Towards a Taxonomy of Hypermedia and Web Application Size Metrics

Emilia Mendes[1], Steve Counsell[2], and Nile Mosley[3]

[1] The University of Auckland, Private bag 92019, Auckland, NZ
emilia@cs.auckland.ac.nz
[2] Brunel University, Uxbridge UB8 3PH, Middlesex, UK
steve.counsell@brunel.ac.uk
[3] MetriQ (NZ) Limited, 19A Clairville Crescent, Wai-O-Taiki Bay, Auckland, NZ
nile@metriq.biz

Abstract. Surveying and classifying previous work on a particular field have several benefits, which are: i) to help organise a given body of knowledge; ii) to provide results that can help identify gaps that need to be filled; iii) to provide a categorization that can also be applied or adapted to other surveys; iv) to provide a classification and summary of results that may benefit researchers who wish to carry out meta-analyses. This paper presents a survey literature of hypermedia and Web size metrics published within the last 12 years and classifies the surveyed studies according to a proposed taxonomy. In addition, we also discuss the changes, mainly in the motivation for size metrics, that have occurred during our review period.

1 Introduction

Within the last 12 years several hypermedia and Web size metrics have been proposed, mainly motivated to help the authoring process of applications or to be used for Web cost estimation. Despite their importance for Web practitioners and those investigating Web cost estimation there is no single reference to date that classifies and compares such metrics.

We are aware of two previous surveys on Web metrics [4],[9] however none looked specifically into hypermedia and Web size metrics or metrics for authoring and cost estimation.

Dhyani et al. [4] concentrates on metrics that belong to one of the following six categories:

- *Web Graph Properties* – Metrics that measure structural properties of the Web on both macroscopic and microscopic scales.
- *Web Page Significance* – Metrics used to assess candidate pages in response to a search query and have a bearing on the quality of search and retrieval on the Web.
- *Usage Characterization* – Metrics that measure user behavior aiming at improving the content, organization and presentation of Web sites.
- *Web Page Similarity* – Metrics that measure the extent of association between Web pages.
- *Web Page Search and Retrieval* – Metrics for evaluating and comparing the performance of Web search and retrieval services.

D. Lowe and M. Gaedke (Eds.): ICWE 2005, LNCS 3579, pp. 110–123, 2005.

- *Information Theoretic* – Metrics that capture properties related to information needs, production and consumption.

Calero et al. [9] provides a survey where Web metrics are classified into three dimensions, all related to Web quality:

- *Web Features Dimension* – Incorporates Content, Navigation and Presentation metrics.
- *Quality Characteristics Dimension* – Incorporates Functionality, Reliability, Efficiency, Portability and Maintainability metrics.
- *Life Cycle Processes Dimension* – Process metrics related to a Web development life cycle.

In addition to the above classification they also assess their surveyed metrics according to a second criteria:

- *Granularity Level* – Whether the metric's scope is a "Web page" or "Web site".
- *Theoretical Validation* – Whether or not a metric has been validated theoretically.
- *Empirical Validation* – Whether or not a metric has been empirically validated.
- *Automated Support* – Whether or not there is a support tool that facilitates the calculation of the metric.

The contribution of this paper is twofold: first, to provide a survey on hypermedia and Web size metrics based on literature published within the last 12 years; second, to provide a taxonomy of size metrics that helps classify this existing body of knowledge. A taxonomy represents a model that is used to classify and understand a body of knowledge [13].

The classification used by our taxonomy was based on basic concepts of software measurement [10], [14],[3].

The remainder of this paper is organised as follows: Section 2 introduces our taxonomy, explaining terms and definitions that are part of this classification. Section 3 presents our literature review, which was based on 15 papers. Section 4 applies the proposed taxonomy to classify each of the papers from our literature review. In Section 5 we discuss the change in trends that have occurred in the area of hypermedia and Web metrics within the last 12 years. Finally, conclusions are presented in Section 6.

