Disposable System Development – A New Paradigm
For Managing ICT Projects in the Innovation Age¹

Part 3: Identify Project Deliverables instead of Project Scope

By Professor Hubert Vaughan
Tsinghua University, China.

Let us consider that you are inside an empty room, and you are allowed to use it anyway you want. As the beneficiary of this empty room, what is the first thing that comes to your mind?

1. The first thing you must determine is what do you want to use it for, i.e. the ultimate objective, or purpose. Do you want to use it as your shop front? Or your business office? Or your little hide-away with your buddies for the weekly poker game?

This is a process to identify the value of your potential investment, i.e. fill up the room with contents (deliverable) that serve your purpose. There are always projects needing development. During the Automation Age, choices were made based on potential saving and operation efficiencies. In the era of Information Age, choices were made based on management objectives and effective management decision making purposes. For the Innovation Age, system will focus more on exploring and identifying potential opportunity for the beneficiaries, such as market penetration, industry leadership, and niche market/product discovery that propel organization grow.

2. Once you know what you want to use the room for, the next step is to identify what should be inside to serve such purpose or objective. A game table? A fridge? A couch? Or maybe a shower room? All these are considered as the final deliverable and each deliverable serves a particular purpose inside the room.

Identify the deliverables of your project that can add value, benefits, and capability to your organization once the project objectives are achieved. Unlike the traditional project management approach of managing changes, project managers for the Innovation Age may be required to embrace changes if such change add value, benefit, and capability to the final outcome, in a way similar to program management.

3. Now that you had identified your deliverable and the purpose each deliverable served. You may wish to consider the components of each deliverable, e.g. the “game table” will include a minimum of six chairs. Therefore each deliverable may consist of single or multiple components to serve your purpose.

¹ This series of articles is by Professor Hubert Vaughan, recently retired from Tsinghua University in Beijing, and is based on his research over the last ten years during which he has developed some new approaches to managing major information and communication technology projects.
Analyze and design the components, not modules (please refer to Part II published in December issue of PMWJ, 2013) of each deliverable with the objectives of serving one single function. This formed the foundation of components replacement for future system maintenance/expansion requirement. This process focus around components interface management for the final project objectives.

4. The next thing you may consider is the appearance of your deliverable so that they can fit the room nicely and comfortably. How do you like your deliverable to appear becomes your next consideration! Do you want your game table with velvet covered top? Should it be round or square? Foldable? How these will fit together?

Create the business procedures how the systems is to be used once it is delivered. Design the UI and Data relationship (this is a form of creating interface management focus between components for the system once implemented, as part of the Incubation Phase of the Disposable System Development and Management Model explained in future articles).

5. Now that you know what components are appropriate for your purpose, you can start building, procure, or search from your own backyard storage. You can even consider finding someone to build it for you if you let that someone become your weekly game player.

This is the Creation Phase of the Disposable System Development and Management Model that enable PM to determine if such component should be in-house develop, out-source, procure, or commission to business partners depends on timeline and budget of the project.

6. Finally, Put the deliverable inside the room and see if you are satisfied. Make adjustments if necessary to make sure you are happy about it. If it all serves your original purpose, then you have a little hide-away with your buddies once a week.

Eventually all components will be integrated and test in live environment before sign-off, as the final Demonstration Phase of the Disposable System Development and Management Model explained in future articles.

Does it sound like a project? It is indeed!

But wait ….. What is the project scope and requirement? Why don’t we manage it like a project by defining the project scope, identify the requirements, plan the execution, build the deliverables and finally deliver?

This is exactly what we face when asked to deliver an ICT project today. Unlike before when the project beneficiary knew exactly what they wanted in the automation age, e.g. build an accounts receivable system to replace the current manual procedure. Or able to identify what information they wished to consolidate for expense identification purpose during the information age.
Most project beneficiary started a project with a vision of what they want to do but really have no idea of what they really need or how to achieve their vision, very much like giving a free hand to determine what to do with the empty room. The project managers will have to guide them step by step to identify their investment purpose, and through the purpose identification, determine the project deliverables and project components.

