

PM WORLD TODAY – FEATURED PAPER – JUNE 2008

Where to look when Requirements do not Exist
in today's Software projects

Prof. Hubert Vaughan

Institute of International Engineering Project management
Tsinghua University, Beijing, China

Anyone managing software development projects will agree it is extremely difficult to identify “Requirements” that enable us to define “specifications” of the solution today, thus causing continuous changes during our development cycle, delaying project delivery and resulting cost over-run.

A lot had been discussed and written about the techniques or ways of identifying requirements during the early stage of system development. It is unfortunate that we don't seem to gain from any of these ideas and experience.

Today's “information age” is very much different from the “automation age” of the past. Most projects Sponsors and Stakeholders really don't have any ideas about what the requirement will be during project initialization stage. We can never identify “Requirement” that do not exist from our customers or users.

On top of the non-existent of “Requirement”, most project managers failed to define a Project Scope that can be managed. Without a manageable scope definition, we can never manage the subsequent changes request during delivery. PMBok made it very clear the important of Scope Management, only if you can manage the project scope, you can manage the Change Requests because Project Scope bounded the requirement of our development.

Scope and Requirement in the 70

It is the early stage of our automation age, that is, changing our manual business processes into computer system that help us improves our productivities in our work environment. Instead of project scope that we know today, we call it “Terms of Reference” that bound our undertaking.

If we were given an assignment to development an Inventory Control System for the warehouse, we may receive an instruction like “Develop a system that identify and keep counts of all goods (Raw materials, semi-finished and Finished-products) that goes into the warehouse, and all goods that leave for the production floor or for delivery to customers”. The Terms of Reference gave us the boundary of our assignment that specified any business processes inside the warehouse should be automated. The System Analyst will then conduct a Fact Finding exercise, by interviewing warehouse management to identify the business processes that took place when goods come into warehouse, how and where it is stored, what triggered goods to be picked up for the production floor or for delivery to

customers. Identify how and where inventories count are recorded, updated, transferred, and removed from the record. Reviewed reports format, contents and forms used inside the warehouse operations.

Armed with all those information gained from our Fact Finding exercise, the Analyst will base on all those processes to identify data capture *requirement*, file format *requirement*, data update *requirement*, and reporting *requirement*. The Terms of Reference basically bounded the Requirement of our development work.

Please take note that Users never gave out requirements, they told the Analyst their operation processes and the Analyst has to define the Requirement after his understanding of the warehouse management operation.

