Software Qual J (2009) 17:125–150
DOI 10.1007/s11219-008-9066-3

An evaluation of the utility of web development methods

Sheridan Jeary · Keith Phalp · Jonathan Vincent

Published online: 10 January 2009

Abstract Although many web development methods exist, they are rarely used by practitioners. The work reported here, seeks to explain why this might be so, and suggests that, for many, the perceived benefits may be outweighed by the difficulty or effort required to learn the method. In attempting to gauge the utility of methods the authors undertook a 2-year study of 23 small web development projects, attempting to use a range of published (academic) methods. Of the 23 projects we found only one case where the developer continued to use an academic web development method throughout the lifecycle. The ability to understand a method and/or its techniques was repeatedly cited as the reason for its abandonment. Our findings also indicate a number of key areas, relating to terminology, completeness, and guidance, where existing methods may be failing their intended users. In attempting to further our understanding of web development methods we completed a comprehensive survey of web development methods; covering 52 web development methods, encompassing a range of different research communities, and drawing upon 63 different sources. Our findings here shed some light upon the confusion of methods for the would-be user. In summary, the findings are that, although there is much of value in a variety of methods, method choice is somewhat bewildering for the newcomer to the field, and many methods are incomplete in some dimension. By providing this work we hope to go some way towards supporting the software engineering community, in both academia and industry, in their understanding of the quality issues that exist with the take up and use of web development methods.

Keywords Web development · Web development methods · Web development projects

S. Jeary (✉) · K. Phalp · J. Vincent
Software Systems Research Centre, Bournemouth University, Fern Barrow, Poole BH12 5BB, UK
e-mail: sjeary@bournemouth.ac.uk
URL: www.sosym.co.uk

K. Phalp
e-mail: kphalp@bournemouth.ac.uk

J. Vincent
e-mail: jvincent@bournemouth.ac.uk

1 Introduction

The premise of web development methods, for example (Gnaho 2000; De Troyer 2001; Schwabe and Rossi 1998; Fraternali and Paolini 2000), is that to build successful, 'good quality', complex web based systems, web developers need to adopt a disciplined and systematic approach (Murugesan and Ginige 2005). However, a recent survey (Lang and Fitzgerald 2005) of web developers found that only 2% of respondents had ever used a web development method. Hence, there is a pressing need to understand why developers do not use web development methods, so that such methods can be adapted or developed to meet the specific requirements of practitioners in the web development industry. Indeed, given that it is argued, within the wider systems development community, that there is already an "*overabundance of…methods that are arcane, impractical and unworkable*" (Lang and Fitzgerald 2005, p. 74), it is reasonable to suspect that web development methods may suffer similar problems. Furthermore, in a domain known to have particularly short development time scales and development teams with mixed (often non-IT) backgrounds, it may be that existing methods, which often assume a particular mind-set, are inappropriate for many web developers.

In exploring these issues further, we first consider the problem of the relatively poor take up of web development methods. In Sect. 3, we then attempt to ascertain the extent to which web development differs from traditional software development, introducing a categorisation by domain of interest. The empirical study is then introduced in Sect. 4, and we outline some of our findings in Sect. 5. Having completed the study our motivation is to investigate the available web development methods and Sect. 6 discusses the review process and sources. Section 7 considers different perspectives on web development and various views on requirements for web development methods are explored. In Sect. 8, a different classification of methods is given, which is intended to support the interests of practitioners, in their selection of methods, and which covers pragmatic concerns such as scope or modelling choice. Finally, Sect. 9 summarises our findings and offers some conclusions before detailing limitations and further work.

2 Use of web development methods

Glass (2001) states that web development is usually part of a much larger, complex systems development, often providing interfaces to other (e.g. legacy) systems. He suggests that, with the size and complexity of such developments, an understanding of software engineering principles is vital. Indeed, Pressman (1998) suggests that, although web development timescales may be shorter, web developments should be carried out in the same manner as traditional systems development projects.

In addition to calls for a disciplined 'software engineering' process, many authors cite ways in which web development differs from traditional software development, and suggest how these differences require further methodological capabilities. For example, Content Management Systems may have specific needs in terms of their information architecture and design, the navigation over that information, and its final presentation (Lang and Fitzgerald 2005).

It would appear, therefore, that web development should proceed with all the discipline of a traditional software engineering process, yet must also provide additional capabilities (to deal with some unique requirements), whilst also integrating with other (often legacy) systems. However, despite an apparently urgent need for web development methods, a

number of surveys have pointed out that they are rarely used, e.g. Barry and Lang (2001), Lang and Fitzgerald (2005), Taylor et al. (2002). In addition, a number of related problems with methods have been noted by other authors, for example:

- There are severe problems in adapting traditional software approaches to web development (Cartensen and Vogelsang 2001).
- Users of 'academic' methods are not web developers, but computer scientists interested in semantics and data analysis (Fernandez et al. 2000).
- Web development methods are not universally applicable and have not been sufficiently tested in live situations (Barry and Lang 2001).
- The technology and tools are immature, and web development reflects an immature discipline (Cartensen and Vogelsang 2001).
- There is a communication gap between academia and industry (Lang and Fitzgerald 2005).

However, despite awareness of these problems, there has been little empirical work to understand the unique requirements of web development methods. Therefore, in attempting to understand this issue, the following section describes particular features of web development, and suggests how these differences may have a bearing upon the suitability of existing methods.

3 Features of web development projects

Rather than simply listing a number of ways in which web development is unique, we attempt to classify development features according to three aspects (or perspectives) of the project; that is, the user, the developer, and the environment. Pertinent features of each category are outlined, followed by a summary of key points for each of these categories, and presented in Table 1.

3.1 Users

Although the user domain is considered important, in terms of requirements for, and use of, any proposed system, it does not appear to have a direct bearing upon methods, since the user is not (directly) part of the development process. Hence, whilst included in the table, the user domain is not discussed in detail here.

3.2 Developers

Lang and Fitzgerald (2005), in their survey directed at hypermedia designers (their definition of hypermedia includes web based systems), found that only 33% of respondents had a software development background, 26% were graphic designers, whilst the remaining 41% had similar proficiency levels, terming themselves information architects, web developers or web designers. Given that the developer completing web developments is therefore not a traditional software developer, then it is possible that they will require a different presentation of the development method. Important differences include: a variety of disciplinary emphases in the project team, immaturity and lack of experience, the developer suffering from cognitive overload and having to deal with rapidly changing technology and markets. For example, in our own experience, many web designers and developers may consider themselves proficient at web development (this can be purely

Table 1 Why web development may differ from traditional software development

Domain	Difference	Source
User domain	The user is divorced from the development and may not be engaged with the process	Holck and Clemmensen (2001)
	There is no traditional entry or exit point to or from web pages causing user disorientation	Barry and Lang (2001)
	The technology is more visible to the user	Lowe (2003)
	There is a high reliance on the user interface	Lowe (2003)
	The non-functional requirements may be primary	Holck and Clemmensen (2001)
	There is volatility in user requirements	Lowe (2003) and Kappel et al. (2006)
	Because of the unpredictable publishing environment, users blame the site when the Internet does not work	Powell et al. (1999)
Developer domain	The development team contains graphic designers, marketers, etc., with different disciplinary emphasis	Overmyer (2000), Holck and Clemmensen (2001), Kappel et al. (2006)
	Developers suffer from cognitive overload	Lowe (2003)
	Developers are not experienced. They are also difficult to use as a resource in estimation	Powell et al. (1999), Vora (1998)
	There is uncertainty in the developer domain	Lowe (2003)
	There are rapidly changing technology and tools	Cartensen and Vogelsang (2001), Lowe (2003)
	There is a lack of useful methods	Cartensen and Vogelsang (2001)
	Web development methods are written for computer scientists and not web developers	Fernandez et al. (2000)
Environment domain	The linkage between the business architecture and the technical design is tighter than conventional software systems	Lowe (2003)
	Web sites are like magazines, particularly in the design of the information structure	Overmyer (2000)
	The web development environment has shorter life cycles, aggressive release demands and shorter time frames for initial delivery	Overmyer (2000)
	Web development is hype-driven	Powell et al. (1999)
	The web development often changes the business model	Lowe (2003)
	Web development is immature	Powell et al. (1999)
	The development is of a sophisticated business model, and both a complex and component based information architecture	Lowe (2003)
	The web is a highly competitive market environment	Lowe (2003)
	Web development involves fine grained evolution and maintenance	Lowe (2003)

HTML coding) but they are often from a marketing or non-computing background, having learned web development and design either ‘on the job’ or from building a personal website.

