

A CASE STUDY: DEVELOPING AN ARCHITECTURAL DESIGN DESCRIPTION FOR THE APPLICATION VIEWPOINT

Annette Lerine Steenkamp
College of Management
Lawrence Technological University
steenkamp@ltu.edu

Kamal Kakish
COBAM, Inc.
kamal@cobaminc.com

Fadi El-Khoury
College of Management
Lawrence Technological University
fkhoury@euclidsoft.com

Andrew Makar
College of Management
Lawrence Technological University
andy@amakar.com

Abstract

This paper presents a case study on the experience of a technical architecture team, the AAV Team, while developing an architectural design description (ADD) for an IT system focused on the application viewpoint of the technical architecture. The project charter and requirements for the assignments of the AAVT Team project are based on a real-world initiative in a global automotive manufacturing company to upgrade its existing Global Purchasing System and deploy a new Global Contract Module. The purpose of the upgrade is to provide all purchasing system users with the required minimum hardware and software prior to implementing the new release of the Global Purchasing System with the ultimate goal to ensure that current users have successful access to the new Global Contract feature and related information. The team assignments performed by the AAV Team formed part of the requirements of a course on IT Systems Architecture in a Doctorate of Management in Information Technology. The AAV Team conducted the architectural project concurrently with four other team projects that focused on the information and business systems, communications engineering, quality of service, and enterprise security viewpoints, respectively. The case study outlines the approach followed by the AAV Team to develop the ADD, and includes the principles that guided the project, the architecture framework, process model and methodology. The paper concludes with a discussion of the project findings and the value of an industry-sponsored learning experience.

Keywords: case study, application viewpoint, architecture design description, architecture framework, architecture process model

1. INTRODUCTION

This case study reports on the application of an architectural approach, known as the S-K Approach (Steenkamp and Kakish, 2004), to develop the architecture design description (ADD) with special reference to the application viewpoint. The application viewpoint contains views of concern to the stakeholders as defined in the ADD, in terms of the major types of applications needed to

manage the data and support business processes (Spewak, 1992). This architectural assignment is formed as part of the course requirements in a Doctorate of Management in Information Technology (DMIT). The course on information technology (IT) system architecture focused on the technical IT architecture, i.e. the software systems, hardware and infrastructure components of IT systems. Due to the complexity of such systems several viewpoints are typically

taken when designing the technical architecture (Boar, 1999; The Open Group, 2000; Burton et al., 2005; and Andrade et al, 2004), and have been proposed in the IEEE1471 Recommended Practice for Architectural Description (IEEE, 2000). Various methodologies, techniques and notations are promoted to develop architectural models representing the architectural viewpoints (Monheit and Tsafirir, 1990; Rehtin, 1997; Boar, 1999; Perks and Beveridge, 2003; Steenkamp and Kakish, 2004), and a range of tools are used in practice (Microsoft, 2004; Proforma, 2004; Popkin Software, 2004; IBM, 2004). The need for an architectural framework providing the organizational context for developing a technical architecture has been advanced by several authors (Zachman, 1987; Orlikowski and Gash, 1994; Cook, 1996; The Open Group, 2003; Frankel, 2003; Schekkerman, 2005; OMG, 2004; Steen et al. 2004). The organizational context was captured in formulating the educational goal of the course, namely to provide a comprehensive perspective of the value of a well-designed technical IT architecture in the enterprise. Objectives are to study the viewpoints of the IT architecture expressed in terms of architecture frameworks. Additional objectives were to review the international standards pertinent to IT system architecture, to explore best practices used in industry, and to apply these in a team project of practical relevance. Individual assignments required students to perform analysis and design at various levels of abstraction using appropriate methods, techniques and supporting tools. Building on this foundation the project teams analyzed, designed and implemented a prototype (where appropriate) for the respective viewpoints, and to document the ADD. The intended outcomes were to impart competencies to students to serve as an IT system architect, with the ability to lead and manage architectural projects that are aligned with the IT strategy.

The team project for the Application Architecture Viewpoint (AAV) Team was based on a real-world project in a global automotive manufacturer. The context of the AAV Team project and the project charter is given in Section 2 and Section 3, respectively. The architecture approach followed by all project teams in the course is summarized in Section 4, focusing on the application viewpoint.

The architecture framework, as instantiated for the application viewpoint, the architecture process model and methodology adopted by the AAV Team are also addressed here. The steps of the methodology followed in the architecture stage of the architecture process model are presented, complemented with a selection of the deliverables. The paper concludes with a discussion of the project experience in Section 5.

2. CONTEXT OF THE AAV TEAM PROJECT

The Heavy Vehicle Division of an automotive manufacturer seeks to upgrade its existing Global Purchasing System and deploy a new Global Contract Module. The existing applications of the Global Purchasing System used by Global Purchasing reside in the legacy client/server architecture, with dependencies on client software layers. All clients of the current system must be upgraded to a pre-specified release of the operating system in order to use the functionality of the new Global Contract Module of the Global Purchasing System. There are 170 existing users and 246 future users at distributed locations (assembly plants and office locations) who will need access to the new system. The purpose of the project is to provide all purchasing system users with the required minimum hardware and software prior to implementing the new release of the Global Purchasing System, with the ultimate goal to ensure that current users have successful access to the new Global Contract feature and related information. The team project requires the AAV Team to define the application architecture for a common intranet portal which provides for single sign-on and user-specific access to relevant information. The team project should utilize web-based products and tools to develop the Global Purchasing System that would increase linkages with employees, purchasing, customers and tens of thousands of suppliers. The team project provides an opportunity to apply an architecture approach and supporting theoretical principles to a practical IT problem in the automotive industry.

3. PROJECT CHARTER

The AAV Team is tasked to develop an ADD from the application architecture viewpoint, within the context of the enterprise architec-

ture for the Global Purchasing System (refer Section 2), incorporating a Global Contract Module with a Global Contract Feature, based on the requirements of a real-world project in the automotive industry. The scope is limited to the project case requirements provided to the AAV Team, although collaboration with the other project teams is required throughout the project to obtain information about the other complimentary viewpoints. The project scope includes analyzing the automotive manufacturing environment, the Global Purchasing applications, the user base, the infrastructure supporting the legacy purchasing system, and all other constraints imposed by corporate IT strategies and policies. The ADD will include the baseline application architecture, supporting viewpoints, and new architectural models that will achieve the desired outcomes and

concerns specified in the project requirements.

