Information and Software Technology 72 (2016) 1–15

Contents lists available at ScienceDirect

Information and Software Technology

journal homepage: www.elsevier.com/locate/infsof

A bibliometric analysis of 20 years of research on software product lines

Ruben Heradio[a,*], Hector Perez-Morago[a], David Fernandez-Amoros[a], Francisco Javier Cabrerizo[a], Enrique Herrera-Viedma[b]

[a] *Department of Software Engineering and Computer Systems, Universidad Nacional de Educacion a Distancia (UNED), Madrid 28040, Spain*
[b] *Department of Computer Science and Artificial Intelligence, University of Granada, Granada 18071, Spain*

ARTICLE INFO

Article history:
Received 11 May 2015
Revised 11 November 2015
Accepted 12 November 2015
Available online 2 December 2015

Keywords:
Software product lines
Bibliometrics
Science mapping
Performance analysis

ABSTRACT

Context: Software product line engineering has proven to be an efficient paradigm to developing families of similar software systems at lower costs, in shorter time, and with higher quality.

Objective: This paper analyzes the literature on product lines from 1995 to 2014, identifying the most influential publications, the most researched topics, and how the interest in those topics has evolved along the way.

Method: Bibliographic data have been gathered from ISI Web of Science and Scopus. The data have been examined using two prominent bibliometric approaches: science mapping and performance analysis.

Results: According to the study carried out, (i) software architecture was the initial motor of research in SPL; (ii) work on systematic software reuse has been essential for the development of the area; and (iii) feature modeling has been the most important topic for the last fifteen years, having the best evolution behavior in terms of number of published papers and received citations.

Conclusion: Science mapping has been used to identify the main researched topics, the evolution of the interest in those topics and the relationships among topics. Performance analysis has been used to recognize the most influential papers, the journals and conferences that have published most papers, how numerous is the literature on product lines and what is its distribution over time.

1. Introduction

A *Software Product Line* (SPL) is an engineering approach to efficient development of whole portfolios of software products [1]. The basis of the approach is that products, instead of being developed from scratch one by one, are built from a core asset base, i.e., a collection of artifacts that have been designed specifically for use across the portfolio.

The SPL approach brings the benefits of economies of scope to software engineering, since less time and effort are needed to produce a greater variety of products. Many companies have exploited the concept of software product lines to increase the resources that focus on highly differentiating functionality and thus improve their competitiveness with higher quality and reusable products while decreasing the time-to-market condition. For instance, van der Linden et al. [2] summarize experience reports from ten different companies working on diverse domains (e.g., Bosch on Gasoline Systems, Nokia on Mobile Phones, Philips on Consumer Electronics Software for Televisions, Siemens on Medical Solutions, etc.).

The goal of this paper is to analyze, using *bibliometric* techniques, the literature on SPLs for the last twenty years in order to determine the main topics and trends of this research area. The outcomes of our analysis provide information regarding the following issues:

1. What are the most influential papers on SPL literature?
2. Who are the most prolific authors?
3. What journals, conferences, etc. have published the majority of the papers?
4. How numerous is the SPL literature? How has paper publication been distributed over time?
5. What are the main topics studied in the area? How has the interest in those topics evolved with time?
6. What are the most impacting papers for a given a topic along a certain period of time?

We have processed 2845 records retrieved from ISI Web of Science (ISIWoS) and Scopus by using two approaches to examine bibliographic data: *performance analysis* [3] and *science mapping* [4].

* Corresponding author. Tel.: +34 913988242.
E-mail addresses: rheradio@issi.uned.es, rheradio@gmail.com (R. Heradio), hperez@issi.uned.es (H. Perez-Morago), david@issi.uned.es (D. Fernandez-Amoros), cabrerizo@issi.uned.es (F. Javier Cabrerizo), viedma@decsai.ugr.es (E. Herrera-Viedma).

http://dx.doi.org/10.1016/j.infsof.2015.11.004

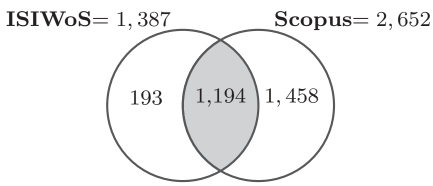


Fig. 1. Number of records retrieved from ISIWoS and Scopus.

Performance analysis tries to quantify the impact of scientific actors (researchers, journals, etc.) on a research field by measuring how often a paper is cited. In particular, this paper uses *H-index* [5] to measure publication impact, which is one of the most popular indicators for citation analysis.

Science mapping attempts to display the structural and dynamic aspects of scientific research, delimiting a research field, and identifying, quantifying and visualizing its thematic subfields. To map the SPL research area, we have used a technique called *co-word analysis*, that measures the association strengths of publication keywords [6].

Our work uses both performance analysis and science mapping in a complementary way. Bibliometric maps supports visualizing the main topics in the literature, their evolutions and inter-relationships. Performance analysis helps to identify the most productive topics (in terms of number of published papers), and the most impacting ones (according to received citations).

The remainder of the paper is structured as follows: Section 2 summarizes the methodology and techniques used to carry out our work. Section 3 reports the performance analysis of the whole SPL research area, identifying those papers that due to their number of citations should be considered as *classics*. Section 4 applies science mapping techniques to identify the cognitive structure and evolution of the SPL literature. Section 5 discusses threats to the validity of our work. Finally, some concluding remarks are provided in Section 6.

2. Materials and methods

To perform the bibliometric analysis described in this paper, the workflow proposed in [7,8] has been followed, which is composed of the following steps:

(1) **Data retrieval.** The data processed in this paper come from ISIWoS and Scopus, which are the most reliable bibliographic databases at the moment [9,10]. On September 2015, the following query[1] was made on Scopus and the ISIWoS Core Collection for the time span 1995–2014:

```
TOPIC =
''software product line*'' or
(
(
''product line*'' or ''mass customization'' or ''product famil*'' or
''program famil*'' or ''software factor*'' or ''product platform*''
) and
(
''domain engineering'' or ''application engineering'' or
''feature model*'' or ''feature diagram*'' or
''decision model*'' or ''decision diagram*'' or
(software and variabilit*)
)
)
```

Venn diagram in Fig. 1 depicts the number of publication records provided by each database: 1387 records from ISIWoS and 2652 records from Scopus. 1194 of those records were common for both Scopus and ISIWos. Thus, the total number of records processed in this paper is 2845.

(2) **Data aggregation.** The scatter plot in Fig. 2 shows the number of citations of the records common to ISIWoS and Scopus, including the corresponding regression line as well. ISIWoS has a more selective procedure to include bibliographic references than Scopus. As a result, Scopus provides more records than ISIWoS, and the citations tend to be higher as well.
For the query we performed, the Pearson's correlation coefficient of the citation counts is 0.87. So the information provided by both databases is rather consistent.
To combine the records, the citation count for the common records was computed as the maximum of the citations given by ISIWoS and Scopus.

(3) **Preprocessing.** The data retrieved from bibliographic databases usually have errors. For instance, references may be duplicated, authors' names may appear in different ways, etc. So, it is necessary to preprocess the data before carrying out any analysis.
To track the evolution of the SPL research area and measure its performance, we have used two approaches that require analyzing publication keywords and citations: Co-Word Analysis and H-index. Hence, we have performed a laborious preprocessing procedure to
   (1) *Correct invalid citations*; e.g., the technical report [11] appears cited in the raw data gathered from ISIWoS as "Bachman F., 2005, CMUSEI2005 TR012" and "Bachma**n**n F., 2005, CMUSEI2005TR012".
   (2) *Standardize keywords*. From the ISIWoS records, a set of 2000 keywords was available, 1667 were authors' keywords and 333 were words provided by ISIWoS *KeyWords Plus* (index terms created by Thomson Reuters from significant, frequently occurring words in the titles of an article cited references). The Scopus records included a set of 9308 keywords. As Fig. 3 summarizes, the initial aggregated set of 11,308 keywords was progressively reduced by applying the following steps:
      (a) Keywords were converted to uppercase, leading and trailing white-spaces were removed, and inner white-spaces were replaced by the character '-'. After that, the repeated keywords were removed and plurals were grouped.
      (b) Keywords useless to identify research topics inside the SPL area were discarded. For example, SOFTWARE-PRODUCT-LINE, PRODUCT-FAMILY, SOFTWARE-ENGINEERING, etc. are applicable to all the records and thus they cannot be used to distinguish particular topics. Therefore, those general keywords were removed.
      (c) Keywords were grouped according to their meaning. To improve the interpretability of the co-word analysis results, the set of keywords was reduced by grouping those words that refer to the same topic. For instance, STAGED-CONFIGURATION, AUTOMATED-CONFIGURATION, FEATURE-BASED-CONFIGURATION, PRODUCT-DERIVATION-TOOL, etc. were grouped as PRODUCT-DERIVATION.

(4) **Analysis.** There are two main approaches to examine bibliographic data [12]:
   (1) *Performance analysis* tries to quantify the impact of scientific actors (researchers, journals, etc.). In particular, the performance analysis we have carried out is based on H-index, which is introduced in Section 2.1.
   (2) *Science Mapping* looks for identifying the cognitive structure and evolution of a research field. Section 2.2 summarizes the techniques used in this paper for mapping the SPL research area.

[1] The asterisk pattern character means zero to many characters; it is used in our query to catch the noun plurals.

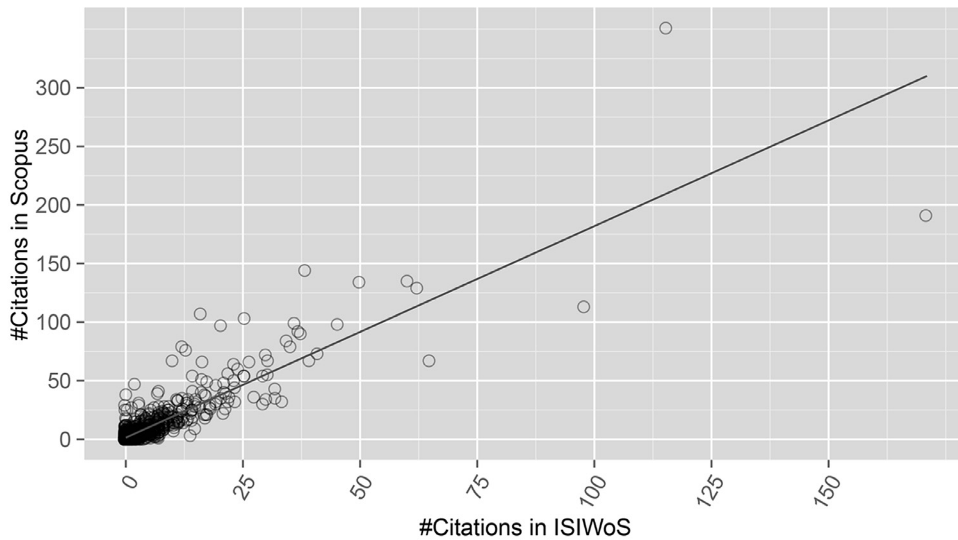


Fig. 2. Number of citations for publications common to ISIWoS and Scopus.