Others also note the relative inexperience of the typical web developer; for example, the average web developer has between 2 and 5 years experience (Powell et al. 1999) and does not come from a computer science background (Lang and Fitzgerald 2005). Hence, whilst the proponents of various methods often assume a particular familiarity with software concepts, such as object oriented analysis and design, many web developers may find such notions far from intuitive.

3.3 Environment

The web development environment could also be a contributory factor in the non-use of development methods. The environment is often highly charged, with short time scales, fast changes to both technology and business models, and fierce competition.

4 The study

There are a number of ways in which one might attempt to gauge the difficulty of a method, including:

- Measuring the complexity of the constructs utilised.
- Measuring the difficulty of the constructs from a cognitive perspective.
- Measuring the difficulty of 'learning the method'.

There appears to be no work on the complexity of the constructs in web development methods, although some work has been done on assessing elements of the Unified Modelling Language (UML) (Siau and Cao 2003). Complexity is clearly very different to difficulty. In terms of the accessibility of methods, approaches such as Cognitive Dimensions appear to offer more potential, and have, for example, been applied to the assessment of elements of the UML (Cox 2000), but, similarly, we are not aware of web methods having been assessed in this manner. Within this paper we wanted to try and understand the process of learning and using a web method, rather than merely examine notational elements, and this required an empirical, as opposed to theoretical, approach. In doing so we attempt to add to a body of work that discusses learning theory with respect to education (e.g. Bigge 1982; Gagne 1977).

4.1 Motivations for the study

The motivation for this work came when arguing for the utility of web development methods to undergraduate students. As Griffiths et al. (2002) and others point out, education is neglected when methods are considered, and this we found true of web development methods. There is little evidence that the usability or otherwise of different web development methods has been considered. Nor has the way the developer would need to approach the learning process in order to understand their use. There is no list of assumptions made as to prerequisite qualifications. Whilst some methods have been formally tested on Masters students, e.g. DPWA (Uden 2002), and others may have been used as part of a degree course by their authors, e.g. WSDM (De Troyer and Leune 1998), there is no empirical evidence to show that methods have been tested in an industrial setting or are useful to web developers.

Arguably, a method that is understandable, learnable and easy to use, and which gives positive benefits to a development team, in terms of process improvements, would be

adopted by a number of organisations for web development. However, despite a time lapse of over 6 years in many cases, allowing the filtering of academic theories into practice, there is little evidence to show that the methods are being considered by industry. Hence, this study investigates whether web development methods are difficult to learn, and whether this may explain their relative lack of use.

4.2 Conduct of the study

The study ran over 2 years, using fourth year students on a BSc (Honours) Business Information Technology degree at a British university. The students spend the third year of their degree on placement within the computing industry. The degree is at the technical end of the Business IT spectrum.

It was felt that the students would have a similar experience level to novice web developers and they would have some knowledge of the development environment, but were not computer scientists. All students on the course study programming for 2 years (mainly Java with some PHP/MySQL), and database design and performance for 3 years, in addition to marketing, accounting and business systems. For their final year project the students have to complete a significant individual piece of development for an independent client in a timescale from early October to mid-March. The final year project is one-third of the students final year marks and thus is an important element of their study. The short time scale and the pressurised development environment would thus tend to mirror that of a web development in industry. Prior to the study, none of the students had heard of any web development methods before, but they were aware of the existence of 'formalised methods' as defined by Fitzgerald et al. (2002).

Each student was self-selecting in that they requested one of the authors as a project supervisor and were undertaking a web development project. Each student was interviewed and given details of three web development methods, which they could use for the development project if they wished. They were given an overview of each of the three methods along with relevant journal papers. The three web development methods were drawn, at random, from those of Table 2. These web development methods were chosen by the authors as, first, being within the scope of the students' understanding, and,

Table 2 List of web development methods

No	Name/description
1	WSDM: a user centered design method for web sites (De Troyer and Leune 1998)
2	RMM: a methodology for structured hypermedia design (Isakowitz et al. 1995)
3	ADWIS: the analysis and design of web-based Information Systems (Takahashi and Liang 1997)
4	FECWAD: a framework for effective commercial web application development (Lu and Yeung 1998)
5	Connallen: modelling web application design with UML (Conallen 1999)
6	DPWA: a design process for web applications (Uden 2002)
7	First year of study: SHDT/W3DT: the structured way of developing WWW sites (Bichler and Nusser 1996a, b) Second year of study: SOHDM: scenario-based object-oriented methodology for developing hypermedia systems (Lee et al. 1998)

second, having enough information available that some sense could be made of the contents.

The students made available their dissertations and evaluations for this research. In addition, some students completed follow up interviews. Thirteen students took part during the first year of the study, and ten in the second year. None were offered any reward for their participation.

5 The findings

Of the 23 students, 13 decided not to use a web development method at all, and used either a traditional method or the Systems Development Life Cycle. Eight students used part of a method; for example, two used RMM slice diagrams and another used scenario analysis, before abandoning the use of the method.

Only one student used a web development method (W3DT) for the whole process. He used a web site that was available at the time, and spoke to Bichler and Nusser via email who referred him to a book by Scharl (2000). He comments that "…the documentation was not freely available…which made the methodology more difficult to understand. Most literature on the methodology was in complex academic language which did not help the developer who was using the methodology for the first time…There was not enough emphasis on the collection of requirements".

Of the 13 students that did not use a web development method, most described the criteria of what they were looking for in a method. They found that the web development methods given to them did not match their criteria.

Comments included:

> …the methodology should make it clear what is required and what is optional, given the tight timeframe and resource available
> The available documentation does not mention the use of iteration
> …are both suitable for small projects but have no way of tracking progress due to lack of time management in the methods
> …it didn't…cover testing or anything like that, it stopped suddenly
> At the beginning of the project I didn't understand why I needed a methodology…I spent a lot of time researching a variety of methods…I did however spend too long in deciding which to use
> …[the method should] cover(s) the [whole] SDLC

One of the issues discussed was the difficulty of following academic papers:

> The most difficult thing about following…the method was the structure…different papers said different things and I didn't know [which to follow]
> I tried to stick to just trying to follow one paper but then you get to something you didn't know…they'd come out with this model and then they'd show you the next model they'd produced but didn't tell you how they produced it
> I decided to cut out aspects of [the method] and eventually decided to remove it…the main issue was in the time consumption and complexity of the techniques
> …the use of the Conceptual Navigation Model involved time consuming research through reasonably complicated literature
> …the main issue was with the time consumption [to understand] the complexity of the techniques within the method

> The design method is very confusing and would take time and experience to implement successfully
> …was also both confusing and possibly difficult to implement without the correct knowledge
> …although the method does provide a technique, Use cases will be used instead. The technique was difficult to understand

Another problem was with the depth of coverage of the method, particularly in the early stages of the lifecycle:

> …they failed to cover the aspects [of feasibility] that [the method] does
> …[the method] did not offer as much support in the early stages of the systems development life cycle
> …the method recommends gathering requirements before the user modelling stage, but did not recommend [a method]. By collecting requirements from a number of people with different interests in the website a good understanding of what the system should include was obtained. However where the interviews were held separately, requirements given by one of the clients were then later rejected by another
> …the approach taken concentrated on gathering requirements from the clients and ignored any requirements that the users may have for the website. I had assumed that by using the method the users requirements would be considered
> …the method did not cover the whole of the lifecycle

Students gave up the method because:

> …it wasn't helping, I couldn't see from the model I'd got at the last stage how it was going to make me a web site. It didn't fit
> …leaving me confused and not knowing how to use the models I had produced
> …the project would have been unnecessarily delayed…to produce more analysis and documentation…
> …[in trying to work out the process] I got confused…
> …used it for the navigation techniques…the outcome is basically commonsense and could have been achieved without the aid of a navigational model
> …[the method] may not have been the best choice of methodology. It concentrated solely on the navigation of the system and neglected the content

The terminology caused problems:

> There were things like attribute design and architecture design I didn't really understand… I understand what attributes are but in the way they were describing them I didn't see what I was going to get from it…I didn't get what they were trying to say

Or the depth of coverage of the method was sketchy in places:

> The lack of literature to support later stages left me with a navigational structure which I did not know how to convert into a website

Fourteen of the students voluntarily completed a Belbin (1981) test. The results showed that for the Completer Finisher personality, none were found to score highly. A Completer Finisher trait would be expected to show a tendency to use a method and complete it. However, 8 of these 14 students scored highly as Resource Investigators, which could

correlate to a personality type with a tendency toward Systems Analysis (Gifford et al. 2003).

Hence, despite a background in Computing, an understanding of common notational and software engineering concepts, and the motivation to complete a successful project, these participants found that the web development methods studied, did not, for the kinds of reasons stated, meet their needs.