4. ARCHITECTURE APPROACH

The architecture approach followed in the course includes consideration of the relevant overarching and viewpoint-specific principles, architecture frameworks, architecture process models and methodologies, along with an appropriate team work approach. The AAV Team adopted the S-K Architecture Approach (Steenkamp and Kakish, 2004) to model the views of the application viewpoint. Figure 1 depicts the model of the S-K Architecture Approach instantiated for the AAV Team and the Global Purchasing System.

Figure 2 shows the instantiation of the IEEE1471 meta model for the AAV Team project.

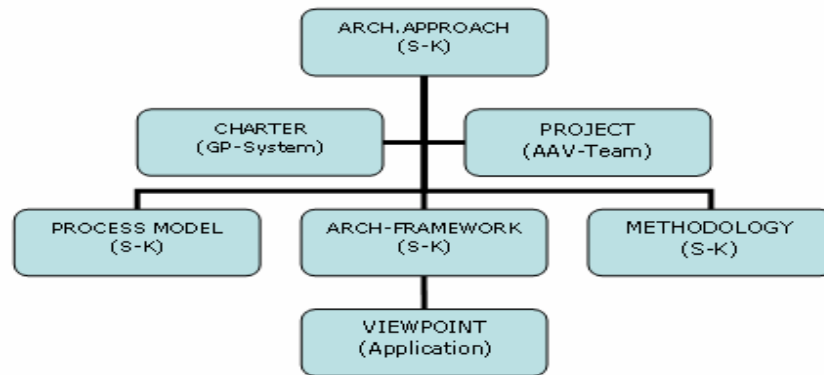


Figure 1. The S-K Architecture Approach adopted for the AAV Team Project

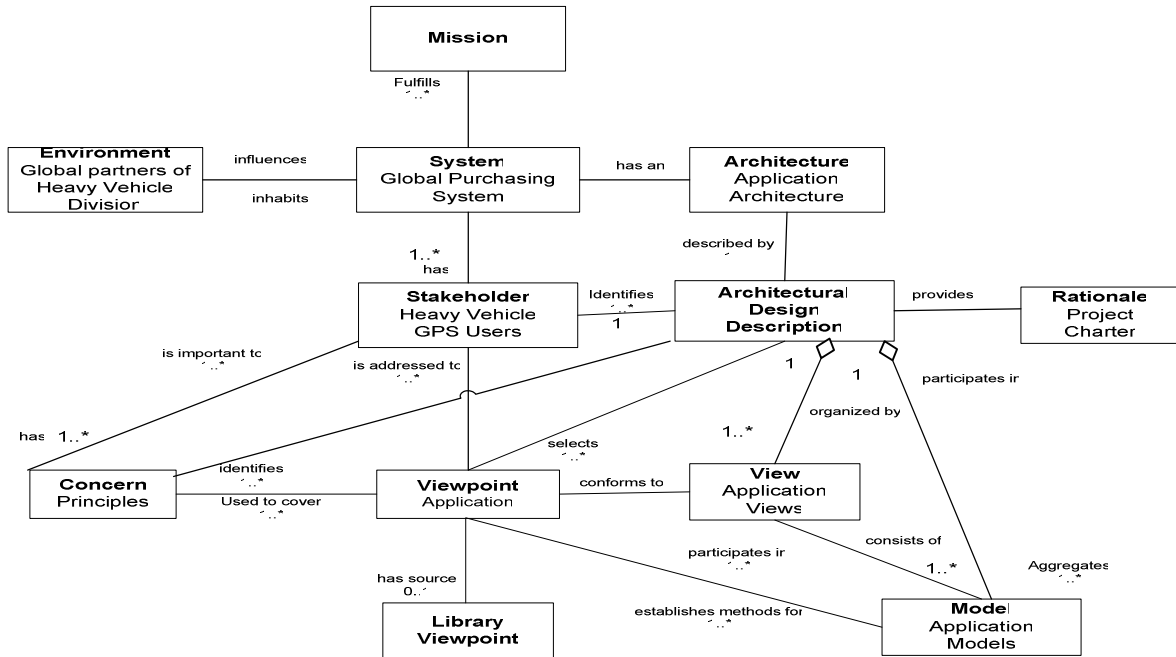


Figure 2. Instantiated IEEE1471 Meta Model of Architectural Description

The key elements of the S-K Approach are summarized below.

Principles

The application architecture principles establish the basis for a set of rules and behaviors to be observed by the architects and application developers within the context of the enterprise architecture (EA). These are principles that govern the EA processes and the implementation of the architecture and af

fect development, maintenance, and use of the EA. The chief architect, in collaboration with the CIO and business managers, identifies the architectural principles that underpin the corporate IT vision and strategic plans. The architectural principles represent fundamental requirements and practices believed to be in the best interests of the organization and should be refined to meet corporate business and IT objectives. Every stage of the architecture process model is supported by the actions necessitated by the architecture principles. The principles are given in the S-K Framework in Appendix 4.

Architecture Framework

An architecture framework provides guidance in terms of architectural attributes when modeling the architectural design description. It establishes terms and concepts pertaining to the structure, content and use of architectural models. The S-K Architecture Framework in Appendix 4 summarizes the attributes for the enterprise and application architecture viewpoints for the new Global Purchasing System. The architect considers the values for the attributes when designing architectural models for the viewpoint, namely:

- The purpose of a view: whether it aids by Informing, for Deciding, or for Designing.
- The content of a view: characterized by the three abstraction levels where it resides: Details, Coherence, and Overview.
- The layer: refers to the layer where the view fits: Business, Application, and Technology.
- An aspect: what the view depicts: Structure, Behavior, and Information.
- Viewpoint language: the Modeling Notation or Representation Scheme used.

- Standards: the best practices adopted when modeling a view.
- A tool: the automated capability used to model a view.