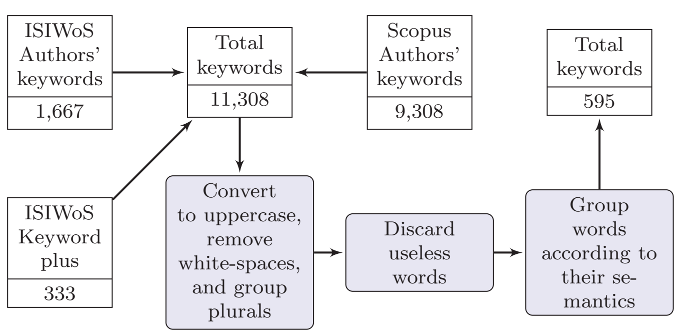


Fig. 3. Summary of the keyword standardization.

To preprocess and analyze the bibliographic data, the open source software tool SciMAT [13] has been used, which is freely available at http://sci2s.ugr.es/scimat/.

2.1. Performance analysis: H-index

The main approach to evaluate research performance is *paper citation analysis* [14], whose main tenet is "the more often a paper becomes cited the greater its influence on the field" [15]. One of the most popular indicators for citation analysis is *H-index*, which was introduced by Hirsch [5] to measure the scientific performance of a researcher through her publications. Its original definition is

"a scientist has index h if h of her n papers have at least h citations each, and the other $(n - h)$ papers have $\leq h$ citations each".

Compared to alternative approaches, H-index has the following advantages:

- It comprises in a single indicator two aspects that traditionally have been measured separately using different indicators: the quantity and the impact of the scientific output of a researcher.
- It has been proven to be robust in the sense that it is insensitive to: (i) a few outstandingly highly cited papers (increasing the H-index requires receiving citations in a large number of papers) [16], and (ii) a set of lowly cited papers (which is usually considered as a drawback)
- It is easy to compute from the citation data available through the scientific databases as Web of Knowledge and Scopus.

Garfield [3] defined the concept of *citation classics* to identify the papers most frequently cited. Analyzing the citation classics of a research field (i) helps recognizing the major advances in the discipline [17], (ii) provides historical perspective of its scientific progress [18,19], and (iii) identifies the main intellectual actors of the research field [20], including not only researchers but also prominent journals, conferences, etc.

Martinez et al. [21] proposed adapting the H-index definition as follows to support the analysis of citation classics:

"a research area has index h, if h of the n papers published in the area have at least h citations each, and the other $(n - h)$ papers have $\leq h$ citations each".

The top h papers of a research area constitute its *H-core*, which identifies the highest-performance publications of the area. An alternative procedure to identify citation classics is establishing a threshold for the number of cites so that a paper is considered classic if it has more citations than the threshold. Unfortunately, the threshold differs for each discipline because the citation counts of a scientific field depend on many factors such as aging of the area, citation distribution, publication and citation practices, the activity rate of the scientific community, the number of scientists, channels of information dissemination, etc [22]. In contrast, H-core is insensitive to the dimension of the research field [5].

2.2. Science mapping

Science mapping can be used to get three kind of maps: thematic networks, strategic diagrams, and maps of conceptual evolution.

2.2.1. Thematic networks

In this paper, a particular science mapping approach known as co-word analysis has been used to identify the main topics of a scientific field and their inter-relationship. It measures the association strengths of terms representative of the publications in the field by analyzing the co-occurrence frequency of pairs of keywords. Several measures have been proposed to estimate the association strength between publication keywords. Van Eck et al. [23] performed an analysis of many of those measures, concluding that *equivalence index* [4] is the most appropriate one for normalizing co-occurrence frequencies. In bibliometrics, equivalence index is also known as *proximity index* [24], *probabilistic affinity index* [25], and *association strength* [6,26].

The equivalence index $e_{A\leftrightarrows B}$ of two keywords A and B is defined by Eq. (1), where $c_{A\leftrightarrows B}$ is the number of publications where A and

B appear together, and c_A and c_B are the total number of documents that include A and B, respectively.

$$e_{A\leftrightarrows B} = \frac{c^2_{A\leftrightarrows B}}{c_A \cdot c_B} \quad (1)$$

The value of $e_{A\leftrightarrows B}$ falls into the interval [0..1]:

- When there is no publication where A and B appear together as keywords, $c_{A\leftrightarrows B} = 0$, and thus $e_{A\leftrightarrows B} = 0$.
- If A and B are keywords that appear always together, $c_A = c_B = c_{A\leftrightarrows B}$, and thus $e_{A\leftrightarrows B} = 1$.

A clustering algorithm can use the equivalence index to identify research topics by looking for groups of strongly linked keywords [4,27,28]. In this paper, the algorithm of *simple centers* [4] has been used to get the clusters, each one representing a topic. This algorithm has been successfully used in a number co-word studies [6,29–33] and has the advantage of producing labeled clusters (in contrast to other alternative approaches that generate unlabeled clusters, which need to be revised by an expert to identify the topics they represent).

Since the equivalence index has the drawback of being high when keywords appear infrequently but almost always together, a clustering algorithm guided by that index might overestimate the importance of those keywords and create irrelevant clusters [6]. To overcome that problem, the simple center algorithm needs to be adjusted through the following parameters:

- The minimum number of publications where keywords are required to appear ($\text{min}_{\text{occurrences}}$).
- The minimum number of publications where a pair of keywords need to appear together to be taken into account ($\text{min}_{\text{co · occurrences}}$). If this parameter is too high, few links may be formed; if it is too low, an excessive number of links may result. In the former case, subspecialities in a field may not emerge; in the latter case, most representative and well-connected topics will be harder to detect due to the noise of less representative and less well-connected ones [6].
- The minimum and maximum number of keywords a cluster can group ($\text{min}_{\text{keywords}}$ and $\text{max}_{\text{keywords}}$). These parameters set the size of the clusters.

Algorithm 1 summarizes in pseudocode how the approach works. In such a way, each detected topic is modeled by a cluster of interrelated keywords known as *thematic network*.

Algorithm 1: simple_centers.

Input $\text{min}_{\text{occurrences}}$; $\text{min}_{\text{co·occurrences}}$; $\text{min}_{\text{keywords}}$; $\text{max}_{\text{keywords}}$;
Output *set of clusters;*
begin
  Remove all keywords included in less than $\text{min}_{\text{occurrences}}$ publications;
  repeat
    Get the link with highest $e_{A\leftrightarrows B}$ from all possible keywords to begin a cluster;
    From that link, form other links in a breadth-first manner, until no more links are possible due to $\text{min}_{\text{co·occurrences}}$ or $\text{min}_{\text{keywords}}$ or $\text{max}_{\text{keywords}}$;
    The keyword which participates in more links is considered the cluster center and so it provides the cluster name;
    Remove all incorporated keywords from the list of subsequent available keywords;
  until *No two remaining keywords co-occur frequently enough to begin a cluster;*

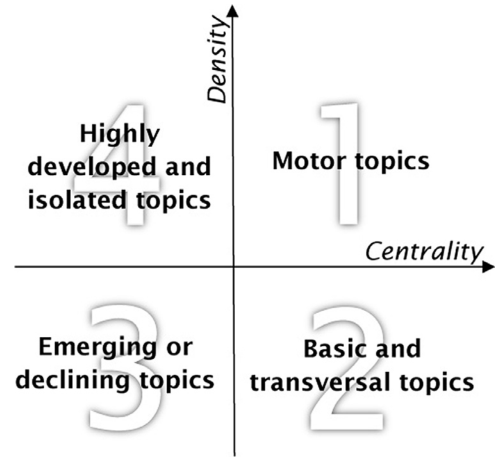


Fig. 4. Distribution of topic roles in a strategic diagram.

2.2.2. Strategic diagrams

The role that a topic plays in a research area may be characterized using the following measures:

- Cluster's *density*, which estimates its internal coherence by measuring the strength of the links that tie together the keywords of the cluster [4]. Density may be computed with Eq. (2), where A and B are keywords belonging to the cluster, and #cluster is the number of keywords of the cluster.

$$\text{density}_{\text{cluster}} = \frac{100}{\#\text{cluster}} \cdot \sum_{A,B \in \text{cluster}} e_{A\leftrightarrows B} \quad (2)$$

- Cluster's *centrality*, which measures its degree of interaction with other clusters. The greater the number and strength of the links of a topic to other clusters, the more central this topic is in the research field [34]. Centrality may be computed with Eq. (3), where A and B are keywords inside and outside the cluster, respectively [4].

$$\text{centrality}_{\text{cluster}} = 10 \cdot \sum_{A \in \text{cluster}, B \notin \text{cluster}} e_{A\leftrightarrows B} \quad (3)$$

To get a global representation of the role of all topics, a *strategic diagram* can be used [35], which is a two-dimensional layout where the x-axis and y-axis represent cluster's centrality and density, respectively. As Fig. 4 shows, topics may be classified according to the quadrant where they are placed in a strategic diagram [4,36,37]:

Quadrant 1 includes topics both well developed and important for the research field. Due to their high centrality and density, those topics are usually known as *motor topics*.
Quadrant 2 contains important but weakly structured topics. They could be transversal topics or underdeveloped topics of considerable significance for the entire research field [33].
Quadrant 3 has both weakly developed and marginal topics, representing emerging or disappearing topics.
Quadrant 4 includes topics that have well developed internal links but unimportant external ties. Since those topics are internally well structured, they indicate that a constituted social group is active in them. Nevertheless, they should be considered peripheral to the work being performed in the global research field [29].

2.2.3. Maps of conceptual evolution

The analysis of co-citation clusters over consecutive periods of years can be used to track the emergence and growth of research

Table 1
Citation classics.

