كلية الهندسة بشبرا
FACULTY OF ENGINEERING AT SHOUBRA
ERJ
ISSN:1687-1340

ENGINEERING RESEARCH JOURNAL (ERJ)
Volume (52),Issue (2)
April 2023, pp:26-31
https://erjsh.journals.ekb.eg

# Enterprise Architect Modelling of Lean Six Sigma Convergence in Supply Chain Operations

**Naser El-Musrati [1,*], Ahmed M. Gaafar [1], Hosam Zakariya [1], El-Awady Attia [1].**

[1] *Mechanical Engineering Department, Faculty of engineering Shubra, Benha University, Cairo, Egypt.*
** Corresponding author*

*E-mail address: nelmosrati@gmail.com , ahmed.marzouk@feng.bu.edu.eg, hossameldin.zakaria@feng.bu.edu.eg, elawady.attia@feng.bu.edu.eg*

**Abstract:** Organizations use standards and frameworks to improve supply chain performance for different reasons of fast changing markets and competition, were methods of dealing with issues is decisive to their success. The development of Enterprise Architecture framework "EA" based on Service Oriented Architecture "SOA", Architect applications define almost everything in enterprises, improves the ways of doing things in industries, services and their supply chains, in a Business-IT alignment dimension. Adding Lean Six Sigma "LSS" features in EA environment is a challenge and opportunity to provide a solid extended supply chain systems network. Modelling this coexistence in an Architect is expected to add a visual follow-up to the supply chain as EA viewpoint and push it to a new level. A conceptual model suggested to depict the Supply Chain Operations, LSS and EA. For this reason a high level Enterprise Architect application used to combine a cyclic activities represented to highlight and emphasis the features, principles and gains of an integrated new tools, supports a knowledge system development and achieves a safe and secure supply chain capability. The research outcomes suggest an industrial engineering era that treats Enterprise Architecture as an effective tool.

**Keywords:** Supply Chain, Lean Six Sigma, Enterprise Architecture. Modelling.

## 1. Introduction

Complex data-oriented supply chains require innovative methodologies to model their components and achieve objectives. The work aims to use Lean Six Sigma "LSS" tools to optimize supply chain on an Enterprise Architecture "EA" work floor [1]. "EA" is concerned with enterprises design, having the Open Group's TOGAF framework. Architecture defined as a "description or detailed plan of an enterprise at component level that guides the implementation thereof" [2]. EA is "a conceptual blueprint that defines the structure and operations" of an organization. The current and future organization objectives are things that EA focuses on, and aims to assist organizations to effectively reach their strategic objectives [3]. According to "The Open Group" it is necessary to develop an EA to provide support as well as essential technology and process structure for any IT strategy. It describes the way in which an organization or part of it, like the supply chain, will function its relationships and structure, its business models as well as the way technology and information systems support the organizational goals and objectives [4]. EA is the archive and information that supports Engineers and decision-makers at various scope levels and details, provides performance score, offers a strategic context for growth and systems' change in response to industry environment, shift in demand, aligns IT and operation. It produces technical and business benefits of; an efficient IT operations, reduced infrastructure complexity and faster procurement.

The Author used an Enterprise Architect application [5], depicted in Fig. 1, to offer the features required to research scope, trace, view optimization points and mechanisms that support Enterprise principles and guidelines. The Architect supports Software as a Service "SaaS" cloud services that offers the following:

- A delivery model makes a reusable elements leased and used for different purposes and platforms via different terms along the SC by cost-for-use only.
- A standard operations and processes to resolve issues and incidents.
- A systematic update of functional and non-functional features, batch follow-up, easier cooperation, communication and global accessibility.
- Resilience to interface external tools such as LSS tools, requirements settings and data flow options in nodes driven by DMAIC stages and ADM iterations.