Depending on the stakeholder the ADD for the application architecture provides multiple views of the new Global Purchasing System. A range of standards, notations and tools were used to develop the models listed under the model portfolio in Appendix 4 that are relevant to the various stakeholders.

Architecture process model

As is the case with system and software process models an IT architecture process model structures the architectural processes into interrelated life cycle stages facilitating the managerial and technical tasks of architects and developers who plan, manage, evaluate, develop and maintain the IT architecture. The architecture

process model followed by the AAV Team is shown in Figure 3. The AAV Team focused on functional analysis, and logical and physical modeling of views for the application architecture viewpoint. The AAV Team (and the other teams) collaborated with the team responsible for the Information/Business System Architecture viewpoints, which focused on the Architecture Stage as highlighted in Figure 3. The Rational Unified Process (RUP) Model was adopted to structure the Build IT Architecture Phase in Figure 3. The RUP 4+1 views, shown in Figure 4, guided the AAV Team to develop models of the logical view (the functionality as seen by the end user), the use-case view (the specification of what is needed in the architecture in the abstract), the process view (as seen by system integrators), the implementation view (the system as seen by the programmers), and the deployment view (seen by system engineering to be used for installation and delivery).

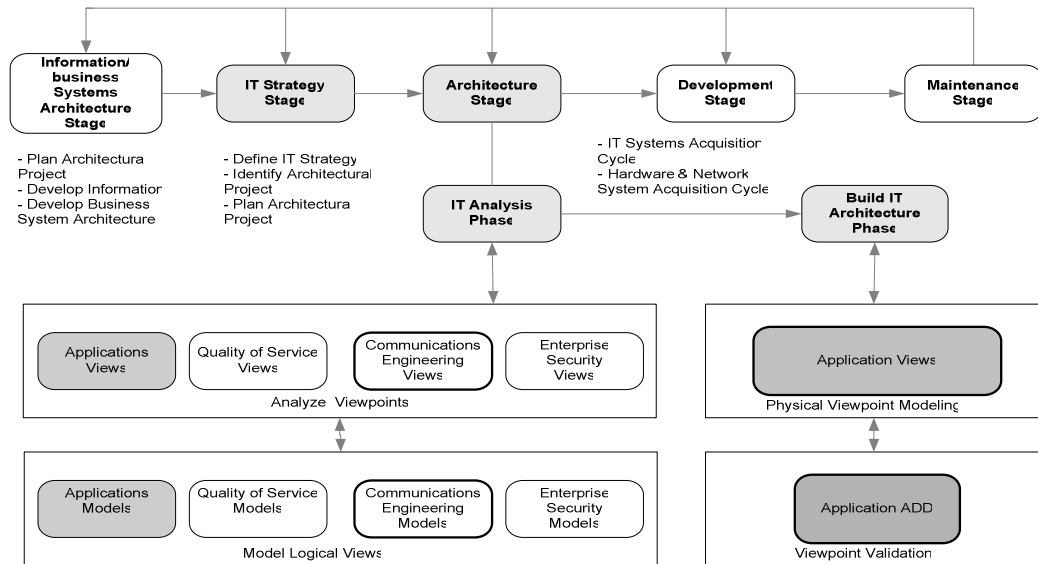


Figure 3. Architecture Process Model

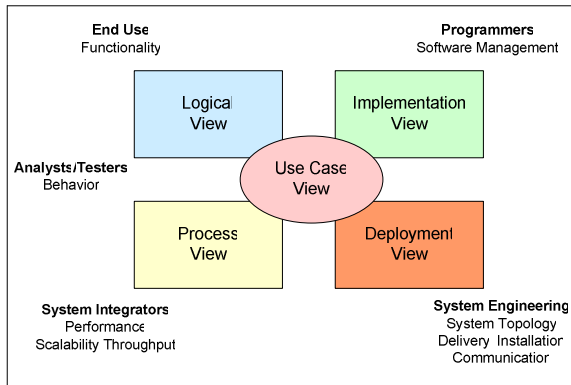


Figure 4. RUP 4+1 Views (from (Kruchten 2000))

Architecture methodology

An architecture methodology prescribes the techniques and steps to be followed to produce the models of the architecture. Table 1 shows the methodology adopted by the AAV Team to develop the ADD for the application viewpoint.

Table 1. Methodology for Architecture Stage

Architecture Stage/Phase	Steps	Application Viewpoint Models
IT Architecture Stage Planning Phase	<ol style="list-style-type: none"> 1. Review project charter 2. Review viewpoint guidelines 3. Adopt architectural framework 4. Prepare Architectural Project Plan 5. Determine Framework/review principles within context of chosen architectural framework for application viewpoint. 	Architectural Project Plan Framework Process Models Overarching and application principles
Architecture Stage IT Analysis Phase	6. Perform functional analysis; analyze project requirements, interpret project charter, and develop viewpoint definitions	Viewpoint definitions: Function tables
	7. Gather information requirements from Team and project sponsors relevant to viewpoint	Viewpoint requirements: Use-cases and scenarios
	8. Choose representation schemes, modeling notations and CASE tool.	Representation schemes/ notations
	9. Adopt documentation method and template.	Documentation method and project folder format
	10. Adopt method for alignment with enterprise and IT strategies.	Performance Matrix and alignment method
	11. Model logical views and document using CASE tool.	System component diagram, UML sequence diagrams, UML class diagram
Target Architecture Stage Build IT Architecture Phase	12. Develop Draft Architectural Design Description	Draft Architectural Design Description (ADD)
	13. Determine draft architectural design/scope; model physical views and document using CASE tool.	Final ADD document Project binder
	14. Present the team project	PowerPoint Presentation

Examples of the RUP 4+1 views, developed by the AAV Team, are presented in the sections below.

Logical view. This view is captured using a function table as in Table 2, providing an example of part of the functionality of the new Global Contract Module.