	Type of source	Year of publication	#Citations	Thematic network
Benavides et al.: Automated analysis of feature models 20 years later: a literature review [43]	Journal	2010	352	AUT-ANLYS
Czarnecki et al.: Formalizing cardinality-based feature models and their specialization [44]	Journal	2005	255	FEATURE-MOD
Czarnecki et al.: Staged configuration through specialization and multilevel configuration of feature models [45]	Journal	2005	227	FEATURE-MOD
Batory: Feature models, grammars, and propositional formulas [46]	Conf.	2005	192	FEATURE-MOD
Coplien et al.: Commonality and Variability in Software Engineering [47]	Journal.	1998	171	SW-ARCH
Smaragdakis et al.: Mixin Layers: An Object-oriented Implementation Technique for Refinements and Collaboration-based Designs [48]	Journal.	2002	157	SW-ARCH
Kästner et al.: Granularity in Software Product Lines [49]	Conf.	2008	144	SW-ARCH
Hallsteinsen et al.: Dynamic software product lines [50]	Journal	2008	135	SW-ARCH
Figueiredo et al.: Evolving software product lines with aspects: an empirical study on design stability [51]	Conf.	2008	134	SW-ARCH
Schobbens et al.: Generic semantics of feature diagrams [52]	Journal	2007	129	FEATURE-MOD
Benavides et al.: Automated reasoning on feature models [53]	Conf.	2005	113	FEATURE-MOD
Schobbens et al.: Feature Diagrams: A survey and a formal semantics [54]	Conf.	2006	107	FEATURE-MOD
Czarnecki et al.: Feature diagrams and logics: There and back again [55]	Conf.	2007	103	FEATURE-MOD
Thaker et al.: Safe Composition of Product Lines [56]	Conf.	2007	103	FEATURE-MOD
Böckle et al.: Calculating ROI for software product lines [57]	Conf.	2004	99	SW-ARCH
Apel et al.: Aspectual feature modules [58]	Journal	2008	98	SW-ARCH, FEATURE-MOD
Classen et al.: Model Checking Lots of Systems: Efficient Verification of Temporal Properties in Software Product Lines [59]	Conf.	2010	98	AUT-ANLYS
Schaefer et al.: Delta-oriented Programming of Software Product Lines [60]	Conf.	2010	97	VARIABILITY-MANAGEMENT, SOFTWARE-DESIGN
Svahnberg et al.: A Taxonomy of Variability Realization Techniques [61]	Journal	2005	95	SW-ARCH, FEATURE-MOD
Czarnecki et al.: Verifying Feature-based Model Templates Against Well-formedness OCL Constraints [62]	Conf.	2006	94	FEATURE-MOD
Metzger et al.: Disambiguating the documentation of variability in software product lines: A separation of concerns, formalization and automated analysis [63]	Conf.	2007	92	FEATURE-MOD
Thüm et al.: Reasoning about edits to feature models [64]	Conf.	2009	90	FEATURE-MOD
Engström et al.: Software product line testing-a systematic mapping study [65]	Journal	2011	84	SW-REUSE
Voelter et al.: Product Line Implementation Using Aspect-Oriented and Model-Driven Software Development [66]	Conf.	2007	79	SW-ARCH, FEATURE-MOD
Bosch et al.: From integration to composition: On the impact of software product lines, global development and ecosystems [67]	Journal	2010	79	SW-DESIGN
Kästner et al.: A Case Study Implementing Features Using AspectJ [68]	Conf.	2007	76	SW-ARCH, FEATURE-MOD
Moon et al.: An approach to developing domain requirements as a core asset based on commonality and variability analysis in a product line [69]	Journal	2005	73	DOMAIN-ENG
Batory et al.: Automated Analysis of Feature Models: Challenges Ahead [70]	Journal	2006	72	FEATURE-MOD
She et al.: Reverse engineering feature models [71]	Conf.	2011	72	AUT-ANLYS
Svahnberg et al.: Evolution in software product lines: Two cases [72]	Journal	1999	67	SW-ARCH
Schmid et al.: A comprehensive product line scoping approach and its validation [73]	Conf.	2002	67	SW-REUSE, SW-ARCH
Northrop: SEI's software product line tenets [74]	Journal	2002	67	SW-REUSE
Czarnecki et al.: Staged configuration using feature models [75]	Conf.	2004	67	FEATURE-MOD
Heidenreich et al.: FeatureMapper: Mapping Features to Models [76]	Conf.	2008	66	FEATURE-MOD
Apel et al.: Type safety for feature-oriented product lines [77]	Journal	2010	66	AUT-ANLYS
Classen et al.: Symbolic model checking of software product lines [78]	C.	2011	64	AUT-ANLYS
Lee et al.: Concepts and Guidelines of Feature Modeling for Product Line Software Engineering [79]	Conf.	2002	64	FEATURE-MOD
Beuche et al.: Variability management with feature models [80]	Journal	2004	61	FEATURE-MOD
Mietzner et al.: Variability modeling to support customization and deployment of multi-tenant-aware Software as a Service applications [81]	Conf.	2009	60	FEATURE-MOD
Liebig et al.: An Analysis of the Variability in Forty Preprocessor-based Software Product Lines [82]	Conf.	2010	58	VARIABILITY-MANAGEMENT
da Mota et al.: A systematic mapping study of software product lines testing [83]	Journal	2011	56	SW-REUSE
Birk et al.: Product line engineering: The state of the practice [84]	Journal	2003	55	SW-ARCH
Alves et al.: Refactoring product lines [85]	Conf.	2006	55	FEATURE-MOD
Fischbein et al.: A foundation for behavioural conformance in software product line architectures [86]	Conf.	2006	55	SW-ARCH
Mendonca et al.: S.P.L.O.T. - Software product lines online tools [87]	Conf.	2009	55	FEATURE-MOD
van Ommering: Software reuse in product populations [88]	Journal	2005	54	DOMAIN-ENG, SW-ARCH
Asikainen et al.: Kumbang: A domain ontology for modelling variability in software product families [89]	Journal	2007	54	DOMAIN-ENG
Dhungana et al.: The DOPLER meta-tool for decision-oriented variability modeling: A multiple case study [90]	Journal	2011	54	AUT-ANLYS, PRODUCT-DERIVATION
Chen et al.: A systematic review of evaluation of variability management approaches in software product lines [91]	Journal	2011	54	VARIABILITY-MANAGEMENT
Berger et al.: Variability modeling in the real: A perspective from the operating systems domain [92]	Conf.	2011	53	AUT-ANLYS
Krueger et al.: Eliminating the adoption barrier [93]	Journal	2002	51	SW-ARCH

Table 2
Most prolific authors.

Author	#Publications	Classics
S. Apel	53	[49,58,68,77,82]
K. Schmid	49	[50,57,73,84]
P. Heymans	49	[52,54,59,63,78]
C. Kästner	45	[49,64,68,77,82]
J.D. McGregor	44	[57,83]
I. Schaefer	44	[60]
U. Kulesza	40	[51,85]
E. S. de Almeida	40	[83]
K. Czarnecki	38	[44,45,55,62,71,75,92]
P. Borba	37	[85]
R. Rabiser	36	[90]
G. Botterweck	35	–
H. Gomaa	31	–
G. Saake	31	[58]
D. Benavides	31	[43,53,70]

areas, and predict their near term change [38]. In particular, the movement of topics throughout the quadrants of strategic diagrams in successive periods of years provides information regarding their evolution. For instance, when a topic passes from Quadrant 3 to 1, it means that an emerging topic has been developed and become central for the field.

The terminology used in a research area also changes over time: whereas some words fall into disuse, other new words come into use. Tracking the set of keywords that is used in each period gives information regarding if the number of researched topics increases (new words are added), decreases (old words are removed), or remains stable. Several measures has been proposed to account for this type of evolution. [39–42] provide discussions on the pros and cons of some of these measures, including *Pearson correlation coefficient, Salton's Cosine formula, Jaccard Index*, and *Inclusion Index*. In this paper, we will follow the recommendations given by Sternitzke and Bergmann [42] and use inclusion index.

The inclusion index between two sets S and S' of words is computed by Eq. (4) as the number of common words to both sets divided by the number of words of the smallest set.

$$\text{inclusion index}_{S \rightleftarrows S'} = \frac{\#(S \cap S')}{\min(\#S, \#S')} \quad (4)$$

In such a way, we build the conceptual evolution maps of the research area analyzed [33].

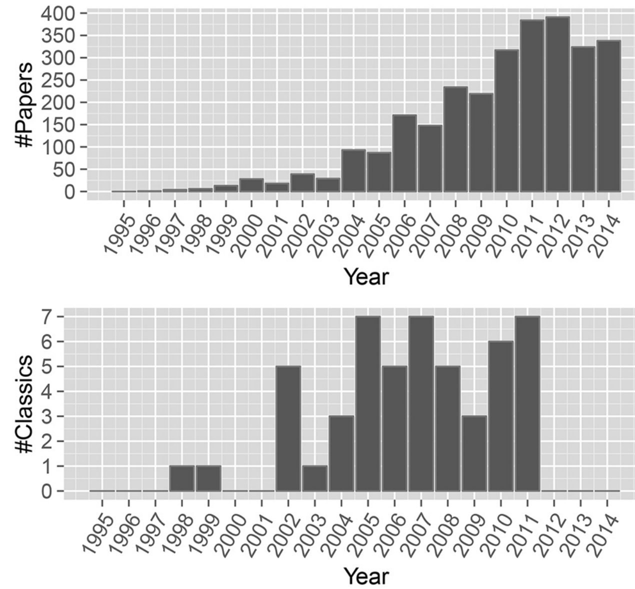


Fig. 5. Number of published papers and classics per year.

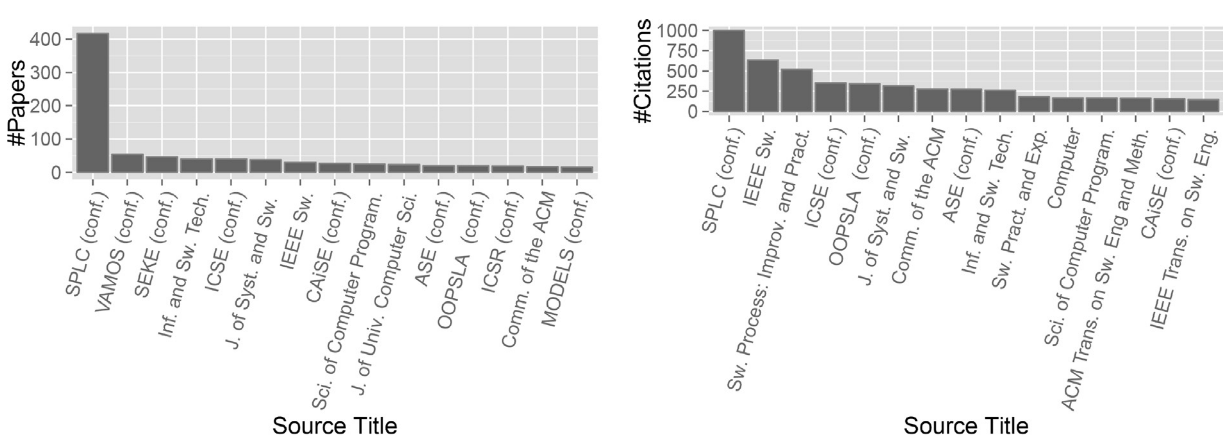
3. Software product line citation classics

This section analyzes the performance of the SPL field. To identify the citation classics in SPLs, the H-index is used (see Section 2.1). The number of papers we have analyzed is 2845, and the total number of citations to all of them is 15,018. Thus, the average number of citations per paper is 5.28. Finally, the H-index of the SPL research area is 51.

