DATA BASE SYSTEM PLANNING TO ERP (ENTREPRISE RESOURCE PLANNING) HIGHER EDUCATION WITH METHOD SOA (SERVICES ORIENTED ARCHITECTURE)

Anief Rufiyanto
(aniefrufiyanto@gmail.com)

Abstract— *ERP system for higher education is a software in order to accommodate the resource needs in an educational institution. ERP (entreprise resource planning) is an information system companies that designed to coordinate all resource, information and activities that need for complete business process. ERP system based in database, at general and software plan modular. Primal condition from ERP system is integration. Integration, it's mean to unite various need in one software in one logical database. Database as base application, built consideringly base SOA (service oriented architecture). Service Oriented Architecture (SOA) is implemented by technologies other than web services, but the m and concepts has gained popularity recently. SOA be software planning that designed in such a manner, up to there flexible moment development phase system and or integration process, where prepare loosely-integrated suit of services that at use in various business domain. The system assesses the development of the utilization of existing resources in an educational institution. Because after all educational institutions should have policies not only make a profit as an economic enterprise. Implemented starting from the data of teachers, students, subject matter to, evaluation of students' learning, including the annual report. Database that support ERP in higher education scope with service need in this time, with guide system SOA in follow fast technology change, the data base system must has integrated, shared, easy and automation. Design and implementation of system 3 components, namely: People (human resources and student), Process (interaction of the learning process and the system) and Product (learning content,output).*

Keywords — *higher education, entreprise resource planning, database, service oriented architecture*

I. INTRODUCTION

In information era in this time, globalization of education world compels higher education increase information management service, so that information need towards stake holder related to education can get fast service response and always follow condition (real time) moment a information wanted. To achieve that things need an existing information technology management according to planned and structured. Governance of information technology is connection structure and process to aim and restrain organization to achieve the aim, with display the process functional value to increases education service institution performance quicker, correct and accurate.

Database development in a higher education undoubtedly very need for organization more complexer with various need to aim software development ERP (entreprise resource planning). ERP solution enterprise entire assets in organization in a system with integration process. [1] ERP be has inwrought to provide resource need in a human resource good education institution (human resource) also according to physical resource (material resource) and non-fisik (immaterial resource), where from all existing resource information displayed institution projection or institution in certain range of time at period to come. Thereby a decision making manajerial in this institution, one of [the] deliberation factual and accurate data about organization goes on. Higher education institution as one of [the] organization non-profit structure, differ from to company profit oriented.

Development database concept to erp based on SOA (services oriented architecture) this aim to support inwrought education service based on education new paradigm at in this time and future, one of [the] factor supporter predominantly service (services) towards all related parties. Implementation and data base planning must threaten in 3 principal components, that is: person (student university, lecturer, employee), process (study process, evaluation, all supporter activity interaction) and result (pass, curriculum, and another).

Concept forwards ERP (entreprise resource planning) for this higher education is developed especially to help stake holder principal an education institution, so that organization mengelolanya can bloom well appropriate existing national education system outline reference. As well as can as soon as may be anticipate insident on institution resource projection happen at period to come. Thereby education institution can increase quality with national standard reference.

ERP system data base based on SOA (services oriented architecture) developed based on a modelling application as composition from nest service that provided a component. SOA a software planning that designed in such a manner so that flexible moment used in development phase a system and or integration process, where with software planning berorientasi service this prepare sevis certain that at use in so many business domain. [2] Database existing, must in easy unitary groups be accessinged. principal characteristic SOA promotings to pass software components merging. between software component interact in one unitary. While profit it from architecture use SOA flexible, scalable, replacability and fault tolerance.

II. BASE THEORY

II.1 DATABASE SYSTEM

Database system has scope broader is compared with data base. Data base system can be interpretted data base collection in a system, as a whole has connection with supported another component. [3] Data base system shaped nest sub system that consist of data base with users that use data bases according to together, with application apart in the supporter computer system. has several important elements, that is: [3]

1. data base as kernel from data base system.
2. software for planning and data base management.
3. hardware as data processing operational supporter.
4. human/brainware, as user, designer and data base manager.