Enterprise Architecture functions achieve normal "SC" chain objectives. The Architect assumed to offer processes' descriptions and requirements in a convergence model. Supply chain stakeholders, partners, customers' packages and services-list use a service menu that utilize a local, central and/or cloud hosted server. Services may be offered through enterprise channels, points of services or by a contracted Partner with The Open Group Architecture Forum "TOGAF" framework. Services requests are offered and tracked with almost no time, cost and quality sacrifices. Improvements are carried out in the service system, continuously optimized by LSS cyclic tools [6]. In this context, data-oriented LSS tools are called as a Solution Building Blocks "SBB" or Architecture "ABB" in data flow, nodes or store points, with schematics, figures, tables... etc., of Enterprise Architect origin are generated and followed to give all the details.

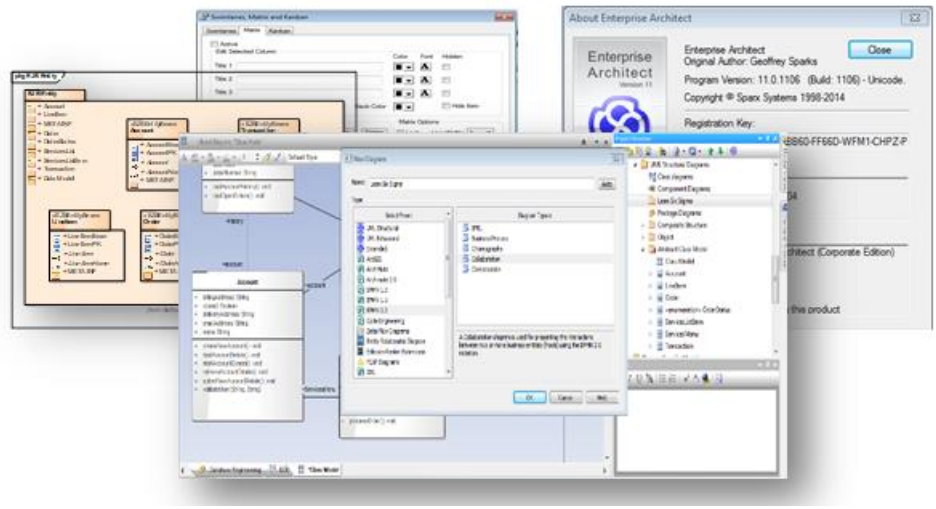


**Fig 1.** Architect. Zachman profiles and packages [5].

## 2. METHODOLOGY

Based on Service Oriented Architecture "SOA" environment and methodology shown in Fig. 2, the Architect offers the opportunity to visualize "viewpoints" in different diagrams and descriptions. The convergence criteria followed uses "SC" as Architecture viewpoint in strategy dimensions, follow TOGAF principles, uses LSS's DMAIC stages for continuous optimization, concentrates on technology strategic dimension and Business-IT alignment [7].

Architect Modelling allows SC clients and staff to trace interrelations of LSS and SC components in a standard UML elements displayed as images and interface in "Resources", "Actors", "Deployment Model" or "Stakeholders", demonstrate SC operations and LSS tools applications functioning, applied with extensions to UML, existing modelling languages and to create Model Driven Generation MDG Technologies. Enterprise Architect supports UML 2 diagram types and modelling of; "Structural" diagrams of high level packages represent SC domain, Formal Requirements and Enterprise Java Bean "EJB". Settings show "Class" with Attributes and Operations as; "Viewable", "not shown" or "only showing the Testing". This shows, Menus of elements, resources allocated in Projects display, status differentiate Colors to indicate requirements, and drawn discussions' diagrams. "Behavioral" diagrams offers; Use Case, Activity, State Machine, Communication, Sequence, Timing or Interaction diagrams to trace functions such as; "Users", "Services", etc. Figure 3, partially shows a behavioral diagram in SC functional requirements View after adding LSS element.

"Class" Diagrams include; C# Model-No Attributes, C# Model, EJB and JAVA Model, generated using Model Driven Architecture "MDA", transform from "Abstract Class" Model, create an "Implementation" model, (Model Dependent Architecture "MDA" and Platform Specific Model "PSM") [5]. Use Case model is an UML of use cases that describes a catalogue of system functionality, each case represents a single, repeatable interaction that an actor experiences when using the system. Figure 4, Shows a Architect-built Use Case includes one or more "scenarios", describe the interactions that go on between the Actor and the System, documents the results and exceptions that occur from user's perspective, defines how services optimization activities are performed.