In contrast, they were able to adopt a fairly pragmatic approach to method; adopting, adapting and synthesising typical software engineering notations within their projects.

To further explore the issues surrounding non-use, a comprehensive survey of web development methods has been completed. The survey is timely, and will enable both academics and practitioners to gain a wider appreciation of the multiplicity of methods that are now available.

6 Survey process

A comprehensive literature review should guide the direction of further research and provide the context for interpreting new findings (Harlen and Schlapp 1998). The review should state its purpose to allow the reader to be aware of the researchers' aims, and describe the information sources searched and the inclusion criteria used in selecting the cited papers (Weed 1997). This section provides such information.

The purpose of this review is to:

- Provide a review of the field of web development methods that will allow the reader to acquire knowledge in the field.
- Allow the identification of some of the gaps in current knowledge.
- Investigate the scope of coverage of individual development methods and the instructions for their use from a pragmatic perspective. That is, whether it is possible for a practitioner to use the methods as detailed, or whether they are written for computer scientists to investigate an interesting phenomenon within the method, as suggested in Fernandez et al. (2000).

The methods and their relevant publications are listed in Table 3.

The criteria for inclusion of a method in the review were that the paper either had to be clear that it contained a web development method, or:

- it had to refer to some part of the Systems Development Life Cycle (SDLC) and contain a framework or technique that would enable some part (or all) of web development
- or, it referred to a web modelling language or method
- or, it contained details of a web development tool that was part of a described development method.

As Lowe (2003) points out, the research literature is extremely fragmented, with few attempts to draw the work together into a cohesive picture. Hence, the quality of the journal or conference was not considered as part of the inclusion criteria because, despite the possible risk of including methods of differing academic quality, it was felt that it was more important to provide as complete a picture as possible.

Table 3 Methods and their publications

Conferences	Methods
Web engineering	Web-based information systems development (WISD) (Gnaho 2000), WebComposition (Gaedke and Graf 2000), rapid service development (RSD) (Janssen and Steen 2000), cross-functional evolutionary methodologies (CFEM) (Norton 2000), functional view of the hypermedia process model (HFPM) (Olsina et al. 2000), RDF/XML framework (Christodoulou et al. 2000), HERA (Houben et al. 2003), agile web engineering (AWE) (McDonald and Welland 2001a)
World wide web conference	Web site design method (WSDM) (De Troyer and Leune 1998), analysis and design of web-based information systems development (ADWIS) (Takahashi and Liang 1997)
International conference on database Technology	Hypertext Design method—lite (HDM-Lite) (Fraternali and Paolini 1998)
VLDB conference	Araneus (Atzeni et al. 1997)
European conference on information systems	Structured hypermedia design techniques (SHDT) (Bichler and Nusser 1996b)
Hypertext and hypermedia	WebML (Olsina 1997), relationship navigation analysis (RNA) (Yoo and Bieber 2000)
IEEE conference on multimedia Computing and Systems	JWeb (Bochicchio et al. 1999), AWARE (Bolchini and Paolini 2004)
SIGMOD	Strudel (Fernandez et al. 1997)
Hawaii international conference on system science	Enhanced object-relationship model (EORM) (Lange 1994), scenario-based object-oriented hypermedia design method (SOHDM) (Lee et al. 1998)
Workshop on software specification and design	Coda (Coda et al. 1998)
Human computer interaction (INTERACT)	Turbo-prototyping (Ghosh 1999)
Software engineering and knowledge engineering	WebML+ (Tongrungrojana and Lowe 2003), simple web method (SWM) (Griffiths et al. 2002)
Workshop on web-oriented software technology	UML-based web engineering (UWE) (Koch and Kraus 2002), object oriented web solution (OOWS) (Pastor et al. 2003)
E-commerce and web technologies	Resource description framework for web engineering (RDF/WE) (Kalpsing and Neumann 2000)
DEXA	ARIADNE (Diaz et al. 2001)
ACM symposium on applied computing	Web application rapid prototyping (WARP) (Bochicchio and Fiore 2004)
International conference on computer documentation	User centred design method (UCDM) (Fuccella 1997)
Journals	
ACM transactions on information systems	AutoWeb (Fraternali and Paolini 2000), hypermedia design method (HDM) (Garzotto et al. 1993)
Journal of database management	View-based hypermedia design method (VHDM) (Lee et al. 1999)
Communications of the ACM	Relationship management methodology (RMM) (Isakowitz et al. 1995), object oriented hypermedia design method (OOHDM) (Schwabe and Rossi 1995), (Nanard and Nanard 1995), UML+ (Conallen 1999)

Table 3 continued

Conferences	Methods
IEEE multimedia	Object oriented hypermedia (OO-H) (Gomez et al. 2001), design process for web applications (DPWA) (Uden 2002), JESSICA (Goeschka and Schranz 2001)
ACM transactions on internet technology	ARANEUS (Merialdo et al. 2003)
Internet research: electronic networking applications and policy	Framework for effective commercial web application development (FECWAD) (Lu and Yeung 1998)
European journal of information systems	Hypermedia application development tool (HADT) (Hatzopoulos et al. 1993)
Information and software technology	Scenario oriented hypermedia design method (SOHDM) (Lee et al. 2004)
Information systems journal	Web information systems development method (WISDM) (Vigden 2002)
IEEE internet computing	WebComposition (Gellerson & Gaedke 1999)
Books	World wide web design technique (W3DT) (Scharl 2000), UML+ (Conallen 1999) WebML (Ceri et al. 2003), hypermedia modeling technique (HMT) (Zoller 2001), user centred web design (UCWD) (McCracken and Wolfe 2004)
PhD thesis	UML-based web engineering UWE (Koch 2000)

7 Perspectives on web development

In this section, we consider issues that have shaped our attempts to understand and categorise methods. In structuring our survey we are informed by arguments from Sect. 2 that suggest that (a) web development has a different (non-Computer scientist) set of developers (b) web methods have poor industrial take up and (c) practitioners needs are not always the primary consideration in the development of methods.

In addition to calls for a disciplined 'software engineering' process, many authors cite ways in which web development differs from traditional software development, and suggest how these differences require further methodological capabilities. For example, Content Management Systems may have specific needs in terms of their information architecture and design, the navigation over that information, and its final presentation (Lang and Fitzgerald 2005).

It would appear, therefore, that web development should proceed with all the discipline of a traditional software engineering process, yet must also provide additional capabilities (to deal with some unique requirements), whilst also integrating with other (often legacy) systems. However, despite an apparently urgent need for web development methods, the conclusions of three separate papers, that have surveyed the practitioner community (Taylor et al. 2002; Lang and Fitzgerald 2005; Barry and Lang 2001), suggest that web development methods are not used in practice. Popular explanations, for such lack of use, include: that developers do not know that the methods exist because they are unaware of the academic literature (Barry and Lang 2001; Taylor et al. 2002), there are problems in adapting traditional approaches to web development (Cartensen and Vogelsang 2001), and the average web developer is inexperienced (Powell et al. 1999) and does not come from a computer science background. In contrast, users of academic methods are, typically, not web developers but computer scientists interested in semantics and data analysis, and in

reality practitioners rely upon the implied processes of Rapid Application Tools (Fernandez et al. 2000). Other reasons suggested are that web development methods are not universally applicable and have not been sufficiently tested in live situations (Barry and Lang 2001), and that the technology and tools are immature and that web development reflects an immature discipline (Cartensen and Vogelsang 2001). The web developer is unlikely to have formal computing qualifications and knowledge. Barry and Lang (2002) point out in their survey that it is not just difficulty in using or understanding formalised methods that is the inhibiting factor against their use, but the fact that they are too cumbersome.

There have only been two surveys published that indicate requirements for a web development method from a practitioner's perspective. Lowe and Eklund (2001) focus on the early part of the life cycle, in the area of client requirements. They find that there is a requirement to establish a client liaison role, such that the actors of the role understand both the technical aspects and the business domain. In addition, clients should understand their role in the development process. The business case should be clarified as early in the process as possible and there should be early evaluation of the design. Developers should accept that requirements will be flexible and should adopt effective configuration management tools and methods. McDonald and Welland (2001a) have a different emphasis in their survey, and find that a web engineering process should be able to cope with short development lifecycle times and with the development of data as well as software components and the inter-dependencies amongst them. The process should take into account the variety of the backgrounds within a developer team and should allow the team to work on a number of tasks in parallel. Their survey also found that practitioners require a focus on the analysis and evaluation stages of web development. In addition, they suggest that the process should also address requirements and testing, and ensure that the maintenance of the product is adequately covered.

