From Comparison Matrix to Variability Model: The Wikipedia Case Study

Nicolas Sannier, Mathieu Acher, Benoit Baudry

From Comparison Matrix to Variability Model
The Wikipedia Case Study

Nicolas Sannier, Mathieu Acher, and Benoit Baudry
University of Rennes 1, Irisa/Inria
Campus Universitaire de Beaulieu, 35042 Rennes cedex, France {firstname.lastname}@inria.fr

***Abstract*—Product comparison matrices (PCMs) provide a convenient way to document the discriminant features of a family of related products and now abound on the internet. Despite their apparent simplicity, the information present in existing PCMs can be very heterogeneous, partial, ambiguous, hard to exploit by users who desire to choose an appropriate product. Variability Models (VMs) can be employed to formulate in a more precise way the semantics of PCMs and enable automated reasoning such as assisted configuration. Yet, the gap between PCMs and VMs should be precisely understood and automated techniques should support the transition between the two. In this paper, we propose variability patterns that describe PCMs content and conduct an empirical analysis of 300+ PCMs mined from Wikipedia. Our findings are a first step toward better engineering techniques for maintaining and configuring PCMs.**

I. Introduction

Numerous organizations and companies maintain and propose tabular data to advertise the list of features supported or not by a given product model or a product. Similarly, open initiatives, like Wikipedia, collect descriptions of a wide range of products in a given domain against different criteria. The so-called *Product Comparison Matrices (PCMs)* provide a convenient and simple formalism to identify the discriminant features of a product compared to another.

We partially introduce a PCM in Figure 1a that deals with the comparison of different webmail providers against different features such as access protocols, automatic forwarding, integration with instant messaging, etc. Such PCMs now abound on the internet and constitute a rich source of information and knowledge. Users can exploit the numerous available PCMs to find and choose an appropriate product that meets their requirements.

Our observations of internet PCMs let us believe that the information and knowledge contained in PCMs can be very useful but, in the meantime, also hard to understand, maintain and exploit. In order to explore this hypothesis, we select Wikipedia, one of the most important source of PCMs from various domains and for different kinds of products.

Variability Models (VMs), like decision models or feature models, are a well-known formalism that can overcome the identified limitations of PCMs. VMs can be employed *i)* to formulate in a more precise way the semantics of PCMs, *ii)* to enable automated reasoning (e.g., for performing assisted configurations) and *iii)* to offer an explicit and compact view of the variability and logical relationships of a product family. Specifically, we consider *Feature Models (FMs)* the most popular notation employed in industry [1] and widely studied by academics for 20+ years [2], [3]. Moreover, FMs are sometimes used in conjunction to tabular data and spreadsheets (like PCMs) when practitioners model variability [1].

In this paper, we address the following research questions: What information is present in PCMs and what is the precise meaning of PCMs w.r.t. variability? What is the gap between PCMs and FMs?

We perform an empirical and rigorous study supported by automated techniques and based on Wikipedia data. We propose a taxonomy of information variability types contained in PCMs. Our observations show quantitative evidences that PCMs of Wikipedia suffer from major drawbacks. In particular, the ambiguous nature of data and the implicit variability patterns we observed complicate the task of reasoning, understanding and exploiting a PCM. This work furthers the understanding of PCMs as well as their relationship to VMs.

The remainder of the paper is organized as follows. Section II introduces some background information and formulates the problem statement. Section III presents the anatomy of a Wikipedia PCM, a taxonomy of variability information and two empirical analyses we performed over a large set of PCMs. Section IV discusses the impacts of our findings and calls for more research effort. Section V discusses threats to validity. Section VI reviews related work while Section VII concludes the paper.

II. Background and Problem Statement

A. Product Comparison Matrices

Product Comparison Matrices (PCMs) provide a simple and convenient way to express properties on products and compare them to several different others from the same *family*. They are provided by open initiatives like Wikipedia or consumers organizations. It allows companies to present and advertise on the different *facets* of their *product series*. PCMs provide a global view on several different competing products, showing the presence, absence, limitations of a facet, expressing *commonality and variability* between products under comparison.