Actors are modeled system Users, each Actor have a role and may perform more than one Actor role although assume only one role in one interaction. Roles may be performed by a nonhuman, such as an application. Human Actor has a large amount of variable data stored in profiles and a database for different uses, some of which are related to performance assessment, other data is for optimum resources use, behavior analysis and mentorship [8]. Use Cases include sub-Cases as part of a larger interaction pattern extended by other use cases to handle exceptional conditions as required by optimization process. "Component" diagrams are models hardware components with composite diagrams to represent "SC" processes. Figure 5, shows structures required to serve entire SC functions.

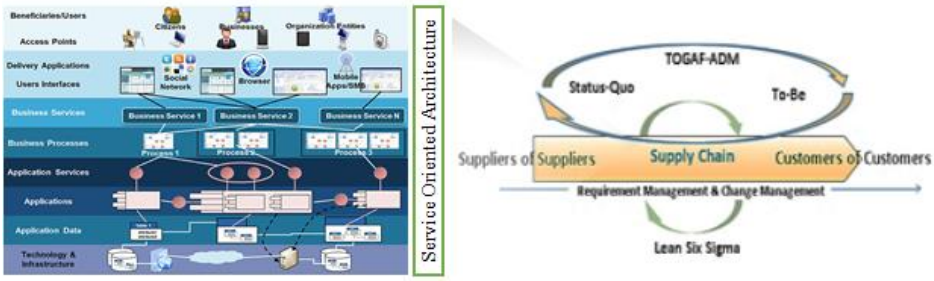


**Fig 2.** Service Oriented Architecture based Methodology.

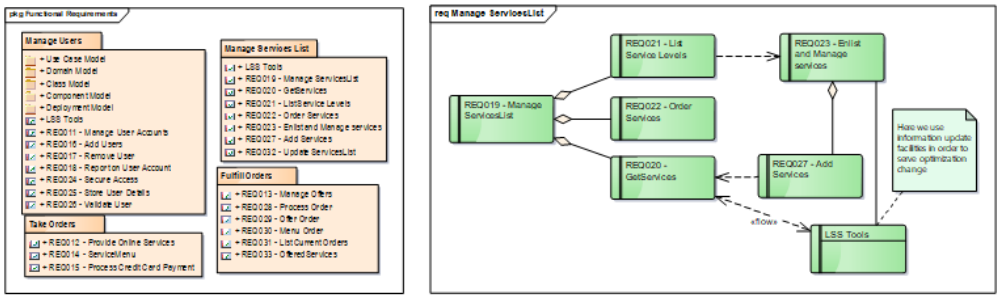


**Fig 3.** Functional requirements and supply chain service list overview

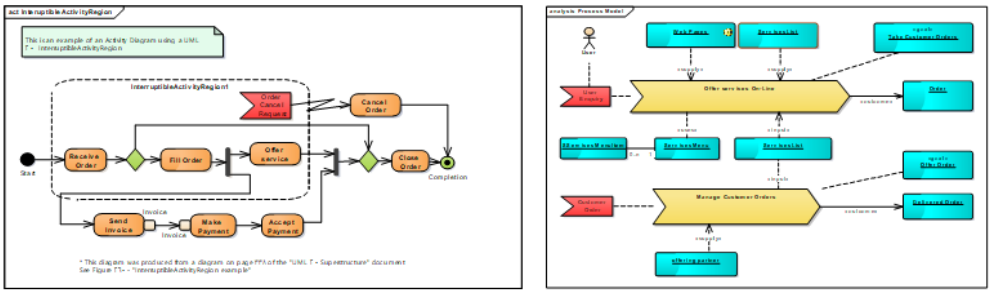


**Fig 4.** Process analysis model and Lean Six Sigma entry nodes

Actors are modeled system Users, each Actor have a role and may perform more than one Actor role although assume only one role in one interaction. Roles may be performed by a nonhuman, such as an application. Human Actor has a large amount of variable data stored in profiles and a database for different uses, some of which are related to performance assessment, other data is for optimum resources use, behavior analysis and mentorship [8]. Use Cases include sub-Cases as part of a larger interaction pattern extended by other use cases to handle exceptional conditions as required by optimization process. "Component" diagrams are models hardware components with composite diagrams to represent "SC" processes. Figure 5, shows structures required to serve entire SC functions.