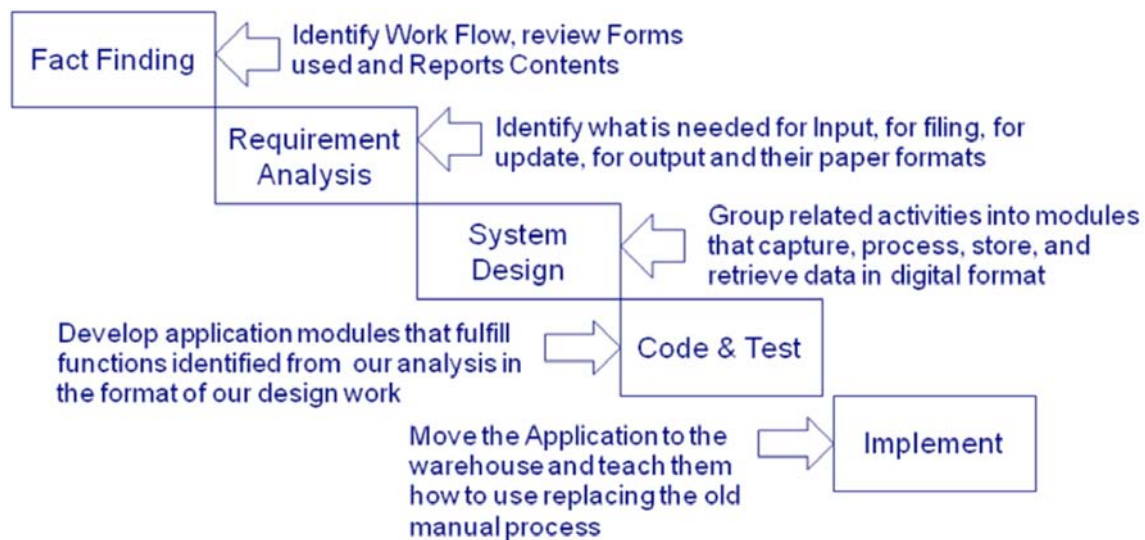


Figure 1: Development Life Cycle of the Automation Age

Once requirements are identify by the Analyst, the Designer will group similar process together to create Application Modules and system logics for development and subsequent implementation.

The User Acceptance was conducted around how the systems delivered can satisfy the Users and provides all the functions that the users performed prior automation. The Users knew the manual business process so well that they certainly can tell if any of the operation functions are missing from the computer system. This gave us the wrong idea that Users Acceptance was based on Functional Requirements and mislead our mind-set until today. In fact, the Users were testing the Functions and determine if the System can meet their operational needs. They accept the system that we delivered, not the functions. Functions were in fact part of the quality we have to deliver in our application.

Scope and Requirement of the 80

The Automation Processes of the 70s was mainly based on isolated departmental operations. From mid-80s onward, most business departments already completed their departmental automation processes. Corporate management found the inconvenience of data spread across the company in multi-department and multi-locations. Similar data in various departments were duplicated and inconsistent. Data accuracy (integrity) and timeliness was in great demand. Introduction of new technology such as remote access, Area-Network, Client/Server computing, database technology and use of personal computers provides the solution of information sharing. System Integration and Data consolidation between business departments and business locations become the next wave of our automation age.

A simple Terms of Reference can no longer define the boundary of our assignment. Therefore we used multiple Term-of-References and rename it to Statements of Work (SOW) describing what we need to deliver at the end. All SOWs formed the Project Scope of our Assignment. In order to ensure we have a well defined project boundary, we use explicit terms such as “Inclusive” and “Exclusive” in our SOWs just to ensure we can manage the Changes that may arise during our development stage.

An example of an assignment to “Develop a Sales Order Processing System” may include initial “Requirement Gathering” process that allow us to discuss with related department heads that enable us to have a clear picture of our project boundary, either building the SOWs or confirming the Inclusive and Exclusive areas of our Work statements. We may come up with something like the following SOWs.

“Develop a Sales Order Processing System with the following functionality:

- SOW-1: Connect Sales Office in City A to the Inventory Data Centre in City B.
- SOW-2: Allow Sales Office terminals in City A to access and enquire quantity of Finished-goods in City B Data Center
- SOW-3: Allow Sales Office in City A to reserve finish-goods in City B warehouse for delivery to Customers
- SOW-4: Allow Sales Office in City A to issue Shipping-orders and instruct Warehouse in City B to ship reserved goods to customer specified by Sales Office in City A
- SOW-5: Calculate shipping charge for shipping goods to Customer and allow Sales Office in Office A to produce Customer Invoice
- SOW-6: Update Inventory count at City B Data Centre when goods were packed for delivery to Customer.
- SOW-7: Transfer Invoice details to Account Receivable system in Head-Office located in City C
- SOW-8: When finished-goods count in City B Data Centre is below minimum stock

level, trigger production planning system for goods production in the Factory located at the designated city where the goods are manufactured.”

Once the above project scope was confirmed and sign-off, Systems Analysts will conduct interviews and determine how the business processes related to each of the SOW was being handled while System Engineers will determine the technical requirements for SOW-1 for connecting the Offices involved.

The handling of each SOW was similar to the way we handled Term of Reference in the 70 and each SOW is known as Sub-project (that is how sub-project was first called within a project). Understanding the business processes of each sub-project will allow our System Analysts to determine the System Requirement of the “Sales Order Processing Systems”.

