

# Agile Software Development Methods Adoption in Thailand – A Survey of Thai Universities

Roy Morien

roymorien@hotmail.com

Research Fellow

Digital Ecosystems and Business Intelligence Institute (DEBII)

Curtin Business School

Curtin University of Technology, Perth, Australia

and

Visiting IS Specialist

Naresuan University, Phitsanulok, Thailand

Dr Orasa Tetiwat

orasat@nu.ac.th

Faculty of Science

Naresuan University, Phitsanulok, Thailand

## ABSTRACT

This paper reports on the commencement of a research project into the adoption of agile software development and agile software project management methods and approaches in universities in Thailand. A preliminary survey, by interviews, of 12 universities in Thailand, including some of the universities considered to be in the Top 5 IT universities in Thailand, was undertaken during June, 2007. Interviews with staff, the presentation of seminars and classes to both undergraduate and post-graduate students, as well as academic staff, were undertaken during this period. Survey questionnaires were given to both staff and students, and a website with these questionnaires has been created. Responses to these questionnaires have yet to be received at this early stage of the survey project. This paper narrates some of the experience of the authors in conducting the interviews and seminars, and reports on the 'state of play' in those universities in regard to the adoption of agile software development and agile software project management methods and approaches in curriculum, and the knowledge base of IS / IT and C/S academic staff of this subject matter.

**Keywords:** agile software development, computer science curriculum, information systems curriculum.

## 1. INTRODUCTION

During June, 2007, an extensive tour of universities in Central, Eastern and Northern Thailand, was undertaken, with the intention of finding out the level of interest in and knowledge of agile software development methods, and agile project management methods, amongst Thai university academics who teach in the areas of Information Sys-

tems, Information Technology and Computer Science, especially those who teach software development methods, systems analysis methods and software project management.

This study is part of a substantially larger research project in to the adoption and use of agile development methods, in Thailand, and subsequently in China and Australia.

Twelve prominent universities were visited, in Mahasarakham and Khon Kaen in the east of Thailand, Chiang Mai, Uttaradit and Phitsanulok in the Central-North, and Bangkok.

Seminars and classes were presented to undergraduate and post-graduate (Masters and PhD) students, and academic staff, at most of these universities. In-depth interviews were conducted with academic staff to ascertain their personal level of knowledge and interest in the subject, and also to ascertain the adoption of these methods in appropriate curriculum in undergraduate and post-graduate courses.

This followed a period of 4 months during 2006 when I was teaching at a major university in Thailand, in Phitsanulok, a small provincial city about 375 kilometres north of Bangkok. That university is considered to be one of the major universities in Thailand, and probably the leading provincial university in Thailand.

**Motivation for the Study**

The motivation for this study is that the subject of agile software development, and agile project management now has sufficient support in the published literature, and published research attesting to the superior effectiveness of these methods, that it seems to be an important topic in the software development knowledge domain, and therefore an essential topic to be included in computer science and information systems curriculum at universities and colleges. Yet that does not seem to be the case. This study is the commencement of a research activity intended, in part, to clarify the position in Thailand, with an intention to encourage and promote this.

Support for this claim of superior effectiveness, and use, is found, *inter alia*, in Ambler (2007), Evans (2006), Shine Technologies (2002), Taylor (2001), Solon (2002), VersionOne (2007) and James Martin (1991) who stated "RAD has been demonstrated ... to be so superior to traditional development that it seems irresponsible to continue to develop systems the old way ...".

**2. CHARACTERISTICS OF AGILE AND LEAN DEVELOPMENT**

The Agile Manifesto (Agile Manifesto, 2007) states the following guiding principles:

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

According to Evans (2002), the essential difference between the traditional approaches to systems development and the agile approaches is the difference between planned iteration and the unplanned rework so common in waterfall-based projects. With the traditional approach, adherence to the prescribed software process is considered the major determinant of success. With the agile approach, adaptation toward achieving the end-goal – working software – is the major factor in success.

This table, taken from Evans (op.cit) summarizes some "contrasts between these remarkably different approaches."

	<b>Waterfall</b>	<b>Agile</b>
<b>Guiding metaphor</b>	Manufacturing /Engineering	Organic /Emergent
<b>Focus</b>	Documentation, Schedule	People, Working Code
<b>Dynamic structure</b>	Cause and Effect, Pre-ventive Approach	Chaordic (Ordered Chaos), Adaptive Approach

**Embrace Change**

If there is one guiding principle of agile development, it is "Embrace Change". This implies the acceptance of the fact that requirements in detail cannot properly, comprehensively or accurately be defined at the beginning of the project (the "Big Bang" approach, or the Big Design Up Front (BDUF) approach (Ambler, 2003a), and are almost certainly subject to change in extended period projects. To quote Highsmith (in Orr, 2002) "By the time a three-year project delivers its first working versions, many of the users have forgotten what they agreed on in year one or have moved on so that the people who have to work with the system have

*little or no idea what it was developed for. One thing is certain, though, if a project takes three years to implement, you can be sure that the requirements will be at least two years out of date by the time it comes into existence.*

Specific definitions of agile development have been attempted by Mahanti (2006: 197) as "a departure from plan-driven traditional approaches, where the focus is on generating early releases of working software using collaborative techniques, code refactoring, and on-site customer involvement". And Melnik and Maurer (2005: 481) as "human centric bodies of practices and guidelines for building usable software in unpredictable, highly-volatile environments". Software development projects are considered to be an unpredictable and highly volatile environment.