2 Size Metrics Taxonomy

The basis for the taxonomy we propose consists of software measurement concepts [10], [14] and literature in software size metrics and measurement [3].

Motivation: Describes the rationale for proposing a given size metric. Examples of motivation can be "to help author hypermedia applications", or "to estimate effort".

Harvesting Time: Describes when in the development life cycle the metric should be measured or estimated. This category can be simply "Early size metric" or "Late size metric", however a longer description can also be given whenever necessary (e.g. "Late size metric to be measured after the implementation is finished").

Metric Foundation: Describes whether the size metric is a Problem-orientated metric or a Solution-orientated metric [3].

- *Problem-Orientated Metric*: A problem-orientated metric assumes that an application size corresponds directly to the size of the problem to be solved in order to deliver a corresponding application. So, the greater the problem, the greater the size. In this context, the problem to be solved is denoted by the functionality of the application to be developed. Problem-orientated size metrics generally take the form of surrogate metrics of functionality. These metrics can be extracted from the specification or design documents. An example of a common problem-oriented metric is Function Points, which aims to measure the size of an application in terms of the amount of functionality within the application, as described by its proposed specification.
- *Solution-Orientated Metric*: In contrast, a solution-orientated metric assumes that an application's size corresponds to the actual delivered size of an application (e.g. Lines of code).

Class: Allows for the organisation of size metrics into either of three possible classes: Length, Complexity, and Functionality [10].

- *Length:* Measures the physical size of a hypermedia or Web application;
- *Functionality:* Measures the functions supplied by the application to the user;
- *Complexity:* Measures the structural complexity of a hyperdocument, where the structure of a hyperdocument is represented by the way nodes are interconnected by links.

According to the descriptions given above, we can say that the foundation for both length and complexity metrics is "solution-orientated", whereas the foundation for a functionality size metric is "problem-orientated".

Entity: Represents the product to which the size metric is associated. Possible values are "Web hypermedia application", "Web software application", "Web application", "Hypermedia application", "Hypertext application", "Media", "Program/Script".

- *Web Hypermedia Application [5]:* A non-conventional software application where chunks of information are generally text/images/video and the structure is static.
- *Web Software Application [5]:* A conventional software application that depends on the Web or uses the Web's infrastructure for execution (e.g. legacy information systems, e-commerce).
- *Web Application:* An application that combines characteristics of both Web software and Web hypermedia applications.
- *Media:* A multimedia component, e.g. graphic, audio, video, animation, photograph.
- *Program/Script:* Code employed to add functionality to an application (e.g. Perl scripts, javasrcipt).

Measurement Scale: Describes the nature of the mapping M from the empirical system to the numerical/symbolic system and determines what sort of manipulations we can apply to a metric. The five scales are Nominal, Ordinal, Interval, Ratio and Absolute [10], as follows:

- *Nominal:* Defines classes or categories, and places entities in a particular class or category, based on the value of the attribute.

- *Ordinal:* Augments the nominal scale with information about an ordering of classes or categories.
- *Interval:* Augments the ordinal scale with information about the size of the intervals that separate the classes.
- *Ratio:* Preserves ordering, the size of intervals between classes, and ratios between classes. Can have one or more associated unit(s) of measurement.
- *Absolute:* The metric always takes the form "number of occurrences of x in the entity E". Has associated only one unit of measurement.

Computation: Describes whether a size metric can be measured Directly or Indirectly [10]. Indirect measurement means that the metric is computed based on other metrics. Conversely, Direct measurement means that the size metric does not rely on other metrics in order to be measured.

Validation: Describes whether a size metric has been validated. Possible values are "validated Empirically", "validated Theoretically", "Both", and "None". This is similar to one of the criterion suggested by Calero et al. [4].

Model Dependency: Represents whether a size metric requires the use of a specific Web methodology or model in order to be measures. Possible values are "Specific", and "Nonspecific".

