processes. Web UI user interface for Frends, from which creating and monitoring processes can be done.

Frends Iron is a stand-alone process engine, with scheduling and parameterizing functionality. It can work together with Biztalk Server, providing several features on top of it. It can be used with server software such as Microsoft SQL Server, Biztalk Server and service bus for Windows Server (Mäkilehto, 2016)

5.3 Reason for migration

Frends Iron 3.3 is becoming old system and there are solutions in the market that have many features that Frends iron 3.3 lacks. Existing integration platform is using Biztalk servers which are inflexible and expensive. Pori Energia needs to change integration architecture design to be based on business model and needs integration platform that supports process integration-oriented application integration.

5.4 Target for the new integration platform

Integration platform where process information flows through a centralized integration system. Integrations are implemented in a coherent and visually modeled language on the integration platform.

The integration platform supports the following ways and functions:

- Data conversion (mapping)
- Transferring between different communication protocols (REST, Web Service, queue systems, SFTPS)
- SOA model
- ESB model
- Deployment chains based on data content or external data

Monitoring of the integration system is very important. In addition to monitoring the flow of data, the monitoring system must be able to monitor the processes of organization from one central point.

Integration platform supports the following ways and functions:

- Traffic light view to workflows.
- Possibility to drill down to workflows
- Possibility to silence control
- Filtered alarm messages

(Mäkilehto, 2016)

5.5   Technical view

Pori Energia plans upgrade integration platform to Frends 4 and migrate all integrations to new integration.
Frends 4 is integration platform designed IPaaS and on premise deployments.
Frends 4 distributed agent architecture with central command components communicating with agents significantly simplifies integration management through one console view and provides full transparency for data flow processing

These central command components include user interface, Microsoft SQL databases and services and Microsoft service bus. This enhances support team and business users by providing advanced control tools and easy-to-manage business processes that can be easily managed by business users. Frends 4 large volume handling is optimized to a level that provides performance, traceability and excellent error management.

Frends 4 also brings the latest software to production, erasing problems with software that is at the end of its life cycle. Frends 4 common components allows Pori Energia to manage integrations in a way where conversion do not cause refactoring projects.

5.6 Economical view

Using Frends 4 will provide centralized management of information flows through a single view, which opens a possibility to control information on business process management. User interface provides a transparent view of management, and report fault conditions via task-level logging, in which access can be limited to a single process enhancing support teams and key users activities. Frends version management has built-in feature that brings agility through a consolidated view of different versions. Deployment processing transfers different versions from one environment to another. Frends enables business processes required for the performance rights, easy access to the performance management, as well as tag views in which only the essential processes in the management console visible to the selected user groups. Frends 4 provides enhanced data base management, and adapter traffic management within the scope of a clear and centralized performance monitoring, Frends is dynamically expandable by adding server agents as business needs change.

The possibility to replace the Microsoft Biztalk server Frends 4 offers a clear potential for cost savings Pori Energia integration environment licences. (Mäkilehto, 2016)

6 MIGRATION PROJECT

According to research by Project management Institute (PMI) with 3000 repliers 28% off all ICT projects failed. 37% off the repliers told that reason for failure was mostly lack of clearly defined and accessible waypoints and milestones that support projects goal. Other reasons for project failure were bad commutation, employee opposition and insufficient funding. (Projekti instituutti, 2006)

This project was decided to be done in tree parts 1.preparation, 2.execution and 3.summarization. Each part contains decision points for guiding and directing the project.
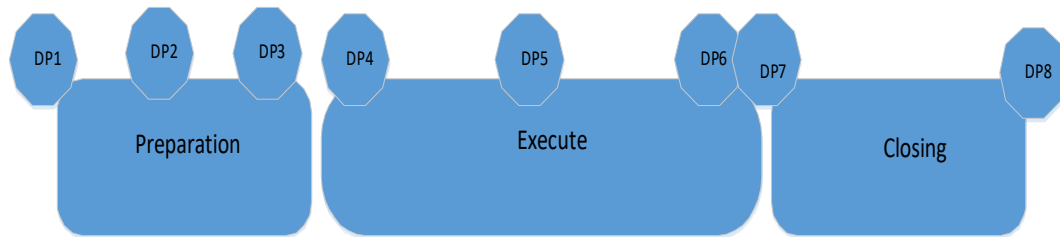
Figure 6.1 Project model

At this project model phases are well-defined and at each DP (decision point) project manager makes decision on how to continue project. Next chapter introduces project parts and decision points.