PCMs propose an important amount of information and knowledge that we want to precisely understand. Based on preliminary observations (presented in section III), we hypothesize PCMs suffer from several major drawbacks:

- **lack of formalization:** cells in a PCM can provide a simple yes/no information describing the presence/absence of a feature, numerical information such as picture quality or screen resolution, length and width. They can

also be filled with unknown values or even implicit empty cells. All these values have to be interpreted w.r.t. variability (is the feature mandatory? optional? are there alternative choices?). Some logical relationships may also exist between column names.

- **lack of automated support:** The lack of formalization hinders the construction of tools and prevents forward efficient and systematic analyses on PCMs. It then becomes hard to propose efficient configurators that could be based on this precious information. Also, internet users, who are filling these matrices should be offered tools and/or good practice guidelines to bridge this formalization gap.
- **understandability and exploitation:** The size and complexity of data can be very important, up to hundreds of products and hundreds of features. Consequently, it can be hard to understand and exploit a PCM of such size. In practice, the more criteria or products there are, the harder the PCM is to read and the less a user can make an effective choice regarding his/her requirements.

B. Variability Models

Variability Models (VMs), like decision models or feature models, share similar goal than PCMs and provide a very synthetic and visual way to describe all possible products (also called *configurations*) in a given domain. VMs are an alternative formalism to overcome the previous identified limitations of PCMs. VMs can be employed to formulate in a more formal way the meanings of PCMs. VMs and their formal semantics enable automated reasoning: VMs come with state-of-the-art satisfiability techniques and solvers that can be used for performing assisted configurations. VMs offer an explicit and compact view of the variability and logical relationships between features.

In this paper we use a specific variability modeling formalism, called *Feature Models* (FMs) [3]–[5]. Using FMs, a family of products can be modeled in terms of mandatory, optional and exclusive features as well as propositional constraints over the features. Feature attributes with different types and range domains can also be used either to document a feature or express complex conditions over the FM.

An example of an FM for email services solutions is given in Fig. 1b. This FM aims at modeling the PCM of Fig. 1a and is one possible and simplified translation.

Choosing a configuration from an FM then means selecting and deselecting features with respect to the FMs constraints while a configuration corresponds to a given product that matches the feature selection.

C. Understanding the Gap between PCMs and VMs

PCMs and VMs are two suitable formalisms for modeling a set of products. Yet, the gap between PCMs and VMs must be precisely understood while automated techniques must support the transition between the two. In [1], the survey shows that FMs are the most popular notation and that tabular data and spreadsheets, such as PCMs, are used in a significant proportion, sometimes in combination to FMs.

Consequently, the research question we want to address is as follows: **RQ: What information is present in PCMs and what is the precise meaning of PCMs w.r.t. variability?**

To address the research question, we consider the Wikipedia case study. Wikipedia manages one of the most important source of PCMs with more than 300 PCMs from various domains and for different kinds of products.

III. An Empirical Analysis of Wikipedia PCMs

We introduce an illustrative example mined from Wikipedia and propose a first set of observations on variability information found in these PCMs. Then we evaluate the observations against a systematic analysis of 50 PCMs and perform a second automatic analysis on a larger set of PCMs.