How we can identify the SOWs that formed project Scope depends how well we discover all business processes related to the so called “Sales Order Processing” functions. As long as the Project Sponsor and Key Stakeholders agreed on the SOWs, anything that are not specified by SOW is considered out of scope and will not be accepted by subsequent change request unless changes was made to the original project scope, which change the project baseline that may affect our schedule, cost, and resources. Change Management in the PMBok was for this purpose that allows project manager to deliver solution according to the scope specified.

Once again, Statement-of-Works are not requirements that we can expect from the Users. It was merely what project Sponsors and Key-Stakeholders want to see when the system was implemented. The Users may be able to inform the Analysts how the current order processing function work. It may involve telephone calls for stock status enquiry and telex transmission for shipping instructions between offices in City A and City B. It may also involve Stock Status Reports delivered to Production Planning Department to evaluate which product was below minimum stock level for production planning actions.

The System Analysts will have to determine how technology application can replaced the telephone calls, telex transmission, report delivery between departments and locations, as well as identifying when and where for data capture, data update, and report production. Therefore, It is up to the System Analysts to identify Systems and functional requirements after his/her understanding of the current business process resulted from interviewing and Analyzing what they had gathered, and determine how technology can apply to these processes improving productivities of business operation, and Requirement Analysis that produced Requirement Statements in our development life cycle.

System Designers will base on the Requirement Statement to create various screen formats, report formats, application modules and business logic for subsequent coding, testing, user training and implementation.

From Automation Age to Information Age in the 90

Automation was for the purpose of improving productivities by use of technology. The next step was to make use of information gathered from the past to provide management with better decision making purpose. It became known as the Information Age.

The complexity of Information Management was to allow management to learn, evaluate, and make appropriate decision based on data available from the past, that help them to identify potential cost saving areas within the organization as well as expanding market share for better profit margin. The use of Technology to deliver **value** is the main scheme of Information Age. At the same time, Internet access and emerging technology integration into application solution introduced additional challenge to the I.T. industries and practicing professional.

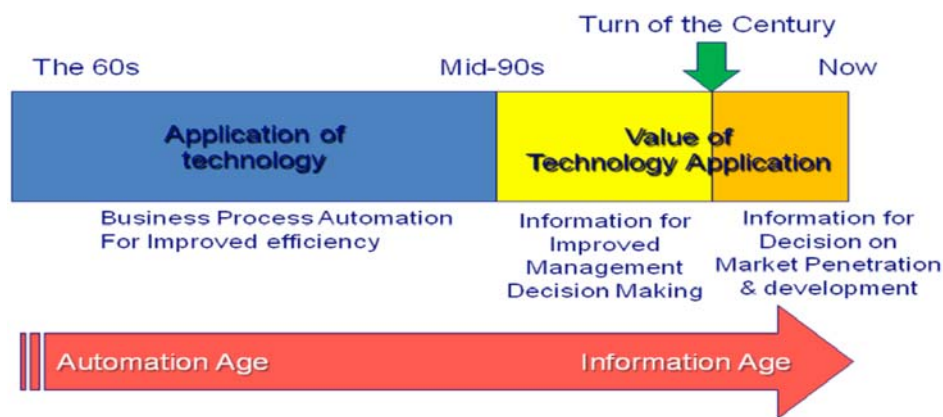


Figure 2: Automation Age to Information Age

I.T. developed its own service industry when most I.T. Department move from the traditional cost centres into profit center during the mid-80 to reign in the explosive technology application requested by department managers. Department have to justify the cost of development that can help them improve productivity and efficiency. Organizations were forced to hold down headcounts to produce efficiency and productivities. Since early 90, I.T. Service organizations mushroom to help organization that does not have the needed resources to do all the development work. The shift toward Value preposition is becoming apparent.

Most projects from mid-90 onward were conceptual based projects that venture into unknown territories, Users want to have application that help them to become more competitive and more profitable but not having a clue how it can be done through technology. A new position was created to address these “unknown territories” in the name of Business Analyst (BA). A Business Analyst is supposed to be a Subject Matter Expert (SME) of the industry and be able to identify business process that need to be develop for system development. BAs were intended to help the technical professionals who know little about the business priority and business practices to define the project scope for

subsequent development work. Unfortunately the gap between business knowledge and technology know-how remain ocean apart and most BA and SA are not talking the same language.