Data base is a data collection mutual connected reasonablely and describe integration between a table with table other, made to fulfill need from a organization. [1] Data base development, according to has profit principal aim likes to follow: [3]

1. *Data can be used by many users*. Data base must can to realize customer between data items from many files. So that different user or different application program can use data base equal to different manner.
2. *Intellectual investment.* application program and structure logik that is on in this time, unnecessary done to repeat when there change in data base. New data can easily mengintegrasikan easily towards data available.
3. *Cost emphasis.* Stock holding cost emphasis be maked because redundancy data voidable. Data base need change cost more complex, unnecessary done from beginning, so that management doesn't want long time.
4. *Cause the loss of double system.* Development sub new system is done permanently threaten in same data base so that prevent the happening of double system development. This concept make possible development can happen in structure, data base volume, and sub data processing system permanently defend integration delivers sub system.
5. *Performance increases*. performance enhanced according to personal reachable with information need that can be fulfilled swiftly, correct, easy and accurate based on from data base.
6. *Data clarity*. Every user can detect clearly data everything available and data authority that can be accessinged.
7. *Use ease*. With application program and structured processing system, unnecessary user thinks technical complexity that is faced moment also pengaksesan that automatic.
8. *Using flexibility*. Data base accesing can be done variously differ, because data base not depending application that worn.
9. *Easy to Change*. Database alterable without influence manners to use data.
10. *Accuracy and consistency*. Accuracy data in data base is done since data catching process up to display information; data consistency can awake because since beginning candid storage from redundance data via data control.

Planning basisdata can be looked at from two sides that is user side and designer side. User person or application program that accessing data base. Designer system designer or program, also manager basisdata. from planning side distinguishable two opinions differ that is conceptually and according to fisis. Conceptual opinion shaped datas that want kept with customer between data one and another. Opinion fisis be applied form related technical, example: data storage technique, data range and storage capacity, storage time database and back up data.

Main database criteria has integrated and shared. Integrated the purpose basisdata shaped merging several different data files, with avoid reduplication a part or entire datas. Explanation shared mean each data file individually usable collectiv by many users, in so many program application without mutual disturbed.

II.2 ERP – ENTREPRISE RESOURCE PLANNING

ERP usually used for planning in manufacturing business and berorintasi profit, while a college not a manufacturing business and should not berorientasi profit. In this watchfulness is system planning concept basisdata college will adopted terminology ERP with architecture based on in servis.

ERP-entreprise resource planning not always threaten to a software. ERP consist of many terminologies that include also company/organization asset as a whole and various process mendalamnya. [4] According to modular, software ERP usually divided on principal module that is:

- *Operation Module*. General Logistics, Sales and Distribution, Materials Management, Logistics Execution, Quality Management, Plant Maintenance, Customer Service, Production Planning and Control, Project System, Environment Management
- *Finacial and Acuntancy Module*. General Accounting, Financial Accounting, Controlling, Investment Management, Treasury, Enterprise Controlling,
- *Human Resource Module*. Personnel Management, Personnel Time Management, Payroll, Training and Event Management, Organizational Management, Travel Management.

ERP projected as entreprise/organization management enhanced step. Marginally, ERP can be described as nest tools management (forecasting, payroll, logistics, inventory, and another), business process use and fast decision making and tested, consumer service in trade according to electronic (customer, supplier), planning and resource scheduling (human, materials, engine, and as it).

Application ERP software that developed to answer business need in the case of integration and business process synchronization. This application will grow otomation function in data presentation and company/organization information according to real time. System ERP can import benefit if we know to how manner get it with facilities exist in ERP. Hook between benefit by get it, draw in table 1: [1]

Table 1. ERP Benefits

Manner Benefits	**Get It TOOLs**
Reliable information	Access DBMS flexible, consistent data, and accurate, reporting system better
Avoid data duplication and operation.	That modules accessing data from basisdata centrally, so that avoid entering process and data modification from various different points, duplication not happen.
Speed up data processing time	Minimased data taking time and report maker
Decrease cost	Saves time, increase control with do analysis comprehensive towards decision organisational.
Adaptation ease	Change in business process can adapted easily
Increase scalability	Has modular system structure and easy to customised.
Support maintenance	Supprted after-sales with a long time system.
Global development	Module extension up to covers supply chain management, customer relationship management .
e-commerce	Internet business, collaborative culture.