For the purpose of this discussion, a definition of Agile and Lean development is "A software development method is said to be an agile software development method when a method is people focused, communications-oriented, flexible (ready to adapt to expected or unexpected change at any time), speedy (encourages rapid and iterative development of the product in small releases), lean (focuses on shortening timeframe and cost and on improved quality), responsive (reacts appropriately to expected and unexpected changes), and learning (focuses on improvement during and after product development)" (Qumer & Henderson-Sellers, 2007).

So, for a software development activity to be agile, it should encompass practices that can be variously described as:

- People Focused: (1) Collaborative: collaboration between developers and clients is continuous and continual. (2) Self-Organising and Self-Managing Teams: Significant responsibility is handed to the team members, rather than the Project Manager, to decide on the work to be done next, in an iteration.
- Empirical and Adaptive: Project management practices that have been published to support 'agile development' practices are described as 'empirical', 'adaptive', 'evolutionary' or 'experiential' rather than 'prescriptive', or 'pre-planned'.
- Iterative: Development is achieved through a series of short iterations each of which

produces a useable enhancement to the system.

- Incremental: Development is achieved through a series of delivered increments to the system, each of which produces a fully developed, fully tested and certified extra feature or component of the system.
- Evolutionary: the system grows in size, the requirements *in detail* are continuously discovered, and are continually emergent during the development period.
- Just-in-Time Requirements Elicitation: Requirements are stated in detail 'just in time' to develop them, in the iteration in which those requirements will be implemented.
- Knowledge-Based: Development activity is decided upon by the knowledgeable, self-managing members of the team, with continual knowledge sharing about the product, the technology and the progress of the project.

### **Agile Development Methods and Approaches**

A number of agile development methods and approaches have been published. These include:

- EVO [www.xs4all.nl/~nrm/EvoPrinc/](http://www.xs4all.nl/~nrm/EvoPrinc/), (Gilb, 1988).
- Spiral Model (Boehm, 1986, 1988).
- Extreme Programming ([www.xprogramming.com/xpmag/whatisxp.htm](http://www.xprogramming.com/xpmag/whatisxp.htm)), (Beck, 2004)
- Scrum ([www.controlchaos.com](http://www.controlchaos.com)), (Schwaber & Beedle, 2001)
- Crystal ([alistair.cockburn.us/index.php/Crystal\\_methodologies\\_main\\_foyer](http://alistair.cockburn.us/index.php/Crystal_methodologies_main_foyer)) (Cockburn, 2003, 2004)
- Function Driven Development (FDD) ([www.featuredrivendevelopment.com/](http://www.featuredrivendevelopment.com/)), (Palmer & Felsing (2002)
- Dynamic Systems Development Method (DSDM) ([www.dsdm.org/](http://www.dsdm.org/)), (Stapleton, 2003)
- Lean Software Development ([www.poppendieck.com/](http://www.poppendieck.com/)), (Poppendieck & Poppendieck 2003, 2006)
- Agile Unified Process ([www.ambyssoft.com/unifiedprocess/agileUP.html](http://www.ambyssoft.com/unifiedprocess/agileUP.html)), (Alhir, 2005)

- Agile Data Modelling Method ([www.agiledata.org/](http://www.agiledata.org/)).
- Rational Unified Process (RUP) ([www-306.ibm.com/software/rational/](http://www-306.ibm.com/software/rational/)), [ootips.org/rup.html](http://ootips.org/rup.html)

A recent industry survey indicated that Scrum was the predominant agile method in use, being applied in 37% of organisations that were using an agile approach in their development activity (VersionOne & APLN, 2007). This survey had over 1680 respondents from 71 different countries.

### **3.COMPUTER EDUCATION IN THAILAND.**

It was apparent from observations of curriculum and courses in 12 different universities in Thailand, that, in general, courses that are variously called Information Systems, Information Technology, Computer Science, Business Computing, are offering contemporary and relevant, and often imaginative, subject matter. For example, one eastern university (Mahasarakham University) has a recently created Department of New Media, offering courses in computer graphics, 3D and 2D graphics design, and a special course in games development. Generally there is a significant emphasis on Internet development, with some lecturers expert in SOA, XML, web services, and OO methods and programming featuring well in curriculum. Expert Systems and AI applications were also taught at one university.

At least at that university, students do a first base-course year in general science, studying mathematics, physics, biology, chemistry and statistics, and then three years of computer-related studies.

#### **English Language Problem**

Problems encountered were specifically related to the problem of language, particularly the knowledge of English amongst Thai students and academic staff. Whilst there seems to be a high proportion of university academic staff with PhD's, most of whom studied abroad in Australia, Britain or the USA and therefore had a good knowledge and fluency of English, many do not. Minimum requirement for a teaching academic is usually a Masters degree, and again some of those teachers had studied abroad. However, there is still a significant number of academic staff who do not speak or understand English. Also, even though most Thai

students learn English in lower school through to university, their fluency is minimal. They may have studied syntax and grammar for 12 years, an hour a week, but their opportunities and experience in face-to-face conversation is very limited, especially with native-English speakers. There is a cultural inhibition that prevents most students from engaging in conversation in English, for fear of being thought foolish – an unnecessary fear certainly, but very real.

#### **No Thai Language Textbooks on 'Agile'**

A significant problem brought about by this lack of knowledge and fluency in English is that most students have no ability to avail themselves of the published literature about agile development.

To the best of my knowledge, there is nothing published in the Thai language on the subject of agile software development, agile project management, or lean software development. Virtually all books, conference proceedings, journals, blogs etc., are published in English, and none in Thai.