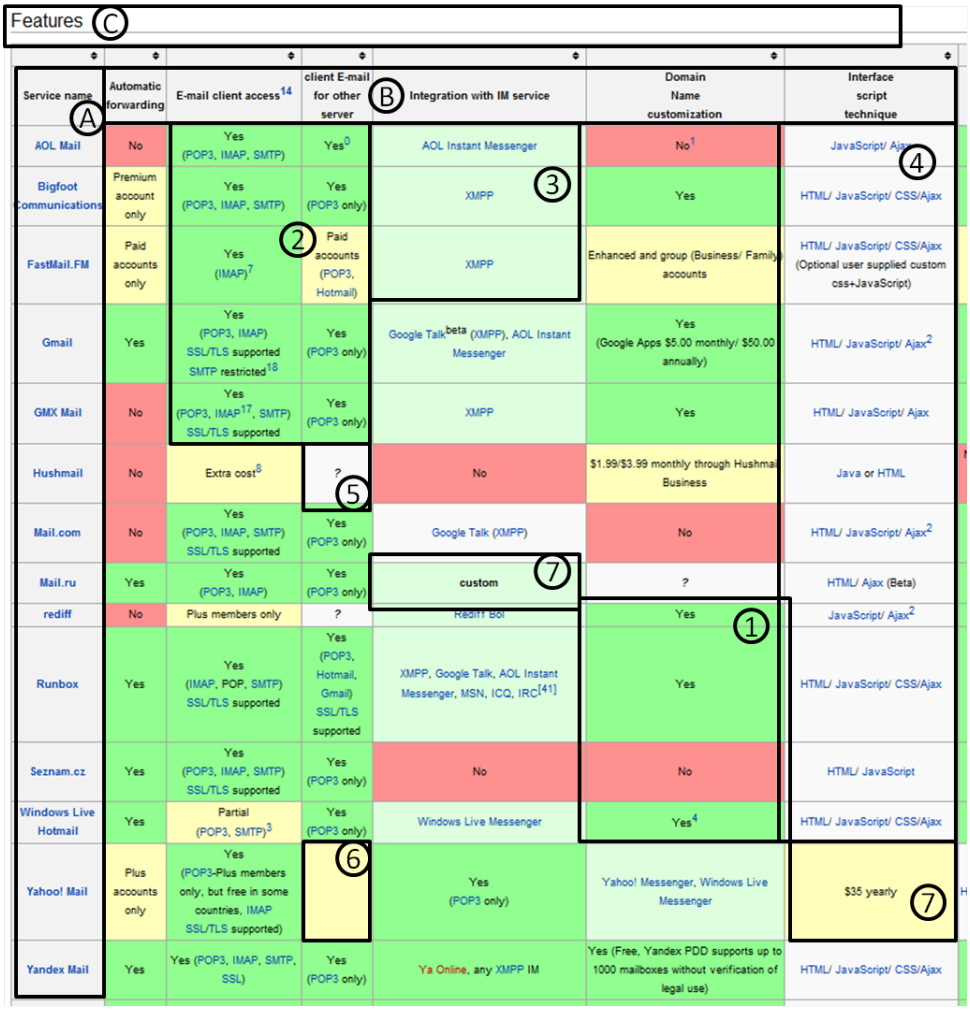
A. Anatomy of a Wikipedia PCM: a First Example

We analyze a PCM about webmail providers mined in Wikipedia[1] and present a sample of the PCM in figure 1a. This PCM compares 15 different products (Ⓐ in the figure) against 12 different criteria (Ⓑ in the figure). This Wikipedia page also proposes different comparison perspectives (Ⓒ) and, consequently, several PCMs related to these perspectives. However, our example focuses on the PCM of figure 1a, which includes 180 different cells to analyze.

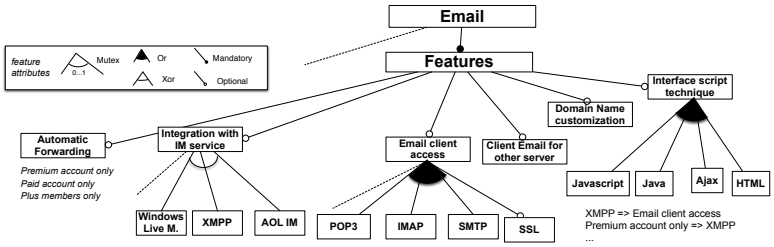
The first observation we make is related to the different comparison criteria, found as headers of the PCM. A PCM is composed of a list of heterogeneous criteria with different levels of precision and flexibility. Consequently, products values regarding these criteria can be a various kind such as:

- ① **Boolean yes/no values**. This kind of variability deals with the straight, non ambiguous, presence or absence of the comparison criteria. We observe that couples of tokens like ”yes/no”, ”true/false”, etc. are potential candidates for this kind of variability information.
- ② **Constrained/Partial/ambiguous yes/no values**. This kind of cells has to be interpreted as: ”the criterion is satisfied under the condition of, with the following limitation, etc”.”Only”, ”if”, ”through”, can be candidate words to recognize this kind of value. The token ”partial” is the most significant evidence of the presence of the value type. One can also see a ”yes” with a footnote or followed by one or several elements that express a condition or limitation.
- ③ **Single-value**. This kind of information has to be interpreted as: ”the criterion is satisfied using this element”. It forms a unique way to satisfy the criteria. The purpose of this information is not to know whether or not the criterion is satisfied but how.
- ④ **Multi-values**. This kind of information has to be interpreted as: ”the criterion is satisfied using these elements”. It forms a set of elements that contributes to satisfy the criterion. It should be noted that there is no homogeneity, within the same matrix, in the way of expressing such enumerations. A same product can be

[1] Available online at http://en.wikipedia.org/wiki/Comparison_of_webmail_providers, last access 10th may 2013



(a) Example of a Wikipedia Product Comparison Matrix



(b) A possible corresponding FM of the PCM (excerpt)

Fig. 1. A family of online emails products: PCMs and FMs

delivered with all of these elements, or deliver different versions for each element.

- ⑤ **Unknown value**. One does not know if the criterion is satisfied. Cells are generally filled with "?", "unknown". This information is rather hard to manage. It cannot be fully interpreted as a boolean "no" answer, as it can prevent the product from being selected, despite the domain reality that is unknown.
- ⑥ **Empty cell**. This information is hard to interpret, i.e., whether it should be analyzed as a strong boolean "no" and accordingly assessed as the absence of the feature or should this be analyzed as an unknown answer ?
- ⑦ **Inconsistent value**. The provided value is partial, ambiguous or lightly related to the analyzed criterion. For instance, in Figure 1a, it is mentioned that "Yahoo! Mail" has a "$35 yearly" interface, whereas all other products mention the underlying technology of their interface.
- ⑧ **Extra Information**. The provided cell value offers additional information such as latest dates, versions. Though not present in Fig 1a, this pattern exists.

It should be noted that the eight information types defined above are not necessarily expressed in a regular way for a

given criteria/header. Specifically, a same header can refer to a specific value for one product, be unknown for another one, or conditionally active in another case, etc. An example is given for the header Client access for email Server (see Fig. 1a).

Further remarks. Beyond the eight information types defined above, we report some qualitative observations. *Colors* of the cell in the PCM have a specific meaning, sometimes undocumented or even non apparent from a user perspective. In our example, "yes" values seem to correspond to green color, "no" values to red colors and "partial" values to yellow colors. This meaning is not explicit neither is systematic over PCMs. Colors can mean more than expected. For instance, software licences of a product may be documented through a specific value (e.g., LGPL or Apache license) complemented with a color. Here the color aims to characterize the kind of software licence (free or proprietary). This kind of information is usually available in the source code of a Wikipedia page.

footnotes are also worth to consider. They may influence the meaning of a cell value, e.g., restricting the validity of a cell value to particular conditions. This makes the PCM information a bit more scattered and ambiguous.

B. A Qualitative Analysis of 50 Wikipedia PCMs

We want to further confirm our intuition over PCM contents. For this purpose, we analyze a sample of 50 Wkipedia's PCMs. We selected the sample according to the following steps:

- We extracted all the pages from Wikipedia using the following search criterion: the page title must contain the following portions "comparison" or "comparison of", "comparison between". We retrieved 381 Wikipedia pages.
- We then analyzed the retrieved pages and rejected the ones that did not contain any comparative table and that were not relevant to our study. We kept 300 "relevant" pages from various domains including economy, linguistic, technology, defense, etc.
- We classified the set of candidate pages according to the number of comparison criteria they have: [1-10],[11-20] ..., [91-100], and [100+] and obtained a PCM distribution. When a comparison page contains more than one table, we consider a page with merged tables with the addition of all criteria, and a maximum value when looking at the products.
- We sampled the candidate set and randomly picked 50 Wikipedia PCMs according to the distribution to have a representative state of practice of PCMs in Wikipedia.
- we manually assessed the 50 retained pages using our catalog of 8 value types.