Furthermore from use activities ERP be got profit among others:

1. *Finance data integration*. Top managemen can see and controls company/organization finance performance eminently, finance data blessing integration.
2. *Operation* process standardization. Operation process is standardized to pass applied and optimal test at field so that happens productivity enhanced, inefficiency depreciation and product quality enhanced.
3. Standardization and information data. Data and information is made in reporting uniformity, especially to company big usually consist of many business unit with total and business various kind.

III.3 SOA – SERVICE ORIENTED ARCHITECTURE

Service-based architectures take legacy application functionality and expose it to the Internet in a reliable, highly available, scalable, flexible, manageable and secure manner, easy and reliable internet-based method to create and access learning. Web service technology has emerged as a new paradigm of distributed computing. [5]

The orchestration service layer provides a powerful means by which contemporary service-oriented solutions can realize some key benefits. The most significant contribution this sub-layer brings to SOA is an abstraction of logic and responsibility that alleviates underlying services from a number of design constraints. [6] For example, by abstracting business process logic:

- Application and business services can be freely designed to be process-agnostic and reusable.
- The process service assumes a greater degree of statefulness, thus further freeing other services from having to manage state.
- The business process logic is centralized in one location, as opposed to being distributed across and embedded within multiple services.

Services can exist at many levels of abstraction, but generally they can be broken into four broad categories: [7]

- Infrastructure Services
- Point-to-Point Services
- Business Services
- Composite Business Services

Service-oriented architecture presents an approach for building distributed systems that deliver application functionality as services to either end-user applications or other services. It is comprised of elements that can be categorized into *functional* and *quality of service*. The architectural stack is divided into two halves, with the left half addressing the functional aspects of the architecture and the right half addressing the quality of service aspects. [8] These elements are described in detail as follows:

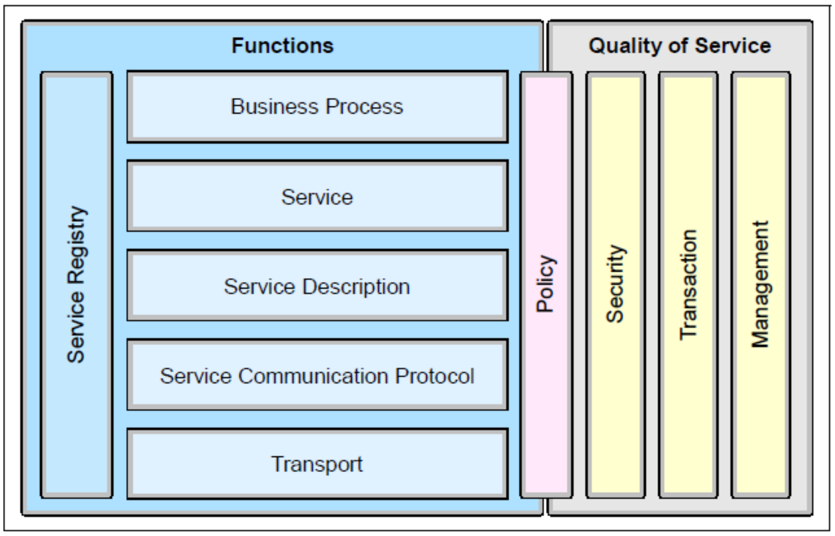

Figure 1. Elements of a service-oriented architecture.

1. **Functional** aspects include:

– *Transport* is the mechanism used to move service requests from the service consumer to the service provider, and service responses from the service provider to the service consumer.
– *Service Communication Protocol* is an agreed mechanism that the service provider and the service consumer use to communicate what is being requested and what is being returned.
– *Service Description* is an agreed schema for describing what the service is, how it should be invoked, and what data is required to invoke the service successfully.
– *Service* describes an actual service that is made available for use.

– *Business Process* is a collection of services, invoked in a particular sequence with a particular set of rules, to meet a business requirement. Note that a business process could be considered a service in its own right, which leads to the idea that business processes may be composed of services of different granularities.
– The *Service Registry* is a repository of service and data descriptions which may be used by service providers to publish their services, and service consumers to discover or find available services. The service registry may provide other functions to services that require a centralized repository.

2. **Quality of service** aspects include:

– *Policy* is a set of conditions or rules under which a service provider makes the service available to consumers. There are aspects of policy which are functional, and aspects which relate to quality of service; therefore we have the policy function in both functional and quality of service areas.
– *Security* is the set of rules that might be applied to the identification, authorization, and access control of service consumers invoking services.
– *Transaction* is the set of attributes that might be applied to a group of services to deliver a consistent result. For example, if a group of three services are to be used to complete a business function, all must complete or none must complete.
– *Management* is the set of attributes that might be applied to managing the services provided or consumed.