#### **Cost of Textbooks on 'Agile'**

A further significant problem is the cost of books on agile development. In an education system where the monthly salary of a qualified and experienced university academic is anything from 15000Baht to 30000Baht (USD\$420 – USD\$850 in round figures), the cost of a book of about USD\$40, or 1300 – 1400Baht is frankly exorbitant for them. Books of this price are beyond the financial means of most students. (This of course does explain the often rampant copying of books, easily achieved at a cost of about 100Baht or so).

Whilst there are many technical books written in Thai on diverse IT subjects, such as Java programming, database, PHP, web development etc., it is a frequent practice for university academics to write their own course notes and 'text books' in Thai, specifically for the subjects that they are teaching. This results in a highly fragmented sector, with individual books at individual universities, and often little universal similarity in curriculum taught. Further, it was stated to me by a Professor of Computer Engineering that many of the books written by Thai authors, especially on more specialized subjects, contain an unfortunate amount of wrong material and facts. This view is not verified at this time, and the professional

opinion of this Professor is accepted, for the purpose of the discussion.

#### **4. A RESEARCH PROJECT – SPREADING THE WORD ON AGILE**

These interviews are a part of research being undertaken on the subject of agile software development and agile project management. The purpose of this research is to ascertain the real level of interest in agile development methods in universities and organisations in Asia and Australia, and to provide training, curriculum and research material in a number of languages, by way of a website and portal.

This will be achieved by four main research and development activities:

1. Research into higher education institutions' inclusion in curriculum of these methods, in Thailand, China and Australia.
2. Research into industry adoption and use of these methods, in Thailand, China and Australia.
3. Research into cultural factors influencing the adoption of agile and lean methods into practice or curriculum in Thailand, China and Australia, at the personal level, the corporate level, and the societal level.
4. Development of a multi-lingual website that will include research materials, reference materials and general information about these methods, together with teaching and learning materials to support 'agile' curriculum, in English, Thai and Chinese.

The recent visit to Thailand was the first move in this research project. This was accomplished by visiting 12 different universities, presenting seminars and classes to undergraduate and post-graduate students, and staff. An online questionnaire has also been created, in Thai, for both staff and students, which it is hoped will be accessed and completed by other universities.

The expectation was to ascertain the level of knowledge about agile software development and agile project management amongst academic staff involved in the teaching of systems analysis, system development, and software project management, and to ascertain whether or not agile soft-

ware development and agile project management was taught in any courses.

Universities visited included

- Mahasarakham University (eastern Thailand)
- Kohn Kaen University (eastern Thailand)
- Chiang Mai University (northern Thailand)
- Maejo University (Chiang Mai)
- Payap University (Chiang Mai)
- Rachipat University (Uttaradit)
- Naresuan University (Phitsanulok)
- Rajamaal University of Technology Lanna Phitsanulok Campus
- Chulalongkorn University, Bangkok
- King Mongkut's University of Technology Thonburi (North Bangkok)
- Kasetsart University (Bangkok)

Other discussions were held with academic staff from Srinakharinwirot University, Bangkok and Silpakorn University (Nakorn Pathom).

A presentation was attended that was made by the development manager of a prominent software development organisation, who has assembled a number of teams now considered to be adept at agile software development methods and project management, held at Software Park, Bangkok, sponsored by the Software Process Improvement Centre.

From these contacts, visits, discussions and interviews, there is now a reasonably good understanding of the 'state of play' in regard to agile software development methods and agile project management methods in Thailand; university curriculum and company adoption.

In summary, there is very little knowledge and understanding about agile software development methods and agile project management methods in Thailand.

#### **5. INTERVIEWS WITH ACADEMIC STAFF**

##### **The First Interview**

In the first interview, with a Professor of Computer Engineering at a major university, first impressions came from the fact that he has Tom Gilb's book (Gilb, 1988) and Scott Ambler's book (Ambler, 2003a), on his table,

and Kent Beck's book (Beck, 2004), amongst others, in his bookshelf.

This Professor in the School of Computer Engineering has been in academia for 10 years, but also has an active consulting practice to industry, especially in the area of software process improvement.

We discussed the acceptance of agile methods in Thailand, and he indicated that there are some software development companies using agile methods, but are following the practice without fully appreciating, understanding or implementing the underlying philosophies and implications of agile. (Extreme Programming (XP) is in use in some companies, but perhaps only about 50% of the precepts of XP are utilized.

Scrum is preferred where agile is implemented, and this is implemented at some large companies, such as Reuters, where CMMI Level 5 is instituted.

His comment was that he considered that Scrum shortens times for meetings, and accelerates the process.

In his teaching he has embedded some elements of agile development, without especially attempting to teach students the specific principles and theories underlying agile. He introduces Test Driven Development from the start of the programming course, and Test First principles, which he considers to be very successful. JUnit is used, his preference is CxxTest, and recently FIT, which will be used to a greater extent in the future.

He has found that organizations which employ his graduates are delighted that the graduates are so knowledgeable of the practices of Test Driven Development and Test First.

In his first year programming unit, he insists that his students develop CxxTest scripts, and student assignments are considered incomplete and inadequate if these scripts are not included, and cannot be successfully translated.

In his database unit, new requirements are 'sprung' on the students during the semester, to inculcate the notion that requirements change, that the software must be adaptable. The students are told 'Software evolves, requirements change'. Iterative planning is an embedded, practical concept with students being required to record their project activities and plans weekly, and to demonstrate their acceptance and under-

standing of the idea that 'Change is useful' ... it indicates learning on the part of the client, as well as being a potential money maker for the developer. Clients will be quoted on a change by change basis on the impact and cost of a requested change.