8 Origins of web development methods by community

There have been no comprehensive surveys of web development methods, although a number of academics from differing communities have carried out a comparison of some of the methods, see for example Koch (1999), Montero et al. (2002) and Gu et al. (2002). It should be noted that none of the comparisons give any criteria for the selection of the methods that were used for the comparison. The papers, journals and books investigated within this paper consist of those listed in Table 3 and those now described; a total of 63 sources. Analysis of the methods compared shows that the most popularly considered methods to appear in papers are RMM (7), HDM (6), OOHDM (6), WSDM (4) and SOHDM (3). Whilst this could suggest that some methods are more widely known and disseminated within the research community, it could also be influenced by the length of time that they have been available (RMM since 1995, HDM 1993, OOHDM 1995, WSDM and SOHDM since 1998).

This survey examines the output of methods from the differing academic communities to attempt to categorise the motivation behind the methods (see Table 4). It is accepted that the categorisation may be flawed in some areas where there is no clarity of background in the individual papers, and some methods might fit equally well into more than one community.

The output from the different academic communities was analysed to determine the differing features that each community considers necessary for a web development

Table 4 Web development methods by academic community

Academic community	Methods
Database	HDM lite (Fraternali and Paolini 1998), STRUDEL (Fernandez et al. 1997), WARP (Bochicchio and Fiore 2004), Autoweb (Fraternali and Paolini 2000), ARANEUS (Merialdo et al. 2003), WebML (Ceri et al. 2003), Jessica (Goeschka and Schranz 2001)
Hypertext and hypermedia	SHDT (Bichler and Nusser 1996), W3DT (Scharl 2000), VHDM (Lee et al. 1999), RMM (Isakowitz et al. 1995), HDM (Garzotto et al. 1993), OOHDM (Schwabe and Rossi 1995), RNA (Yoo and Bieber 2000), ADWIS (Takahashi and Liang 1997), EORM (Lange 1994), HADT (Hatzopoulos et al. 1993), ARIADNE (Diaz et al. 2001), SOHDM (Lee et al. 1998), HERA (Houben et al. 2003), SchemaText (Kuhnke et al. 2000), WSDM (De Troyer and Leune 1998)
Object oriented analysis and design	OO-H (Gomez et al. 2001), UML+ (Conallen 1999), WebComposition (Gellerson and Gaedke 1999), OOWS (Pelechano et al. 2003)
Modelling/notation	WISD (Gnaho 2000), RSD (Janssen and Steen 2000), UWE (Koch 2000), FECWAD (Lu and Yeung 1998), HMT (Zoller 2001)
Multimedia	JWeb (Bochicchio et al. 1999), MATILDA (Lowe et al. 1996)
Information systems/software engineering methods	WISDM (Vigden 2002), AWE (McDonald and Welland 2001b), Turbo-prototyping (Ghosh 1999)
Human–computer interaction	Macweb (Nanard and Nanard 1995), UCDM (Fuccella 1997), URMDP (Alaa and Fitzgerald 2004), UCWD (McCracken and Wolfe 2004), DPWA (Uden 2002)
Other	CFEP (Norton 2000), WOOM (Coda et al. 1998), RDF/WE (Kalpsing and Neumann 2000), RDF/XML framework (Christodoulou et al. 2000), SWM (Griffiths et al. 2002), WebML+ (Tongrungrojana and Lowe 2003), AWARE (Bolchini and Paolini 2004), QEM (Olsina et al. 2000), HFPM (Olsina 1997)

method. Our initial thoughts were to synthesise these criteria, to produce a list of requirements for a web development method that may be applicable across community boundaries. However, this did not further our understanding of the methods. First, such a synthesis covered the very differences among communities which revealed insights into their perspective. Second, and more importantly, the practitioner requires methods suitable for their specific developments, and needs to assess methods according to other criteria, such as scope, focus, etc. Hence, the following section attempts to categorise methods in a manner that is likely to aid their selection (or rejection) for development, rather than simply categorise by a theoretical (or purely academic) dimension. Any attempt at methodology categorisation is recognized as a difficult problem (Glass 2004) and comparison of methods is also a complex issue, which has been the subject of much research over a number of years within the Information Systems community, as summarised by Avison and Fitzgerald (2006).

9 Classifications of web development methods

There can be seen to be a chasm between the methods that academics are developing, and even their thoughts on requirements for those methods (Gu et al. 2002) and the methods that practitioners require in commercial settings. It appears that either there is a lack of

understanding of research outcomes by practitioners, or as suggested above, that research outcomes are not yet suited to practical development (Lowe 2003) and the field has not yet matured enough to have a viable approach to applied research.

This section examines methods by adopting a variant of the approach used by Lee et al. (2004) which reflects the authors' practice-based approach. A practitioner would be likely to consider methods on the basis of their scope and this offers a logical way in which to structure a survey of methods. Other questions that a practitioner will raise when initially considering which method to adopt have also been used to structure the survey notably: the underlying modelling technique, the primary focus of the method, and its teachability or learnability.

9.1 Scope

One difficulty with disparate strands of research, from differing academic communities, is that different terminology is often used. Thus, it is not always easy to understand to which development phase an author may be referring. In addition, the majority of methods adopt different notations. These two factors contribute to making web development methods a particularly complex area to navigate. In addition, even the notion of a method has different connotations and meanings. Many methods such as ARANEUS (Mecca et al. 1998) and WebComposition (Gellerson and Gaedke 1999) concentrate on the modelling aspects alone, and whilst they may mention other parts of the lifecycle, they do not provide enough detail to allow implementation of those stages. The Modelling/Notation community believes that conceptual, logical and physical models are the route from requirements to implementation, and the greater the level of model detail the closer the method is to implementation. Further, a web application development is characterised by three major design dimensions, which the Web Engineering community has recognised should be kept separate. Structure, (describing the organisation of the information managed by the application), navigation (concerning the ability to access the information) and presentation (allowing the content to be produced for the user) (Fraternali 1999). The methods surveyed have been examined and categorised according to their scope of the Systems Development Life Cycle as discussed next.

9.1.1 Those methods that do not prescribe a particular methodological approach

For example, RSD (Janssen and Steen 2000) offers an integrated framework on two dimensions—business and service oriented models—but does not specify a particular methodological approach nor associated techniques. WebComposition (Gellerson et al. 1997) specifies a component model based on the design, implementation and maintenance of a web application. Although it discusses a number of approaches it has an open process model focussing on reuse. WISDM (Vigden 2002) covers requirements and the software model, but does not prescribe an approach, nor an implementation. In QEM (Olsina et al. 2000) the user of the method specifies which part of the product lifecycle they intend to investigate.

9.1.2 Those methods covering the full life cycle

WOOM (Coda et al. 1998), WSDM (De Troyer 2001), UWE (Koch 2000) using an iterative approach, and WebML (Ceri et al. 2003), cover the full lifecycle from

requirements to implementation, with some, or most of, the issues that need to be considered focussing on a modelling framework. OOHDM describes domain analysis, navigational design, abstract interface design and implementation. It has since evolved to encompass the full lifecycle (Guell et al. 2000). JWeb starts at the definition of an HDM schema and provides an environment to assist in the whole process, including implementation (Bochicchio et al. 1999). DPWA (Uden 2002) focuses on requirements (using Applied Cognitive Task Analysis (Militello and Hutton 1998)), usability requirements and Relational Navigation Analysis but does cover the whole lifecycle including maintenance. OOWS (Pelechano et al. 2003) covers requirements (use cases and scenarios) taken from the OO-Method (Pastor et al. 1998). It also covers conceptual modelling, navigational and presentation modelling, architecture design and implementation, adding patterns for presentation and services for architecture.

9.1.3 Those methods discussing a lifecycle approach but not covering some aspects in any detail

A number of methods, such as HERA (Houben et al. 2003), XWMF (Kalpsing and Neumann 2000) and VHDM (Lee et al. 1999), discuss a full lifecycle approach but do not explicitly cover feasibility, requirements, implementation or maintenance. ADWIS (Lu and Yeung 1998) does not explicitly cover them either, but adds Scenario Analysis. Similarly for SHDT/W3DT (Bichler and Nusser 1996), which does mention some techniques for requirements gathering, but not explicitly. SOHDM (Lee et al. 1998) covers construction but not feasibility, requirements gathering or maintenance. HFPM (Olsina 1997) describes the process of a hypermedia design method, functionally, using the concept of views. It outlines a full lifecycle approach but provides few details except in the main area of focus. RDF/XML Framework (Christodoulou et al. 2000) does not cover feasibility nor requirements, but does cover development and maintenance. SchemaText (Kuhnke et al. 2000) advocates analysis, design, implementation, test and maintenance but only provides an overview of document engineering techniques and navigation structures. SWM (Griffiths et al. 2002) covers the whole lifecycle but not in great depth, concentrating on an Integrated Project Support Environment for teaching.