Service oriented architecture (SOA) desirable design principle to builds system distributing reliable and send service fungsionalitas, with add emphasis in loose coupling deliver user service. in this case service has several characteristics:

a. Logical view, service that seen from level business operation that identified as independent interfaces.
b. Message orientation, a service that client that swop message.
c. Menycription orientation, servive mendekripsikan as processor engine metadata.
d. Network orientation, service demanded to worn in network. this matter is menenkan in need service automatically with easyly is found.
e. Platform neutrality, message is submitted to pass interfaces that use platform neutral (multi platform) and data format standart like xml.

III. METHODS

From various review hit data base and ERP on that applicable in higher education scope with service need in this time for college, with guide in various system superiority SOA in follow system change so the data base system concept must has easy and automation

Universities differ from other organizations because they have different environments and circumstances, and they use ERP technologies for academic purposes. Faculty and staff commonly interact with core institutional activities through ERPs, and students need more information and better E-learning environments. In sum,this means that the system is, by definition, critical to the institutions’ mission. Furthermore, these organizations are governmental organizations and do business for profit and non-profit purposes, which might make ERP systems in these organizations with different concomitances especially with the high percentage of failure implementations.[9]

All of these issues lead to critical questions about the success and the benefit of ERPs for theses organizations. the core part of these issues centralizes the study's aim on whether or musical note the system improves user performance, and also whether ERP systems meet staff requirements in higher education environments.

Data base concept for this need is that can to support criteria ERP and as according to system architecture based on service, all at once multiplatform doesn't depend on operating system and or application, permanent data base not influenced. but such permanent the data base system can to developed at period to come, unchanged significant in itself data. only system need and application that bloom appropriate organization complexity need.

IV. DISCUSSION

ERP systems are often the largest software application adopted by universities with significant amounts allocated to their implementation. Higher education institutions are persisting in the information system era by adopting and implementing ERP system. The need to evaluate their benefits and impacts on organizations and individuals are increasingly essential.

The success of ERP systems can be classified into two categories: the success of ERP adoption and the success of ERP systems implementation. For the successful ERP adoption, this research used already proven user acceptance models for information system such as technoloy acceptance model and success model as the starting point. The model developed the rationale for the causal relationship based on these combined theoretical backgrounds and incorporated three main dimensions for identifying the truth about the success of ERP systems: success factors, intermediate constructs, and success indicators. The model also considered the success of ERP implementation based on the reviews on the project management fundamentals. Technoloy acceptance model is formulated in an attempt to achieve these goals by identifying a small number of primary variables suggested by previous research dealing with the cognitive and affective determinants. A construction project varies from one context to another depending on determinants including complexity, duration, budget, and quality. In ERP projects, the complexity depends on the project scope that includes the number of business functions affected and the extent to which ERP systems implementation changes business processes.

Four success factors for ERP implementation as follows: 1. Top-management support, planning, training, and team contributions; 2. Software-selection efforts; 3. Information systems area participation; and 4. Consulting capability and support. They also provided five outcome questions, which were shown to be significantly correlated and should, therefore, be combined to form a single outcome factor, effectiveness. Their regression analysis proved that all the success factors can affect the outcome significantly; therefore, these factors can be considered the representative success factors in ERP implementation. [10]

Enterprise Resource Planning systems are key to optimizing organizational performance; however, choosing which system to implement, when to perform the implementation, and how to minimize costs while maximizing system acceptance are all important issues to businesses considering an ERP implementation.

Designing ERP-enterprise resource planning appliedly at college, as according to paradigm now must based on service. By because in this time and future at period to come, want capacity and complexity excelsior. Such, database system that wanted correcter accustom to configuration simple but easy grow, without change to repeat as a whole system and the process.