However, he made it clear that he personally embraces agile methods but within a well structured project approach. We discussed the problem of clients wanting up front estimates, and quotes, and he overcomes this problem wherever he can by having an initial requirements determination phase to gather enough information to be able to give a reasonably accurate estimate and quote.

This professor was questioned about the attitude, knowledge and practices of the other academics in his department, and he indicated that he is 'the odd man out' in his thinking, although agile methods are accepted as appropriate to teach by the other academics. They do not, however, teach them. He is also the senior academic, so is able to include this in his own teaching without obstruction or adverse comment. The other academics are able to follow him, where they accept that agile methods are better and therefore appropriate to teach, but he does not spend time trying to sell the idea.

He said that there are very few books on information systems written in Thai (and none on agile development). Most lecturers prepare their own workbooks and reference material and notes. He indicated that where there are Thai language books, they are often wrong in the material being presented, because the authors do not have a good knowledge of the subject area.

He agrees that it is a problem that many staff are 'recycled' students, without industry experience and deep knowledge. He refuses to employ academic staff without at least 2 years' industry experience, for this reason.

This Professor was studying and working in the US for an extensive period of some years. He acknowledges that there is a cultural difference, and he has changed his perspectives because of that. In the US communications and interaction between people is more direct and 'up-front', while in Thailand 'we don't like to speak out'. Managers might feel that their authority and status has been diminished. This results in only some

aspects of agile development being able to be introduced in the course, and into companies, although some companies 'love it'. It was suggested that perhaps the situation was a culture of 'obedience' rather than 'innovativeness', but he said 'conformity' is a better word to use.

A personal observation was that this does raise the matter of a 'cycle of conformity' within which it is very difficult to introduce agile methods into organizations. So where to start? It is my view that it must start at the university level and wait for the informed graduates to become the next generation of IS managers? But, there is still a barrier in Thai industry generally.

### **Subsequent Interviews**

Subsequent discussions and interviews were conducted with twenty three other academics, at 12 universities. Unfortunately, the first interview where the interviewee was clearly knowledgeable about agile methods, was never repeated.

From the interview notes taken at the time, here is a brief description of the various interviews that were conducted.

### **Second Interview**

This was conducted with a PhD graduate from Newcastle-on-Tyne University in England. This academic's expertise was in Industrial Engineering, who conducts software project management classes for Computer Science students. He is also involved in curriculum development for a Masters course, to be conducted in English, in manufacturing systems engineering and automation, which he admitted was 'not famous because Thai students want Thai language courses'. He includes scheduling practices such as PERT and CPM in his software project management course, with concepts such as Just-in-Time scheduling being a very small part of what he teaches. During discussion, he admitted that he realised that software projects were characteristically different to manufacturing scheduling, which were easy to estimate, unlike software projects. He also realises that production project planning is not appropriate for software projects, where his major expertise and interest was in production scheduling for warehouse inventory, primarily. He also reinforced the opinion that had been expressed previously that most curriculum publications are prepared by individual lecturers, and there are

few if any publications on agile development in the Thai language.

Needless to say, it was clear that agile project management methods and approaches were not included in the software project management courses that he conducted.

### **Interview #3**

This was conducted with a senior teaching academic who holds a PhD from Leeds University in England, whose primary teaching interest is in networking and network security. His personal non-academic experience included working with companies as a consultant on logistics and warehousing systems. He personally had little knowledge of agile development methods, but admitted that, from what he knew and understood, agile development methods were superior to the traditional plan-oriented approaches. His approach to systems projects is 'not 100% SDLC, but not agile either'. He agreed that it was just not possible for clients to state all their requirements up front, but usually undertakes an analysis phase to gain an overview of requirements before quoting. He is also fully aware of the fact that client's 'keep changing their mind during the development phases'. His approach includes development by phases, where each phase is about 100,000Baht worth of development. This equates to about four man-months of development, given that a competent and experienced programmer in Thailand can earn about 25,000Baht per month.

Again, it was obvious that agile systems development and agile project management methods are not included in the curriculum in any courses at this university.

### **Three More Interviews – Same University**

The next teaching academic interviewed was currently responsible for the development of a new degree; a Bachelor of Science (Computer Software Development). This is in a relatively new department called the Department of New Media, which is in the Faculty of Informatics. This faculty encompasses many aspects of information processing, including Information Sciences, Information Retrieval and Analysis, as well as the more traditional Computer Science, Information Technology and Information Systems. This arrangement is impressive, given the clear acknowledgement of the now tightly knit relationship amongst and between these

disciplines. The creation of the Department of New Media, which includes courses in 2D and 3D Graphics processing, and Games Development, clearly indicated the contemporary and forward-looking culture of the Faculty. But with the obvious exception of the inclusion of agile methods in any curriculum! Even in the development of a new degree course in Computer Software Development, currently underway, agile development methods are not included. Various subjects to be included were Object-Oriented Programming, Software Requirements and Specification, Systems Analysis using UML, and Database design and Management; presumably all based on traditional structured approaches.

A further interview was conducted with a young, newly graduated academic, who had been educated at a Thai university. He had no knowledge whatever of the term 'agile development', nor any of the 'component methods', such as iterative development, incremental development or even software prototyping.

Similarly a third member of that faculty, whose level of English was very sparse. This academic did know about agile development because he had attended lectures conducted in 2006, when one author was a teacher at a prominent Thai university, and included seminars and classes in agile development for Masters students, of which he was one. Unfortunately his interest in incorporating agile methods was very limited, and he certainly did not teach anything about this currently.