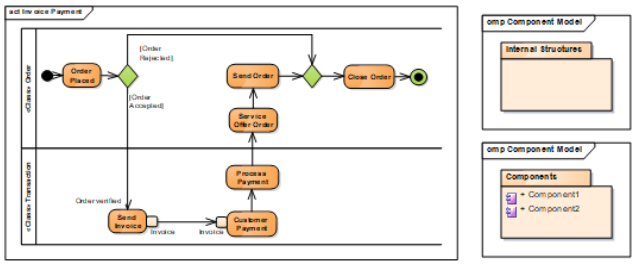


**Fig 5.** Payment process partition diagram, components and Composite Structure.

"Components package" contains modeled SC components and structural interfaces, ports and other gateways elements or internal structural components. Packages in Fig. 6 use "use cases" to define an Actor interaction with the system, specify scenarios, sequence, interface, dynamic diagrams and descriptions to clarify the system view as user interacts.

Connectivity and internal structure are further modeled in "Internal Structures" and "Connections packages" in a flexible configuration that accepts changes asked by "requirements". SC "Component Model" defines how classes, artifacts and other low level elements are collected into a high level components and interfaces and connections between them, deployed to a replaceable and adjustable hardware platforms described in a "Deployment Model" and impacts optimization demanded changes. "Internal Structures" provide a component's internal tasks and dependencies view in Fig. 7. It illustrates how the component fulfills its behavioral contracts and interface to other components within the SC. "Internal Structures" contain models of how components are wired for internal and external services [9]. A "composite" diagrams used to detail the connections and dependencies within components, between classes (objects or parts) and interfaces (services), provide a specific and optimum behavior to other components. The "component" exposes other "SC" components' "interfaces" for further component's services to perform its work. "Dependency" relationships show functionality met by other components and how the whole system works.

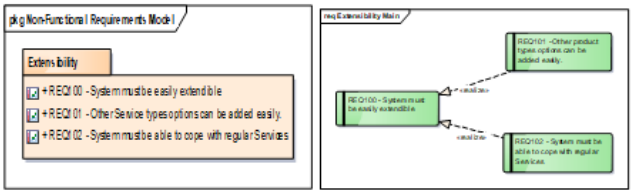


**Fig 6.** Non-functional requirements representation.

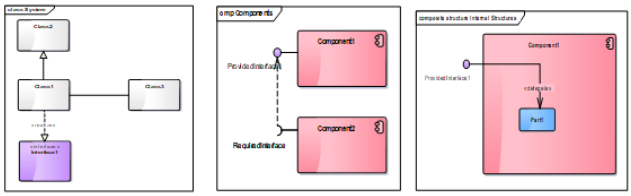


**Fig 7.** Package of Supply Chain modelled components and structures.

## 3. RESULTS

SCOR model followed, where clients -whoever they are- have tasks or services list to select from, and get a list of them from a server deployed either by a local server or a contracted partner. These services offered, demanded or traced with almost no time, cost and quality sacrifices. Improvements are recommended, approved and implemented in an on-line service system. SPARX Architect application sets the diagrams followed with hints and relations. In this context, LSS Tools are configured locally or on cloud and called-in as a Solution or Architecture Building Block, "SBB's" or "ABB's" to process in data flow, node or store. The Architect allows to document requirements graphically, using "Requirement element" that is available from the "Requirements" Toolbox folder to visualize package content architecture. Figure 8. Shows Tools insert options as vertical lines for "LSS" to trace changes in a visual view [10].