Project scope was hard to define under such circumstances. Users want the computers to tell them how to expand and capture additional sales, how to identify their customer's behavior and their taste of existing products through the use of Artificial Intelligent (AI); they want to be able to identify where they can save operation cost and better resources utilization via Enterprise Resource Planning (ERP); they want to expand their sales network using the Internet; and they also want to retain customer royalty and deliver better customer service by use of Customer Relation Management (CRM) application software. All these solutions are in fact looking for value through the use of technologies that replaced the technology application for productivity gained during the automation age.

It was during this time that the "Sponsor" and "Stakeholder" function emerged. Sponsors knew what they want to achieve without knowing how to get there, while Stakeholders will always support the Sponsor's initiative if it does not cause too much changes to their functional areas. The Stakeholders want to make sure all new systems implementation will bring benefits and recognition of their operational environment and sign-off any propose operation process or process change.

The software solutions may involved not only multi-operating departments, may also involve multiple geographic regions, and various business entities. Most unfortunate, there is no roadmap available how to achieve the business objective within the organization. Without a recognized business process in place, technology application was practically impossible to deliver such objective. Which means developing a precise project scope is rather difficult and it is impossible for the IT professional to develop Requirement for the application.

The practicing I.T. professional of that era was unable to determine how to develop a system that can satisfy our customer or users to meet their business objectives. We still rely on technology to drive the end result, thus Rapid Application Development (RAD) approach, and 4GLs products come into play by teaming the users and the technical team together to bring out ideas from the User's mind, and hopefully by working together can derive some forms of consensus toward the final solution. This kind of development approach turn the initial focus of defining project scope into defining user requirements, and that mind set continue to haunt the profession until to-date.

The Y2K problem in the late 90 distracted most of our inability of delivering Information Systems to Users. It is not until after the turn of the century that we come back and face the same challenge in application development during the mid-90.

From a Professional to a Craftsman

A decade had past, I.T. professional still face with the same problem of development application based on visions and concepts in this Information Age. The customer doesn't always have a business process that can leads to their business objective. Requirement for the application is Non-existent, and most project manager ignores that fact that we have to identify project scope before we can control subsequent changes.

Instead of finding ways of defining project scope that we can base on for analyzing the requirement, most of us hoping customers can tell us something, anything that we can considered as requirement. As a result, most projects are facing continuous changes during development stage. It is apparent that we are facing more change requests in our development stage than ever before, causing project over-run and long delay in system delivery.

Some of the most common excuses that I have heard from project managers during the last few years was their blame for their own project failure by pointing their fingers at their customer by saying "they don't not know their requirement".

"If they can tell us their requirements, we have no problem in delivering their solution....."

"Our users keep changing their mind....they want this function yesterday, and then these two additional functions tomorrow. I bet they have more requirements tomorrow....."

All these statements only indicated the failure of the project manager not having a well defined project scope, and cannot manage change request during development phase.

These individuals are becoming more like a craftsman instead of software professional. Tried to imagine when we buy our first home, and have some money to decorate the interior, we would spent days in identifying what need to be done. Once we know our requirement, we will find a carpenter to do the job for us. The Carpenter will deliver what we instruct them to do, no more, no less and doesn't care if the job is inside the house or outside the house, as long as you tell them what to do. The carpenter will not alter our instruction because they know well enough they may have to pull it down and re-do the job as we originally specified. The Carpenter is a craftsman by trade, very much like our software practitioners described above awaiting customer's requirement to deliver what is needed.

On the other hand, if you are a successful businessman just brought yourself a villa along the water-front. Instead of spending days of determining your need for decoration, you will find a Interior Designer and tell the Design your vision of your dream villa. The Designer will recommend to you and try to convince you're the recommendation will fulfill your vision of a dream home. The Designer will explain to you the kind of design and decoration you need to have the villa meet your living expectation. Once you agreed on the recommendation, craftsman will be called in to do the job for you based on the recommendation of the

Designer. The Designer is a Professional and I hope all our I.T. Practitioners are professional instead of a Craftsman within the I.T. industry.