### **Interviews at a 'Top 5' University**

Four interviews were conducted at this university, and found some very interesting, and quite contradictory facts were revealed. One Assistant Professor, who teaches Software Engineering and Principles of Programming Languages in C#, to Masters students, includes concepts of Extreme Programming in these courses. He also includes this in the undergraduate course in Object-Oriented programming. The two major aspects of XP that he teaches are iterative development and timeboxing. There is some enthusiasm about agile methods in his Masters classes. Other concepts of Test-Driven Development, and the use of test harnesses such as a small amount of JUnit are in-

cluded, but Continuous Integration concepts are not. He agrees with the principles and practices of agile, whilst at the same time not including most of those in his courses. He uses Larman's book (Larman, 2004, *Applying UML and Patterns*) as his text.

Interviews with two other teaching academics in Computer Science were quite short, because they immediately indicated that they knew nothing about agile development, did not know the term, and did not include any aspects of agile in their teaching. In discussion rather contradictory understandings by them were revealed, such as 'it is impossible to find out all of the requirements up front', and 'clients always change their minds and ask for changes and new things during the development activity'. However, this view apparently did not deter them from teaching the 'waterfall' method, where all requirements are ostensibly gathered up front, in the initial phase, and change is to be deterred and controlled as much as possible in the interest of keeping to 'the plan'.

This contradiction was found in most of the interviews that were conducted.

### **A Rare Find – An Application Generator**

The final interview that was conducted at this university revealed that this teacher had developed, over a number of years, a quite excellent application generator, for the creation of web sites, in PHP, with MySQL as the associated DBMS. An enormous amount of coding and development work had gone into this product, and it has been used internally in the university to develop some browser-based systems for other faculties. It was an impressive effort. This author having used application generators since 1987, considers himself to have significant experience and understanding of this type of software, and this product was indeed considered to be quite impressive. However, the developer had made no attempt to commercialise the software, nor was it used in any of his teaching.

The question was asked 'Did you specify this product in full at the beginning, and then develop it exactly according to that specification?'. The answer was 'No'. He explained to me how the idea had grown and developed (perhaps evolved, emerged) over time, and he had adapted and modified it as new ideas came to him, and he had learned from using it. He looked somewhat perplexed



when he was asked why, then, did he still continue to teach students that this was not the 'proper' way to develop systems? Why did he not incorporate his hard-earned personal knowledge in his teaching, and teach his students reality, instead of a 30 year-old and apparently clearly flawed methodology?

### Summary of Further Interviews

In the interests of brevity, and given the sameness of the information gathered at most of the further interviews that I undertook, a few comments of the noted comments during those various interviews are listed, without further discussion.

- I have some knowledge of agile development, but I do not include it in my teaching at all.
- SDLC and DFD's are taught in all of our subjects, but not anything about agile.
- There are no text books in Thai on agile
- There is a big problem teaching 'agile' because there are no teachers with experience or knowledge of agile.
- We have a software testing subject, but we do not include Test-Driven development, Continuous Testing, Continuous Integration and other topics in that
- I have heard about (agile) but know nothing about what it is.
- I have never heard of it.
- I learned about it in Australia when I was doing my Masters, but I only have a vague memory of it now.
- I did study agile development when I was doing a software engineering Masters at (a Bangkok university).
- The professor at that university (in Bangkok) focused on agile project management.
- We do not include any of that in our curriculum.
- (three interviews) ... on a scale of 0-5, 0 is our level of knowledge of agile.
- We include some concepts of iterative development, but nothing about agile development.
- We would be interested in including it in our future curriculum, but we need much more knowledge about it first.
- I cannot see the process of agile development. When I read about it, it talks

about adaptive development, but I cannot see the 'real' agile method.

### Overall Impressions

Only one of the academics who was interviewed was competent or especially knowledgeable about agile development. A few admitted that they had 'learned something about agile' in their Masters studies, in Australia, and some in the US, but mostly, the Thai-educated academics had not heard of the term. It is apparent that agile software development methods and agile project management methods are just not part of university curriculum in Thailand. Given that I interviewed staff at 5 universities that are considered to be either 'prominent' or are in what are reputed to be the 'Top 5' universities in Thailand, this came as somewhat of a disappointment.

What it did highlight was the problem of the current generation of teaching academics being the previous generation of students, often at the same university at which they were now teaching, and in the same Faculty. It was obvious that many of these new, and often reasonably inexperienced teaching academics, were teaching what they had been taught, and not teaching what had been missing from their undergraduate and post-graduate courses (University academics in Thailand always have at least a Masters degree, and a high proportion have PhD's, many earned from universities in Australia, the USA and Britain). Comments by interviewees implied that you cannot teach what you have never been taught, or studied, and how can we teach it if we know nothing about it and there is nothing written in Thai on the subject.

There are a number of important points that I perceived in many of these interviews; common points expressed or implied by most of the academics interviewed.

### Breaking the 'Ignorance' Cycle

The question of why Thai academics were not aware of agile methods was discussed. Of primary significance in this is the lack of availability of published material in the Thai language. To my best knowledge, probably 90%-95% of research publications, journals, books, conference proceedings etc. are written in English, and none is written in Thai. (I believe that there may be 2 or 3 books translated into Chinese ... but that is another research journey).

Obviously, for those teaching academics who did not learn anything about agile methods in their courses, it is therefore very difficult for them to find information on agile development, especially if their level of English is poor. (Some of the teaching academics, especially those who studied at Thai universities, spoke little to no English, and the interviews were conducted through an interpreter). Therefore the cycle of exclusion of agile development from university curriculum continues.