First part of Table I provides some global information about the 381 pages we automatically retrieved. More than half of the pages have between 1 and 20 criteria (Families 1 and 2). Surprisingly, there exist very large PCMs. 17 PCMs have more than 100 criteria. The largest comparison page is the "Comparison of Nvidia graphics processing units" with 55 different tables for a maximum of 64 products under comparison and a total amount of criteria of 1387. 11 analyzed pages contain over 1000 cells, which make these pages *understandability and usage* even harder.

TABLE II
VALUE TYPES FREQUENCIES FOR 300+ WIKIPEDIA PCMS

	①	②	③	④	⑤	⑥
amount	111309	1788	45903	33922	16823	15279
%	49.4	0.8	20.4	15.1	7.5	6.8

Table I provides a summary of our analysis of the 50 Wikipedia pages, the number of tables, cells, and values frequency[2]. These 50 pages contained 165 tables and about 29500 different cells. The 50 pages mainly deal with computer systems, architectures, programming at various levels but also include topics like linguistic, mechanics, politics, defense, among others.

We observe a large variety of value types frequencies at the individual pages level and family level (① varies from 21.02% to 73.13%, ⑤ varies from 2.54% to 27.66%). This is due to the large variety of comparison criteria and their level of precision. This heterogeneity also reflects Wikipedia's diversity in terms of domains, collaborative authors, etc.

Concerning "uncertainty", information that is not a straightforward variability information (②, ⑤, ⑥, ⑦, and ⑧), it represents a mean of 25.6%. It represents a significant number of cells that cannot stand as-is in a FM. On the other hand, around 75% of PCMs content is rather direct information and allow a direct mapping to FMs.

C. A Quantitative Analysis of 300+ Wikipedia PCMs

To gain further statistical evidence about the frequency of the eight patterns, we implemented an automated extraction process for operating over 300+ Wikipedia pages. We used the state of the art parser *Sweble* [6] to process the source of each Wikipedia page. In addition, we implemented automated techniques to recognize the pattern of a cell value, following the observations of the qualitative study. We do not seek to automatically detect patterns ⑦ and ⑧ since they are mainly based on human perception.

In total, we analyzed 31097 products and 225024 cell values. The results are reported in Table II.

We now compare the results with those previously obtained in the qualitative study. The frequency of Boolean values has slightly increased (49.4 *versus* 47.3) and still important, confirming the importance of the pattern ① . Similarly, the frequency of single values (pattern ③) remains important (slight decrease with 20.4 *versus* 22.75). The frequency of multi-values ④ has increased to a large proportion (15.1 *versus* 4.37). We can hypotesize that part of the values can actually belong to pattern ⑦ or ⑧ (two patterns we do not detect and that are usually constituted of multiple values). The frequency of pattern ② has decreased significantly (0.8 *versus* 3.71) but still constitutes a minor pattern.

[2]More detailed information for each page is available online at http://tinyurl.com/WikipediaPCM