Table 2 Function Table

Function #	Function	Function Description
F01.01000	Login	All users need to sign on to use the GPS system.
F01.02000	Logout	When finished with the purchasing business process, user clicks "log-out" to sign out.
F01.03000	Requirement Cycle	Follow corporate statement of work and cost estimates processes and maintain relevant data
F01.04000	Requisition Cycle	Access module to perform following functions: <ul style="list-style-type: none"> • Evaluate specifications • Confirm sources • Review past performance of sources • Produce solicitation package (bid documents, qualified vendor, proposal evaluation)
F01.05000	Solicitation Cycle	Access module to support solicitation process and create: <ul style="list-style-type: none"> • Request for Quote • Request for Information • Request for Proposal
F01.06000	Award Cycle	<ul style="list-style-type: none"> • Support processes to award contract and store digital contract information
F01.07000	Global Contract Administration	Access global contract function module

Use Case View: This view captures business requirements of the system at a high level. It depicts the collaboration between the external actors (the purchasing manager and vendor) and

the individual use cases that comprise the system. Figure 5 shows a use case model for the system.

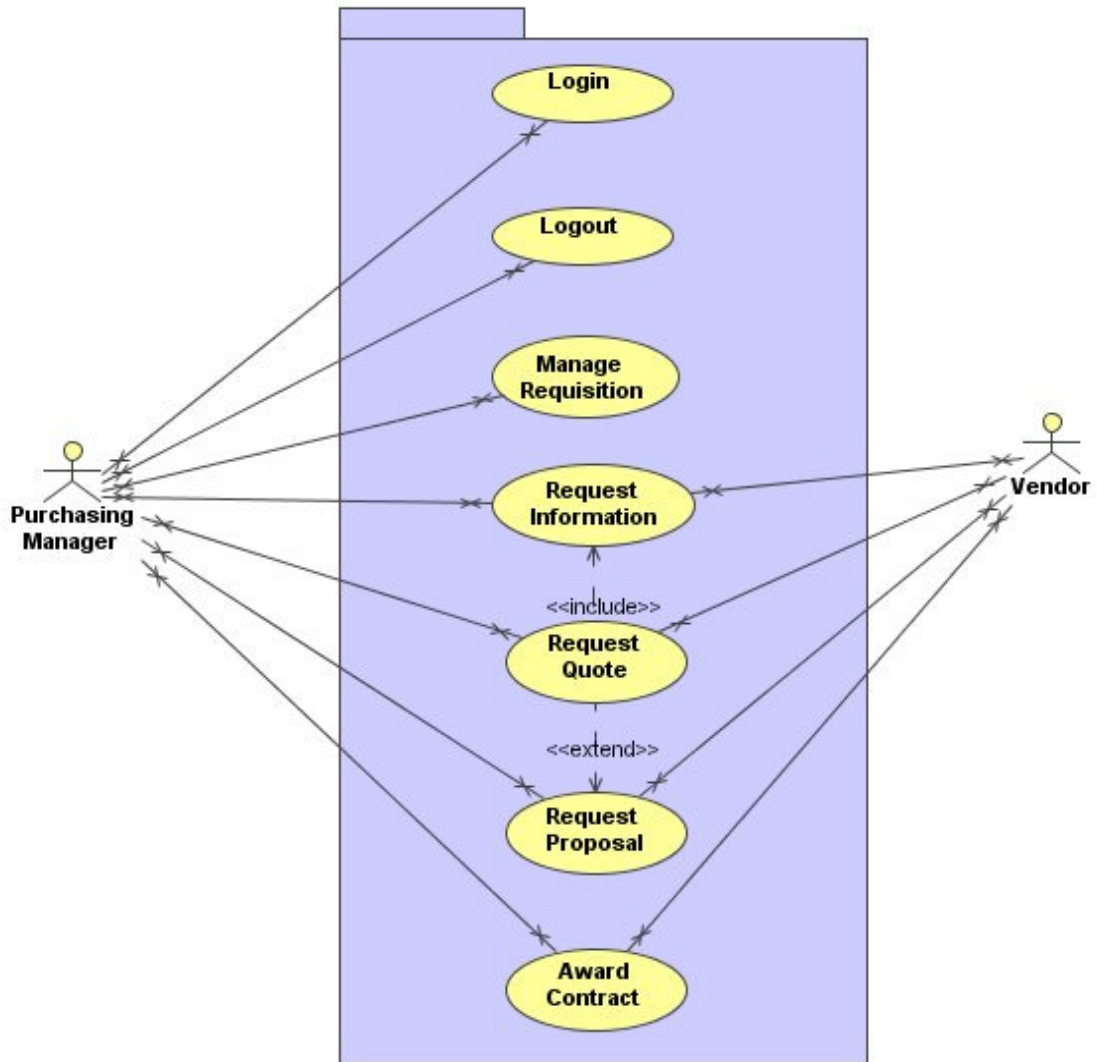


Figure 5. Use Case Model

Process View: The Process view for the application is captured externally and internally. The external behavior is the workflow configuration for the purchase requisitions process of the pur

chasing process, shown in Appendix 2 . The internal behavior is the application object process and messaging represented by the sequence diagram shown in Figure 6.