### "Oh, You Mean Iterative Development (or 'The Spiral Model')"

While most of the interviewees did not know the term 'agile development', after I explained some of the characteristics of it to them, some of them responded in this manner "Oh, you mean Iterative Development. Yes, I teach students that". However, this was rare, and only seemed to be said when that person was using a textbook that had some information about that in it. Two academics specifically referred to Boehm's Spiral Model.

One academic uses Larman's book (Larman, 2004) in his classes, and does include some discussion of agile development, supported by the content of that book.

### "But I can't see the process in Agile"

One person interviewed made this comment, but it was also implied in what some others said, too. There seems to be a perception, once some of the basics of agile development are explained, that it is too simple to be taken seriously. A high level of ceremony is equated with being a serious development approach.

Whenever, during a seminar or when I was presenting a class, when the table of Agile Values is displayed, there is always the argument that "Oh, you can't ignore documentation ... you need documentation" or "But what if the customer can't be continually involved". Adherence to the project plan was also seen as a major factor in the development project being 'properly' managed.

Another obstacle to understanding about agile development was always the question of estimating, and up-front quoting of project costs. It was always interesting to see that the estimating errors of 200%, 300%, up to 700% in some of the famous development disasters, presumably undertaken using a traditional, plan-driven approach, with de-

velopment taking place over the course of many years, were never questioned. The problems of estimating were often seen as being of significance when considering the efficacy of agile approaches, but were not seen as a serious undermining of the traditional approaches.

Some examples of these 'famous development disasters' include the relatively recent Australian Customs Service. In 2005, the Australian Customs Cargo IS was launched: The original cost estimate was \$33 million in 2000, but the cost by 2005 was \$250 million, system delivered 3 years late (supposed to be 2002). It failed substantially and caused major problems and costs for Australian importers and exporters (The Australian, 2005).

In 2002 the Canadian government discovered that its national gun registry project went 500 times over budget: Originally estimated at \$2 million, the agency spent \$1 billion before it got caught (Ambler, 2003b).

Other infamous project development failures include (but are by no means all):

- 1980's: Bank of America (USD\$70 million + substantial lost business)
- 1992 '*...Westpac's (Banking Corporation) highly ambitious CS90 project was described as 'under review' after five years of development and an investment believed to be \$200-\$300 million....CS90 is no longer a project*' (Source Plunkett, S., *Banks Pay for the Wasteful Years, Business Review Weekly, Feb.21, 1992*)
- 1993: London Stock Exchange share settlement system (terminated after the expenditure of £75 million, and an additional £400 million in collateral expenditure in the broking industry)
- 2002 : State of Florida Child Welfare System: original cost estimate in 1990, \$32 million, delivery by 1998. Cost by 2002 \$170 million, system predicted to be delivered by 2005 at a cost of \$230 million. (Source: Poppendieck & Poppendieck, *Lean Software Development: An Agile Toolkit*, Addison-Wesley, 2003)

This interesting phenomenon has been seen in many places. Agile development was questioned as to its viability, usefulness and appropriateness in a variety of ways, but mostly the same questions were as appropriate to be asked of the traditional approaches. Also, questions have been asked

which, frankly, were just entirely irrelevant, and seemed to be asked in almost a desperate attempt to 'poke holes' in the agile approaches. For example, 'But what if programmers refuse to follow the standards?', and 'But what if the client, late in the project, wants to change from, say a Windows platform to a Linux platform'. The reader is left to formulate an answer to these two questions, and which are sure to be very similar that those answers made at the time. Just please remember that public seminars were being presented at the time, and tactful responses were demanded.

### **Why Don't Thai Universities Teach Agile?**

In all of the interviews conducted, once it had been established that agile development was not included in the curriculum at that university, the direct question was asked; Why not? Why is agile not included in your curriculum?

The one expected answer was Because organisations don't use it, so why should we teach it? It would then have been a matter of the perceived relevance of the curriculum. However, this answer was never given, presumably because that answer was pre-stalled by the usual answer of 'We don't know anything about agile development'.

It also seemed that where one teacher was including some element of agile in their teaching, this was not in any way shared by other teachers in the same department. It was always a very individual effort. This was also exacerbated by the situation where many teachers wrote their own course book, and this was not shared with others, nor was it used in other universities, as a published textbook may have been.

### **6. MOTIVATION FOR THE RESEARCH, PERSONAL AND PROFESSIONAL**

I have studied, and practiced, aspects of agile software development, and agile project management, for about 25 years. My prior qualifications and experience before entering the IS industry were in corporate management, corporate law, and accounting. It seems that this has enabled me to view the software development process much more from a business perspective, as distinct from the apparent view of the technically trained practitioners, graduates from Computer Science or Software Engineering courses. Software development has been viewed as a

human activity, not as a technical activity. Significant efforts have been made for more than 20 years, to have software prototyping, iterative and incremental development methods incorporated into university curriculum. This endeavour has substantially failed.

It seems that this is a major failing of tertiary education institutions which present degree and post-graduate courses in computer science, information systems, information technology, business systems development; call it what you will.

To date, there are nearly 90 user groups listed on the Agile Alliance web site ([www.agilealliance.org/show/1641](http://www.agilealliance.org/show/1641)). There are at least 11 books written that I have grouped under the heading of Agile Software Development, General; 5 books on Lean Software Development; 7 on Agile Project Management; 17 on Agile Software Engineering; 22 on Extreme Programming. There are 5 books on associated management disciplines, such as The Toyota Way (Liker, 1997). There are many other books that discuss matters such as Lean Product Development, Lean Six Sigma, and so forth. There are three major international conferences on agile development, and other international conferences on lean manufacturing and associated subjects, which are considered relevant to our study of agile development and agile project management. A search of the Internet with the key phrase 'agile software development' reveals 3,340,000 'hits'; 'lean software development' returns 2,200,00 'hits'; 'agile project management' 2,150,000 'hits'.