Preparation

At the preparation phase orderer, owner and the result off the project were determined. At preparation phase also project schedule with milestones and completed project organizations was compiled. In the project plan working methods and resource requirements for the project were set. At this phase also risk analysis and overall strategy for attaining the result was made. Biggest risk were determined to be risk of non-working integrations which would effect on time schedule and budget. Most often this is because of lack of time in testing. To overcome risks project plan was drafted in a way that several milestones were applied to each phase. Each phase would have several testing milestones and for each application multiple iteration phases were planned from the beginning.

Project recourses were comprehensively defined by individuals from different organization units to ensure commitment from different parts of the organization.

Last state at preparation phase was to complete project plan and have the project plan be accepted by project owner before execution phase can start.

Decision point DP1
- At DP1 Decision to initiate the project is made.
- If decision to proceed with project is made, basic planning for of the project directive in started.

Decision point DP2

- At DP2 decision to continue or interrupt preparation work is made.
- The basis for decision is made with preliminary project plan and schedule.

Decision point DP3

- At DP3 decision to approve project plan is made.
- The basis for the decision consists of the project plan, risk list and requirement/solution descriptions.

Execute

At the Execution phase project management handled the attaining of the results to bring the project to close .Project manager hold meetings to manage work towards next decision point. Project manager also checks that results and schedule are moving like planned at the project plan. Execution phase was divided to several milestones to help and guide project organization to attain project goals.

Decision point DP4

- At DP4 decision to start execution is made.
- The basis for decision consists of the approved project plan, approved requirement and solution descriptions for the result.

Decision point DP5

- At DP5 decision to continue or change the project is made.
- The basis for decision consists of project analysis, risk analysis and status reports.

Decision point DP6

- At DP6 decision to approve the result is made.
- The basis for decision consists of delivery meeting.

Decision point DP7

- At DP7 decision to transfer responsibly from project organization to owner.
- The basis for decision consists of verification of transfer documentation.

Closing

Examination on how the project was done was done at closing phase of the project, the fulfilment of objective and experience gained was put together so that it can be used in future projects. At closing phase phase project management also handed back the resources that project had.

Final report on the project was made and project was closed.

Decision point DP8

- At DP8 decision to close the project is made.
- The basis for decision consists of final report and the list of filed documents.

7 CONCLUSIONS

The scope of this work was to study integration theory and construct new integration model for Pori Energia.

Pori Energia like many others companies today have multiple individual business critical applications that each have their own software vendor and are running on their own system platforms performing only limited tasks. In order to optimize business processes and to manage business data efficiently must core business application be able to change data and communicate with each other.

Pori Energia is using multiple systems simultaneously, and critical business information is located in many different systems, and the data provided is not always consistent. The need for integration is usually observed when the software in use lacks the desirable features and some other system contains the desired features. The need for integration in the company is too often solved in ways that are based solely on technical solutions or cost-effectiveness. Too often, integration projects are only integrating two system or at worst creating point to point transfers, thus only adding to the complexity of systems and worsening fault tolerance.

Designing and constructing well working system integration platform can be heavy and long project for a company if the basics of integration are not in order.

Project commission is best be done in well planned project model so that schedule, costs and recourses can be estimated comprehensively. If the organization does not have sufficient knowledge and resources to comprehensively understand the requirements of system integration it is recommended to use consulting company. It is important to reserve adequate amount of time and resource that integration project is able to produce all necessary documentation and verify the accuracy of the processes.