3 Literature Review of Hypermedia and Web Size Metrics

This Section presents a literature review of hypermedia and Web size metrics proposed within the past 12 years, described in chronological order. We have not detailed too much these metrics due to shortage of space.

3.1 1992 – Size Metrics by Botafogo et al.

Botafogo et al. [1] proposed size metrics to be used to help identify problems with the hyperdocument being created. Their focus was on the hyperdocument's navigation rather than on its content.

- *Compactness:* Measures how well connected (by links) a hyperdocument is. Its value varies between zero (completely disconnected) and one (completely connected).
- *Stratum:* Measures to what degree the hyperdocument is organised into a single reading path. Its value varies between zero (no imposed reading order) and one (single path).

3.2 1995 – Size Metrics by Yamada et al.

Yamada et al. [21] proposed size metrics to measure authoring and maintenance problems.

- *Interface Shallowness:* Cognitive load on users. Assumes that applications are structured hierarchically, each level corresponds to a cognitive "layer", and moving from one layer to another increases the cognitive load on users.

- *Downward Compactness:* Structural complexity of reaching the n^{th} node from the root.
- *Downward Navigability:* Measures hypermedia navigability, where an easily navigable hypermedia application (1) has a shallow interface layer from the root to the n^{th} node and (2) is compact from the root (that is, it is structurally simple to reach the n^{th} node from the root).

3.3 1995 – Size Metrics by Hatzimanikatis et al.

Hatzimanikatis et al. [12] proposed size metrics to measure the readability and maintainability of hypermedia applications.

- *Path Complexity:* The number of different paths or cycles that can be found in a hyperdocument, assuming it to be a graph. The *path complexity* of a linear hyperdocument is minimal.
- *Tree Impurity:* The extent to which a graph deviates from being a tree.
- *Modularity:* Measures if the nodes are self-contained and independent.
- *Individual Node Complexity:* Complexity that a single node imposes on the overall structure.

3.4 1996 – Size Metrics by Bray

Bray [2] proposed size metrics to measure the size of Web applications.

- *Page Size:* Measured in three different ways:
  1. The sum of space used (Kbytes) by its Web pages (PS1);
  2. The sum of the number of words in its Web pages (PS2);
  3. The sum of the number of image references in its Web pages (PS3).
- *Outbound Connection:* Number of links that point to another Web application/site.
- *Inbound Connection:* Number of links from other applications pointing to application *w*.

3.5 1997 – Size Metrics by Fletcher et al.

Fletcher et al. [11] proposed size metrics to predict effort to develop multimedia applications[1].

- *Media Type:* Number of graphics, audio, video, animations, photographs.
- *Media Source:* If media is original or reused.
- *Component Duration:* Duration of an animation, sound or video.
- *Number of Objects* (including sounds): Number of objects on the screen.
- *Screen Connectivity:* Number of links between a screen and other screens.
- *Screen Events:* Number of events on a screen.
- *Actions per Event:* Average number of actions per event.

[1] Although this work targets at multimedia applications, the strong similarities allow for the same assessment to be applied to hypermedia applications

3.6 1998; 2000 – Size Metrics by Cowderoy

Cowderoy [7],[8] proposed size metrics to predict effort to develop Web applications.

Web application

- *Web Pages:* Number of Web pages in an application.
- *Home Pages:* Number of major entry points to the Web application.
- *Leaf Nodes:* Number of Web pages in an application that have no siblings.
- *Hidden Nodes:* Number of Web pages excluded from the main navigation buttons.
- *Depth:* Number of Web pages on the second level that have siblings.
- *Application Paragraph Count:* Number of PPC for all Web pages in an application.
- *Delivered Images:* Number of unique images used by the Web application.
- *Audio Files:* Number of unique audio files used in a Web application.
- *Application Movies:* Number of PMs for all the Web pages in an application.
- *3d Objects:* Number of files (inc;. 3D objects) used in a Web application.
- *Virtual Worlds:* Number of files (incl. virtual worlds) used in a Web application.
- *External Hyperlinks:* Number of unique URLs in the Web application.