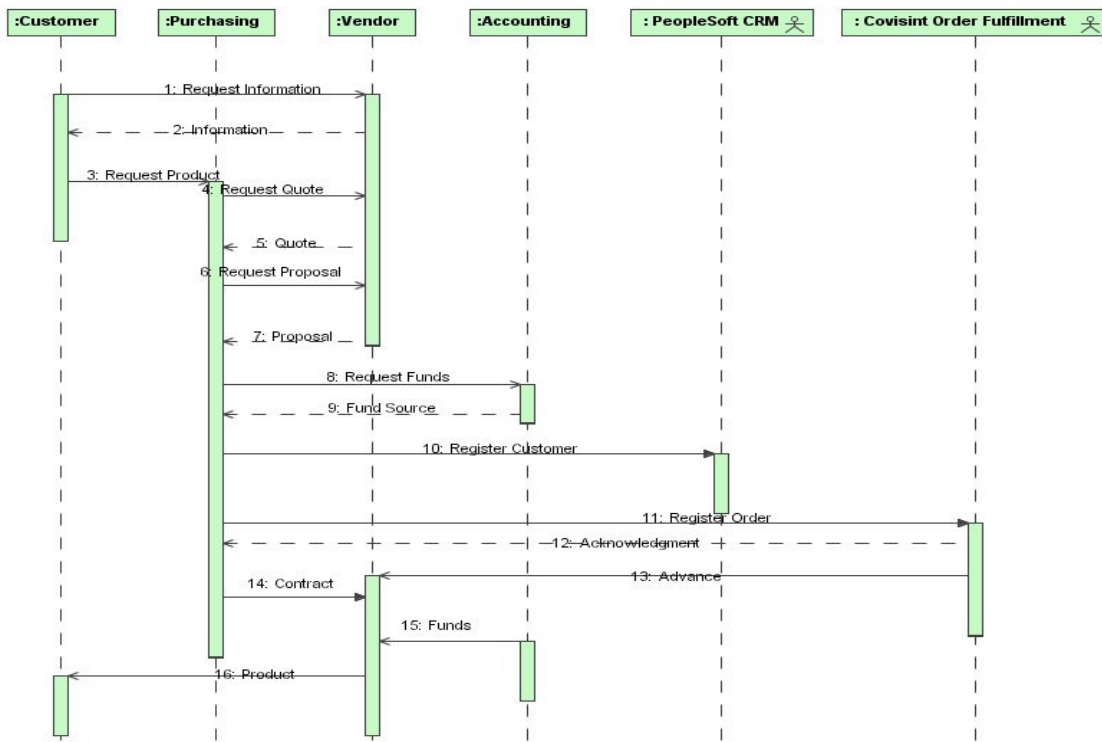


Figure 6. Sequence Diagram of the Purchasing Process

Implementation view: The Implementation view describes the organization of static software modules in the development environment in terms of layering and packaging; and in terms of the structural building blocks of the system objects. By separating the application into layers (refer Figure 7), the business logic would be captured at the application layer. Each of these layers is meant to be an independent entity that does not include any repetitions, and all system functionality is encapsulated in layers.

A lower level from the application layers in the implementation view is where packages are situated. Packages help in organizing applications as well as model elements into groups, making the implementation simpler, easier, and more organized. File folders depict packages and are most common on class diagrams because these models have a tendency to grow. Figure 8 shows a package diagram for the system where the code is distributed into four packages.

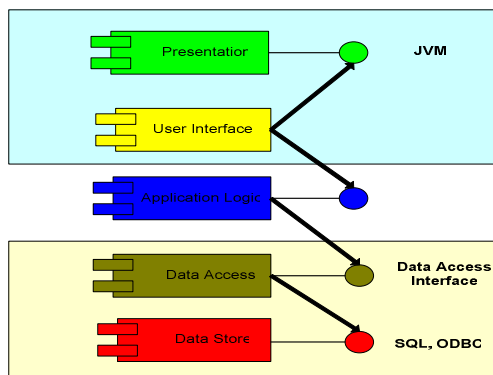


Figure 7. Application Layers

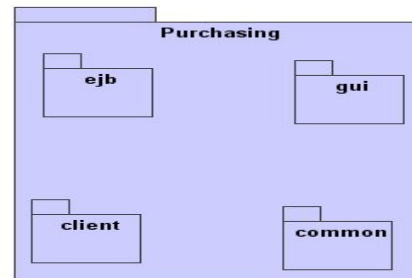


Figure 8. System Package Diagram

The class diagram shown in Figure 9 represents the structure for the purchasing system. In the process view, the sequence diagram (refer Figure 6) showed the internal

behavior of the system objects, whereas in this view, the class diagram represents the

structural aspect of the system objects and their relation to each other.

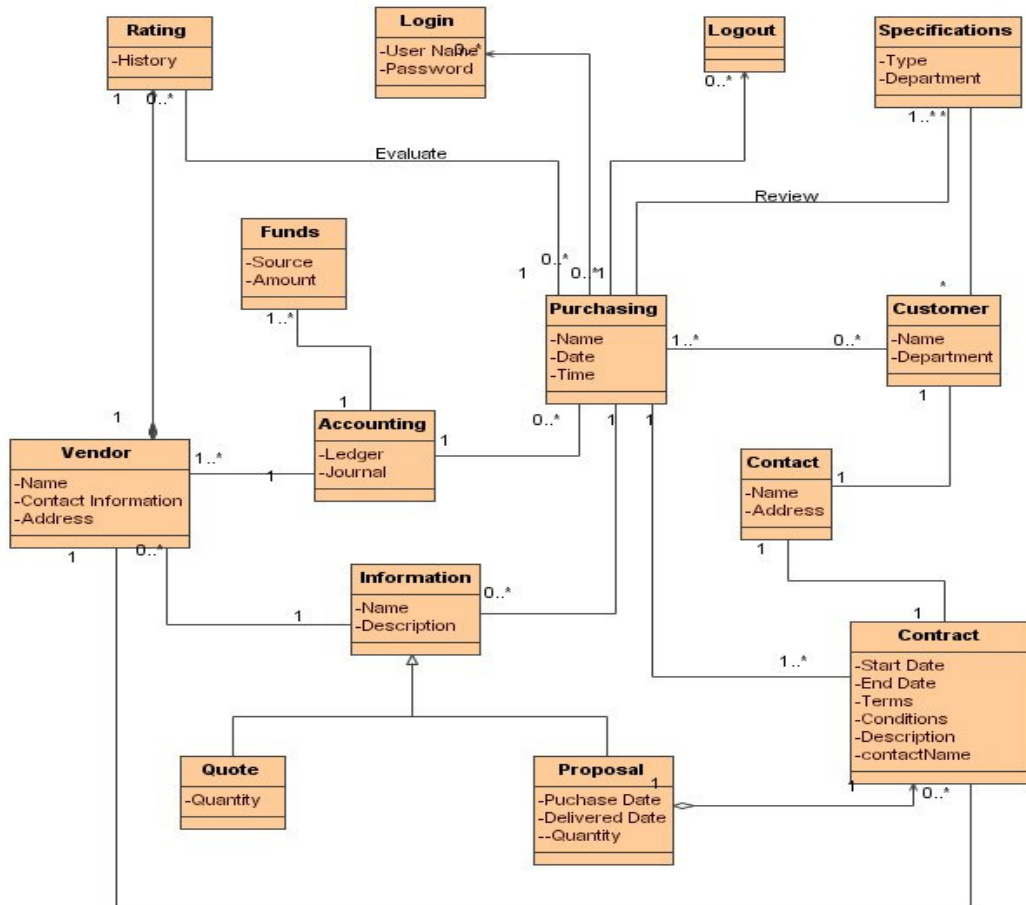


Figure 9. System Class diagram

Deployment View: The Deployment view describes all activities performed by the system engineer in deploying the system to all customers. The proposed deployment environment includes the hardware required to run the system, shown in Figure 10. Appendix 3 describes the conceptual infrastructure architecture for the system.