TABLE I
EVALUATION OF 50 WIKIPEDIA PCMS

	retrieved documents	distribution	# PCMs in the 50 set	# tables	# cells	Value Type %							
						1	2	3	4	5	6	7	8
Family 1	102	34 %	17	21	2226	21,02%	2,02%	31,85%	9,61%	9,16%	11,37%	1,26%	13,70%
Family 2	71	23,67%	12	24	4576	19,62%	1,14%	42,33%	6,86%	11,76%	10,29%	0,44%	7,56%
Family 3	32	10,67%	5	20	1874	52,99%	2,08%	25,77%	6,19%	4,06%	2,51%	1,65%	4,75%
Family 4	34	11,33%	6	23	3487	25,67%	5,05%	40,78%	10,35%	4,96%	4,39%	0,52%	8,29%
Family 5	15	5 %	2	10	2462	50,28%	1,30%	17,67%	6,62%	6,34%	11,33%	0,12%	6,34%
Family 6	12	4 %	2	11	1733	54,07%	9,41%	19,22%	1,44%	13,91%	0,00%	0,63%	1,33%
Family 7	6	2%	1	5	1822	73,16%	5,27%	4,56%	0,38%	13,23%	0,05%	0,38%	2,96%
Family 8	4	1,33%	1	10	1965	68,60%	5,39%	15,17%	1,32%	2,54%	3,46%	0,10%	3,41%
Family 9	6	2%	1	6	2840	73,13%	4,37%	8,13%	1,02%	8,42%	0,00%	1,20%	3,73%
Family 10	5	1,67%	1	11	2162	53,56%	7,31%	6,24%	0,42%	27,66%	1,53%	0,05%	3,24%
Family 11	13	4,33%	2	24	4312	59,97%	2,39%	14,73%	0,56%	15,84%	2,74%	0,16%	3,62%
Total	300		50	165	29459	47,29%	3,71%	22,75%	4,37%	10,86%	4,83%	0,55%	5,64%

The most important result is that we confirm patterns ① and ③ are by far the most widely used, constituting almost 75% of the content of PCMs.

IV. Implications of Findings and Future Work

A. Towards Better Management of PCMs

As we hypothesized in Section II, our empirical study of Wikipedia PCMs show that they suffer from different drawbacks. The most important of them is the lack of formalization and the latent ambiguity of some cells in PCMs (around 25% in our qualitative study), which have to be heavily interpreted.

This limitation can be explained as these PCMs do not initially target formal analysis or automatic tool support. They emerge as an open initiative from different internet end users. This positive behavior is however limited by the lack of methodology and homogeneity when *contributors* fill the Wikipedia form. As there exist no canvas or framework to assist the contributors are free to build the PCM they want, disregarding the quality, granularity, or precision of the information they provide. This set an important challenge to propose a flexible yet formal canvas to fill and maintain PCMs for:

- **end users** who consult PCMs, would benefit from such improvements as PCM information could then be more homogeneous, less ambiguous, in other words more readable ;
- **contributors** can benefit from such flexible frameworks and be guided in the authoring task, provide better information, while keeping the freedom inherent of the open initiative world. Such framework should provide information that is: (1) easier to analyze and review, (2) easier to maintain and update. Ideally, the solution should be non intrusive w.r.t. existing languages and tooling support offered by the Wikipedia initiative ;
- **developers** can benefit from better PCMs as it would be easier to provide tool support. Ad hoc tools exist to analyze and mine the source code of such pages, but none of them are generic enough to propose quick, practical and efficient analyses of these PCMs.

B. Towards Deriving Configurators

A second limitation comes from the Wikipedia table and page format itself. Tables are a very simple way to present this kind of information. Easy to create, easy to read. However it does hardly scale up in terms of readability and understandability. The more features or the more products there are, the larger the PCM is. Authors typically split their PCMs for the sake of readability, but also scatter the information.

From a configuration perspective, end users should benefit from the interactive choice of selecting their features of interest and progressively filter and narrow their configuration space, possibly in multiple steps. This sets the challenge to propose a complete tool chain from these PCMs, to their formalization and usage through efficient *configurators* that guide users to desirable - and valid - product configurations [7].

To promote efficient usages, we also have to guide the user in the configuration task. A configurator cannot propose a full set of hundreds of unordered features. A configurator should be able to identify and propose the main features of interest. Choosing a particular feature can be very discriminant in the configuration process, imposing important configuration limitations, or not, when the choice is common.

In Section III, we analyzed that around 25% of the cells were containing uncertain information. This information is very different from the traditional binary (all-or-nothing) approach while using variability models. We also have to deal with uncertainty at the configuration level where a user can select feature with different degrees of desirability or maintain uncertainty to avoid early restrictions and provide more diverse choices and just-in-time decisions.