Web page

- *Actions:* Number of independent actions by use of Javascript, Active X etc.
- *Page Paragraph Count (PPC):* Number of paragraphs in a Web page.
- *Word Count:* Number of words in a Web page.
- *Navigational Structures:* Number of different structures in a Web page.
- *Page Movies (PM):* Number of movie files used in a Web page.
- *Interconnectivity:* Number of URLs that link to other pages in the same application.

Media

- *Image Size (IS):* Computed as width * height.
- *Image Composites:* Number of layers from which the final image was created.
- *Language Versions:* Number of image versions that must be produced to accommodate different languages or different cultural priorities.
- *Duration:* Summed duration of all sequences within an audio file.
- *Audio Sequences:* Number of sequences within the audio file.
- *Imported Images:* Number of graphics images imported into an audio file.

Program

- *Lines of Source Code:* The number of lines of code in a program/script.
- *McCabe Ciclomatic Complexity:* The structural complexity of a program/script.

3.7 1999; 2000; 2001 – Size Metrics by Mendes et al.

Mendes et al. [16]-[18] proposed size metrics initially to estimate effort to develop Hypermedia applications [17] and later to estimate effort for Web applications [16],[18].

Hypermedia application

- *Hyperdocument Size:* Number of files (e.g. HTML files).
- Complexity
- *Connectivity:* Number of non-dynamically generated links within a hypermedia application.
- *Compactness:* Measures how inter-connected the nodes are.
- Stratum: Measures to what degree the application is organised for directed reading.
- *Link Generality:* Measures if the link applies to a single or multiple instances.

Web application

- *Page Count:* Number of HTML or SHTML files .
- *Media Count:* Number of unique media files.
- *Program Count:* The number of CGI scripts, JavaScript files, and Java applets.
- *Total Page Allocation:* Space (Mbytes) allocated for all HTML or SHTML pages.
- *Total Media Allocation:* Space (Mbytes) allocated for all media files.
- *Total Code Length:* Number of lines of code for all programs.
- *Reused Media Count:* Number of reused or modified media files.
- *Reused Program Count:* Number of reused or modified programs.
- *Total Reused Media Allocation:* Space (Mbytes).allocated for all reused media files.
- *Total Reused Code Length:* Number of lines of code for all reused programs.
- *Code Comment Length:* Number of comment lines in all programs.
- *Reused Code Length:* Number of reused lines of code in all programs.
- *Reused Comment Length:* Number of reused comment lines in all programs.
- *Total Page Complexity:* Average number of different types of media used, excluding text.
- *Connectivity:* Number of internal links, not including dynamically generated links.
- *Connectivity Density:* Computed as *Connectivity* divided by *page coun*t.
- *Cyclomatic Complexity:* Computed as *Connectivity -page coun*t) + 2.

Web page

- *Page Allocation:* Allocated space (Kbytes) of a HTML or SHTML file.
- *Page Complexity:* Number of different types of media used on a page, not including text.
- *Graphic Complexity:* Number of graphics media.
- *Audio Complexity:* Number of audio media.
- *Video Complexity:* Number of video media.
- *Animation Complexity:* Number of animations.
- *Scanned Image Complexity:* Number of scanned images.
- Complexity
- *Page Linking Complexity:* Number of links.

Media

- *Media Duration:* Duration (minutes).of audio, video, and animation
- *Media Allocation:* Size (Kbytes) of a media file.

Program

- *Program Code Length:* Number of lines of code in program.

3.8 2000 – Size Metrics by Rollo

Rollo [21] did not suggest any new size metrics. However, he was the first, as far as we know, to investigate the issues of measuring functionality of Web applications aiming at cost estimation, using numerous function point analysis methods.

- *Functional Size:* Number of function points associated with a Web application. Function points were measures using COSMIC-FFP[2], Mark II and Albrecht [21].