Enterprise computing systems (that's a m we'll use to refer to the computer systems that is run a company's business) has grown up in relatively isolated "silos" over the last 40 or 50 years or so. that is means that when business functions were "computerized" many years ago, separate programs2 were written for all those particular functions and run on a computer. whenever a particular function was desired, that program was invoked by the user. essentially, there was a specific program for each specific purpose. that is meant that is there were many separate programs, each designed and used for a relatively narrow scope of functions. [12]

Service oriented architecture (SOA) desirable design principle to builds system distributing reliable and send functionality service, with add emphasis in loose coupling deliver user service. in this case service has several characteristics:

1. *logical view*, service that seen from level business operation that identified as independent interfaces
2. *message orientation*, a service that client that swop message.
3. *menycription orientation*, servive mendekripsikan as processor engine metadata.
4. *network orientation*, service demanded to worn in network. This matter is optimized in need service automatically with easyly is found.

Enterprise processes are usually distributed across multiple applications and hardware/software systems. These processes are also event-based in the sense that the sub processes are linked by a series of events. For example, the depletion of inventory at a manufacturer may lead to an eventtrigger that's automatically generated and propagated to one or more suppliers to replenish the depleted inventory items. [13]

Each system's software architecture reflects the different principles and set of tradeoffs used by the designers. Service-oriented software architecture has these principles, such :

1. *Services are discoverable and dynamically bound.*
   SOA supports the concept of service discovery. A service consumer that needs a service discovers what service to use based on a set of criteria at runtime. The service consumer asks a registry for a service that fulfills its need. The best way to explain dynamic binding and discover is to use an example. For example, a banking application (consumer) asks a registry for all services that perform credit-card validation. The registry returns all entries that support this. The entries also contain information about the service, including transaction fees. The consumer selects the service (provider) from the list based on the lowest transaction fee. Using a pointer from the registry entry, the consumer then binds to the provider of the credit card service. The description of the service consists of all the arguments necessary to execute the service. The consumer formats a request message with the data, based on the description provided by the directory pointer. The consumer then binds the message to a transport type that the service expects and sends the service the request message over the transport. The service

provider executes the credit-card validation and returns a message, whose format is also specified by the service description. The only dependency between producer and consumer is the contract, which the third-party registry provides. The dependency is a runtime dependency and not a compile-time dependency. All the information the consumer needs about the service is obtained and used at runtime. This example shows how consumers execute services dynamically.

2. *Services are self-contained and modular.*
   Services are self-contained and modular. One of the most important aspects of SOA is the concept of modularity. A service supports a set of interfaces. These interfaces should be cohesive, meaning that they should all relate to each other in the context of a module. The principles of modularity should be adhered to in designing the services that support an application so that services can easily be aggregated into an application with a few well-known dependencies. Since this is such an important concept when creating services, we will explain some of the principles of modularity and, in particular, how they apply to the creation of services. Outlined the following five criteria for determining whether a component is sufficiently modular. These criteria apply equally well when determining whether a service is sufficiently modular. ***Modular Decomposability,*** The *modular decomposability* of a service refers to the breaking of an application into many smaller modules ***Modular Composability,*** The *modular composability* of a service refers to the production of software services that may be freely combined as a whole with other services to produce new systems. ***Modular Understandability,*** The *modular understandability* of a service is the ability of a person to understand the function of the service without having any knowledge of other services. ***Modular Continuity,*** The *modular continuity* of a service refers to the impact of a change in one service requiring a change in other services or in the consumers of the service. An interface that does not sufficiently hide the implementation details of the service creates a domino effect when changes are needed. ***Modular Protection,*** The *modular protection* of a service is sufficient if an abnormal condition in the service does not cascade to other services or consumers. For instance, if an error in the checking account service causes invalid data to be stored on a database, this could impact the operation of other services using the same tables for their data.
3. *Services stress interoperability.*
   Service-oriented architecture stresses interoperability, the ability of systems using different platforms and languages to communicate with each other. Each service provides an interface that can be invoked through a connector type. An interoperable connector consists of a protocol and a data format that each of the *potential* clients of the service understands. Interoperability is achieved by supporting the protocol and data formats of the service's current and potential clients.
   Techniques for supporting standard protocol and data formats consist of mapping each platform's characteristics and language to a mediating specification. The mediating specification maps between the formats of the interoperable data format to the platform-specific data formats. Sometimes this requires mapping character sets such as ASCII to EBCDIC as well as mapping data types. For instance, Web services is a mediating specification for communicating between systems. JAX-RPC and JAXM map Java data types to SOAP. Other platforms that supportWeb services mediate betweenWeb service specifications and their own internal specifications for character sets and data types.
4. *Services are loosely coupled.*
   Coupling refers to the number of dependencies between modules. There are two types of coupling: loose and tight. Loosely coupled modules have a few wellknown dependencies. Tightly coupled modules have many unknown dependencies. Every software architecture strives to achieve loose coupling between modules. Service-oriented architecture promotes loose coupling between service consumers and service providers and the idea of a few well-known dependencies between consumers and providers. A system's degree of coupling directly affects its modifiability. The more tightly coupled a system is, the more a change in a service will require changes in service consumers. Coupling is increased when service consumers require a large amount of information about the service provider to use the service. In other words, if a service consumer knows the location and detailed data format for a service provider, the consumer and provider are more tightly coupled. If the consumer of the service does not need detailed knowledge of the service before invoking it, the consumer and provider are more loosely coupled. SOA accomplishes loose coupling through the use of contracts and bindings. A consumer asks a third-party registry for information about the type of service it wishes to use. The registry returns all the services it has available that match the consumer's criteria. The consumer chooses which service to use, binds to it over a transport, and executes the method on it, based on the description of the service provided by the registry. The consumer does not depend directly on the service's implementation but only on the contract the service supports. Since a service may be