Table 1 summarizes the H-core of the SPL field, i.e., its citation classics. The last column describes the thematic network for each classic according to the science mapping results reported in Section 4.

Table 2 summarizes the most prolific authors, i.e., those who have published a largest number of papers. The last column lists the citation classics each author has written.

Fig. 5 summarizes the number of papers and classics per year. Fig. 6 identifies the most prominent sources of publication. Fig. 6(a) shows the conferences and journals that have published more papers. Fig. 6(b) represents the sources whose papers have received more citations.



(a) Sources with most papers (b) Sources with most citations

Fig. 6. Top publication sources.

Table 3
Parameters for Algorithm 1.

Parameter	Period 1 (1995–1999)	Period 2 (2000–2004)	Period 3 (2005–2009)	Period 4 (2010–2014)
$min_{occurrences}$	2	3	4	4
$min_{co\cdot occurrences}$	2	3	3	3
$min_{keywords}$	2	2	2	2
$max_{keywords}$	5	5	5	5

4. Science mapping and longitudinal study for software product lines

To analyze the structure and dynamics of the SPL research area, the bibliographic data were divided into four consecutive periods of time: 1995–1999, 2000–2004, 2005–2009, and 2010–2014. To detect the main topics of each period, the algorithm of simple centers was run using the parameters summarized in Table 3. Out of a total of 2845 documents, 23 were published in Period 1, 208 in Period 2, 902 in Period 3, and 1712 in Period 4. Note that Periods 1 and 2 have rather less documents than Periods 3 and 4. As recommended by [6,33], parameters $min_{occurrences}$ and $min_{co\cdot occurrences}$ were reduced for those periods to accommodate the lesser volume of data.

Fig. 7 offers a global representation of the simple center algorithm outcomes. The topics detected in each period are arranged on a strategic diagram according to their centrality and density. Topics are depicted as nodes whose volume is proportional to the number of publications they have associated.

4.1. Period 1

Period 1 includes just one topic: Software Architecture (SW-ARCH). Fig. 8 shows the structure of its associated cluster as a graph.

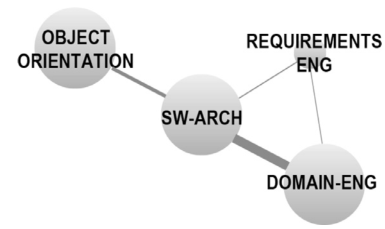


Fig. 8. Thematic network SW-ARCH for Period 1 (1995–1999).

Table 4
Topic performance for Period 1 (1995–1999).

Topic	#Publications	#Citations	Average Citations	H-index	H-core
SW-ARCH	13	157	12.08	7	[47,72,94–96] [97,98]

Keywords are represented as nodes whose volume is proportional to their associated number of publications. The equivalence index of two keywords *A* and *B* is depicted by the thickness of the edge that links *A* to *B*. Table 4 summarizes the performance of the period, i.e., the number of publications on topic SW-ARCH, the number of citations they received in total and on average, the H-index for SW-ARCH, and those publications that should be considered as classics for SW-ARCH.

According to Fig. 8 and Table 4, the SPL field comes from a confluence of research on Software Architecture, Domain Engineering, Requirements Engineering, and Object Orientation. From 1995 to 1999 there were published a reduced number of papers with a high number of citations on average, shaping a topic with high density and

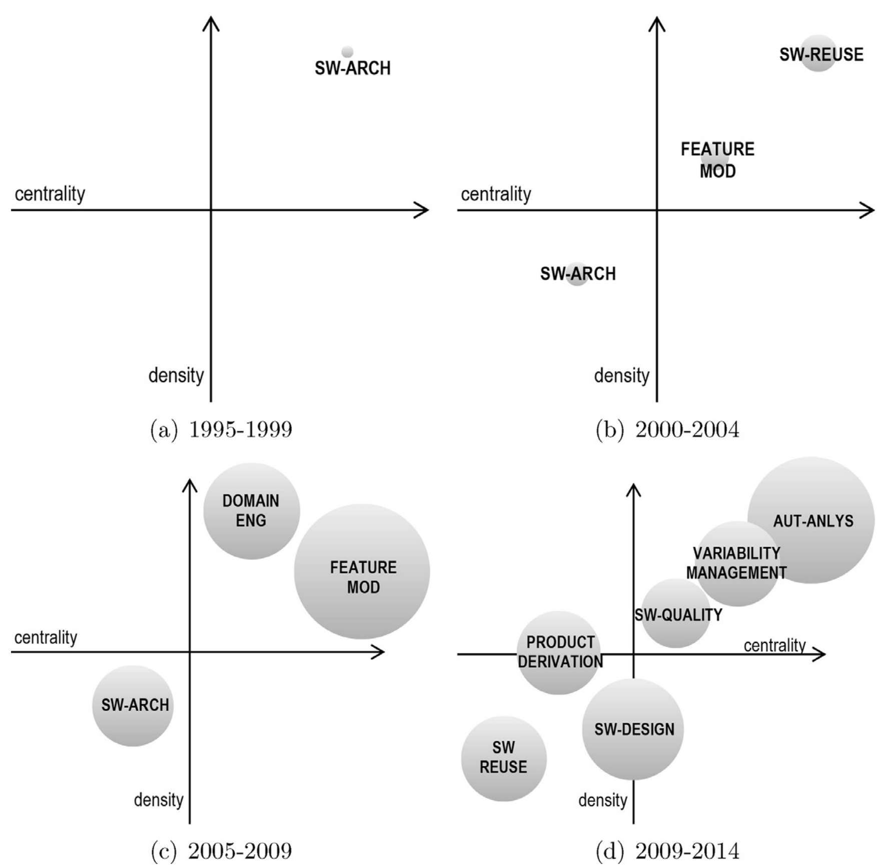
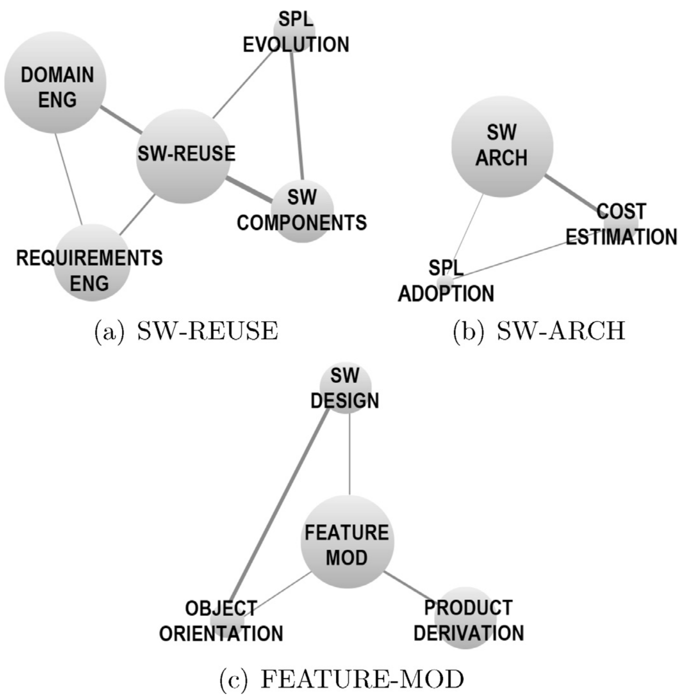


Fig. 7. Strategic diagrams for Periods 1, 2, 3, and 4.



(a) SW-REUSE (b) SW-ARCH

(c) FEATURE-MOD

Fig. 9. Thematic networks for Period 2 (2000–2004).

centrality that acted as motor for the research carried out in the subsequent periods.

4.2. Period 2

Period 2 encompasses three topics: Software Reuse (SW-REUSE), SW-ARCH, and Feature Modeling (FEATURE-MOD). Fig. 9 shows the structure of their associated clusters. Table 5 sums up the performance of the period.

Table 5
Topic performance for Period 2 (2000–2004).

Topic	#Publications	#Citations	Average citations	H-index	H-core
SW-REUSE	68	678	9.98	12	[73,74,99–101] [102–106] [107,108]
FEATURE-MOD	52	562	10.81	12	[75,79,80,99,102] [108–112] [113,114]
SW-ARCH	47	433	9.21	9	[48,57,73,84,93] [106,107,110,115]

According to Fig. 7(b), SW-REUSE and FEATURE-MOD replaced SW-ARCH as motor topic. In particular, SW-ARCH fell into quadrant 3, becoming a declining topic.

From 2000 to 2004, most published papers where about SW-REUSE, a topic which bound together research on Domain Engineering, Requirements Engineering, SPL Evolution, Software Reusability, and Software Components. Moreover, those papers where the most cited on average.