**Project Components Decomposition Method**

At the turn of the Century, I was tasked with the development of a Customer Relations Management System for the Retail Banking Division of a banking group in Canada. Without knowing how Customer Relations 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.

The main purpose of the PCDM is to decompose **Project** into components which form the final deliverable of an application. The Components consist of 4 levels of decomposition processes. Except the top level “Purpose Statement” that requires detail discussion with the Sponsor, all other levels utilize small team consisting 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 for final approval and acceptance. 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 **Purpose Statements** 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. Some Purpose Statements may look like:

- “Enable customers’ portfolio management capability”.
- “Able to analyst customers’ income and expanse models and trends”.
- “Able to deliver comprehensive financial management and benefits to customers’ financial portfolio”.
- “Develop formal and informal communication with customers on their financial investment status”.
- “Identify and develop new business opportunity for customers”.
- “Develop and improve customer service satisfaction and loyalty”.

**Level Two** should be analyzed by the Project Manager or Senior Analysts for 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 Project Scope of our Automation Age**

It was relatively simple and straightforward to identify project scope during the Automation Age (circa early 70s to early 90s). Most computerization projects simply developed software applications replacing the manual processes within a department of an organization that helped us improves our productivities and efficiency in our work environment. Automation Age was an era for technology replacing manual processes to reduce errors and eliminate redundant activities.

If we were given an assignment to develop an Inventory Control System for the warehouse, we may receive an instruction like “*Develop a system that identifies and keeps count of all goods (Raw materials, semi-finished and Finished-products) that go into the warehouse, and all goods that leave for the production floor or for delivery to customers*”. This instruction was known as “Term of References” that drew a boundary of functions that needed to be converted from manual processes to automated procedures. Term of References was later
called “Project Definition” or “Scope” in the early 80s when project management was officially introduced to commercial entities. During automation age, scope was easily identified by knowing the starting point and ending point of the business processes to be replaced.

Senior management and departmental staff knew exactly what manual procedures needed to be replaced by the automation processes. Functional requirements were the business processes that serve specific business objectives within the department and it can easily discovered through fact finding, 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. From the operational staff, System Analyst can 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 the Fact Finding exercise, the Analyst will base on all those business processes to identify data capture requirement, file format requirement, data update requirement, and reporting requirement. The Terms of Reference basically formed the boundary of our development work.

Requirement were never provided by Users, they were developed by the System Analysts after exploring the six Ws (what, where, when, who, how and why) each function the manual procedures performed within the boundary of project scope. Once “what need to be done by the computer to replace manual processes were identified” by the Analyst, and computerized solutions were designed based on these so these “Requirement Statements”. This became the Functional Requirement of our project and the System Designer will group similar processes together to create Application Modules and system logics for program coding and testing (development), and subsequent implementation.

As more and more departmental business procedures migrated from manual processes to computerized solutions, more and more IT projects involved new application development and system integration with existing solutions. Projects involved cross department solutions, possibly with multi-location business operations. Scope is manageable within the boundary of the business entity in order to control subsequent change requests.

The Project Scope of our Information Age

System development during the Automation Age was mainly based on isolated departmental operations. From late-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 focus
of our automation age. Until the present day, Information Age focused on value of technology implementation. Which means technology application should give them the much needs information for making business decision, or provide them with a way of expanding their market shares, or help them reduce operation overheads.

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 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 requirement 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”.

© 2014 Hubert Vaughan
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 were able to inform the Analysts how the manual 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.

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.
From Automation Age to Information Age in the 90s

I.T. developed its own service industry when most I.T. Department move from the traditional cost centers 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 objective, thus enable the technical team to formulate 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 involve 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 mindset 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-90s.

We drifted from Professionals to Craftsmen

A decade had pass, IT professionals still face 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 requirements, most of us hope customers can tell us something, anything that we can consider as requirement. As a result, most projects are facing continuous changes during the 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 fingers at their customers by saying “they don’t not know their requirements”.