9.1.4 Those methods not discussing a lifecycle approach and covering only part of the lifecycle

Methods that focus on the requirements aspect of the SDLC include RNA (Yoo and Bieber 2000) which describes a process for finding and modelling the links between information domains and CFEP (Norton 2000), which describes a current practitioner approach to web development and focuses on the requirements from a user and product viewpoint using Joint Application Development (Soltys and Crawford, Undated). AWARE (Bolchini and Paolini 2004) uses goals to assist in the identification of requirements and helps with content, interaction, navigation and presentation. UCDM covers audience definition and content identification and validation, highlighting usability aspects (Fuccella 1997).

Whilst UML+ (Conallen 1998) provides extensions to the UML notation to take account of different web page requirements on both client and server, the method used is the Unified Process (Kruchten 2000). OO-H (Gomez et al. 2001) gives details of requirements gathering using use case diagrams and a business class diagram. The navigation requirements are modelled from a class diagram and the top level Navigation Access Diagram is automatically generated. The scope of this method extends from

problem space to solution space, but with an emphasis on design and implementation. WISD (Gnaho 2000) concentrates on user modelling and navigational modelling. ARANEUS (Atzeni et al. 1998) starts at the database conceptual schema design and continues to page generation. HMT (Zoller 2001) covers design and does mention requirements. EORM describes an iterative design method (Lange 1994). Nanard and Nanard (1995) cover the design process as does ADM (Diaz et al. 2001). Finally, STRUDEL (Fernandez et al. 2000) is a web site implementation tool focussing on the management of different types of data from differing sources.

The scope of the methods is tabulated in Jeary (2005) which emphasises the mixed scope of the methods researched for this survey and the fact that many methods do not cover the SDLC from end to end.

9.2 Method approach

In this section, web development methods are classified according to the underlying modelling concept.

9.2.1 Methods taking an ER approach

A number of methods have been developed based on the Entity-Relationship (ER) model. Since ER models are widely used and understood, these methods have the advantage that a new data modelling language does not need to be learnt (Ceri et al. 2003), in addition to the associated other techniques. Although HDM is included, it differs from the ER approach, but is considered to be a mix between the ER model and the Dexter model (Garzotto et al. 1993) (Table 5).

9.2.2 Methods taking an object-oriented approach

Many of the methods surveyed in this paper adopt an object-oriented approach. Although some methods are based directly on the Unified Modelling Language (UML), others use either a proprietary notation or bespoke extensions to the UML (Table 6).

9.2.3 Methods based on neither the entity relationship nor object-oriented approaches

A number of methods do not advocate a specific approach to modelling. For some, this is due to their focus on early stages of the lifecycle, which does not necessitate a modelling approach. RNA (Yoo and Bieber 2000) only considers the analysis of data relationships and hence is not based on a data model. FECWAD (Lu and Yeung 1998) provides a

Table 5 Methods taking an ER approach

HDM (Garzotto et al. 1993)	ARANEUS (Atzeni et al. 1998)	HMT (Zoller 2001)
VHDM (Lee et al. 1999)	WebML (Ceri et al. 2003)	JESSICA (Goeschka and Schranz 2001)
RMM (Isakowitz et al. 1995)	Hera (Houben et al. 2003)	SHDT and W3DT (Bichler and Nusser 1996a, b)
HMT (Zoller 2001)	ADWIS (Lu and Yeung 1998)	

Table 6 Methods taking an object-oriented approach

WISD (Gnaho 2000)	OOHDM (Schwabe and Rossi 1995)	DPWA (Uden 2002)
WSDM (De Troyer and Leune 1998; De Troyer 2001)	UML+ (Conallen 1998)	UWE* (Koch 2000)
HFPM (Olsina 1997)	WebComposition (Gellerson et al. 1997)	JESSICA (Goeschka and Schranz 2001)
OO-H (Gomez et al. 2001)	WOOM (Coda et al. 1998)	EORM (Lange 1994)
Nanard & Nanard (Nanard and Nanard 1995)	XWMF (Kalpsing and Neumann 2000)	SOHDM (Lee et al. 1998)
OOWS (Pelechano et al. 2003)	Partly Schematext (Kuhnke et al. 2000)	

framework for feasibility. AWARE is a framework for requirements identification (Bolchini and Paolini 2004). QEM (Olsina et al. 2000) looks at the quality of the artefact produced in different phases of the lifecycle.

eW3DT (Scharl 2000) is based on a visualisation approach which complements either OO or ER approaches. Neither RSD (Janssen and Steen 2000) CFEP (Norton 2000) nor ADM (Diaz et al. 2001) specify any specific technique. SWM (Griffiths et al. 2002) is based on a process of stages and deliverables but does not prescribe a modelling approach. SchemaText (Kuhnke et al. 2000) is based on a hypertext approach, UCDW (McCracken and Wolfe 2004) is based on an HCI approach and UCDM is based on a usability approach, which deals with requirements and audience definition (Fuccella 1997).

STRUDEL (Fernandez et al. 1997) provides a data management system, whilst UWE (Koch 2000) uses the Unified Process as a basis, as does Conallen (1999), and RDF/XML Framework (Christodoulou et al. 2000) provide a generic theoretical framework that can be applied to any development.

9.3 Method focus

Many of the surveyed development methods are focussed on a specific area, as their authors investigate phenomena of interest to them. In this section methods are classified according to focus.

9.3.1 Focus on pre-requirements

FECWAD (Lu and Yeung 1998) is a framework for assessing the feasibility or merit of a project before development begins. WebML+ (Tongrungrojana and Lowe 2003) is a modelling language for forming a bridge between web business and information modelling. WISDM (Vigden 2002) could be partly considered however as it considers organisational analysis early in the development (Table 7).

Table 7 Focus on pre-requirements

FECWAD (Lu and Yeung 1998)	WebML+ (Tongrungrojana and Lowe 2003)	WISDM (Vigden 2002) (partly)

Table 8 Focus on user modelling/requirements

ADM (Diaz et al. 2001)	SOHDM (Lee et al. 1998)—customer analysis	UCDW (McCracken and Wolfe 2004)
AWARE (Bolchini and Paolini 2004)	TURBO (Ghosh 1999)	WISD (Gnaho 2000)
CFE (Norton 2000)	UCDM (Fuccella 1997)	WSDM (De Troyer and Leune 1998)
DPWA (Uden 2002)		

Table 9 Focus on conceptual models/design models

ADM (Diaz et al. 2001)	HMT (Zoller 2001)	OOWS (Pelechano et al. 2003)
ADWIS (Takahashi and Liang 1997)	JESSICA (Goeschka and Schranz 2001)	SHDT/W3DT/eW3DT (Bichler and Nusser 1996a, b)
Araneus (Atzeni et al. 1998)	Jweb (Bochicchio et al. 1999)	UML+ (Conallen 1998)
EORM (Lange 1994)	MacWeb (Nanard and Nanard 1995)	UWE (Koch 2000)
HDM (Garzotto, et al. 1993)	OOHDM (Schwabe and Rossi 1995)	WSDM (De Troyer and Leune 1998)
HDM-Lite (Fraternali and Paolini 1998)		

9.3.2 Focus on user modelling/requirements

Some methods have requirements and user modelling as their stated focus, such as UCDM (Fuccella 1997), WSDM (De Troyer and Leune 1998), TURBO (Ghosh 1999) and AWARE (Bolchini and Paolini 2004). Other methods have been placed in this category as it was felt that they have some specific and useful focus in this area, such as DPWA (Uden 2002) for its Applied Cognitive Task Analysis and SOHDM (Lee et al. 2004) for its customer analysis (Table 8).

9.3.3 Focus on conceptual models/design models

A large number of methods focus on the modelling stages, with particular interest in the separation into the data layer, navigation layer and the presentation layer. Of particular concern has been the means of transformation of the models between the layers, and ensuring that modelling information is not lost in such transformations (Table 9).

9.3.4 Aiming to automate/part automated

Some methods specifically aim to automate the process, or parts of it, such as STRUDEL (Fernandez et al. 2000), which provides a tool that can manage disparate data sources. Others, such as WSDM (De Troyer 2001), aim to turn the user (audience) requirements into a high level formal description in the conceptual design, which can later be used to automatically or semi-automatically generate effective web sites. It concentrates on the information design and the navigation structure, using a proprietary notation. In OO-H (Gomez et al. 2001), from a UML compliant class diagram, personalised navigation access diagrams are produced for each user type. The default interface is then generated, and improved using a pattern catalogue.