3.9 2000 – Size Metrics by Cleary

Cleary [6] proposed size metrics to estimate effort to develop Web applications.

Web hypermedia application

- *Non-textual Elements:* Number of unique non-textual elements within an application.
- *Externally Sourced Elements:* Number of externally sourced elements.
- *Customised Infra-structure Components:* Number of customised infra-structure components.
- *Total Web Points:* Size of a Web hypermedia application in Web points.

Web software application

- *Function Points:* Functionality of a Web software application.

Web page

- *Non-textual Elements Page:* Number of non-textual elements.
- *Words Page:* Number of words.
- *Web Points:* Length of a Web page. Scale points are "Low", "Medium" and "High". Each point is attributed a number of Web points, previously calibrated to a specific dataset.
- Complexity
- *Number of Links into a Web Page:* Number of incoming links (internal or external links).
- *Number of Links out of a Web Page:* Number of outgoing links (internal or external links).
- *Web Page Complexity:* Complexity of a Web page based upon its *number of words*, and combined *number of incoming* and *outgoing links*, plus the *number of non-textual elements*.

[2] COSMIC-FFP = COmmon Software Measurement International Consortium-Full Function Points

3.10 2000 – Size Metrics by Reifer

Reifer [20] proposed size metrics to be used to estimate effort to develop Web applications.

- *Web Objects:* The number of Web Objects in a Web application using Halstead's equation for volume, tuned for Web applications. The equation is as follows:

$$V = N \log_2(n) = (N_1 + N_2) \log_2 (n_1 + n_2) \quad (1)$$

where:

N = number of total occurrences of operands and operators
n = number of distinct operands and operators
N_1 = total occurrences of operand estimator
N_2 = total occurrences of operator estimators
n_1 = number of unique operands estimator
n_2 = number of unique operators estimators
V = volume of work involved represented as Web Objects

Operands are comprised of the following metrics:

- *Number of Building Blocks:* Number of components, e.g., Active X, DCOM, OLE.
- *Number of COTS:* Number of COTS components (including any wrapper code).
- *Number of Multimedia Files:* Number of multimedia files, except graphics files.
- *Number of Object or Application Points [7],[8]:* Number of object/application points etc.
- *Number of Lines:* Number of xml, sgml, html and query language lines.
- *Number of Web Components:* Number of applets, agents etc.
- *Number of Graphics Files:* Number of templates, images, pictures etc.
- *Number of Scripts:* Number of scripts for visual language, audio, motion etc.

3.11 2003 – Size Metrics by Mangia and Paiano

Mangia and Paiano proposed size metrics to estimate effort to develop Web applications modelled using the W2000 methodology [15].

Web application

- *Macro:* Macro-functions required by the user.
- *DEI:* Input data for each operation.
- *DEO:* Output data for each operation.
- *Entities:* Information entities which conceptually model the database.
- *AppLimit:* Application limit of each operation.
- *LInteraction:* Level of interaction various users of the application have in each operation.
- *Compatibility:* Compatibility between each operation and application's delivery devices.
- *TypeNodes:* Types of nodes which constitute the navigational structure.
- *Acessibility:* Accessibility associations and pattern of navigation between node types.

- *NavCluster:* Navigation cluster.
- *ClassVisibility:* Visibility that classes of users have of the navigational structure.
- *DeviceVisibility:* Visibility that delivery devices have of the navigational structure.

3.12 2003 – Size Metrics by Mendes et al.

Mendes et al. [19] proposed size metrics to estimate effort to develop Web applications.