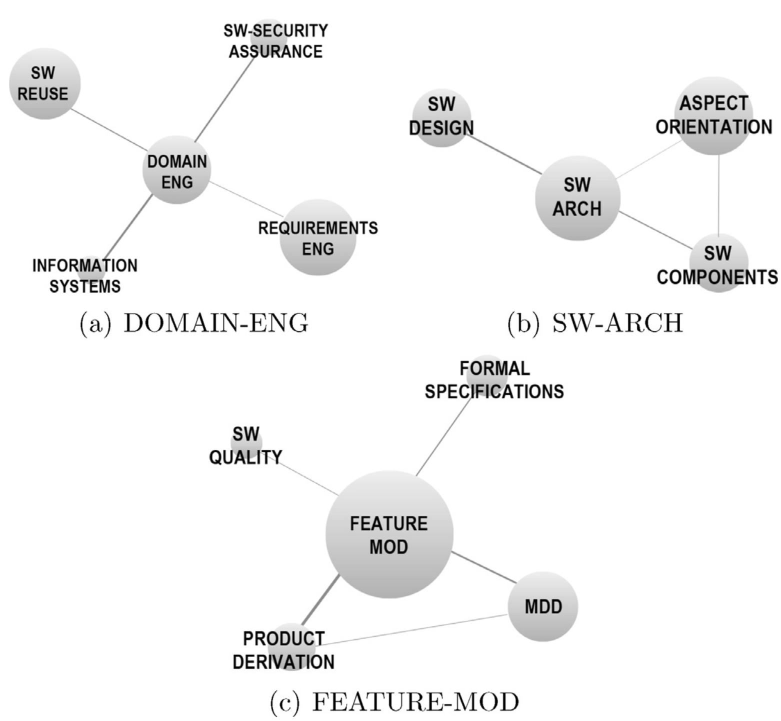
In addition, FEATURE-MOD emerged as a motor topic that grouped research on Software Design, Feature Modeling, Object Orientation, and Product Derivation on SPLs.

Notice that the relation between papers and topics is N: M, that is, a paper may talk about many topics, and a topic may be covered by more than one paper. For instance, Ref. [102] is a core paper of both SW-REUSE and FEATURE-MOD (see Table 5).

4.3. Period 3

Period 3 has three main topics: Domain Engineering (DOMAIN-ENG), SW-ARCH, and FEATURE-MOD. Fig. 10 shows the structure of their associated clusters. Table 6 summarizes the performance of the period.

From 2005 to 2009, FEAUTURE-MOD became the most important topic in SPL research, not only in terms of quantity (being the topic



(a) DOMAIN-ENG (b) SW-ARCH

(c) FEATURE-MOD

Fig. 10. Thematic networks for Period 3 (2005–2009).

Table 6
Topic performance for Period 3 (2005–2009).

Topic	#Publications	#Citations	Average citations	H-index	H-core
DOMAIN-ENG	198	1,865	9.42	12	[69,88,89,116,117] [118–122] [123,124]
FEATURE-MOD	282	2,923	10.36	23	[44–46,52,53] [54–56,58,61] [62–64,66,68] [70,76,81,85,87] [116,125,126]
SW-ARCH	162	1,180	7.28	11	[49–51,58,61] [66,68,86,88,127] [128]

about which more papers were published), but also in terms of quality (being the topic with the most citations).

4.4. Period 4

Period 4 displays an explosion of topics. It has twice the topics than Periods 2 and 3: Automated Analysis of Variability Models (AUT-ANLYS), SW-DESIGN, VARIABILITY-MANAGEMENT, SW-QUALITY, SW-REUSE, and PRODUCT-DERIVATION. Fig. 11 shows the structure of their associated clusters. Table 7 summarizes the performance of the period.

Table 7
Topic performance for Period 4 (2010–2014).

Topic	#Publications	#Citations	Average citations	H-index	H-core
AUT-ANLYS	264	1,451	5.50	13	[43,59,71,77,78] [90,92,129–131] [132–134]
SW-DESIGN	213	796	3.74	11	[60,67,77,91,135] [136–140] [141]
VARIABILITY MANAGEMENT	182	583	3.20	9	[60,82,91,137,140] [142–145]
SW-QUALITY	146	310	2.12	5	[146–150]
PRODUCT DERIVATION	175	524	2.99	9	[90,132,151–153] [154–157]
SW-REUSE	186	693	3.72	10	[65,83,130,158,159] [160–164]

According to Fig. 7(c), from 2010 to present days, three topics have played a motor role: AUT-ANLYS, VARIABILITY-MANAGEMENT, and SW-QUALITY. As Fig. 11(a) shows, FEATURE-MOD is the keyword with most associated publications (not only for its cluster, but for all the clusters of Period 4). The high equivalence index between FEATURE-MOD and AUT-ANALYS reflects the works on formal methods and algorithms to support the automated analysis of feature diagrams. According to Fig. 11(c), VARIABILITY-MANAGEMENT binds together research on Software Components, Domain Specific Languages (DSL),

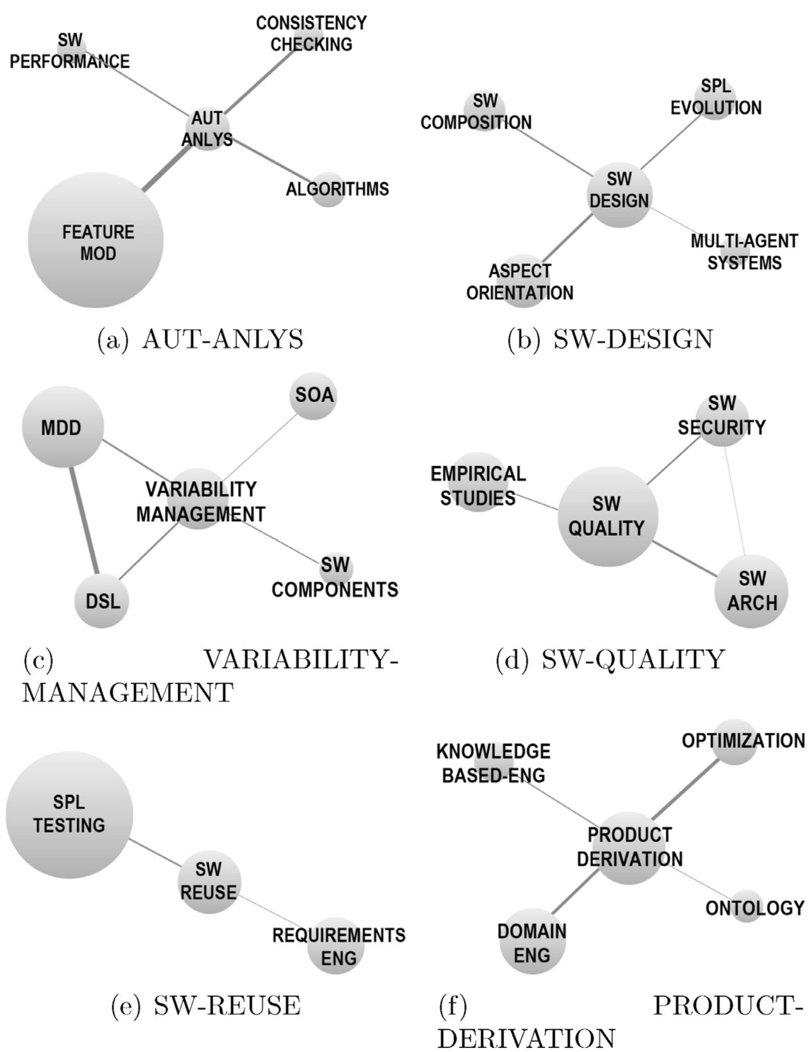


Fig. 11. Thematic networks for Period 4 (2010–2014).

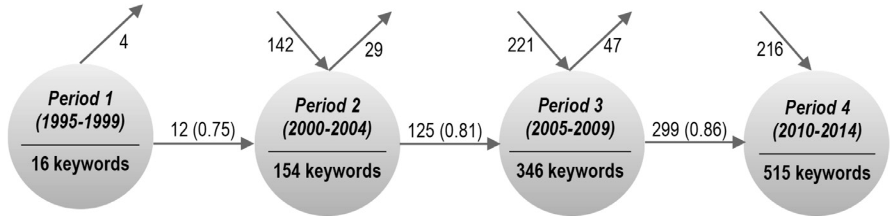


Fig. 12. Number of keywords per period.

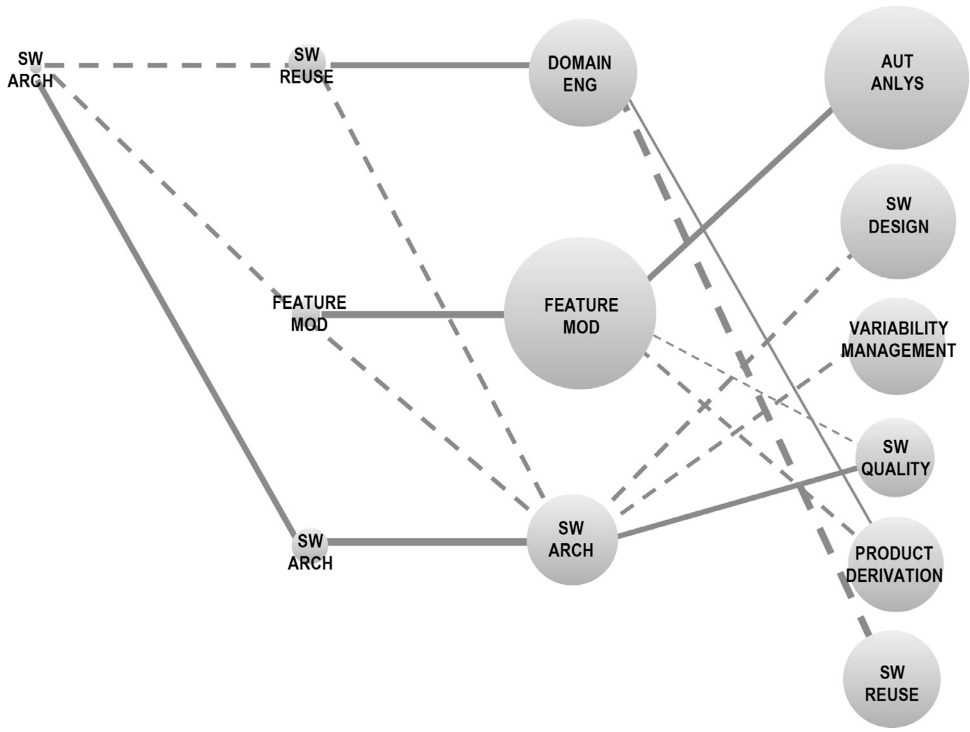


Fig. 13. Topic evolution between periods.

Model Driven Development (MDD), and Service-Oriented Architecture. Finally, as Fig. 11(d) shows, SW-QUALITY groups research on Software Architecture (SW-ARCH), Software Security, and Empirical Studies where the SPL paradigm is validated.

According to Figs. 9(c) and 10(c), the derivation of particular products from a SPL platform has been strongly associated to research on feature modeling from 2000 to 2009. Nevertheless, Fig. 11(f) and the position of PRODUCT-DERIVATION in the strategic diagram in Fig. 7(c) show that this subject is gaining independence and, in the current period, it binds together research on Ontologies, Domain Engineering, Optimization problems and Knowledge-Based Engineering.