The I.T. project Challenge in the 21st Century

While most I.T. professional are still considering the use of technology, customers are looking for value out of technology application. Instead of defining project scope which is the most important issue in Project management, we dive into requirements as soon as project start. How can customers or users tell you anything that they don't even know themselves?

It is important that we, who practice in the technology industry, should adjust our mind set in considering business value through technology application. Use of technology is no longer our major challenge. The end can have many means of getting there and it should be the ultimate end that we have to consider. As long as we know what we have to deliver, there are many ways we can achieve such end result. The end result of any project is "deliverable". Our project is officially closed as long as the customers accept and sign off our final deliverable. Therefore we should consider the deliverable as the project objectives

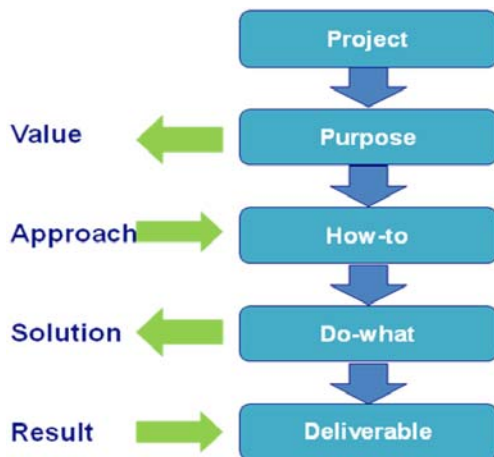
It was back in early 96 when I was tasked with the development of an internet sales application for a department store back in Melbourne, Australia. I really have no ideas how to finish my assignment. I start spending time with my project sponsor and hope to understand the motives behind his investment. Afterward, we started to develop a business process for achieving the final objectives. We have to make a lot of assumption during business process development because we don't really know the behavior of their potential customers at the other end of the internet connection.

The business process we developed became our project scope. Once the business processes were completed and accepted by the sponsor, we were able to develop the requirement statement and completed the development for implementation without too many changes. It was a brand new experience for the sponsor and myself acting as project manager.

After this initial encounter, I started looking at ways of delivering conceptual projects. It is unrealistic to spend a lot of time to define the future business process to come up with the project scope, yet the project scope has to be defined as soon as possible so that requirements can be identified for subsequent design and development work. In view of the competitive business challenge in today's environment, development cycle for common projects had been forced to finish within 6 months or less. Therefore it is not practical to spend a month or so in determining the project scope alone. We need a new way of identify project scope so that we can move on to the development as soon as possible.

Project Components Decomposition Method

At the turn of the Century, I was once again tasked with the development of a Customer Relation Management System for the Retail Banking Division of a banking group in Canada. Without knowing how Customer Relation should be managed, we formed a project team that involved Branch Managers and a few of our Major Customers to brainstorm our deliverable.



Knowing what the customer expect to feel satisfy with the banking services, and our branch manager expect to find from existing database on customer information in order to develop new service and new businesses, we start looking for ways of delivering such expectations. The Brainstorming process formulated the initial version of the Project Components Decomposition Method (PCDM).

Since then, it had matured into the current stage that I have been teaching my graduate students in China during the last 3 years.

Figure 3: Project Components Decomposition hierarchy

The main purpose of the PCDM is to decompose **Project** into components which form the final deliverable of an application. The Components consists of 4 levels of decomposition processes. Except the top level “Purpose Statement” require detail discussion with the Sponsor, all other levels utilize small team consists of the Project Manager, System Analysts, Business Analysts, and a number of key stakeholders to brain-storm each and every How-to statements, Do-what statements and formulate deliverable definition. All these require team consensus whereby the final result is accepted by all party concern.

Level One should be developed along with Project Sponsor to identify the **Purposes statement** of the Project that brings value to the investment. Any project will have multiple purpose statements and we have to understand the mind-set of the sponsor for investing the application solution. It is not exactly the Project Objective Statement (POS) that we know of. It is more on capability development, productivity improvement, manual process refinement, and other value that contribute to the betterment of organizational operation.