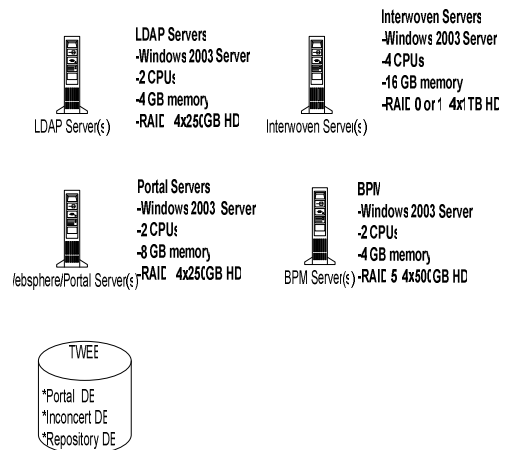


Figure 10. Deployment Hardware Requirement

Author email addresses are encouraged to appear as part of the authorship information, together with the author name. However, at your discretion you can include them as endnotes. Do NOT use footnotes.

5. DISCUSSION

The S-K approach provided the AAV Team with an analytical framework and systematic methodology to develop the models of the ADD for the application viewpoint. The AAV Team summarized their findings in a case study (used to author this paper), documenting the project experience, and created a project binder to store the project deliverables for future reference. At the end of the project, the AAV Team debriefed and presented their finding to the project sponsors and the other teams. Due to the time constraint the AAV Team did not build a prototype of the enhanced Global Purchasing System.

This case study describes the AAV Team project experience to develop the application viewpoint ADD for a real-world problem in the Heavy Vehicle Division of a global automotive manufacturer. The project provided an academic opportunity to apply the S-K architecture approach to an existing Global Purchasing System, and taught the AAV Team how to analyze the project charter and requirements for the application viewpoint using an analytical architecture framework. This framework provided a greater appreciation for the value of a common language, terminology, principles and role of modeling tools to create an enterprise architecture, which responds to business needs and the strategic vision of a large enterprise. Like the other project teams the AAV Team followed a systematic methodology to model the views of the ADD. The team also used a project plan to manage the teamwork, met bi-weekly to review deliverables and plan the next steps. They also had regular face-to-face meetings with the project sponsors to review progress and clarify issues that emerged during project sessions. Collaboration with the other teams also gave the AAV Team valuable insight into the complexities of the enterprise architecture. Reviewing models for views of other viewpoints en-

hanced understanding of the needs of the stakeholders. Teamwork was supported by the Blackboard Learning System and use of a number of automated tools for modeling the architecture. Skype was used for Internet-based communication between team members.

6. ACKNOWLEDGEMENTS

The authors wish to express their appreciation to the sponsors of the AAV Team project for providing an opportunity to apply IT architectural theory to practice and collaborating on this educational project. The AAV Team also wishes to thank the other project teams for the rewarding collaboration during the project.