In Fig. 7(c), SW-DESIGN is halfway between the third and second quadrant. Jointly with Fig. 11(b), it could be interpreted as that, from 2010 to 2014, a considerable effort is being made to apply general research on Software Development (Software Design, Aspect Orientation, Multi-Agent Systems, Software Composition) to SPLs.

According to Table 7, the most performing topic in the last years has been the automated analysis of feature models.

4.5. Longitudinal analysis of topic evolution

Fig. 12 shows the number of keywords used in each period. The nodes represent periods; the arrows between two consecutive periods are labeled by (i) the number of keywords shared by both periods, and, in parentheses, the inclusion index between periods. Upper-incoming arrows are labeled by the number of new keywords for the pointed period, and upper-outgoing arrows are labeled by the number of keywords that are not present (i.e., discontinued) in the next period. For instance, Period 2 includes 154 keywords. 125 of them are also included in the next period. The inclusion index between Periods 2 and 3 is 0.81. Finally, in Period 3, 221 new key words were included, and 47 old ones were discarded.

Fig. 12 shows that the SPL research action area has been progressively expanded since (i) the number of publication keywords has been incremented incessantly from period to period (see nodes), and (ii) the number of new keywords is much bigger than the number of discarded ones for every period (see upper arrows). On the other hand, the inclusion index between periods is rather high, pointing out that the SPL community uses a consolidated terminology.

Fig. 13 describes topic evolution. In particular, Fig. 13 depicts topics as nodes whose volume is proportional to the number of publications they have associated. Nodes are vertically aligned by periods (e.g., the second column includes the topics present in Period 2). The inclusion index of two topics T and T' is represented by the thickness of the edge that links T to T'. Whenever the keyword that gives name to T' is also a keyword in cluster T, the edge is represented by a solid line. Otherwise, the edge is depicted as a dashed line.

According to Fig. 13, the evolution of the SPL area has behaved properly, growing smoothly and continuously. The number of publications has increased in each period. Moreover, no topic has abruptly disappeared; on the contrary, original topics have progressively been consolidated and, especially in the last five years, they have been branched out to more specific research themes.

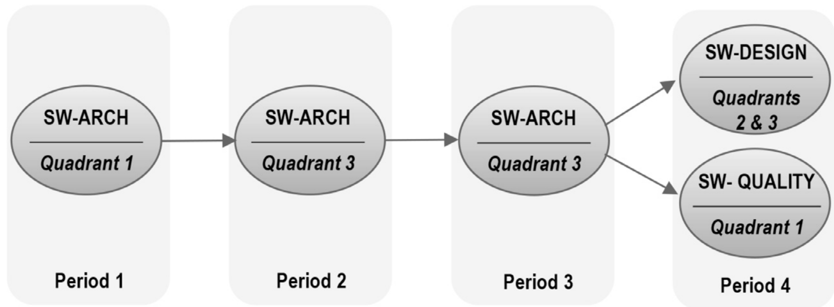


Fig. 14. Movement of SW-ARCH through the strategic diagrams in Fig. 7.

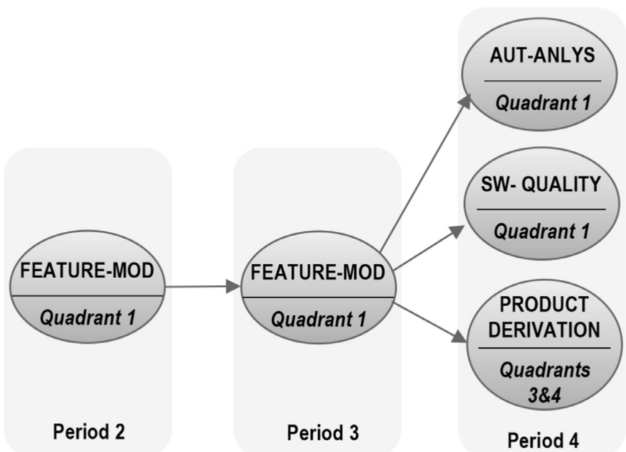


Fig. 15. Movement of FEATURE-MOD through the strategic diagrams in Fig. 7.

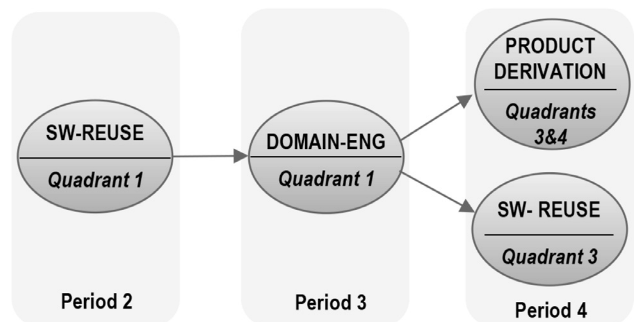


Fig. 16. Movement of SW-REUSE through the strategic diagrams in Fig. 7.

Feature modeling has been the most important topic in the whole SPL research area. It presents the best evolution behavior and the best quality indicators (see the H-index of FEATURE-MOD, AUT-ANLYS, SW-QUALITY, and PRODUCT-DERIVATION in Tables 5, 6, and 7). Research on software architectures and software reuse has also been essential for the development of the area. In particular, the following conclusions may be drawn from Figs. 7 and 13:

1. Fig. 14 depicts the movement of SW-ARCH through the quadrants of the strategic diagrams in Fig. 7. SW-ARCH was the initial motor topic, which inspired research on SPLs. Nevertheless, it moved from the first quadrant in Period 1, to the third quadrant in Periods 2 and 3, becoming a peripheral topic.
2. The evolution of FEATURE-MOD is described by Fig. 15. From Period 2, feature modeling has behaved as an essential motor topic for the development of the SPL field. As the size of the nodes in Fig. 13 shows, the number of publications on this topic has grown dramatically. In the last five years research on feature modeling has spread out to several subareas: AUT-ANLYS, PRODUCT-DERIVATION, and SW-QUALITY (two of them playing a motor role).
3. The evolution of SW-REUSE is described by Fig. 16. A main goal for the SPL paradigm is shifting from opportunistic to systematic reuse of software. Accordingly, SW-REUSE and DOMAIN-ENG have worked as motor topics from 2000 to 2009. However, nowadays SW-REUSE may be considered a peripheral topic.

5. Threats to validity

The following points summarize the main threats to the validity of our work, and the strategies we have followed to try to overcome them:

1. **Bibliographic database**. Although there are several freely available databases, such as Google Scholar (GS), CiteSeerX, Microsoft Academic Search, getCITED, etc., the paid subscription databases ISIWoS and Scopus are currently the most reliable [9,10].
   If the query in Section 2 had been performed against another database, the gathered information would have been different. For instance, as Franceschet [165] notes, ISIWoS and GS have distinct philosophies. To provide a safeguard against low-quality or low-impact material being indexed, ISIWoS uses a selective inclusion procedure where in-house editors assess candidate publication outlets using criteria such as timeliness, peer-review process, international diversity of editors and authors, citation impact, and self-citation rate. In contrast, GS includes everything that resembles scholarly work, based on automatic format inspection rather than content inspection, thereby risking inflation of its record [166]. It has been argued that, because of its automatic inclusion process, GS is susceptible to errors in metadata [167] and to indexing of non-scientific works [165,168–170]. Moreover, according to Winter et al. [166]: “the ISIWoS quality control mechanism is especially important today in an era where some scientists seem to suffer from writing incontinence [171] and where predatory publishers are on the rise [172]”.
   By combining data from ISIWoS and Scopus, we have tried to achieve a good balance between:
   - *Retrospective coverage*: ISIWoS offers the most complete retrospective quality coverage for all scientific disciplines [173,174], which is specially appropriated for the rigorous science mapping analysis carried out in this paper.
   - *Conference coverage*: the scientific community in Computer Science is highly driven by conferences, which are inadequately covered by ISIWoS. In contrast, Scopus provides a more complete conference coverage, indexing some of them specifically focused on software product line engineering, such as the Software Product Line Conference (SPLC) and the International Workshop on Variability Modelling of Software-intensive Systems (VaMoS).
2. **Keyword standardization**. If the input to the simple centers algorithm included too many keywords, the output might be hard

to interpret: there would be a high number of clusters, composed of many keywords interrelated with low equivalence indices. To overcome this problem, the standardization described in Section 2 was performed.

To properly undertake such standardization, it is needed a deep knowledge of the SPL research area, specially to discard and group the keywords. In our case, this task was carried out by two of the authors of this work (Ruben Heradio and David Fernandez-Amoros), who are experts on SPLs and regularly publish research on this area. Whenever they had doubts regarding the meaning of particular keywords, they carefully read the corresponding papers to put the words into context.

3. **Period setting and parameterization of the simple centers algorithm**. To get good performance in the analysis of co-citation clusters over consecutive periods of years, it is needed a balanced period length: short enough to prevent smoothing excessively the data and long enough to include sufficient publications for the analysis [33]. In our work, we decided to group the publications in periods of five years, as [28,30,32,175], and [33] also do in similar science mapping analyses.

Equivalence index has the drawback of being high when keywords appear infrequently but almost always together. As a result, the simple centers algorithm may overestimate the importance of those keywords and create irrelevant clusters. To avoid that problem, two of the authors of this paper (Enrique Herrera-Viedma and F. Javier Cabrerizo), who are experts on bibliometrics and regularly publish research on this field experimentally adjusted the simple centers algorithm parameters as summarized in Table 3.

To ease the review of our work, the records retrieved from ISIWoS, all the raw keywords included in such records, the standardized keywords, a description on how keywords were grouped, and the sources where the papers where published are available on:

http://rheradio.github.io/SPL-Bib-Anlys/

6. Concluding remarks

Two bibliometric techniques have been used to provide a structured view of the SPL literature. Thanks to science mapping, we have identified the main researched topics, the evolution of the interest in those topics and the relationships among topics. Thanks to performance analysis, we have identified the most influential papers, the journals and conferences that have published most papers, how numerous is the literature on product lines and what is its distribution over time.

Moreover, it has been detected that software architecture was the initial motor of research in SPLs, that work on software systematic reuse has also been essential for the development of the area, and that feature modeling has been the most important topic for the last fifteen years, having the best evolution behavior in terms of number of published papers and received citations.

Acknowledgment

This work has been supported by the Spanish Government under the CICYT project DPI-2013-44776-R, the Comunidad de Madrid under the RoboCity2030 excellence research network S2013MIT-2748, the FEDER funds under grant TIN2010-17876 and grant TIN2013-40658-P, and the Andalusian Excellence Projects under grant TIC-05299 and grant TIC-5991.