Level Two should be analyzed by the Project Manager or Senior Analyst the various approaches it should take to come up with the value or the **How-to statements** in direct response to each purpose statement. “How-to” is in fact “**How** are we going **to** deliver the value of Purpose Statement”. Once again, each purpose statement may consist of multiple Approaches that lead to different How-to statements. Also, one How-to statement may satisfy multiple Purpose Statements.

Level Three is once again developed by the Project Manager or Senior Analyst the various solutions for each of the approach, i.e. **Do-What statements** to satisfy each and every How-to statement.

Since How-to statements are the approach we determine could deliver the value of each and every Purpose statement. With each approach we take, we need to have one or more (multiple) solutions that can accomplish each and every How-to statement. Likewise, each solution can support one or more approach in return.

The **final level** is in direct response to each of the solution specified from the Do-What statements and it forms the final **deliverable statements** of the Project. All these deliverables become the Project Scope Definition.

The following case study will illustrate how to use PCDM to determine the final deliverable as soon as project is initialized.

Case Study: Tour booking Management System Development

A hostel owner in a seaside resort town would develop local tours to attract customers and like to develop a tour booking management system for out of town trips. It is a new attempt at the hostel and hope it can help capture additional business from hostel guests and non-hostel guests from around the resort town. The solution should:

Initially offer three trips:

- **Deep Sea World Experience** (US\$45) – which can be booked with optional scuba diving (US\$30). Users should check those booking scuba diving have a diving qualification – this should be checked and recorded.
- **Safari Experience** (US\$25) – display a message to warn bookers that this involves an early start at 7am
- **Mountain biking** (US\$45 with bike hire, US\$30 without) – you should take a note of whether or not users have their own bikes at the time of booking
- Take the booking – which trip, what date, number of adults, names, deposits etc
- Save all the booking information into electronic files
- Offer an accounting facility which will tell the hostel owner the value of all the bookings in the system

- Be easy to use and data can be ported to PDAs/ smart phones or accessed via Internet connection etc
- A normal Desktop application, rather than a Web application is desired. However, hostel owner is regularly away from the hostel, so it's desirable if you can develop several web pages to show him the current booking status from his phone browser.

Case Study Analysis:

Most practitioners will consider the above information given by the customer is part of the Requirements. Then we have to identify the rest of the requirement, how do we know if we have all the requirement to start development work?

Some will say it is the project scope. Then we have to develop the business process within the scope, and it is a business venture that is not yet implemented, the customer was hoping the application can provide them with the business processes. Should we develop the business process first? Or develop the system and come up with the business process for the customer?

I would consider the information as part of the expectation from the customer for the application system. It is my consideration that the information provided by the customer is part of the quality requirement of the application. I would start defining the project scope, and then develop the requirement for the application development thereafter. What are the Scope definitions anyway?

The Purpose Statements

The first level of decomposition is to work with Project Sponsor to identify the value of their investment, and Value should be the ultimate purpose of their investment objectives. Some of these purposes could be a vision, or it could serve an expectation, or certain intangible gain of business operation.

Case Study Review: Purpose Statements

1. Provide staff members with accurate Tour Information, booking & condition of Tours
2. Better booking management and tour management
3. Improve front-desk effectiveness, provide tour booking conditions to avoid misleading information distribution, minimize potential risks causing accidents during tour
4. Provide up-to-date tour details, booking & financial status
5. Provide management with PDA/Smart Phone access to booking and financial details
6. Attract additional customers to the hostel
7. Attract additional hostel-guests and non-hostel-guests business

Case Study Analysis: Validity of the Purpose Statements

By working with the Project Sponsor, we came up with 7 purpose statement. These are the ultimate purposes for the investment of the application solution. Before we accept the validity of purpose statements, we better compare how it map against the original information issued (Expectation or Quality Requirements) that some considered Requirements.