“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 lack of analytical skills of our IT practitioners and the failure of the project manager of formulating a well-defined project scope, and cannot manage change requests during development phase.

These individuals are becoming more like craftsmen instead of software professionals. Try to imagine when we buy your first home, and have some money to decorate the interior; we would spend 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 an Interior Designer and tell the Designer your vision of your dream villa. The Designer will recommend to you and try to convince you that 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 for the villa to 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 IT Practitioners are professional instead of a Craftsman within the IT industry.

We struggled for nearly two decades during this so called information age to identify project scopes and functional requirements in the development and integration of automated solutions. Finally scope definition was supported by Statement of Work (SOW) to make certain we understand what were included and what were not included in our projects. However, ICT projects could involve every department in every region the company is operating within its commercial entity in order to provide the management information for better decision making and operational efficiency. The larger the organization, the more complex the scope definition became, the more difficult to lock down and manage.

**Scope for the Innovation Age?**

While most IT professionals are still considering the use of technology, customers are looking for value out of technology application. Instead of defining project scope which is considered 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.

Traditionally, UAT is based on full functionality of the system we developed. Users can tell if any one of the functions is missing, if so, have to deliver it before sign off.

Today, most users want a system without any idea how it can achieve its investment objective. They cannot tell if a function is critical to the application or any function is missing (because system is to fulfill a strategic objective or conceptual ideas) from the system until they hit on it accidentally. The more functions are available to them, the more they want from you.....

Requirement in today’s system is the basic quality that should be built into the system. Acceptance is based on what the user received from you that meet his/her strategic purpose of the system, namely, the deliverable

The IT project Challenge in the 21st Century

At the beginning of this document, the scenario of an empty room for a beneficiary to determine what to do with is a simple illustration of how system development is envisioned in organization today. Management knows they have the capability to build computer system for their purpose, but lack of focus of what to achieve. They gather ideas of what could be done and then select the one they feel could give them the benefit of expanding their market and business opportunities. Yet they are not sure how to get there.

As a Professional ...

- We need to be able to understand the concept or vision behind the Needs of the development, i.e. The ultimate purpose
- We must equip ourselves with the knowledge, process, and ability to convert intangible visions or ideas into tangible deliverable
- We needs to be pervasive, convincing and consultative
- We needs to let our customers feel that they are giving us our ideas.

The Disposable System Development and Management Model illustrated below provides a straightforward approach in delivering user systems that meet their investment objectives. It follows similar logic of the empty room presented at the beginning of this paper.

This month we focus around the “Exploration Phase” by applying the PCDM to identify project deliverable. The next few issues will focus around subsequent phases so that we can have a more comprehensive view of how this model can help us resolve some of the development issues in today’s projects.
Figure 3: Disposable System Development and Management Model for ICT Projects

Stage 1 Execution (Exploration Phase) use the Project Components Decomposition Methods to identify project deliverable as well as major components. The following case study will illustrate how to use PCDM in the Exploration Phase to determine the project components and final deliverable as soon as project is initialized. A section in Part II (published in December 2013) explained the differences between application modules and application components.

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 requirements 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.

<table>
<thead>
<tr>
<th>Customer Expectation</th>
<th>Purpose Statements</th>
</tr>
</thead>
<tbody>
<tr>
<td>• 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.</td>
<td>1, 2, 3, 4</td>
</tr>
<tr>
<td>• Manage Safari Experience (US$25) – display a message to warn bookers that this involves an early start (7am)</td>
<td>1, 2, 3, 4</td>
</tr>
<tr>
<td>• 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</td>
<td>1, 2, 3, 4</td>
</tr>
<tr>
<td>• Take the booking – which trip, what date, number of adults, names, deposits etc</td>
<td>2, 3, 4</td>
</tr>
<tr>
<td>• Save all the booking information into a data file</td>
<td>2, 3, 4</td>
</tr>
<tr>
<td>• Offer an accounting facility which will tell the hostel owner the value of all the bookings in the system</td>
<td>3, 4, 5</td>
</tr>
<tr>
<td>• Be easy to use and data can be ported to PDAs/ smart phones, etc.</td>
<td>5</td>
</tr>
<tr>
<td>• 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 him the current booking status in the phone browser.</td>
<td>2, 4, 5</td>
</tr>
</tbody>
</table>