V. Threats to Validity

External threats. Our study is based on an empirical analysis of Wikipedia comparison pages. We analyzed the complete set of candidate comparison pages (381), compute their distribution into different families (11 analyzable families) and respect the distribution in our randomly chosen 50 pages analysis set. This allows us to have a global independent and diverse analysis set. However, we have not evaluated the relevance of Wikipedia as a representative and generalizable case study for the whole PCM domain. We plan to extend the

study to the numerous PCMs available on the internet and also consider PCMs specified in an industrial context.

Internal threats. We identified and evaluated eight variability patterns based on a mix between manual inspection and automated techniques. To reduce the user effort, we restrict the scope of the study to a manual investigation of 50 Wikipedia pages. Yet, we may have missed some patterns or performed incorrect qualitative analyses. We encourage other researchers to replicate our study and made our studies available.

VI. Related Work

Spreadsheets and PCMs. Errors in spreadsheet are common but non trivial and considerable research effort has been devoted to the study of spreadsheets [8]. The effort is still ongoing around automated techniques for fault localization and guidelines toward well maintainable spreadsheets [9]–[12]. PCMs can be seen as a special form of spreadsheets with a different domain. Whereas previous work aim at tackling programming errors or code smells in spreadsheets, we address here variability information and provide qualitative and quantitative evidences that PCMs from Wikipedia suffer as well from major drawbacks, calling for more research.

Reverse Engineering FMs. In [13], we proposed a semi-automated procedure to support the transition from product descriptions (expressed in a PCM) to FMs. Our initial study was rather informal and conducted on a synthetic and limited data sample. In particular we missed four variability patterns. ① is a merged of the initial "mandatory" and "dead feature" patterns. ② is an extension of the "optional" that did not initially take into account constraints or partial answers. ③ and ④ are identical with real value and multi-value patterns. ⑤, ⑥, ⑦ and ⑧ are new patterns, dealing with the current state of practice and lack of formalization of PCMs.

The product line research community has shown significant interest in the ability to automatically generate (boolean) FMs from existing data. As established by our empirical study, information in PCMs contains more than simple Boolean values and products themselves have variability. Dumitru et al. [14] and Czarnecki et al. [15] for instance, rely on product-by-feature matrices, but with boolean features. FM synthesis techniques [4], [13], [16], [17], [17]–[21] should be revised accordingly, for example, to take numeric values into account.

VII. Conclusion

PCMs potentially provide lots of rich and useful information but present many drawbacks such as lack of formalization, lack of tool support and understandability. One possibility to tackle these concerns is to translate PCMs into feature models (FMs), giving a clear semantics and enabling the automatic analysis of a family of product.

We proposed a catalog of 8 variability information, retrieved in Wikipedia PCMs, and conducted a qualitative and automatic analysis of a set of 300 Wikipedia comparison pages that allowed us to evaluate and understand the gap between PCMs and FMs. Around 75% of PCMs content can be directly translated to Boolean-based FMs but the handling of numeric attributes or uncertainty requires more effort to fit with the current state of practice of PCMs.

In the middle of the bridge between PCM makers and PCM users, perspectives are many. The major challenge is to be able to leverage the community's collaborative work and to propose tools to create, maintain, navigate more efficiently in PCMs. Another research direction is to build configurators that assist end-users in selecting a product via a dependable and controlled feature selection process.

Acknowledgements

This work is partially supported by the French BGLE Project CONNEXION, and the French Ministry of Defense project MOTIV.

References

[1] T. Berger, R. Rublack, D. Nair, J. M. Atlee, M. Becker, K. Czarnecki, and A. Wasowski, "A survey of variability modeling in industrial practice," in *VAMOS'13*. ACM, 2013, p. 7.

[2] D. Benavides, S. Segura, and A. Ruiz-Cortes, "Automated analysis of feature models 20 years later: a literature review," *Information Systems*, vol. 35, no. 6, 2010.