- *Web Pages:* Number of Web pages in a Web application.
- *New Web Pages:* Number of Web pages created from scratch.
- *Customer Web Pages:* Number of Web pages provided by the customer.
- *Outsourced Web pages:* Number of outsourced Web pages.
- *Text Pages:* Number of text pages (A4 size) that had to be typed.
- *Electronic Text Pages:* Number of reused text pages in electronic format.
- *Scanned Text Pages:* Number of reused text pages that had to be scanned with OCR
- *New Images:* Number of new images/photos/icons/buttons created.
- *Electronic Images:* Number of reused images/photos in electronic format.
- *Scanned Images:* Number of reused images/photos that need to be scanned.
- *External Images:* Number of images obtained from an image/photo library or outsourced.
- *New Animations:* Number of new animations (Flash/gif/3D etc) created from scratch.
- *External Animations:* Number of reused animations (Flash/gif/3D etc).
- *New Audio:* Number of new audio/video clips created.
- *External Audio:* Number of reused audio/video clips.
- *High Fots:* Number of High-effort[3] features off-the-shelf (FOTS), i.e., reused as is.
- *High FotsA:* Number of High-effort FOTS adapted to local circumstances.
- *High New:* Number of new High-effort Feature/ Functionality developed from scratch.
- *Fots:* Number of Low-effort FOTS, i.e., reused as is.
- *FotsA:* Number of Low-effort FOTS adapted to local circumstances.
- *New:* Number of new Low-effort Feature/ Functionality developed from scratch.

4 Application of Taxonomy to Surveyed Size Metrics

This Section discusses the literature review presented in Section 3 in light of the taxonomy proposed in Section 2. In order to provide a more effective discussion, we have summarised the main findings from the literature review, presented as Table 1. The literature review was based on 15 studies, where 133 metrics were proposed in

[3] High effort means that a single feature used at least 12 person hours to be created from scratch or four person hours to be adapted

total. The detailed results for the application of the taxonomy to the size metrics can be downloaded from http://www.cs.auckland.ac.nz/~emilia/detailedtable.pdf.

Eleven studies (73%) proposed size metrics motivated by their use to estimate effort for developing applications. This suggests that, at least for the studies motivated towards effort estimation, size metrics should be harvested early in the development cycle to be of use for estimating effort and costs. However, out of the 109 metrics proposed for effort estimation, only 33 metrics (30%) are *Early* metrics, all of which were proposed by only two studies [19], [15]. Most of the proposed metrics are solution-orientated (83%) and length (62%) metrics. Thirteen (64%) metrics, out of a total of 19 functionality metrics, measure functionality using some of the function points analysis methods, and the remaining six base their measurement on a list of features/functions to be provided to customers at the start of the development [19].

Table 1. Summary of Literature review findings

Category	**Values**	**studies**	**%**
Motivation	Help author hypermedia applications	1	6.6%
	To give feedback on possible improvements that will lead to better authoring and maintenance	1	6.6%
	Measure readability and maintainability	1	6.6%
	to measure the size of Web applications	1	6.6%
	estimate effort to develop multimedia applications	1	6.6%
	to estimate effort to develop Web applications	9	60%
	to estimate effort to develop hypermedia applications	1	6.6%
Category	**Values**	**metrics**	**%**
Harvesting Time	Early	33	25%
	Late	100	75%
Metric foundation	Problem-orientated	23	17%
	Solution-orientated	110	83%
Class	Length	82	62%
	Functionality	19	14%
	Complexity	32	24%
Entity	Web software application	1	1%
	Web hypermedia application	4	3%
	Web application	76	57%
	Hypermedia application	14	11%
	Hypertext application	0	0%
	Web page	22	16%
	Media	11	8%
	Program/Script	5	4%
Measurement Scale	Nominal	4	3%
	Ordinal	4	3%
	Interval	0	0%
	Ratio	118	89%
	Absolute	7	5%
Computation	Direct	103	77%
	Indirect	30	23%
Validation	Empirically	69	52%
	Theoretically	0	0%
	Both	5	4%
	None	59	44%