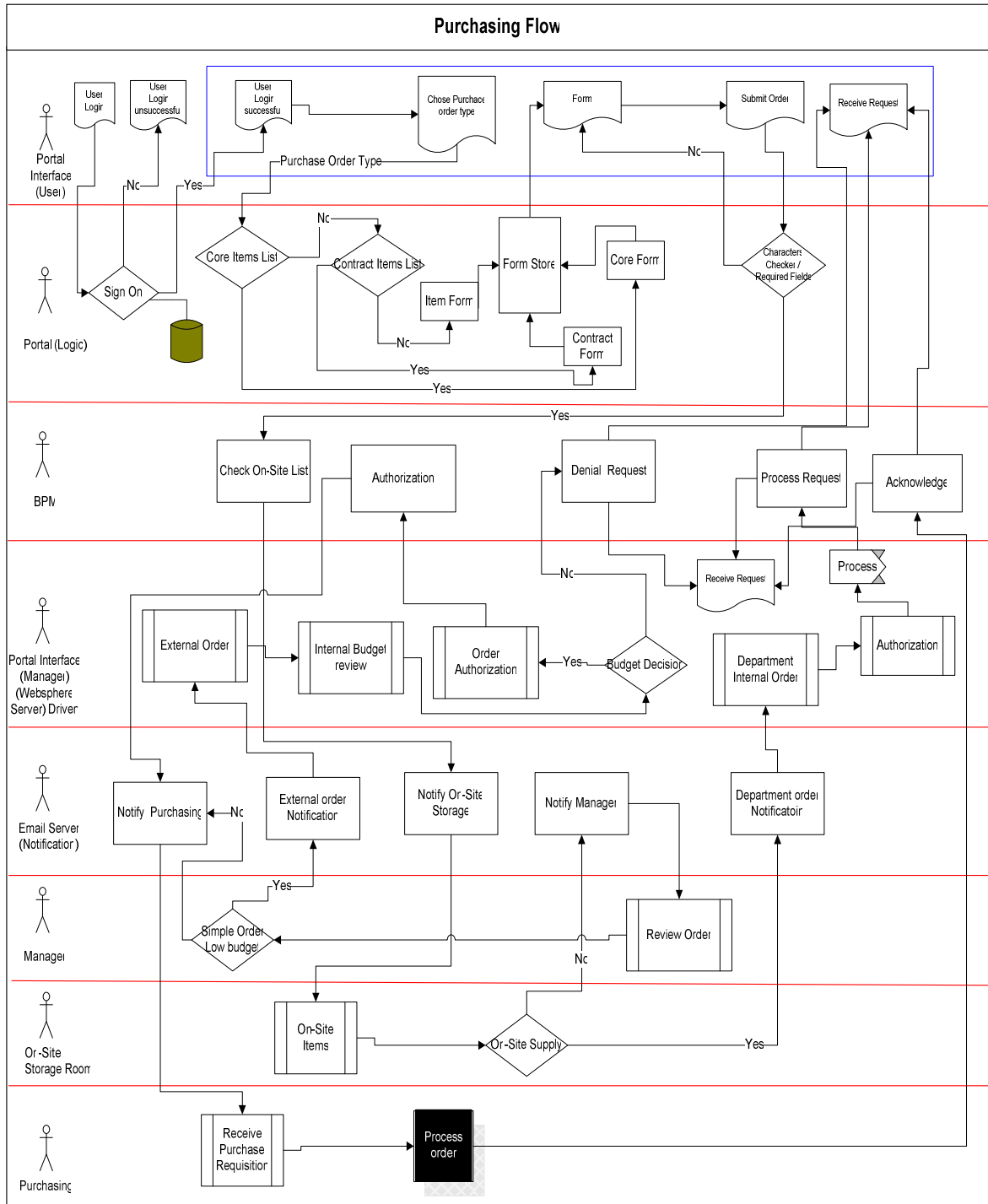
7. REFERENCES

- Andrade, J., Ares, J. Garcia, R. and Pazos, J. (2004) "A methodological framework for viewpoint-oriented conceptual modeling," IEEE Trans. on Software Engineering, 30, 5.
- Boar, B.H. (1999) Constructing Blueprints for Enterprise IT Architectures. John Wiley & Sons.
- Burton, B, Phifer, G., Valdes, R. and Lapkin, R. (2005) "Use Gartner's Framework to Assess Interoperability," Gartner Research Paper.
- Cook, M. (1996) Building Enterprise Information Architecture. Prentice-Hall.
- Frankel, D. S. (2003) Model Driven Architecture: Applying MDA to Enterprise Computing. Wiley & Sons.
- IBM Corporation (2006) Retrieved April 2006 from <http://publib.boulder.ibm.com/series/v5r1/ic2924/index.htm?info/rzakh/rzakh000.htm>.
- IBM Corporation (2006). Retrieved April 2006 from <http://www.ibm.com>
- IBM Corporation (2006) Retrieved April 2006 from <http://publib.boulder.ibm.com/series/v5r1/ic2924/index.htm?info/rzakh/rzakh000.htm>.
- IEEE (1998) Std 1058-1998 IEEE Standard for Software Project Management Plans.

- IEEE (2000) IEEE Std 1471-2000 IEEE Recommended Practice for Architectural Description of Software-Intensive Systems. New York: Institute of Electrical and Electronics Engineers.
- Kruchten, P. (2000) The Rational Unified Process, An Introduction. Upper Saddle River, Addison-Wesley.
- Monheit, M. and Tsafir, A. (1990) "Information Systems Architecture: a consulting methodology," In CompEuro'90. Proc. of the IEEE Int. Conf. on Computer Systems and Software Engineering.
- OMG (2004) Model-Driven Architecture Standards. Retrieved April 2006 from <http://www.omg.org>.
- Orlikowski, W.J. and Gash, D.C. (1994) "Technological Frames: Making sense of Information Technology in Organizations," ACM Transactions on Information Systems, 12, 2.
- Perks, C., & Beveridge, T. (2003) Guide to Enterprise IT Architecture. New York: Springer-Verlag, New York, USA.
- Phifer et al. (2003) Portals and Web Services will boost business value in 2004. Gartner Whitepapers, Gartner. Retrieved April 2006 from http://mediaproducts.gartner.com/gc/w/ebletter/bea_reprints/article1/article1.html
- Popkin Software (2004) Retrieved April 2006 from http://www.popkin.com/products/system_architect.htm.
- Proforma Corporation (2004) ProVision Workbench Whitepaper. Retrieved April 2006 from <http://www.proformacorp.com/>.
- Rechtin, E. (1997) "Teaching systems architecting: Science and Art," IEEE Spectrum, 29, 10.
- Schekkerman, J. (2005). How to Survive in the Jungle of Enterprise Architecture Frameworks – Creating or Choosing an Enterprise Architecture Framework, 3rd Edition, Trafford Publishing, Victoria, B.C., Canada.
- Spewak, S.H. (1992) Enterprise Architecture Planning, New York: Wiley & Sons.
- Steen, M. W. A., Akehurst, D.H., ter Doest, H.W.L. and Lankhorst, M.M. (2004) "Supporting Viewpoint-Oriented Enterprise Architecture," in Proc. of the 8th IEEE Intl Enterprise Distributed Object Computing Conf. (EDOC 2004), IEEE.
- Steenkamp, A. L. and Kakish, K. (2004) "An Approach to Developing Information Technology Architecture," in Proc. of the Intl Conference on Informatics Education Research (ICIER2004), Washington, DC.
- Steenkamp, A.L. (2005) DMIT Research Notes Series 05/03.
- The Open Group (2003) TOGAF The Open Group Architecture Framework Version 8.1 "Enterprise edition". Retrieved April 2006 from <http://www.opengroup.org/public/arch/p1/oview>.
- Wikipedia (2006) Retrieved April 2006 from <http://en.wikipedia.org/>
- Zachman, J. A. (1987) "A Framework for Information Systems Architecture", IBM Systems Journal, 3.
- Zachman, J. A. And Sowa, J.F. (1992) "Extending and Formalizing the Framework for Information Systems Architecture." IBM Systems Journal, 31, 3. Retrieved April 2006 from <http://www.infoed.com/Open/Resource.s.html>