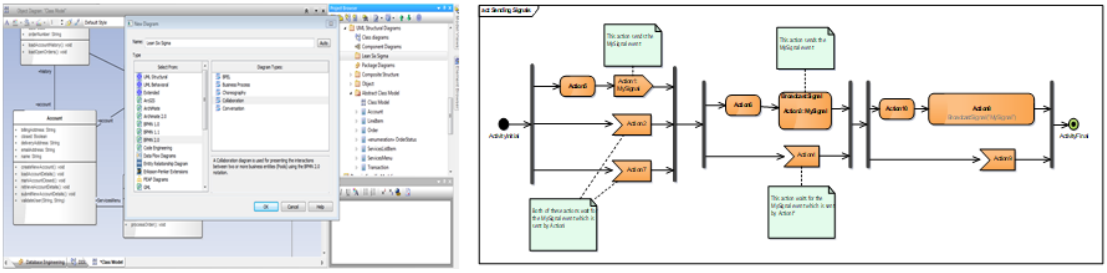

**Fig 8.** Hosting Tools in nodes (vertical lines) in content architecture flow.

Using a "Requirement" element in "UML" model, allows drawing relationships between requirements, capture weakness points and "SC" direct traceability to other model aspects as "Use Cases", "Test Cases" and other Analysis or Design elements required for best performance and document formal, nonfunctional, performance or security requirements as shown in Fig. 9 SC data flow.

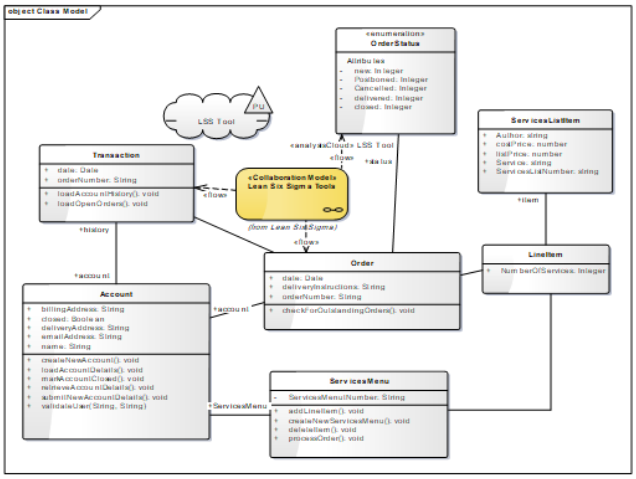


**Fig 9.** Data flow to and from Lean Six Sigma tools Class Diagram [11].

"Requirements" of supply chain or a sub-system are defined using the Custom Element of type "Requirement", with "element" views the properties. It can have relationships with other elements such as other "Requirements", "Use Cases" etc. A "Hierarchy" window views "Requirement" traceability accessed from the main menu and LSS tools are available to be called automatically to send messages as required by tool selection in every data processing or store nodes [12].

## 4. ANALYSIS (DEPLOYMENT MODEL)

The Architect viewed model describes how and where to deploy quality system in a supply chain. Machines, tools and devices are reflected as nodes, and the internal construction embed additional nodes as "artifacts", like service applications assigned to nodes that model system's run-time configuration, guided deployment specifications. Final system's physical locations, artifacts and connectivity deployment nodes is set in a Topology package, models system's connectivity, networks, physical locations and spatial characteristics along the supply chain, making a map of how services are handled and by which actors in "Kaizen" improvement [1]. This resolves weaknesses captured during optimization check cycles. Nodes package contains all "SC" hardware and software components models required to be utilized in the final implementation, such as LSS tools in Fig. 10, where almost all data elements are subject to LSS tools definitions, analysis, improvement and control. Automation substitutes human complex variables of uncertainity with those of a stable simpler Actors. System Boundary shows the logical interface between users and the system.

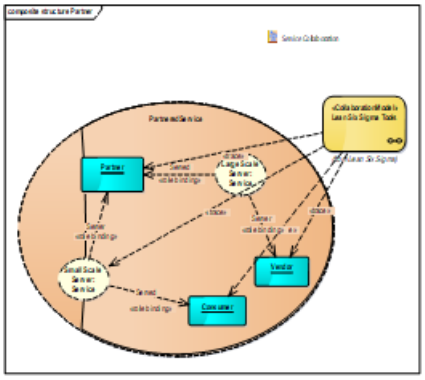


**Fig 10.** Objects Class Model with Lean Six Sigma tools optimization [11].

TOGAF framework interfaces an up-to-date references in which optimization guided, systems migrated and changes implemented. The Enterprise and Supply Chain need to model the activities carried out by entities running operations, involve departments or divisions of a larger entity as Enterprise defines and describes SC activity, by

focusing on inputs, outputs, goals and events driving the operation. This model is subject to an optimization, innovation and improvement changes as required to remain on optimum performance. Optimum Process Model captures the next changes required using EA developer such as TOGAF.