Slightly more than half of proposed size metrics (57%) relate to the entity Web application, which suggests they can be used for static as well as dynamic Web applications. Only 38 size metrics (28%) are bottom-up metrics, allowing for the measurement of "parts" of an application (e.g. Web page, media). The remaining 72% target at the whole application, where application can either be hypermedia (11%), Web

hypermedia (3%), Web software (1%), or Web (57%). The majority of metrics are measured on a ratio scale (89%), not surprising given that most metrics are solution-orientated. This is also reflected on the number of metrics that can be computed directly (77%), as opposed to indirectly (23%). A comparatively high number of metrics have been proposed without either empirical or theoretical validation (44%), which unfortunately makes their corresponding studies "advocacy research". Empirical and/or theoretical validation are fundamental to building our scientific knowledge [10]. Despite the small number of size metrics measured using either the nominal (3%) or ordinal scale (3%), researchers and practitioners alike should take care when applying these metrics since their measures cannot be employed arithmetically, without being in violation of the Representational Theory of measurement, a fundamental concept which is often ignored (e.g. [6], [20]).

5 Change in Trends

In the years 1992 to 1996 size was measured solely using complexity size metrics. In 1997 came the first publication that demonstrated the use of hypermedia/multimedia size metrics for cost estimation. From 1998 to 2000 more work was devoted to size metrics applicable to cost estimation; three of these were by industry practitioners [7],[8],[6],[20] who proposed metrics and exemplified their use with very small data sets or development practices from just one Web company for each practitioner. Regrettably, their findings may not be applicable to other Web companies work and practices and cannot be considered an empirical validation, so hampering the external and internal validity of their findings, respectively.

Except for [19] and [15], all size metrics proposed for cost estimation presented in Section 3 have been related to implemented Web applications, represented predominantly by solution-orientated size metrics. Even when targeted at measuring functionality based on function point analysis, researchers only considered the final Web application, rather than requirements documentation generated using existing Web development methods. This makes their usefulness as early effort predictors questionable. Except for Rollo [21], all literature cited in Section 3 employed at least one Solution-Orientated type of metric. This may be explained by the difficulty in using early size metrics, gathered at the start of the Web development life cycle.

Length and complexity metrics are classes used respectively by 62% and 24% of the 133 size metrics presented in Section 3. Functionality was used as a class of only 14% of the size metrics. The small amount of previous work using functionality size metrics may be explained by the fact that until recently the highest volume of Web applications developed used solely static pages, written in HTML, with graphics and Javascript. Therefore both researchers and practitioners would have focused on size metrics that were adequate for this type of Web application.

6 Conclusions

This paper presented a survey literature of hypermedia and Web size metrics published in the literature within the last 12 years, and classified the surveyed studies according to a proposed taxonomy.

Eleven studies proposed a total of 133 size metrics to be used for effort estimation. However, only 33 can be harvested early in development cycle, necessary criterion for estimating effort and costs. 83% and 68% of the metrics are solution-orientated and length metrics, respectively. Close to two-thirds of the functionality size metrics measure functionality using function points analysis methods. The other third uses a list of features/functions to be provided to customers at the start of the development. 89% of the metrics are measured on a ratio scale and 77% can be computed directly. 44% of the metrics have not been validated empirically or theoretically.

Regarding the change in trends we have observed that from 1996 onwards, the majority of size metrics were geared towards Web applications, rather than hypermedia applications, illustrating a shift in the focus not only from the research community but also by practitioners. Most size metrics were aimed at cost estimation, except for those proposed between 1992 and 1996.

Recent work [19],[20] showed that complexity size metrics do not seem to be as important as functionality and length size metrics. This may be due to the motivation behind the proposition of such metrics. However, it may also point towards a change in the characteristics of Web applications developed in the past, compared to those developed today. Many Web applications are moving to be "dynamic" applications, where pages are generated "on-the-fly". This may indicate that looking at an application's structure, represented by its links, ceases to be as important as in the past. This also explains the gradual exclusion of complexity size metrics from recent literature.