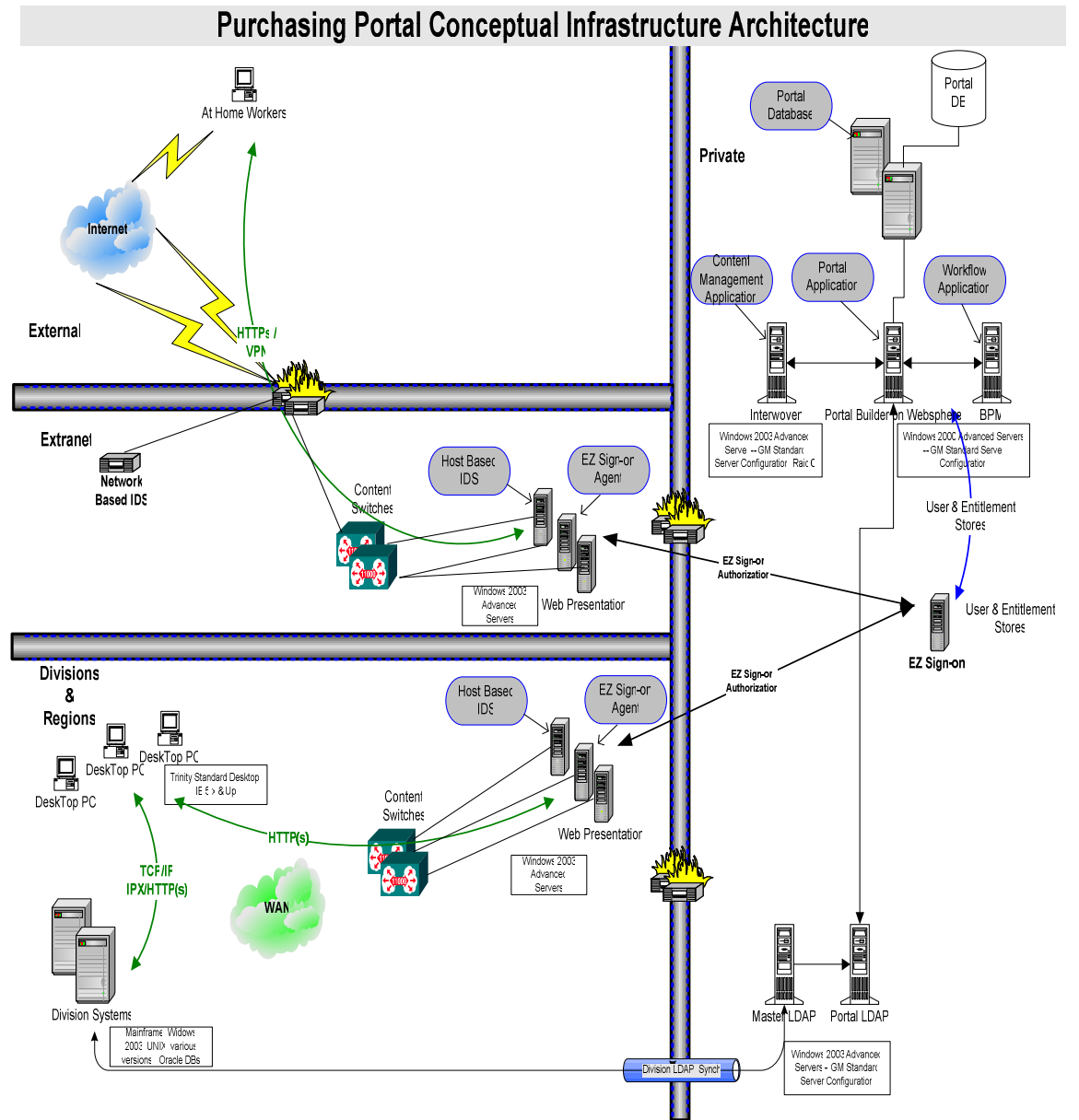
Appendix 2. Process Model

Appendix 2 shows the external process view of the flows across various actors for requisitions in the purchasing process.



Appendix 3. Conceptual Infrastructure Architecture

The model of the conceptual infrastructure architecture shown in Appendix 3 captures the conceptual view of servers, networks devices and LAN/WAN connectivity, and is part of the deployment view.



Appendix 4. S-K Architecture Framework for the AAV Team Project

The S-K architecture framework, instantiated for the application viewpoint, is shown in Appendix 4.

Stakeholder	Purpose	Concern/ Principle	Content	Layer	Aspect	Viewpoint Language	Model Portfolio	Standard	Tools
Strategist (CFO)	Deciding	Business Principles	Overview	Business	Structure		Enterprise Strategic Plan System Block Diagram	COBIT	MS Visio
Global CIO Purchasing Project Man- ager	Informing Deciding	Business/ Strategy Timeliness Efficiency	Coherence- Scope Details	Business, Tech- nology Application, Technology	Structure Behavior Information		IT Strategic Plan System Block Diagram Figure 3: S-K Arch. Proc- ess Model, Figure 4: 4+1 RUP	COBIT IEEE 12207 IEEE 1058, Pmbok, IBM RUP	MS Visio MS Project
Anal- ysts/Tester	Designing	Accessibility to contract data	Details	Application, Technology	Information	Unified Model- ing Language	Figure 5 : Business use case model for GPS. Function table	OMG UML 2.0	MagicDraw MS Word
Programmers	Designing	Desired func- tionality Reduced complexity	Details	Application	Information	Unified Model- ing Language	Figure 7: Implementation model (architecture lay- ers) Figure 8: Package dia- gram, Figure 9 : Class diagram	OMG UML 2.0	MS Visio MagicDraw
System Inte- grator	Designing	Dependency on other projects	Details	Application	Structure Information	Unified Model- ing Language	Figure: Business Process for GPS Figure 6 : Sequence dia- gram for GPS Figure 7: Implementation model	OMG UML 2.0	MS Visio MagicDraw
System Engi- neering	Designing	Enhanced func- tionality Main- tainability	Details	Application, Technology	Structure Information	Unified Model- ing Language	Figure 10: Deployment diagram	OMG UML 2.0	
End User	Informing	Ease of use	Details	Application	Information	User Training guide	Figure 5: Use case model		
Procurement	Deciding	Strategy	Details	Business	Structure Information		System Block Diagram	COBIT	MS Visio