References

[1] P. Clements, L. Northrop, Software Product Lines: Practices and Patterns, Addison Wesley Professional, 2001.
[2] F.J. van der Linden, K. Schmid, E. Rommes, Software Product Lines in Action: The Best Industrial Practice in Product Line Engineering, Springer-Verlag Berlin Heidelberg, 2007.
[3] E. Garfield, Introducing citation classics. The human side of scientific reports, Current Comments (1977) 5–7.
[4] M. Callon, J. Courtial, F. Laville, Co-word analysis as a tool for describing the network of interactions between basic and technological research: the case of polymer chemistry, Scientometrics 1 (1) (1991) 155–205.
[5] J.E. Hirsch, An index to quantify an individual's scientific research output, Proc. Natl. Acad. Sci. USA 46 (2005) 16569–16572.
[6] N. Coulter, I. Monarch, S. Konda, Software engineering as seen through its research literature: a study in co-word analysis, J. Am. Soc. Inf. Sci. 13 (1998) 1206–1223.
[7] K. Börner, C. Chen, K.W. Boyack, Visualizing knowledge domains, Annu. Rev. Inf. Sci. Technol. 37 (1) (2003) 179–255.
[8] M. Cobo, A. Lopez-Herrera, E. Herrera-Viedma, F. Herrera, Science mapping software tools: review, analysis, and cooperative study among tools, J. Am. Soc. Inf. Sci. Technol. (7) (2011) 1382–1402.
[9] V. Gomez-Jauregui, C. Gomez-Jauregui, C. Manchado, C. Otero, Information management and improvement of citation indices, Int. J. Inf. Manag. 2 (2014) 257–271.
[10] E.S. Vieira, J.A. Gomes, A comparison of Scopus and web of science for a typical university, Scientometrics 81 (2) (2009) 587–600.
[11] F. Bachmann, P.C. Clements, Variability in software product lines, Technical Report, Software Engineering Institute, CMU/SEI-2005-TR-012, 2005.
[12] E. Noyons, H. Moed, A. Van Raan, Integrating research performance analysis and science mapping, Scientometrics 46 (3) (1999) 591–604.
[13] M. Cobo, A. Lopez-Herrera, E. Herrera-Viedma, F. Herrera, Scimat: a new science mapping analysis software tool, J. Am. Soc. Inf. Sci. Technol. 63 (8) (2012) 1609–1630.
[14] H.F. Moed, New developments in the use of citation analysis in research evaluation, Archivum Immunologiae et Therapiae Experimentalis 57 (1) (2009) 13–18.
[15] E. Garfield, Citation Indexing—Its Theory and Application in Science, Technology and Humanities, Institute for Scientific Information Press, 1979.
[16] J.K. Vanclay, On the robustness of the *h*-index, J. Am. Soc. Inf. Sci. Technol. 58 (10) (2007) 1547–1550.
[17] W. Tam, E. Wong, F. Wong, D. Hui, Citation classics: top 50 cited articles in "respiratory system", Respirology 18 (1) (2013) 71–81.
[18] D.R. Smith, Ten citation classics from the New Zealand medical journal, J. N. Z. Med. Assoc. 120 (1267) (2007) 2871–2875.
[19] S. Stack, Citation classics in suicide and life threatening behavior: a research note, Suicide Life Threat Behav. 42 (6) (2012) 628–639.
[20] A. Baltussen, C. Kindler, Citation classics in critical care medicine, Intensiv. Care Med. 30 (5) (2004) 902–910.
[21] M. Martinez, M. Herrera, J. Lopez-Gijon, E. Herrera-Viedma, *H*-classics: characterizing the concept of citation classics through *h*-index, Scientometrics 98 (3) (2014) 1971–1983.
[22] P. Albarran, I. Ortuno, J. Ruiz-Castillo, The measurement of low- and high-impact in citation distributions: technical results, J. Informetr. 5 (1) (2011) 48–63.
[23] N.J. van Eck, L.L. Waltman, How to normalize cooccurrence data? An analysis of some well-known similarity measures, J. Am. Soc. Inf. Sci. Technol. 60 (8) (2009) 1635–1651.
[24] A. Rip, J. Courtial, Co-word maps of biotechnology: An example of cognitive scientometrics, Scientometrics 6 (6) (1984) 381–400.
[25] M. Zitt, E. Bassecoulard, Y. Okubo, Shadows of the past in international cooperation: collaboration profiles of the top five producers of science, Scientometrics 47 (3) (2000) 627–657.
[26] N.J. van Eck, L. Waltman, Bibliometric mapping of the computational intelligence field, Int. J. Uncertain. Fuzziness Knowl. Based Syst. 15 (5) (2007) 625–645.
[27] M. Callon, J. Law, A. Rip, Mapping the Dynamics of Science and Technology, The MacMillan Press Ltd, in: Ch. Qualitative Scientometrics, 1986, pp. 103–123.
[28] V. Kandylas, S.P. Upham, L.H. Ungar, Analyzing knowledge communities using foreground and background clusters, ACM Trans. Knowl. Discov. Data 4 (2) (2010).
[29] Q. He, Knowledge discovery through co-word analysis, Libr. Trends 48 (1) (1999) 133–159.
[30] R. Bailon-Moreno, E. Jurado-Alameda, R. Ruiz-Baños, J. Courtial, Analysis of the scientific field of physical chemistry of surfactants with the unified scienctometric model. fit of relational and activity indicators, Scientometrics 63 (2) (2005) 259–276.
[31] R. Bailon-Moreno, E. Jurado-Alameda, R. Ruiz-Baños, The scientific network of surfactants: structural analysis, J. Am. Soc. Inf. Sci. Technol. 57 (7) (2006) 949–950.
[32] A. Lopez-Herrera, M. Cobo, E. Herrera-Viedma, F. Herrera, R. Bailon-Moreno, E. Jimenez-Contreras, Visualization and evolution of the scientific structure of fuzzy sets research in Spain, Inf. Res. 14 (4) (2009) http://www.informationr.net/ir/14-4/paper421.html.
[33] M. Cobo, A. Lopez-Herrera, E. Herrera-Viedma, F. Herrera, An approach for detecting, quantifying, and visualizing the evolution of a research field: a practical application to the fuzzy sets theory field, J. Informetr. 5 (1) (2011) 146–166.
[34] S. Bauin, B. Michelet, M. Schweighoffer, P. Vermeulin, Using bibliometrics in strategic analysis: understanding chemical reactions at the CNRS, Scientometrics 22 (1) (1991) 113–137.
[35] J. Law, S. Bauin, J.-P. Courtial, J. Whittaker, Policy and the mapping of scientific change: a co-word analysis of research into environmental acidification, Scientometrics 14 (3–4) (1988) 251–264.
[36] W. Turner, F. Rojouan, Evaluating input/output relationships in a regional research network using co-word analysis, Scientometrics 22 (1) (1991) 139–154.