The analysis Process in Fig. 11, shows how LSS tools utilize Enterprise Architect features to continuously monitor, suggest and implement the SC required change. The process uses Architecture goals as reference entry and trace feature, call LSS tools to handle the relevant data and use control feature to implement the change. EA allows standard UML elements to be displayed with alternatives. This is useful to communicate concepts to clients or non-technical staff, in forms of; Resources overview, Actors, Deployment Model or Stakeholders reports.

## 5. DISCUSSION

The Architect is a proven powerful tool to express supply chain interrelations, activities, capture weaknesses and visually depicts relations and testing required by extended supply chain but connected sub-entities, components and Actors.

It models Enterprise activities as definitions mention, as part of normal activity, focusing on inputs, outputs, goals and key events that drive operations. Figure 12, shows optimization tools, innovation loops and improvement changes required to capture the up-coming changes required. The model uses "LSS" tools utility and Enterprise Architect's features to continuously monitor, suggest and deploy "SC" required traces and changes, making a record of activities in a log database.

It uses aligned Architecture–Strategy goals as a reference to "trace" features and timely call "LSS" tools to handle the relevant data and use Control feature to implement the change [13]. "EA" UML elements displayed with alternative images is useful for communicating concepts to beneficiaries including non-technical staff. This encourages using Architects in data-oriented enterprises and empirical frameworks convergence. Enterprise Architects needs resilient and skillful environment for Optimum Target Process Model for high level benefit to express hundreds of variables impact on supply chain performance.

It uses aligned Architecture–Strategy goals as a reference to "trace" features and timely call "LSS" tools to handle the relevant data and use Control feature to implement the change [13]. "EA" UML elements displayed with alternative images is useful for communicating concepts to beneficiaries including non-technical staff. This encourages using Architects in data-oriented enterprises and empirical frameworks convergence. Enterprise Architects needs resilient and skillful environment for Optimum Target Process Model for high level benefit to express hundreds of variables impact on supply chain performance.