Table 10 Methods aiming to automate the web design process

Method and author	Tool name	Notes
ADM (Diaz et al. 2001)	Ariadne tool	
ADWIS (Takahashi and Liang 1997)	Pilot boat and web architect	Not available
Araneus (Atzeni et al. 1997)	ULIXES, PENELOPE	
RDF/XML Framework (Christodoulou et al. 2000)		Applying tool support using RDF/XML
EORM (Lange 1994)		
HDM Lite (Fraternali and Paolini 1998)	Autoweb	
HERA (Houben et al. 2003)		
HMT (Zoller 2001)	WebCon	
JESSICA (Goeschka and Schranz 2001)		
Jweb (Bochicchio et al. 1999)		
MacWeb (Nanard and Nanard 1995)		
OOHDM (Schwabe and Rossi 1995)	OOHDM-Web	
QEM (Olsina et al. 2000)	WebQEM_Tool	
Schematext (Kuhnke et al. 2000)		
SHDT/W3DT (Bichler and Nusser 1996a)	Web Designer	No longer available
STRUDEL (Fernandez et al. 2000)		
SWM (Griffiths et al. 2002)		
UWE (Koch and Kraus 2002)	ArgoUML	
WebComposition (Gellerson et al. 1997)		
WebML (Ceri et al. 2003)	WebRatio	
WOOM (Coda and al. 1998)		
XWMF (Kalpsing and Neumann 2000)	GRAMTOR, RDF-Handle,WebObjectBrowser	

JWeb (Bochicchio et al. 1999) is a development environment which assists, from the development of the definition of the conceptual schema using HDM constructs, to implementation of an application, whereas OOWS (Pelechano et al. 2003) is devised to use the OO-Method and is aiming for a fully automated environment (Table 10).

9.4 Teachability/learnability

Recent work highlights the complexity of web development and suggests that there are a number of factors which lead to it requiring greater cognitive skills than traditional software development (Kushwaha et al. 2006). Any development which is, in itself complex, will need methods that will assist with removing that complexity. In addition if, as reported, a large number of web developers are not computer scientists or hypermedia specialists (Holck and Clemmensen 2001; Overmyer 2000), then any notation or method needs to be clearly defined and simple to use, that is it needs to be learnable by its intended audience. We also found that the majority of the methods in this survey did not give enough detail to enable them to be used in their entirety. It can be seen (Jeary 2005) that even when methods are covered in their entirety, there is rarely enough information given about the method to allow the method's practical use. In addition, the explanations of how to use a number of methods may be difficult for the non-computer scientist to understand. For example, in Araneus (Atzeni et al. 1997) the developer needs an understanding of both hypertext and

database theory, and similarly with Strudel (Fernandez et al. 2000), which adds differing data types and sources to the method and has not been designed with the practitioner in mind. Other methods that also make assumptions on user expertise and understanding are OO-H (Gomez et al. 2001) and Jessica (Goeschka and Schranz 2001). DPWA (Uden 2002) covers Applied Cognitive Task Analysis (Militello and Hutton 1998), which was found to be cumbersome (Uden 2002), and WISD (Gnaho 2000) is clearly to be used by developers with knowledge and understanding of fuzzy logic, which is important in defining the user model. These methods are unlikely to be taken up by web developers. The explanations of the techniques are overly technical and complex and with web development time scales being shorter than traditional development cycles with aggressive release demands (Overmyer 2000) there is little time to be spent on learning complex techniques.

A number of methods have explained their phases or techniques in enough detail to make them teachable or learnable. For example: FECWAD (Lu and Yeung 1998), WSDM (De Troyer and Leune 1998), WebML (Ceri et al. 2003), SHDT/W3DT (Bichler and Nusser 1996), eW3DT (Scharl 2000) and RNA (Yoo and Bieber 2000). UCDM (Fuccella 1997) is described as having 'quick and dirty' approaches to getting user feedback for usability processes and TURBO (Ghosh 1999) is a practitioner based method.

10 Conclusions

The study in this paper attempted to ascertain, from a practical perspective, whether web development methods were difficult to use, and, if so, to try and understand what aspects of the methods proved problematic. In a study of 23 participants, all but one abandoned their chosen web development method, finding them difficult to learn and apply. Feedback from the study suggests that the main issues with methods were:

(a) Both the formality of web development methods, and ambiguity of terminology, caused problems in their application. If this is true of participants with specific training and knowledge in computing, then it suggests that typical practitioners, who often have no formal computing background, may also view development methods as difficult.
(b) Many of the methods were incomplete. For example, many mention feasibility and requirements but assume the method user will have the knowledge to undertake those aspects. In addition, they assume that the emphasis taken by the method user will be the same as that envisaged by the method author. Again, the variety of backgrounds of web developers means that such assumptions may be inappropriate.
(c) There is a lack of detailed practical guidance. The description often focuses on the rationale rather than the application. Although, the authors note that novices may feel more comfortable with a higher level of guidance, whereas the practitioner may feel the opposite, it may be that more complete guidance would increase users' confidence in web development methods.

Our survey of the web development methods shows that they have emerged from a number of different and largely distinct academic communities, and have been disseminated in a wide range of conferences and publications. Hence, our first aim was to examine the origins and evolution of the methods, by community. In doing so we found that each community had a different perspective on web development and distinct views on the features that methods should provide. Furthermore, there was little agreement over terminology and notation across such communities, and many methods gave only partial coverage of the development lifecycle.

We performed a comprehensive survey of over 50 web development methods (or techniques), and provide various perspectives (and classifications). However, since users of web development methods would, as a minimum, need to know the scope of any method they chose to adopt, and the particulars of the notations found within it, we have provided classifications of methods not only by the dominant notational approach, but also by their coverage of the lifecycle. In doing so, we find that, despite assertions to the contrary, very few methods cover the entire lifecycle, from requirements to maintenance, in sufficient detail for the method to be followed in all phases.

In summary, we suggest that, in order for web development methods to see large scale industrial usage, they must take more account of both the nature and needs of web developers. Specifically, to adopt and define clear terminology; to use notations and elements that do not assume familiarity with formal approaches; to provide a complete (whole lifecycle) solution; and to provide clear guidance as to how the method and its constituent elements should be applied.

Hence, whilst there is a proliferation of methods it appears that few match, or even attempt to match, the requirements of the practitioner. We, therefore, suggest that further work in this area needs to be carried out in order to ascertain industrial needs and investigate the extent to which existing methods meet those needs. Only then are we likely to see a reasonable take-up of web development methods within industry.

11 Limitations and further work

This work has provided some evidence to suggest that academic web development methods are difficult to use. However, the study only assesses the reported difficulty, rather than having any underlying theoretical model. In order to progress, one might, as suggested previously, attempt to apply learning theory to discover, more precisely, further causes for such difficulty, other than the suggestions of the participants. In addition, measures of cognitive difficulty may well highlight particular issues. In defence of our approach, however, ultimately, it is the perceived difficulty of a method that is the real barrier to its adoption, rather than any theoretical measure.

A further issue is that we have only used student practitioners for small development projects. An industrial study may provide different views on web development methods. However, it is clear that there are problems with existing methods, and that further work needs to be conducted to ascertain where web development methods can be improved to assist the practitioner. In doing so, it would seem sensible, in adopting good software engineering practice, that a first step would be to attempt to ascertain the requirements for web development methods, either in general, or, perhaps, for particular classes of development projects. One possible approach may be an adaptation of problem frames to describe web frames for different types of web developments or different domains, as suggested by Jeary and Phalp (2004).

References

Alaa, G., & Fitzgerald, G. (2004). A proposal of an upfront requirements modelling and design practice for e-Commerce projects. In *37th Hawaii International Conference on System Sciences*. Los Alamitos: IEEE Computer Society.