both a consumer and a provider of some services, the dependency on only the contract enforces the notion of loose coupling in service-oriented architecture.

5. *Services have a network-addressable interface.*
   The role of the network is central to the concept of SOA. A service must have a network-addressable interface. A consumer on a network must be able to invoke a service across the network. The network allows services to be reused by any consumer at any time. The ability for an application to assemble a set of reusable services on different machines is possible only if the services support a network interface. The network also allows the service to be location–independent, meaning that its physical location is irrelevant. It is possible to access a service through a local interface and not through the network, but only if both the consumer and service provider are on the same machine. This is done mainly to enhance performance. Although a service may be configured for access from a consumer on the same machine, the service must
   also simultaneously support a request from across the network. Because of this requirement, service interface design is focused to a large extent on performance.
6. *Services have coarse-grained interfaces.*
   The concept of granularity applies to services in two ways. First, it is applied to the scope of the domain the entire service implements. Second, it is applied to the scope of the domain that each method within the interface implements. The levels of granularity are relative to each other. For instance, if a service implements all the functions of a banking system, then we consider it coarsegrained. If it supports just credit-card validation, we consider it fine-grained. In addition, if a method for inquiring about a customer returns all customer information, including address, this method would be coarser-grained than a method that does not return the customer's address.
   The appropriate level of granularity for a service and its methods is relatively coarse. A service generally supports a single distinct business concept or process. It contains software that implements the business concept so that it can be reused in multiple large, distributed systems.
7. *Services are location-transparent.*
   Location transparency is a key characteristic of service-oriented architecture. Consumers of a service do not know a service's location until they locate it in the registry. The lookup and dynamic binding to a service at runtime allows the service implementation to move from location to location without the client's knowledge. The ability to move services improves service availability and performance. By employing a load balancer that forwards requests to multiple service instances without the service client's knowledge, we can achieve greater availability
   and performance. As mentioned earlier, a central design principle in object-oriented systems is
   separation of implementation from interface. This means that an object's interface and its implementation may vary independently. The primary motivation for this principle is to control dependencies between objects by enforcing the interface contract as their only means of interaction. Service-oriented architecture takes this principle one step further, by reducing
   the consumer's dependency on the contract itself.
8. *Services are composable.*
   A service's composability is related to its modular structure. Modular structure enables services to be assembled into applications the developer had no notion of when designing the service. Using preexisting, tested services greatly enhances a system's quality and improves its return on investment because of the ease of reuse.
9. *Service-oriented architecture supports self-healing.*
   With the size and complexity of modern distributed applications, a system's ability to recover from error is becoming more important. A *self-healing* system is one that has the ability to recover from errors without human intervention during execution. *Reliability* measures how well a system performs in the presence of disturbances. In service-oriented architecture, services will be up and down from time to time. This is especially true for applications assembled from services from multiple organizations across the Internet. The extent to which a system is self-healing depends on several factors. Reliability depends on the hardware's ability to recover from failure. The network must also allow for the dynamic connection to different systems at runtime. Modern Internet networking protocols inherently provide this capability.
   Another aspect of self-healing is the architecture from which the application is built. Architecture that supports dynamic binding and execution of components at runtime will be more self-healing than one that does not. For instance, service-based systems are self-healing to a greater degree than previous

architectures, because services are bound to and executed dynamically at runtime. If a service fails, the client may find, bind, and execute a different service, as long as the other service provides the same or a similar interface contract.