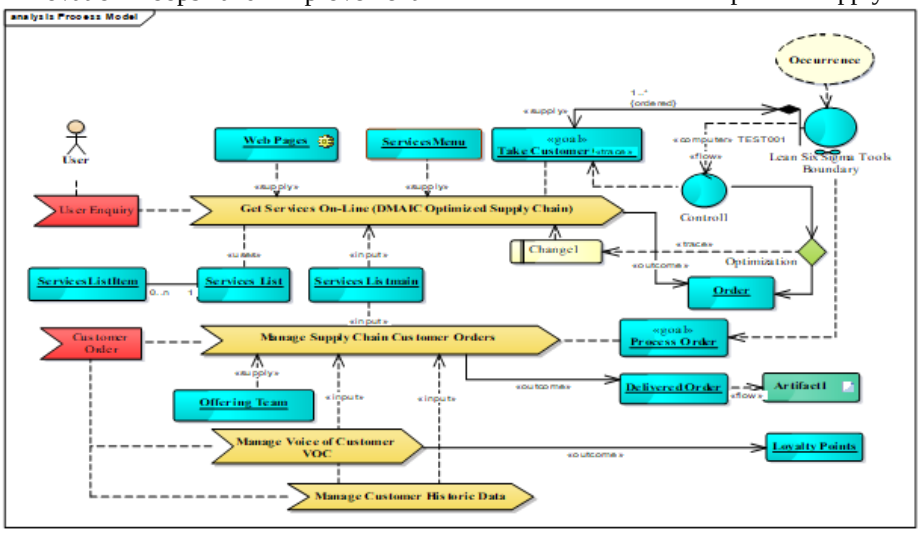


**Fig 11. Lean Six Sigma Optimized SC Process Model.**

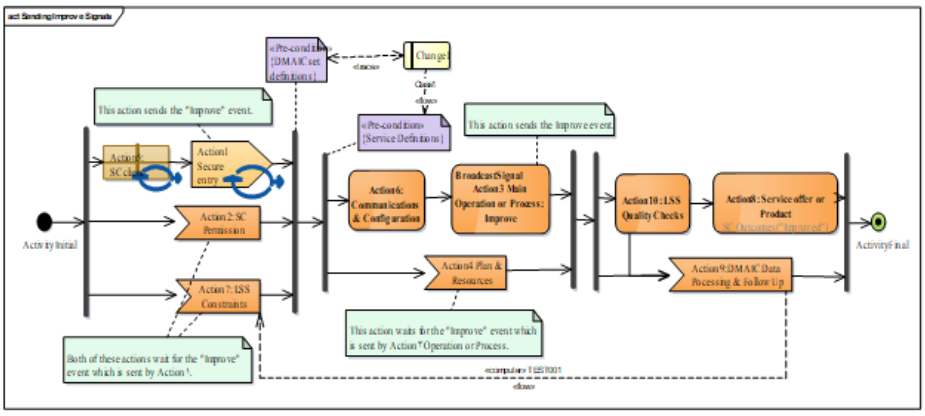


**Fig 12.** Model layers and activity with commenting feature.

## REFERENCES


[1] P. Flynn, "P. Flynn, "https://patrickflynn.info," Simply Kaizen PTY ltd. ABN 59 633 947 805, 2021.," 2021. [Online]. Available: https://patrickflynn.info/what-is-kaizen/..

[2] D. JA., The power of linking analysis and TOGAF® to achieve IT results, 2017.

[3] T. P. James Marsh and V. R. Gamini Lanarolle, "Lean six sigma: Exploring future potential and challenges," Sheffield Hallam University, UK. University of Moratuwa, Sri-Lanka., Moratuwa, 2020 .

[4] B. C. Ladd, "Supply Chain Management and Lean Six Sigma in a Retail Environment .," North Dakota State University, North Dakota, 2013.

[5] G. Sparks, *SPARX Systems Enterprise Architect V. 11.0 of 2014*, Sparx systems, 2014.

[6] S. Furterer, "A Framework Roadmap For Implementing Lean Six Sigma In Local Governmental Entities. Electronic Theses and Dissertations, 2004-2019.," 2019. [Online]. Available: https://stars.library.ucf.edu/etd/91..

[7] J. A. DiIanni, "The power of linking analysis and TOGAF® to achieve IT results.," *White Paper. https://www.jdiianni.com/.*, 2017.

[8] X. C. Lianying Zhanga, "Role of lean tools in supporting knowledge creation and performance in lean construction.," in *International Conference on Sustainable Design, Engineering and Construction*, College of Management and Economics, Tianjin University, 92 Weijin Road, Nankai District, 300072, Tianjin, P.R. China, 2016 .

[9] R. B. S.-F. et.al, "Sustainable Supply Chain Management—A Literature Review on Emerging Economies Sustainability; www.mdpi.com/journal/," *sustainability*, vol. 12, no. doi:10.3390/su12176972, 2020.

[10] S. A. J. A. R. &. D. J. A. Vinodh, "Integration of continuous improvement strategies with Industry 4.0: A systematic review and agenda for further research. The TQM Journal, 33(2), 441–472.," 2020. [Online]. Available: https://doi.org/10.1108/TQM-07-2020-015.

[11] N. M. Elmusrati, "A perception of LSS and TOGAF ADM Optimized Enterprise Supply Chain Cloud services," *Connecting Asia*, 7 2022.

[12] R. B. C.-S. S. E. O.-B. S. &. R.-B. M. E. Sánchez-Flores, "Sustainable Supply Chain Management—A Literature Review on Emerging Economies.," Sustainability, 12(17), 6972. , 2020. [Online]. Available: https://doi.org/10.3390/su12176972.. [Accessed 2020].

[13] M.-H. &. W. C.-T. Liao, "Using Enterprise Architecture to Integrate Lean Manufacturing, Digitalization, and Sustainability: A Lean Enterprise Case Study in the Chemical Industry. Sustainability, 13(9), 4851.," 2021. [Online]. Available: https://doi.org/10.3390/su13094851..