In many organizations the commissioning of system integration requires a lot of changes to current processes. At the the implementation stage it is necessary to ensure continuous communication the organization's staff to comply with project management principles and to avoid the change resistance. Employees should be included to develop processes and participate in planning as much as possible and from as many business process area units as possible.

At Pori Energia the starting point for integration development was good, since the company had invested in system integration for several years and had sufficient competence for designing new integration solutions. Project model used in information system projects also supported integration project. Project team which consisted of best in integration professionals in the province guaranteed wide-ranging awareness on the demands and quality of work.

What comes to case Pori Energia integration platform development project it can be stated that predetermined goals of the project were accomplished. New integration platform and processes were generated and taken into service in schedule.

As further action Pori Energia must continue the development of integration platform according company's integration road map.

8   REFERENCES

6is-ta. (2017, 11 02). *Middleware*. Retrieved 11 2, 2017, from 6ix-ta.com: http://6ix-
     ta.com/middleware

Athenaitsol. (2017, 12 2). *System integration services*. Retrieved 12 12, 2017, from
     www.athenaitsol.com: http://athenaitsol.com/system-integration-services

Enoro. (2017, 12 09). *generis-platform*. Retrieved 12 09, 2017, from Generis:
     http://www.enoro.com/generis-platform

Finlex.   (2009).   *Valtioneuvoston   asetus   sähköntoimitusten   selvityksestä   ja
     mittauksesta*.   Retrieved   09   04,   2017,   from   www.finlex.fi:
     http://www.finlex.fi/fi/laki/alkup/2009/20090066

Linthicum, D. (2001). *B2B Application Integration.* Boston: Addison-Wesley.

Mäkilehto, P. (2016). *Integraatio määrittely.* Pori: Pori Energia.

Mäkinen, T. (2014). *Etäluennan loppuraportti.* Pori: Pori energia.

Mäkinen, T. (2014). *Etäluennan tietojärjestelmät.* Pori: Pori Energia.

Paunonen, M. &.–J. (1998). *Hoitotieteen tutkimusmetodiikka.* Juva: WSOY.

Pori Energia. (2016, 10 05). *Pori energia toimintakertomus*. Retrieved 09 04, 2017,
     from   www.porienergia.fi:
     https://www.porienergia.fi/Global/Vuosiraportit/PoriEnergia_TOIMINTAKE
     RTOMUS_2016_web.pdf

Projekti instituutti. (2006, 12 10). *miksi_projektit_epaonnistuvat*. Retrieved 12 10,
     2017,   from   www.projekti-instituutti.fi: https://www.projekti-
     instituutti.fi/blogi/miksi_projektit_epaonnistuvat

Tekla. (2015, 12 20). *NIS*. Retrieved 12 20, 2017, from www.tekla.com:
     http://www.tekla.com/fi/tuotteet/trimble-nis

Tieke. (2010, 06 30). *Tiedonsiirtosuositus*. Retrieved 12 16, 2017, from www.tieke.fi:
     https://www.tieke.fi/display/Sanomarekisteri/Energiateollisuus-
     Tiedonsiirtosuositus+2010-06-30

Tieto.   (2015,   12   9).   *Forum*.   Retrieved   12   9,   2016,   from   Tieto:
     http://www.tieto.fi/toimialat/energia/it-ratkaisut-
     sahkoverkkoyhtioille/energiayhtioiden-asiakashallinta-ja-
     laskutusjarjestelmat-tieto/forum-asiakaspalvelujarjestelma-
     sahkoverkkoyhtioille-tieto

Tähtinen, S. (2005). *Järjestelmäintegraatio – Tarve, Vaihtoehdot, Toteutus.* Jyväskylä: Gummerus Kirjapaino Oy.

w3. (2017, 10 1). *XML*. Retrieved 12 20, 2017, from www.w3.org: https://www.w3.org/XML/

YIN, R. K. (2003). *Case study research: design and methods.* Thousand Oaks: Sage Publications.