“Database is a self-describing collection of integrated tables”, mean database a data collection that describe integration between table one with the other table. “Database is a self-describing”, here explained that data structure mutual integration in a place that known as data dictionary or metadata. follow to be reason from database use: [1]

1. Solid. unnecessary again make paper archives in large size.
2. Speed. engine can get to return and change far data quicker than human that can do.
3. Decrease donkey work. go against the stomach from process take care archives - archives shaped paper deductible.
4. Recent. newest information and accurate always available menyetiap when wanted time.

Database with data warehouse based on object oriented project (command sql) that produced slimer and will accessing data manifestly quicker, that is total record that processed fewer and process join decrease. from overall increase efficiency prosentase if united will produce average increase efficiency prosentase 461.801,84%. show that data use warehouse more handal and efficient compared old fashion database use. [11]

Data model relational often also called basisdata relational. this model will show a mechanism with will managed /mengorganisasi data physically in secondary memory that will affect also in pengelpmpo overall data in system. [3] DBMS (database management system) software system that make possible user to define, make, take care and controls access to database. database management system special software that is used to make, accessing, controls, and regulate a database. database management system be data aggregate mutual connected and also contain program collection to accessing data.

So, database management system interacted software with user application program and database. database management system prepare several facilities as follows: [1]

1. DDL (data definition language) make possible user to define data type, structure, and limitation - limitation (constraints) in data that kept into database.
2. DML (data manipulation language) make possible user to put into (insert), change (update), wipe off (delete), and display (retrieve) data from database
3. Access control prepare access kontrol to database, like security system, integrity system, concurrency control system, recovery control system, and user-accessible catalog.

DBMS has several functions, that is:

a. Data storage, retrieval, and update. A DBMS must equip/prepare user ably storage, penelusuran return, and change data in database.
b. An user-accessible catalog. A DBMS must prepare catalog that describe data storage location and can be accessinged by user.
c. Transaction support. DBMS must prepare a mechanism that will guarantee every from all activities will change that also not transaction.
d. Concurrency control service. DBMS must prepare a mechanism to guarantee that database can be changed truly when several users change when does concurrent database.
e. Recovery services. DBMS must prepare a mechanism to repair data base that broken because a certain insident.
f. Authorization services. DBMS must prepare a mechanism to guarantee that only user that given authority that can accessing data base.
g. Support for data communication. DBMS must can ber-integrasi with communication software.
h. Integrity services. DBMS must prepare a manner to guarantee that data in data base and data change, both follow correct rules.
i. Services to promote data independence. DBMS must cover facilitiess that support independency programs from recent data base structure.
j. Utility services. DBMS should prepare facilities service nest.

In this time, semantic object model is one of [the] data model maker technique often worn web service technology, because modesty in the form of words. this data model later menransformasikan will be database design. data from form, report and query represented in the form of semantic object. same like entity, semantic object consist of attribute - attribute. attribute semantic object consist of 3 kinds:

a. s*i*mple attribute. single element.

b. *group* attribute. collection from single element that has similar character, for example: address consists of road, city, provinsi, zip code.

c. *semantic* object attribute. attribute that connect between semantic object one with another.

Every has cardinality with format n. m that is n minimal cardinality and m maximal cardinality. semantic object model doesn't has one way customer, if a object contains object other, so second object also contain first object, this called paired attributes. found object identifier be one or more object attribute that used to identify object and group identifier be identifier that has more than 1 attribute.

1. There 7 object types from semantic object diagram: [31. simple objects. object has only simple attribute or group attribute, and the value single-value.
2. Composite objects. object has only simple attribute or group attribute, and the value there multi-value.
3. Compound objects. object that contain most a little one object attribute (may augmenting one / two attribute kinds other) but value single attribute and value group attribute- single-value. There 4 types compound objects, picture under this describe fourth type.

Object 2 can *Object 1 contain* *can contain*	*One*	*Many*
One	1:1	1:N
Many	M:1	M:N

4. Hybrid objects. object that be combination between composite objects and compound objects, has one sub-attribute object in group attribute- the value multi-value.
5. Association objects. a object that two or more object attribute other and keep unique data for customer.
6. Parent/subtype objects. same like concept class, parent class principal and subtype descendant from object other and contain entire attributes from object.
7. Archetype / version objects. semantic object that make semantic object other that represent version, rilis, or edition from archetype.