There are many hundreds, if not thousands, of articles that have been published since the mid-1970's on iterative development, incremental development, rapid application development, evolutionary development, software prototyping. IBM now publishes a regular e-magazine (The Rational Edge) which is comprised substantially of articles on agile development and project management. There are numerous other subscription publications, from Dr. Dobbs, from Methods and Tools, and others.

The question must be asked How can universities and colleges NOT include this subject matter in their courses? How can universities and colleges ignore such a long-standing and well published area of software development practice? How can they justify

almost totally ignoring what must be seen as a valid, and rapidly expanding, body of knowledge and practice that has actually been in play for 30 years?

The term 'agile' has now been in existence since 2001; more than 6 years. Why have university academics not heard of this term?

#### **Economic Justification**

Research has shown that, over the past 30 years, literally billions of dollars have been wasted on failed system development projects. This was money thrown away on projects that failed to be delivered, or when delivered, failed to be useful. It does not include the collateral damage of business losses, and loss of business, caused by software system failure.

If the practice of software system development can be improved to the extent even of a reduction of 30% of those losses (a totally arbitrary figure for my purpose), then the cost savings, and savings on wasted development expenditure, will be enormous. We are not talking a trivial amount of millions here.

#### **The Outsourcing Industry – from both sides**

During his attendance at the Agile Development Conference in 2005, in Denver, Colorado, one author became aware of a movement that is recalled as 'Bring it Back Home' or similar. This was a movement in the US to bring back outsourced work to the local IT industry. One train of thought in this was that founding the decision to outsource on a costing model based on agile development, rather than on the traditional phased, plan-driven approach, would alter the economics considerably, in favour of retaining the work in the US.

Another recent news item about the outsourcing industry in India was that that industry sector in India was now finding it difficult to recruit staff because of the burgeoning demand for IT/IS staff that could no longer be met locally. Indian firms were, in their turn, seeking to outsource their development work. The question arose Could Thailand benefit from this situation? if they were properly prepared for it, and were able to be competitive. Thailand has an excellent education system, and a substantial pool of well-trained and skilled technical people; except for this unfortunate and widespread lack of education about agile development

methods. It seems to me that the adoption of such approaches may give budding Thai outsourcing consultancies a competitive edge. This thinking was reinforced by a seminar presented at Software Park in Bangkok, through the Software Process Improvement Centre. The presenter of that seminar was the development manager at a major development consultancy in Thailand, where he had currently developed seven project teams now skilled and experienced in agile methods. This had been done, it appeared, to meet exactly the demand for outsourcing services that I had contemplated.

#### **5. CONTRIBUTING TO THE SOLUTION**

As indicated above, it is the intention of the authors to promote the inclusion of agile development as a significant topic in university curriculum in Thailand.

This will be achieved by communicating with interested academics in Thailand, and creating a community of interested academics in Thailand.

A website and curriculum portal, in English, Thai, and also Chinese, dedicated to research and publications on the subject of agile software development, and agile project management, will be developed.

It is also intended to have some leading books on these subjects translated into the Thai language.

It is early days yet, but these initiatives will support the spread of information to Thai academics, and support their efforts to include agile in their curriculum.

This research and promotion activity will be extended to China, with a similar survey and similar fact-finding tour of prominent universities there.

In addition, the second arm of my strategy is to contact software development organisations and promote agile development thinking and possibly joint research activities between such organisations and universities.

#### **Language and Cultural Issues**

Why are 35% of the Agile User Groups in the USA, and 35% in Europe? (Calculated from the list of 90 Agile User groups there are nearly 90 user groups published on the Agile Alliance web site (Agile Alliance, 2007). There are only 3 User Groups listed as being in Australia, none in Thailand, and 1 in China and 1 in Japan. This implies to me that outside of the USA, and Europe, there is very

little interest in agile development. Why is this? Is it because of the lack of information in English? If so, this does not explain the lack of interest in Australia, where, of course, English is the native language.

Is it a cultural matter, where the organisational and personal culture in Thailand is against the adoption of agile? Two views on this were expressed during the interviews. One was that graduates going into organisations are perhaps too subject to the culture of conformity, so they do not attempt to influence their new work environment. A somewhat opposite view was that agile development, with its culture of 'welcome change' suited the Thai temperament and culture, where negotiation and willingness to oblige are part of the culture. This of course is not necessarily the same in Australia, where there is, it is thought, a similar lack of adoption of agile methods in universities and organisations.

It is intended to try to find out more about this in further research activity and study, in the future.

## 6. CONCLUSION

Whilst I did find there is a certain level of understanding and knowledge about agile development amongst Thai academics, and some inclusion of this in Thai university curriculum, this can only be seen as very meagre indeed. It is safe to say that, as a general and widespread rule, there is little to no knowledge of agile software development methods, and agile project management methods, amongst Thai IS / IT or Computer Science teaching academics. Similarly, I can safely say that these topics are not included in curriculum in general. What is included certainly does not encompass a comprehensive coverage of the subject area.

I also perceived a nascent interest in the topics, once I had had the opportunity to present a class or seminar. At one university, three Masters students were sufficiently interested in the topic that they chose a dissertation topic based on some aspect of agile development.

Finally, what is obvious is that the lack of reading and learning material in the Thai language is a major constraint.

It is hoped that contributions to overcoming that problem can be made, by the authors.