Customer Expectation	Purpose Statements
<ul style="list-style-type: none"> • Manage Deep Sea World Experience (US\$45) – which can be booked with optional scuba diving (US\$30). Users should check those booking scuba diving have a diving qualification – this should be checked and recorded. 	1, 2, 3, 4
<ul style="list-style-type: none"> • Manage Safari Experience (US\$25) – display a message to warn bookers that this involves an early start (7am) 	1, 2, 3, 4
<ul style="list-style-type: none"> • Manage Glentress mountain biking (US\$45 with bike hire, US\$30 without) – you should take a note of whether or not users have their own bikes at the time of booking 	1, 2, 3, 4
<ul style="list-style-type: none"> • Take the booking – which trip, what date, number of adults, names, deposits etc 	2, 3, 4
<ul style="list-style-type: none"> • Save all the booking information into a data file 	2, 3, 4
<ul style="list-style-type: none"> • Offer an accounting facility which will tell the hostel owner the value of all the bookings in the system 	3, 4, 5
<ul style="list-style-type: none"> • Be easy to use and data can be ported to PDAs/ smart phones etc 	5
<ul style="list-style-type: none"> • A normal Desktop application, rather than a Web application. However, hostel owner is regularly away, so it's desirable if you can develop several web pages to show 	2, 4, 5

him the current booking status in the phone browser.	
--	--

Figure 4: Mapping of Purpose Statements with Expectations

It is apparent that Purpose Statement 6 and 7 were not specified from the Quality Requirement. It is time for us to check with the sponsor and resolve the differences.

The Purpose Statements are the source of our component decomposition processes. It is concurred by the Project Sponsor and is the main reasons behind the project initiation. If we have any information contradiction with the Purpose Statements, we will have difficulty in getting users sign-off at the end of the delivery.

A simple Gap Analysis has to be performed to determine if the two extra Purpose Statements are vital to the success of the project's expected value. By checking with project sponsor that to achieve purpose statement 6 and 7, the How-to Statement will become advertising of the tours in other media and the Do-What statement may become Tour Brochures printing for distribution. These 2 purpose statements do not have anything to do with the application development and should not be considered as part of the project scope.

Now that we have secure Five Purpose Statements, we can proceed with the next level of components decomposition.

The How-to Statements

The Approach we formulate to address each of the Purpose Statements becomes our How-to Statements. The best way of decomposing Purpose Statement is to keep asking yourselves "how to satisfy such purpose, what approach should I take?" Each Purpose Statements may consist of multiple How-to Statements. You may come up with one or more answers for each purpose and sometimes each approach may satisfy multiple purposes.

Case Study Review: The How-to Statements

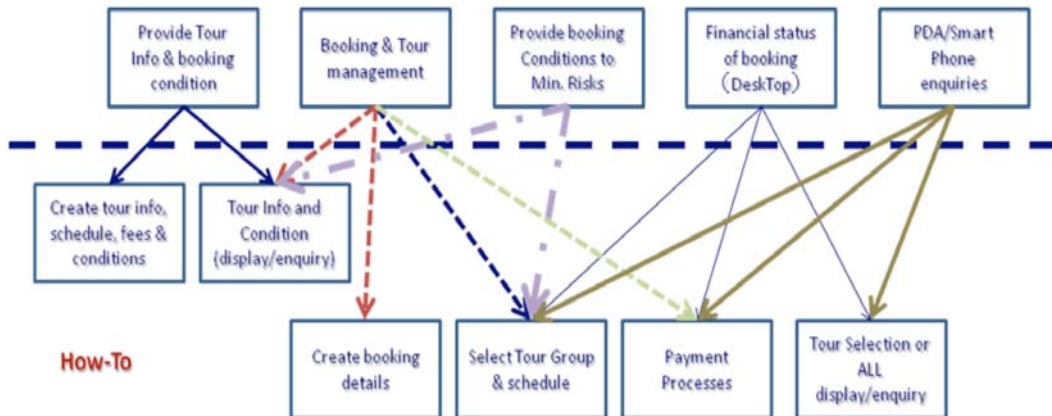


Figure 5: From Purpose Statements to Ho-to Statements

The Do-What Statements

To arrive with the solution of each and every How-to Statement, it is necessary to identify actions that address individual “How-to”. There may be multiple actions that satisfy each Approach (known as “How-to”) and each action can be part of the total solution. The best way to decompose “How-to” Statement is for the project manager or Senior Analyst to find answers for “What are the solution I can deliver that address the How-to Statement”. Similarly, each solution (Do-What) may satisfy multiple How-to Statements.