Atzeni, P., Mecca, G., & Merialdo, P. (1997). To weave the web. In *International Conference on very Large Databases*, Athens, Greece.
Atzeni, P., Mecca, G., & Merialdo, P. (1998). Design and maintenance of data-intensive web sites. In *Conference on Extending Database Technology*, Valencia, Spain.
Avison, D., & Fitzgerald, G. (2006). *Information systems development: Methodologies techniques and tools*. London: McGraw-Hill.
Barry, C., & Lang, M. (2001). A survey of multimedia and web development techniques and methodology usage. *IEEE Multimedia, 8*, 52–59.
Barry, C., & Lang, M. (2002). A comparison of 'traditional' and multimedia information systems development practices. *Information and Software Technology, 45*, 217–227.
Belbin, M. R. (1981). *Management teams*. New York: Addison-Wesley.
Bichler, M., & Nusser, S. (1996a). Developing structured WWW-Sites with W3DT. In *WebNet 96*, San Francisco, CA, 15–19 October.
Bichler, M., & Nusser, S. (1996b). SHDT—the structured way of developing WWW-sites. In *Proceedings of the 4th European Conference on Information Systems*. Lisbon, Portugal, 2–4 July.
Bigge, M. (1982). *Learning theories for teachers*. New York: Harper and Rowe.
Bochicchio, M., & Fiore, N. (2004). WARP: Web application rapid prototyping. In *2004 ACM Symposium on Applied Computing*, Nicosia, Cyprus, 14–17 March.
Bochicchio, M. A., Paiano, R., & Paolini, P. (1999). JWeb: An HDM environment for fast development of web applications. In *IEEE Conference on Multimedia Computing and Systems*, Florence, Italy. Los Alamitos: IEEE Computer Society.
Bolchini, D., & Paolini, P. (2004). Goal-driven requirements analysis for hypermedia-intensive web applications. *Requirements Engineering, 9*, 85–103.
Cartensen, P. H., & Vogelsang, L. (2001). Design of web-based information systems—new challenges for systems development? In *9th International Conference on Information Systems*. Bled, Slovenia, 27–29 June.
Ceri, S., Fraternali, P., Bongio, A., Brambilla, M., Comai, S., & Matera, M. (2003). *Designing data-intensive web applications*. Amsterdam: Morgan Kaufmann Publishers.
Christodoulou, S. P., Zafiris, P. A., & Papatheodorou, T. S. (2000). Web engineering: The developers view and a practioners approach. In *Web Engineering 2000*. Berlin: Springer.
Coda, F., Ghezzi, C., Vigna, G., & Garzotto, F. (1998). Towards a software engineering approach to web site development. In *9th International Workshop on Software Specification and Design*. Los Alamitos: IEEE Computer Society.
Conallen, J. (1998). Modelling web application design with UML. Retrieved June 27, 2003, from http://www.rational.com/products/whitepapers/100462.jsp.
Conallen, J. (1999a). *Building web applications with UML*. Boston: Addison-Wesley.
Conallen, J. (1999b). Modelling web application architectures with UML. *Communications of the ACM, 42*, 63–70.
Cox, K. (2000). Cognitive dimensions of use cases—feedback from a student questionnaire. In *Proceedings of the 12th Annual Meeting of the Psychology of Programming Interest Group (PPIG-12)*. Cosenza, Italy, 10 April.
De Troyer, O. M. F. (2001). Audience-driven web design. In M. Rossi & K. Siau (Eds.), *Information modelling in the new millennium*. Hershey: IDEA Group Publishing.
De Troyer, O. M. F., & Leune, C. J. (1998). WSDM: A user centered design method for web sites. In *7th International World Wide Web Conference*. Elsevier: Computer Networks and ISDN Systems.
Diaz, P., Aedo, I., & Montero, S. (2001). Ariadne, a development method for hypermedia. In *DEXA, 2001*. Munich, Germany. Berlin: Springer.
Fernandez, M., Florescu, D., Kang, J., Levy, A., & Suciu, D. (1997). STRUDEL: A web site management system. In *ACM SIGMOD Conference on the Management of Data*. Tucson, AZ.
Fernandez, M., Florescu, D., Levy, A., & Suciu, D. (2000). Declarative specification of web sites with strudel. *VLDB Journal, 9*, 38–55.
Fitzgerald, B., Russo, N. L., & Stolterman, E. (2002). *Information systems development: Methods in action*. London: McGraw-Hill.
Fraternali, P. (1999). Tools and approaches for developing data-intensive web applications: A survey. *ACM Computing Surveys, 31*, 227–263.
Fraternali, P., & Paolini, P. (1998). A conceptual model and a tool environment for developing more scalable, dynamic, and customisable web applications. In *6th International Conference on Extending Database Technology*, Valencia, Spain, 23–27 March.
Fraternali, P., & Paolini, P. (2000). Model driven development of web applications: The AutoWeb system. *ACM Transactions on Information Systems, 18*, 323–382.

Fuccella, J. (1997). Using user centered design methods to create and design usable web sites. In *Proceedings of 15th International Conference on Computer Documentation*, Salt Lake City, UT.
Gaedke, M. & Graf, G. (2000). Development and evolution of web-applications using the WebComposition process model. In *WebEngineering 2000*. Berlin: Springer.
Gagne, R. (1977). *The conditions of learning*. New York: Holt, Rhinehart and Wilson.
Garzotto, F., Paolini, P., & Schwabe, D. (1993). HDM—a model-based approach to hypertext application design. *ACM Transactions on Information systems, 11*, 1–26.
Gellerson, H. W., & Gaedke, M. (1999). Object-oriented web application development. *IEEE Internet Computing, 3*, 60–68.
Gellerson, H. W., Wicke, R., & Gaedke, M. (1997). WebComposition: An object-oriented support system for the web engineering lifecycle. *Computer Networks and ISDN Systems, 29*, 429–437.
Ghosh, G. (1999). Turbo-prototyping: Ultra rapid user centred web development. In *Human–Computer Interaction INTERACT 99*. Edinburgh, Scotland IFIP.
Gifford, S. S., Henry, S. M., & Schoenhoff, P. K. (2003). Personality and type considerations for programming teams in computer science classes. In *7th World Multiconference on Systems, Cybernetics, and Informatics*, Orlando, FL, 27–30 July.
Glass, R. (2001). Who's right in the web development debate? *Cutter IT Journal, 14*, 6–10.
Glass, R. L. (2004). Matching methodology to problem domain. *Communications of the ACM, 47*, 19–21.
Gnaho, C. (2000). Web-based information systems development—a user centered engineering approach. In *WebEngineering 2000*. Berlin: Springer.
Goeschka, K. M., & Schranz, M. W. (2001). Client and legacy integration in object-oriented web engineering. *IEEE Multimedia, 8*, 32–41.
Gomez, J., Cachero, C., & Pastor, O. (2001). Conceptual modelling of device-independent web applications. *IEEE Multimedia, 8*, 26–39.
Griffiths, G., Hebbron, B. D., Lockyer, M. A., & Oates, B. J. (2002). A simple method and tool for web engineering. In *SEKE 2002*. Ischia, Italy: ACM.
Gu, A., Henderson-Sellers, B., & Lowe, D. (2002). Web modelling languages: The gap between requirements and current exemplars. In *Australian World Wide Web Conference*, Queensland, Australia, 6–10 July.
Guell, N., Schwabe, D., & Vilain, P. (2000). Modeling interactions and navigation in web applications. In *WWW and Conceptual Modelling Workshop*. Lecture Notes in Computer Science. Salt Lake City, USA.
Harlen, W., & Schlapp, U. (1998). *Spotlights* (pp. 1–5). Edinburgh: Scottish Council for Research in Education.
Hatzopoulos, M., Vazirgiannis, M., & Rizos, I. (1993). HADT: Hypermedia application development tool for tourist applications. *European Journal of Information Systems, 2*, 91–100.
Holck, J., & Clemmensen, T. (2001). *What makes web-development different?* Denmark: Department of Informatics, Copenhagen Business School Copenhagen.
Houben, G.-J., Barna, P., Frasincar, F., & Vdovjak, R. (2003). Hera: Development of semantic web information systems. In *International Conference of Web Engineering*. Oviedo, Spain. Berlin: Springer.
Isakowitz, T., Stohr, A., & Balasubramanian, E. (1995). RMM: A methodology for structured hypermedia design. *Communications of the ACM, 38*, 34–44.
Janssen, W., & Steen, M. (2000). Rapid service development: An integral approach to e-Business engineering. In *WebEngineering 2000*. Berlin: Springer.
Jeary, S. (2005). Web development methods—a survey. Technical Report: SSM W01/05 Bournemouth University, Bournemouth.
Jeary, S., & Phalp, K. (2004). On the applicability of problem frames to web-based applications. In *First International Workshop on Advances and Applications of Problem Frames*, Edinburgh, Scotland.
Kalpsing, R., & Neumann, G. (2000). Applying the resource description framework to web engineering. In *1st International Conference on Electronic Commerce and Web Technologies*, Greenwich, London, 4–8 September.
Kappel, G., Proll, B., Reich, S., & Retschitzegger, W. (2006). *Web engineering*. New York: Wiley.
Koch, N. (1999). A comparative study of methods for hypermedia development. Technical Report: 9905 Ludwig Maximilians, Universitat Munchen.
Koch, N. (2000). Software engineering for adaptive hypermedia systems: Reference model, modelling techniques and development process. *Fakultat fur Mathematik und Informatik* (p. 371). Munchen: Ludwig-Maximilians-Universitat.
Koch, N., & Kraus, A. (2002). The expressive power of UML-based web engineering. In *Second International Workshop on Web-oriented Software Technology*, Valencia, Spain, June 2002, CYTED.
Kruchten, P. (2000). *The rational unified process: An introduction*. Boston: Addison-Wesley.