[3] S. Apel and C. Kästner, "An overview of feature-oriented software development," *Journal of Object Technology (JOT)*, vol. 8, no. 5, pp. 49–84, July/August 2009.

[4] K. Czarnecki and A. Wasowski, "Feature diagrams and logics: There and back again," in *SPLC'07*. IEEE, 2007, pp. 23–34.

[5] T. Thüm, D. Batory, and C. Kästner, "Reasoning about edits to feature models," in *ICSE'09*. ACM, 2009, pp. 254–264.

[6] H. Dohrn and D. Riehle, "Design and implementation of the sweble wikitext parser: unlocking the structured data of wikipedia," in *WikiSym'11*, ser. WikiSym '11. ACM, 2011, pp. 72–81.

[7] E. Khalil Abbasi, A. Hubaux, M. Acher, Q. Boucher, and P. Heymans, "The anatomy of a sales configurator: An empirical study of 111 cases," in *CAiSE'13*, ser. LNCS, vol. 7908, 2013, pp. 162–177.

[8] R. R. Panko, "Thinking is bad: Implications of human error research for spreadsheet research and practice," *CoRR*, vol. abs/0801.3114, 2008.

[9] R. Abraham and M. Erwig, "Ucheck: A spreadsheet type checker for end users," *J. Vis. Lang. Comput.*, vol. 18, no. 1, pp. 71–95, 2007.

[10] F. Hermans, M. Pinzger, and A. van Deursen, "Automatically extracting class diagrams from spreadsheets," in *ECOOP'10*, ser. LNCS, vol. 6183. Springer-Verlag, 2010, pp. 52–75.

[11] J. Cunha, J. Visser, T. L. Alves, and J. Saraiva, "Type-safe evolution of spreadsheets," in *FASE'11*, ser. LNCS, vol. 6603, 2011, pp. 186–201.

[12] F. Hermans, M. Pinzger, and A. v. Deursen, "Detecting and visualizing inter-worksheet smells in spreadsheets," in *ICSE'12*. IEEE, 2012, pp. 441–451.

[13] M. Acher, A. Cleve, G. Perrouin, P. Heymans, C. Vanbeneden, P. Collet, and P. Lahire, "On extracting feature models from product descriptions," in *VaMoS'12*. ACM, 2012, pp. 45–54.

[14] H. Dumitru, M. Gibiec, N. Hariri, J. Cleland-Huang, B. Mobasher, C. Castro-Herrera, and M. Mirakhorli, "On-demand feature recommendations derived from mining public product descriptions," in *ICSE'11*. ACM, 2011, pp. 181–190.

[15] K. Czarnecki, S. She, and A. Wasowski, "Sample spaces and feature models: There and back again," in *SPLC'08*, 2008, pp. 22–31.

[16] E. N. Haslinger, R. E. Lopez-Herrejon, and A. Egyed, "On extracting feature models from sets of valid feature combinations," in *FASE'13*, ser. LNCS, vol. 7793, 2013, pp. 53–67.

[17] U. Ryssel, J. Ploennigs, and K. Kabitzsch, "Extraction of feature models from formal contexts," in *FOSD'11*, 2011, pp. 1–8.

[18] S. She, R. Lotufo, T. Berger, A. Wasowski, and K. Czarnecki, "Reverse engineering feature models," in *ICSE'11*. ACM, 2011, pp. 461–470.

[19] N. Andersen, K. Czarnecki, S. She, and A. Wasowski, "Efficient synthesis of feature models," in *SPLC'12*, 2012, pp. 106–115.

[20] M. Acher, B. Baudry, P. Heymans, A. Cleve, and J.-L. Hainaut, "Support for reverse engineering and maintaining feature models," in *VaMoS'13*. ACM, 2013, p. 20.

[21] J.-M. Davril, E. Delfosse, N. Hariri, M. Acher, J. Cleland-Huang, and P. Heymans, "Feature model extraction from large collections of informal product descriptions," in *ESEC/FSE'13*, 2013.