Case Study Review: The Do-What Statements

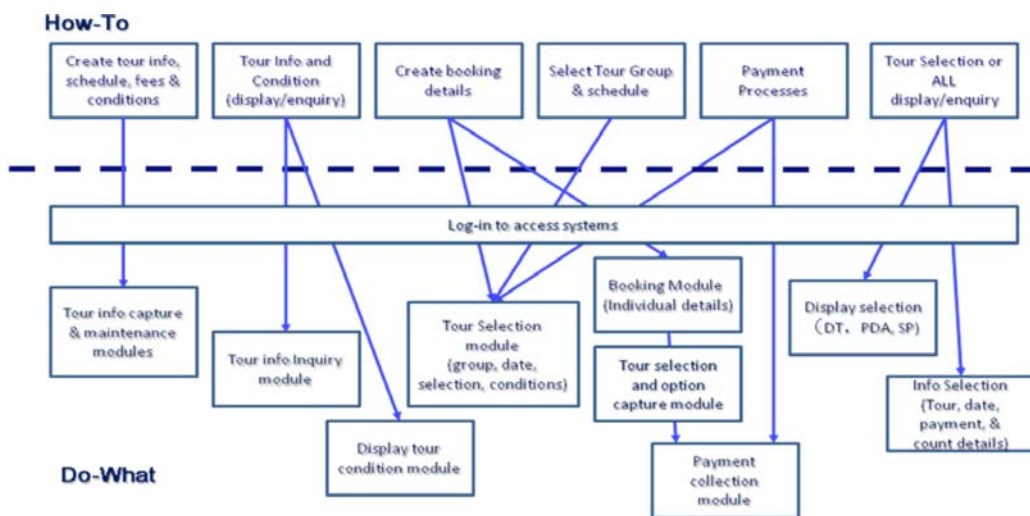


Figure 6: From How-to Statements to Do-what Statements

challenges and we shall find ways to resolve any challenge in the future. So far, we are confident that Project Components Decomposition Method will enable us to identify project scope that allow us to manage software development projects in a professional way,

**Thanks to Dr. J. Davidson Frame, Academic Dean of University of Management & Technology, Virginia, USA, whose input to this article made it most valuable to readers to track the IT history and have more insight of using PCDM.*

About the Author:



Hubert Vaughan

Professor of Project Management

Tsinghua University, Beijing, China

Author



Professor **Hubert Vaughan** is a graduate of Melbourne University in Australia, with an MBA from York University in Canada.

Professor Vaughan commenced his career with National Mutual Insurance in the field of Information Technology in 1972. During the last 30+ years, he has lived on all five continents and held senior technical and management positions with such international organizations as IBM, DEC, Unisys, Tandem, Cable & Wireless, Bell Canada, ANZ Banking Group and Bank of Montreal. His career has covered software development, professional services, technology consulting, project/program management, strategic planning as well as business development. Mr. Vaughan is a Professor at the Institute of International Engineering Project Management of Tsinghua University, Beijing, China. He also teaches PM at the Graduate School of the China Academy of Science; Software College of Beijing University of Astronautics and Aeronautics; Software College of Nankai University, and the School of Software Engineering of the Harbin Institute of Technology. Apart from his teaching engagements, Hubert also acts as consultant for several State owned organizations in China. Professor Hubert Vaughan is an International Editorial Advisor for PMForum and PM World Today; he can be contacted at hubertvaughan@pm.tsinghua.edu.cn. Additional information can be found at http://www.pmworldtoday.net/team/editorial_advisors.htm.