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

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

The Project Deliverable

The end result of each “Do-what” statement will turn up to be a tangible deliverable definition. Building the relationships between all deliverable will become the project deliverable that outlines what we have to deliver at the end of the development in order to meet the customer expectation. This will become the Project Scope that the development will based on that lead back to the Purpose Statements given and agreed by the project sponsor at the early stage of our project initialization.

Case Study Review: Deliverable Definitions

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

Figure 7: From Do-what Statement to Deliverable Definitions
The End Result

Using PCDM that define project deliverable formulating the project scope took one day to complete. Additional three days to identify functional requirement of the application and 8 person-weeks for the development. It took 2 persons in just six weeks to complete the assignment meeting customer expectation.

During the implementation stage, customers’ requests changes to certain Interface layout causing one additional day to fix. Overall, PCDM help us identified the project scope that allow us to firmly identify requirement necessary for the application.

We are now in the process of expanding the use of PCDM that will enable us to identify project scope for integration projects and other major undertakings. We may come across 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 allows us to manage software development projects in a professional way.

I shall make use of the next few issues here at the PMWJ to explain the Incubation Phase, the Creation Phase, and the final Demonstration Phase. I shall also discuss with more details on identifying components that can be dispose of when its function is no longer relevant to the business objectives. System maintenance is no longer required because new components can be developed to replace the old.

Furthermore, I shall explain how to search for development partners replacing in-house development team or out-sourcing assignment to minimize development cost as well as operation overhead.

This Disposable System Development and Management Model is still in its infant stage. I shall try to explain how this model has the potential to change the landscape in the IT industry. It addresses most of the issues we face when managing ICT projects, and I welcome those who may want to work with me on refining the processes of this model in the near future.

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

Professor Hubert Vaughan (Retired)
Beijing, China

Hubert Vaughan commenced his career in the field of computer technology in early 1972. For thirty years, Hubert practiced and served a number of International technology and financial Organizations including IBM, DEC, Unisys, Tandem, Bell Canada, Andersen Consulting, Lucent Technologies, National Mutual, ANZ Banking Group and Bank of Montreal; holding senior management positions in Technology related services. His career covered the five major continents around the world as Department Manager, Director, Assistant Vice President, and Vice President that spanned across software development, professional services, product development, technology consulting, project/program management, strategic planning as well as business development.

The last ten years, Hubert joined the Academic Institutions in China as Professor at the Institute of International Engineering Project Management (IIEPM) of Tsinghua University. Hubert also lectured at the Graduate School of China Academy of Science, the Beijing University of Aeronautic and Astronautic, teaching Innovation Management, Management of Technology, Program Management, Project management, and Software Engineering.

Apart from his teaching engagements, Hubert is a Research Fellow at the China Academy of Management Science, a member of the International Society of Professional Innovation Management (ISPIM), a former member of PMI’s Certification Governance Council (CGC); a co-founder of First International Innovation Management Alliance (FiMA), and an Editorial Advisor of professional e-journal PM World Journal. Hubert is a Program Consultant to a number of multi-billion dollars projects run by State-Owned technology organizations and financial institutions in China.

Hubert is a regular presenter at international conferences and seminars in North America, Europe, Middle-East and Asia-Pacific. He had published more than fifty papers related to Software Engineering, Project Management, Program Management, and Innovation Management subjects both in China and in various international professional journals. Retired from his academic engagement in July 2013, Hubert continues his research work in Innovation Engineering and presents at international events about his research findings throughout his career. He can be contacted at hubertvaughan@gmail.com