## REFERENCES

- Agile Alliance (2007),  
[www.agilealliance.org/show/1641](http://www.agilealliance.org/show/1641)
- Agile Manifesto (2008),  
<http://agilemanifesto.org/>
- Alhir, Sinan Si, "The Agile Unified Process (AUP)", [home.comcast.net/~salhir/TheAgileUnifiedProcess.PDF](http://home.comcast.net/~salhir/TheAgileUnifiedProcess.PDF), Accessed March 1st, 2007
- Ambler, Scott (2003a), *Agile Database Techniques: Effective Strategies for the Agile Software Developer*, Wiley Application Development Series
- Ambler, Scott (2003b), "Something's Gotta Give", *Dr Dobbs Architecture & Design*, March 1st, 2003,  
<http://www.ddj.com/architect/184414962>
- Ambler, Scott, (2007), "Agile Has Crossed the Chasm", July, [www.ddj.com/dept/architect/200001986](http://www.ddj.com/dept/architect/200001986)
- Barker, Garry (2005), "New Customs IT cargo system fails to deliver", *Technology Editor, The AGE newspaper*, [www.theage.com.au/news/national/new-customs-it-cargo-system-fails-to-deliver/2005/10/20/1129775901865.html](http://www.theage.com.au/news/national/new-customs-it-cargo-system-fails-to-deliver/2005/10/20/1129775901865.html) accessed 21st October, 2005
- Beck, Kent (2004), *Extreme Programming Explained: Embrace Change*, 2nd Edition, Addison-Wesley, 2nd edition, ISBN 0321278658)
- Boehm, Barry (1986), "A Spiral Model of Software Development and Enhancement", *ACM SIGSOFT Software Engineering Notes*, August, 1986.
- Boehm, Barry, "A Spiral Model of Software Development and Enhancement", *IEEE Computer*, vol.21, #5, May 1988, pp 61-72.
- Cockburn, Alistair, *Agile Software Development: The Cooperative Game*, Addison-Wesley, 2006, 2nd edition, ISBN 0321482751
- Cockburn, Alistair, *Crystal Clear: A Human-Powered Methodology for Small Teams (The Agile Software Development Series)*, Addison-Wesley, 2004, ISBN 0201699478
- Evans, Gary K, (2002), "The CASE (Tool) for Agility: Rational XDE", *The Rational Edge*,

- September, 2002,  
[www.therationaledge.com/content/sep\\_02/f\\_caseForAgility\\_ge.jsp](http://www.therationaledge.com/content/sep_02/f_caseForAgility_ge.jsp)
- Evans, Ian (2006), "Agile Delivery at British Telecom, Methods & Tools", Summer 2006:20, [www.methodsandtools.com/mt/download.php?summer06](http://www.methodsandtools.com/mt/download.php?summer06)
- Fowler, Martin (2003),  
[www.martinfowler.com/articles/evodb.html](http://www.martinfowler.com/articles/evodb.html), accessed May 19th, 2005
- Gilb, T. (1988), Principles Of Software Engineering Management, Addison-Wesley
- Highsmith, Jim, "Retiring Lifecycle Dinosaurs- Using Adaptive Software Development to meet the challenges of a high-speed, high-change environment", Software Testing and Quality Engineering, [www.stqemagazine.com](http://www.stqemagazine.com), July/August 2000.
- Larman, Craig (2004) Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development, 3rd Edition, Prentice-Hall.
- Liker, Jeffrey K., Becoming Lean: Inside Stories of U.S. Manufacturers, Productivity Press, 1997 ISBN: 978-1-56327-173-1
- Mahanti, A. (2006). "Challenges in enterprise adoption of agile methods - A survey". Journal of Computing and Information Technology 14(3): 197-206.
- Martin, J. Rapid Application Development, Maxwell McMillan, 1991
- Melnik, G. and F. Maurer (2005). "Agile methods: A cross-program investigation of student's perceptions of agile methods". Proceedings of the 27th international conference on Software engineering ICSE'05, ACM Press, IEEE Computer Society.
- Orr, Ken (2002), "Agile Requirements", Agile Project Management Advisory Service, Executive Report, Vol. 3, No. 12, Cutter Corporation
- Palmer, Stephen R. & John M. Felsing (2002), A Practical Guide to Feature-Driven Development (The Coad Series), Prentice Hall, , 2002, ISBN 0130676152
- Poppendieck, Mary & Tom Poppendieck, Lean Software Development: An Agile Toolkit, Addison-Wesley, 2003, ISBN 0321150783
- Poppendieck, Mary & Tom Poppendieck, Implementing Lean Software Development: From Concept to Cash, Addison-Wesley, 2006, ISBN 0321437381
- Qumer A., B. Henderson-Sellers, (2007), "An Evaluation of the Degree of Agility in Six Agile Methods and its Applicability for Method Engineering", Information and Software Technology, 2007
- Schwaber, Ken & Mike Beedle, Agile Software Development with Scrum, Prentice Hall, 2001, ISBN 0130676349
- Shine Technologies Pty Ltd, (2002), "AGILE METHODOLOGIES - Survey Results", [www.shinotech.com/agile\\_survey.jsp](http://www.shinotech.com/agile_survey.jsp)
- Solon R. (2002), "Benchmarking the ROI for Software Process Improvement", The DoD SoftwareTech News, Nov.2002, USA DoD October 21, 2005
- Stapleton, Jennifer (2003), DSDM: Business Focused Development, DSDM Consortium, Second Edition, 2003
- Taylor, Andrew (2000), "IT projects: sink or swim", The Computer Bulletin, Volume 42, Number 1, January 2000, pp. 24-26(3), Oxford University Press
- VersionOne and APLN, 2nd Annual Survey "The State of Agile Development", conducted June-July, 2007