Kuhnke, C., Schneeberger, J., & Turk, A. (2000). A schema-based approach to web engineering. In *Thirty-third Hawaii International Conference on System Sciences*. Hawaii: IEEE.

Kushwaha, D. S., Singh, R. K., & Misra, A. K. (2006). Cognitive web based software development process: Towards a more reliable approach. *ACM SIGSoft Software Engineering Notes, 31*, 1–6.

Lang, M., & Fitzgerald, B. (2005). Hypermedia systems development practices: A survey. *IEEE Software, 20*, 68–75.

Lange, D. (1994). An object-oriented design method for hypermedia information systems. In *Twenty-seventh Annual Hawaii International Conference on System Sciences*, Maui, Hawaii.

Lee, C., Suh, W., & Lee, H. (2004). Implementing a community web site: A scenario-based methodology. *Information and Software Technology, 46*, 17–33.

Lee, H., Kim, J., Kim, Y. G., & Cho, S. (1999). A view based hypermedia design methodology. *Journal of Database Management, 10*, 3–13.

Lee, H., Lee, C., & Yoo, C. (1998). A scenario-based object-oriented methodology for developing hypermedia information systems. In *31st Annual Hawaii International Conference on System Sciences*. Maui, Hawaii: IEEE.

Lowe, D. (2003). Web system requirements: An overview. *Requirements Engineering, 8*, 102–113.

Lowe, D., & Eklund, D. (2001). Commercial issues in specification of web systems. In *6th Australian Workshop on Requirements Engineering*, Sydney, 22–23 November 2001.

Lowe, D., Ginige, A., Sifer, M., & Potter, J. (1996). The MATILDA data model and its implications. In *Third Literational Conference on Multimedia Modelling*, Toulouse, France.

Lu, M.-T., & Yeung, W.-L. (1998). A framework for effective commercial web application development. *Internet Research: Electronic Networking Applications and Policy, 8*, 166–173.

McCracken, D. D., & Wolfe, R. J. (2004). *User-centered website development: A human-computer interaction approach*. Upper Saddle River: Pearson Prentice Hall.

McDonald, A., & Welland, R. (2001a). Agile web engineering process. Technical report: TR-2001-98. Department of Computing Science, University of Glasgow, Glasgow.

McDonald, A., & Welland, R. (2001b). Web engineering in practice. In *Proceedings of the 4th WWW10 Workshop on WebEngineering*, Hong Kong, 1 May 2001.

Mecca, G., Atzeni, P., Masci, A., Sindoni, G., & Merialdo, P. (1998). The ARANEUS web-Base management system. *Sigmod Record, 27*, 544–546.

Merialdo, P., Atzeni, P., & Mecca, G. (2003). Design and development of data-intensive web sites: The ARANEUS approach. *ACM Transactions on Internet Technology, 3*, 49–92.

Militello, L. G., & Hutton, R. J. B. (1998). Applied cognitive task analysis (ACTA): A practitioner's toolkit for understanding cognitive task demands. *Ergonomics, 41*, 1618–1641.

Montero, S., Diaz, P., & Aedo, I. (2002). Requirements for hypermedia development methods: A survey of outstanding methods. In *CAiSE 2002*. Berlin: Springer.

Murugesan, S., & Ginige, A. (2005). Web engineering: Introduction and perspectives. In A. Ginige (Ed.), *Web engineering*. Hershey: Idea Group.

Nanard, J., & Nanard, M. (1995). Hypertext design environments and the hypertext design process. *Communications of the ACM, 38*, 49–56.

Norton, K. S. (2000). *Applying cross-functional evolutionary methodologies to web development in web engineering 2000*. Berlin: Springer-Verlag.

Olsina, L. (1997). Applying the flexible process model to build hypermedia products. In *Hypertext & Hypermedia: Products, Tools, Methods*, Paris, France, Hermes Editorial.

Olsina, L., Lafuente, G., & Rossi, G. (2000). Specifying quality characteristics and attributes for websites. In *WebEngineering 2000*. Berlin: Springer.

Overmyer, S. P. (2000). Whats different about requirements engineering for web sites? *Requirements Engineering, 5*, 62–65.

Pastor, O., Fons, J., & Pelechano, V. (2003). OOWS: A method to develop web applications from web-oriented conceptual models. In *Third Literational Workshop on Web-oriented Software Technologies*, Oviedo, Spain, 15 July 2003.

Pastor, O., Pelechano, V., Insfran, E., & Gomez, J. (1998). From object oriented conceptual modelling to automated programming in Java. In *17th International Conference on Conceptual Modeling*, Singapore, June 1998. Berlin: Springer.

Pelechano, V., Fons, J., Manoli, A., & Pastor, O. (2003). Developing web applications from conceptual models. In *Caise 2003*. Berlin: Springer.

Powell, T. A., Jones, D. L., & Cutts, D. C. (1999). *Web site engineering: Beyond web page design*. Upper Saddle River: Prentice Hall.

Pressman, R. S. (1998). Can internet-based applications be engineered? *IEEE Software, 15*, 104–109.

Scharl, A. (2000). *Evolutionary web development*. London: Springer.

Schwabe, D., & Rossi, G. (1995). The object-oriented hypermedia design model. *Communications of the ACM, 38*, 45–46.
Schwabe, D., & Rossi, G. (1998). An object oriented approach to web-application design. *Theory and Practice of Object Systems, 4*, 207–225.
Siau, K., & Cao, Q. (2003). How complex is the unified modelling language? In *Advances in database research* (Vol. 2, pp. 294–306). Idea Group.
Soltys, R., & Crawford, A. (Undated). JAD for business plans and designs. Retrieved 1 June 2008, from www.thefacilitator.com.
Takahashi, K., & Liang, E. (1997). Analysis and design of web-based information systems. In *6th International World Wide Web Conference*, Santa Clara, CA.
Taylor, M. J., McWilliam, J., Forsyth, H., & Wade, S. (2002). Methodologies and website development: A survey of practice. *Information and Software Technology, 44*, 381–391.
Tongrungrojana, R., & Lowe, D. (2003). WebML+: A web modeling language for forming a bridge between business modeling and information modeling. In *Fifteenth Iiternational Conference on Software Engineering and Knowledge Engineering*. San Francisco.
Uden, L. (2002). Design process for web applications. *IEEE Multimedia, 9*, 47–55.
Vigden, R. (2002). Constructing a web information system development methodology. *Information Systems Journal, 12*, 247–261.
Vora, P. R. (1998). Designing for the web: A survey. *Interactions, 5*, 13–30.
Weed, D. L. (1997). Methodologic guidelines for review papers. *Journal of the National Cancer Institute, 89*, 6–7.
Yoo, J., & Bieber, M. (2000). Finding linking opportunities through relationship-based analysis. In *Hypermedia & hypertext*. San Antonio, TX. New York: USAACM Press.
Zoller, P. (2001). HMT: Modelling interactive and adaptive hypermedia applications. In M. Rossi & K. Siau (Eds.), *Information modelling in the new millennium*. Hershey: IDEA Group Publishing.

Author Biographies

Sheridan Jeary is co-Director of the Software Systems Research Centre at Bournemouth University and has research interests that span Web Development Methods, Web Systems and the alignment of Information Technology with business (strategy and process). She had several years of management and systems experience across a variety of domains before moving into academia. She is the BU Project Manager for the EU Commission funded VIDE project which is financed under Framework 6 and is looking at Model Driven Development for business users.

Keith Phalp is Associate Dean: Head of Software Systems and Psychology and co-Director of the Software Systems Research Centre at Bournemouth University. His research encompasses software engineering, process modelling and requirements engineering. Dr Phalp originally read for a first degree in Mathematics, which he then taught for a few years, before completing a Masters in Software Engineering in 1991, followed by a PhD in process Modelling in 1994. He then spent 3 years as a post-doctoral research fellow at the University of Southampton, again in the area of process modelling. In 1997, Dr Phalp took up a lectureship at Bournemouth, and has been there ever since. He became Reader in Software Engineering in 2006 and was made Head of Group in 2007.

Jonathan Vincent is a Reader in Software Systems. His research interests are in Artificial Intelligence and Natural Computing. Dr Vincent has a BEng in Electrical and Electronic Engineering, an MSc in Computing (Software Engineering) and a PhD in Computer Science. He has wide ranging research interests and has published in a variety of areas within software engineering and computer science, including component based software engineering, software quality, modelling, evolutionary computation and neural networks.