References

1. Botafogo, R., Rivlin, A.E. and Shneiderman, B. Structural Analysis of Hypertexts: Identifying Hierarchies and Useful Metrics, ACM TOIS,10(2), (1992), 143–179.
2. Bray, T. Measuring the Web. Proc. Fifth WWW Conference, May 6-10, Paris, France, (1996) http://www5conf.inria.fr/fich_html/papers/P9/Overview.html.
3. Briand, L.C., and Wieczorek, I. Software Resource Estimation. Encyclopedia of Sw. Eng., 2, P-Z (2nd ed.), Marciniak, John J. (ed.) NY: John Wiley & Sons, (2002),1160–1196.
4. Calero, C., Ruiz, J., and Piattini, M. A Web Metrics Survey Using WQM, Proceedings ICWE04, LNCS 3140, Springer-Verlag Heidelberg, (July 2004), (2004), 147–160.
5. Christodoulou, S. P., Zafiris, P. A., Papatheodorou, T. S. WWW2000: The Developer's view and a practitioner's approach to Web Engineering, Proc. 2nd ICSE Workshop Web Eng. (2000), 75–92.
6. Cleary, DWeb-based development and functional size measurement. Proc. IFPUG 2000 Conference, (2000).
7. Cowderoy, A.J.C., Donaldson, A.J.M., Jenkins, J.O. A Metrics framework for multimedia creation, Proc. 5th IEEE Metrics Symposium, Maryland, USA, (1998).
8. Cowderoy, A.J.C., Measures of size and complexity for web-site content, Proc. 11th ESCOM Conference, Munich, Germany, (2000), 423–431.
9. Dhyani, D., Ng, W.K., and Bhowmick, S.S. A survey of Web metrics, ACM Computing Surveys, 34(4), (December 2002), (2002), 469–503.
10. Fenton, N. E. and Pfleeger, S. L., Software Metrics, A Rigorous & Practical Approach, 2nd edition, PWS Publishing Company and International Thomson Computer Press, (1997).
11. Fletcher, T., MacDonell, S. G., and Wong, W. B. L. Early experiences in Measuring Multimedia Systems Development Effort. In: Multimedia Technology and Applications, Hong Kong: Springer-Verlag, (1997), 211–220.

12. Hatzimanikatis, A. E., Tsalidis, C. T., and Chistodoulakis, D. Measuring the Readability and Maintainability of Hyperdocuments, J. of Software Maintenance, Research and Practice, 7, (1995), 77–90.
13. IEEE, Standard Taxonomy for Software Engineering Standards (ANSI). The Institute of Electrical and Electronics Engineers Inc., (1986).
14. Kitchenham, B.A., Hughes, R.T., and Linkman, S.G. Modeling software measurement data, IEEE TSE, 27, 9, (Sept. 2001), (2001), 788–804.
15. Mangia, L., and Paiano, R. MMWA: A Software Sizing Model for Web Applications, Proc. WISE'03, (2003).
16. Mendes, E., and Mosley, N.. Web Metrics and Development Effort Prediction, Proc. ACOSM 2000, (2000).
17. Mendes, E., Hall, W., Harrison, R. Applying measurement principles to improve hypermedia authoring, NRHM, 5, Taylor Graham Publishers, (1999),105–132.
18. Mendes, E., Mosley, N., and Counsell, S. Web Metrics – Estimating Design and Authoring Effort. IEEE Multimedia, Special Issue on Web Engineering, Jan.-Mar., (2001), 50–57.
19. Mendes, E., Mosley, N., and Counsell, S. Investigating Early Web Size Measures for Web Costimation, Proc. EASE'2003 Conference, Keele University, (2003).
20. Reifer, D.J. Web development: estimating quick-to-market software. IEEE Software, (Nov/Dec), (2000), 57–64.
21. Rollo, T. Sizing e-commerce. Proc. ACOSM 2000, Sydney, Australia, (2000).
22. Yamada, S., Hong, J., and Sugita, S. Development and Evaluation of Hypermedia for Museum Education: Validation of Metrics, ACM Transactions on Computer-Human Interaction, 2(4), (1995), 284–307.