V. CONCLUSION

Based on analysis result and concept planning that done, so can be taken conclusion as follows this:

1. ERP (enterprise resource planning) can be adopted in information system planning at higher education, information system that developed to use terminology ERP supposed can produce repair continue (continous improvement) from activity and penyediaan resource.
2. Service Oriented Architecture (SOA) is implemented by technologies other than web services, but the m and concepts has gained popularity recently. web services acres refocusing organizations on the concepts of service-oriented architecture. although highly reusable, loosely coupled architectures has been a field goal for many organizations. web services acres fostering interest in and providing the technology to implement service-oriented architectures that is enable them to realize their is vision.
3. Database that support ERP in higher education scope with service need in this time, with guide system soa in follow fast technology change, the data base system must has integrated, shared, easy and automation.

REFERENCES

[1] Robby, Owen Kwanentent, Frans Mei Wardana ; ANALISIS DAN PERANCANGAN BASIS DATA UNTUK MENDUKUNG APLIKASI ERP EDUCATION PADA UNIVERSITAS BINA NUSANTARA (Studi Kasus : Academic Management and Content Preparation), *Bina Nusantara University*

[2] Prasetyo, Hendro Joko; IMPLEMENTASI SERVICE ORIENTED ARCHITECTURE (SOA) MENGGUNAKAN TEKNOLOGI WEB SERVICE; (Hendrojogja@yahoo.com),*Fakultas MIPA Universitas Widya Dharma Klate*

[3] Sutanta, Edhy, *BASIS DATA DALAM TINJAUAN KONSEPTUAL*, CV. Andi Offset, Yogyakarta, 2011;

[4] Choldun R, Muh. Ibnu; PERANCANGAN SISTEM INFORMASI AKADEMIK DENGAN MENGIMPLEMENTASIKAN ERP; ibnucholdun@yahoo.co.uk ; Laboratorium Sistem Informasi Politeknik Pos Indonesia; Jalan Sariasih 54 Bandung, 40151

[5] Jabr, Mohammed A. Al-Omari, Hussein K.; *e-Learning Management System Using Service Oriented Architecture*;1Department of Computer Science,Amman Arab University for Graduate Studies, Amman, Jordan 2King Abdulaziz University for Science and Technology, Riyadh, Saudi Arabia

[6] Erl, Thomas; *Service-Oriented Architecture: Concepts, Technology, and Design*; Prentice Hall Professional Technical Reference, ISBN 0-13-185858-0, July 2005.

[7] Brown Paul C. Ph.D.; *Service-Oriented Architecture : A View From the Field Principal Software Architect*; TIBCO Software, 2006.

[8] Endrei, Mark, Ang,Jenni, etc*; Patterns: Service-Oriented Architecture and Web Services*; IBM Red Books, April 2004.

[9] Ahed Abugabah, Louis Sanzogni ; ENTERPRISE RESOURCE PLANNING (ERP) SYSTEM IN HIGHER EDUCATION: A LITERATURE REVIEW AND IMPLICATIONS; *International Journal of Human and Social Sciences* 5:6, 2010

[10] Chung,BooYoung; Skibniewski, Mirosław J.; Kwak, Young Hoon; DEVELOPING ERP SYSTEMS SUCCESS MODEL FOR THE CONSTRUCTION INDUSTRY**;** Journal Of Construction Engineering And Management. © ASCE / March 2009

[11] Warnars, Spits; TATA KELOLA DATABASE PERGURUAN TINGGI YANG OPTIMAL DENGAN DATA WAREHOUSE; ISSN: 1693-6930 _ 25, Fakultas Teknologi Informasi, Universitas Budi Luhur Jl. Petukangan Selatan, Kebayoran Lama, Jakarta 12260, 021-5853753,e-mail: spits@bl.ac.id

[12] Block, Jim; SERVICE-ORIENTED ARCHITECTURE: THE IMPORTANCE AND RELEVANCE OF SOA IN THE FINANCIAL ENTERPRISE; Diebold Global Advanced Technology, January 2007

[13] SERVICE ORIENTED ARCHITECTURE; IMPLEMENTATION FRAMEWORKS; Fiorano Software, Inc. Copyright 2005.