[37] T. Cahlik, Comparison of the maps of science, Scientometrics 49 (3) (2000) 373–387.
[38] H. Small, Tracking and predicting growth areas in science, Scientometrics 68 (3) (2006) 595–610.
[39] L. Hamers, Y. Hemeryck, G. Herweyers, M. Janssen, H. Keters, R. Rousseau, A. Vanhoutte, Similarity measures in scientometric research: the jaccard index versus Salton's cosine formula, Inf. Process. Manag. 25 (3) (1989) 315–318.
[40] H.P.F. Peters, R.R. Braam, A.F.J. van Raan, Cognitive resemblance and citation relations in chemical engineering publications, J. Am. Soc. Inf. Sci. 46 (1) (1995) 9–21.
[41] P. Ahlgren, B. Jarneving, R. Rousseau, Requirements for a cocitation similarity measure, with special reference to Pearson's correlation coefficient, J. Am. Soc. Inf. Sci. Technol. 54 (6) (2003) 550–560.
[42] C. Sternitzke, I. Bergmann, Similarity measures for document mapping: a comparative study on the level of an individual scientist, Scientometrics 78 (1) (2009) 113–130.
[43] D. Benavides, S. Segura, A. Ruiz-Cortes, Automated analysis of feature models 20 years later: a literature review, Inf. Syst. 6 (2010) 615–636.
[44] K. Czarnecki, S. Helsen, U. Eisenecker, Formalizing cardinality-based feature models and their specialization, Softw. Process: Improv. Pract. 10 (1) (2005) 7–29.
[45] K. Czarnecki, S. Helsen, U. Eisenecker, Staged configuration through specialization and multilevel configuration of feature models, Softw. Process: Improv. Pract. (2) (2005) 143–169.
[46] D. Batory, Feature models, grammars, and propositional formulas, in: Proceedings of the International Conference on Software Product Line, Rennes, France, 2005, pp. 7–20.
[47] J. Coplien, D. Hoffman, D. Weiss, Commonality and variability in software engineering, IEEE Softw. 15 (6) (1998) 37–45.
[48] Y. Smaragdakis, D. Batory, Mixin layers: an object-oriented implementation technique for refinements and collaboration-based designs, ACM Trans. Softw. Eng. Methodol. 11 (2) (2002) 215–255.
[49] C. Kästner, S. Apel, M. Kuhlemann, Granularity in software product lines, in: Proceedings of the International Conference on Software Engineering, New York, NY, USA, 2008, pp. 311–320.
[50] S. Hallsteinsen, M. Hinchey, S. Park, K. Schmid, Dynamic software product lines, Computer 41 (4) (2008) 93–95.
[51] E. Figueiredo, N. Cacho, C. Sant'Anna, M. Monteiro, U. Kulesza, A. Garcia, S. Soares, F. Ferrari, S. Khan, F. Castor Filho, F. Dantas, Evolving software product lines with aspects: an empirical study on design stability, in: Proceedings of the International Conference on Software Engineering, Leipzig, Germany, 2008, pp. 261–270.
[52] P.-Y. Schobbens, P. Heymans, J.-C. Trigaux, Y. Bontemps, Generic semantics of feature diagrams, Comput. Netw. 51 (2) (2007) 456–479.
[53] D. Benavides, P. Trinidad, A. Ruiz-Cortes, Automated reasoning on feature models, in: Proceedings of the International Conference on Advanced Information Systems Engineering, Porto Portugal, 2005.
[54] P. Schobbens, P. Heymans, J.-C. Trigaux, Feature diagrams: a survey and a formal semantics, in: Proceedings of the International Conference on Requirements Engineering, Minneapolis, MN, USA, 2006, pp. 139–148.
[55] K. Czarnecki, A. Wasowski, Feature diagrams and logics: there and back again, in: Proceedings of the Software Product Line Conference, Kyoto, Japan, 2007, pp. 23–34.
[56] S. Thaker, D. Batory, D. Kitchin, W. Cook, Safe composition of product lines, in: Proceedings of the International on Generative Programming and Component Engineering, New York, NY, USA, 2007, pp. 95–104.
[57] G. Böckle, P. Clements, J.D. McGregor, D. Muthig, K. Schmid, Calculating ROI for software product lines, IEEE Softw. 21 (3) (2004) 23–31.
[58] S. Apel, T. Thomas Leich, G. Saake, Aspectual feature modules, IEEE Trans. Softw. Eng. (2) (2008) 162–180.
[59] A. Classen, P. Heymans, P.-Y. Schobbens, A. Legay, J.-F. Raskin, Model checking lots of systems: efficient verification of temporal properties in software product lines, in: Proceedings of the International Conference on Software Engineering, New York, NY, USA, 2010, pp. 335–344.
[60] I. Schaefer, L. Bettini, F. Damiani, N. Tanzarella, Delta-oriented programming of software product lines, in: Proceedings of the International Conference on Software Product Lines, Jeju Island, South Korea, 2010, pp. 77–91.
[61] M. Svahnberg, J. van Gurp, J. Bosch, A taxonomy of variability realization techniques, Softw. Pract. Exp. 35 (8) (2005) 705–754.
[62] K. Czarnecki, K. Pietroszek, Verifying feature-based model templates against well-formedness OCL constraints, in: Proceedings of the International Conference on Generative Programming and Component Engineering, New York, NY, USA, 2006, pp. 211–220.
[63] A. Metzger, K. Pohl, P. Heymans, P. Schobbens, G. Saval, Disambiguating the documentation of variability in software product lines: a separation of concerns, formalization and automated analysis, in: Proceedings of the International Requirements Engineering Conference, New Delhi, India, 2007, pp. 243–253.
[64] T. Thüm, D. Batory, C. Kästner, Reasoning about edits to feature models, in: Proceedings of the International Conference on Software Engineering, Washington, DC, USA, 2009, pp. 254–264.
[65] E. Engstrom, P. Runeson, Software product line testing - a systematic mapping study, Inf. Softw. Technol. 53 (1) (2011) 2–13.
[66] M. Voelter, I. Groher, Product line implementation using aspect-oriented and model-driven software development, Proceedings of the International Software Product Line Conference, Washington, DC, USA, 2007, pp. 233–242.
[67] J. Bosch, P. Bosch-Sijtsema, From integration to composition: On the impact of software product lines, global development and ecosystems, J. Syst. Softw. 83 (1) (2010) 67–76.
[68] C. Kästner, S. Apel, D. Batory, A case study implementing features using aspectJ, in: Proceedings of the International Software Product Line Conference, Washington, DC, USA, 2007, pp. 223–232.
[69] M. Moon, K. Yeom, H.S. Chae, An approach to developing domain requirements as a core asset based on commonality and variability analysis in a product line, IEEE Trans. Softw. Eng. 31 (7) (2005) 551–569.
[70] D. Batory, D. Benavides, A. Ruiz-Cortes, Automated analysis of feature models: challenges ahead, Commun. ACM 49 (12) (2006) 45–47.
[71] S. She, R. Lotufo, T. Berger, A. Wasowski, K. Czarnecki, Reverse engineering feature models, in: Proceedings of the International Conference on Software Engineering, New York, NY, USA, 2011, pp. 461–470.
[72] M. Svahnberg, J. Bosch, Evolution in software product lines: two cases, J. Softw. Maint. 30 (6) (1999) 391–422.
[73] K. Schmid, A comprehensive product line scoping approach and its validation, in: Proceedings of the International Conference on Software Engineering, Orlando, FL, USA, 2002, pp. 593–603.
[74] L. Northrop, SEI's software product line tenets, IEEE Softw. 19 (4) (2002) 32–40.
[75] K. Czarnecki, S. Helsen, E. Ulrich, Staged configuration using feature models, in: Proceedings of the International Software Product Line Conference, Boston, USA, 2004, pp. 266–283.
[76] F. Heidenreich, J. Kopcsek, C. Wende, FeatureMapper: mapping features to models, in: Proceedings of the International Conference on Software Engineering, New York, NY, USA, 2008, pp. 943–944.
[77] S. Apel, C. Kästner, A. Größlinger, C. Lengauer, Type safety for feature-oriented product lines, Autom. Softw. Eng. 17 (3) (2010) 251–300.
[78] A. Classen, P. Heymans, P.-Y. Schobbens, A. Legay, Symbolic model checking of software product lines, in: Proceedings of the International Conference on Software Engineering, New York, NY, USA, 2011, pp. 321–330.
[79] K. Lee, K. Kang, J. Lee, Concepts and guidelines of feature modeling for product line software engineering, in: Proceedings of the International Conference on Software Reuse, Austin, Texas, USA, 2002, pp. 62–77.
[80] D. Beuche, H. Papajewski, W. Schröder-Preikschat, Variability management with feature models, Sci. Comput. Program. 53 (3) (2004) 333–352.
[81] R. Mietzner, A. Metzger, F. Leymann, K. Pohl, Variability modeling to support customization and deployment of multi-tenant-aware software as a service applications, in: Proceedings of the ICSE Workshop on Principles of Engineering Service Oriented Systems, 2009, pp. 18–25.
[82] J. Liebig, S. Apel, C. Lengauer, C. Kästner, M. Schulze, An analysis of the variability in forty preprocessor-based software product lines, in: Proceedings of the International Conference on Software Engineering, New York, NY, USA, 2010, pp. 105–114.
[83] P.A. da Mota Silveira Neto, I. do Carmo Machado, J.D. McGregor, E.S. de Almeida, S.R. de Lemos Meira, A systematic mapping study of software product lines testing, Inf. Softw. Technol. 53 (5) (2011) 407–423.
[84] A. Birk, G. Heller, I. John, K. Schmid, T. von der Massen, K. Muller, Product line engineering, the state of the practice, IEEE Softw. 20 (6) (2003) 52–60.
[85] V. Alves, R. Gheyi, T. Massoni, U. Kulesza, P. Borba, C. Lucena, Refactoring product lines, in: Proceedings of the International Conference on Generative Programming and Component Engineering, New York, NY, USA, 2006, pp. 201–210.
[86] D. Fischbein, S. Uchitel, V. Braberman, A foundation for behavioural conformance in software product line architectures, in: Proceedings of the Workshop on Role of Software Architecture for Testing and Analysis, New York, NY, USA, 2006, pp. 39–48.
[87] M. Mendonca, M. Branco, D. Cowan, S.P.L.O.T.: software product lines online tools, in: Proceedings of the ACM SIGPLAN Conference Companion on Object Oriented Programming Systems Languages and Applications, New York, NY, USA, 2009, pp. 761–762.
[88] R. van Ommering, Software reuse in product populations, IEEE Trans. Softw. Eng. 31 (7) (2005) 537–550.
[89] T. Asikainen, T. Männistö, T. Soininen, Kumbang: a domain ontology for modelling variability in software product families, Adv. Eng. Inform. 21 (1) (2007) 23–40.
[90] D. Dhungana, P. Grünbacher, R. Rabiser, The DOPLER meta-tool for decision-oriented variability modeling: a multiple case study, Autom. Softw. Eng. 18 (1) (2011) 77–114.
[91] L. Chen, M.A. Babar, A systematic review of evaluation of variability management approaches in software product lines, Inf. Softw. Technol. 53 (4) (2011) 344–362.
[92] T. Berger, S. She, R. Lotufo, A. Wasowski, K. Czarnecki, Variability modeling in the real: a perspective from the operating systems domain, in: Proceedings of the International Conference on Automated Software Engineering, New York, NY, USA, 2010, pp. 73–82.
[93] C. Krueger, Eliminating the adoption barrier, IEEE Softw. 19 (4) (2002) 29–31.
[94] W. Lam, A case-study of requirements reuse through product families, Ann. Softw. Eng. 5 (1998) 253–277.
[95] D. Dikel, D. Kane, S. Ornburn, W. Loftus, J. Wilson, Applying software product-line architecture, Computer 30 (8) (1997) 49–55.
[96] D.C. Rine, R.M. Sonnemann, Investments in reusable software. a study of software reuse investment success factors, J. Syst. Softw. 41 (1) (1998) 17–32.
[97] K.C. Kang, S. Kim, J. Lee, K. Lee, Feature-oriented engineering of PBX software for adaptability and reuseability, Softw. Pract. Exp. 29 (10) (1999) 875–896.

[98] R. Lutz, G. Helmer, M. Moseman, D. Statezni, S. Tockey, Safety analysis of requirements for a product family, in: Proceedings of the International Conference on Requirements Engineering, Colorado Springs, CO, USA, 1998, pp. 24–31.
[99] K. Lee, K. Kang, Feature dependency analysis for product line component design, in: Proceedings of the International Conference on Software Reuse, Madrid, Spain, 2004, pp. 65–69.
[100] M. Ardis, N. Daley, D. Hoffman, H. Siy, D. Weiss, Software product lines: a case study, Softw. Pract. Exp. 30 (7) (2000) 825–847.
[101] J. van Gurp, J. Bosch, M. Svahnberg, On the notion of variability in software product lines, in: Proceedings of the Working IEEE/IFIP Conference on Software Architecture, Amsterdan, Netherlands, 2001, pp. 45–54.
[102] S. Park, M. Kim, V. Sugumaran, A scenario, goal and feature-oriented domain analysis approach for developing software product lines, Ind. Manag. Data Syst. (4) (2004) 296–308.
[103] R.R. Lutz, Extending the product family approach to support safe reuse, J. Syst. Softw. 53 (3) (2000) 207–217.
[104] A. Taulavuori, E. Niemel, P. Kallio, Component documentation-a key issue in software product lines, Inf. Softw. Technol. 46 (8) (2004) 535–546.
[105] H. Zhang, S. Jarzabek, B. Yang, Quality prediction and assessment for product lines, in: Proceedings of the International Conference on Advanced Information Systems Engineering, Klagenfurt, Austria, 2003, pp. 681–695.
[106] J. Greenfield, K. Short, Software factories: assembling applications with patterns, models, frameworks and tools, in: Proceedings of the ACM SIGPLAN Conference on Object-oriented Programming, Systems, Languages, and Applications, New York, NY, USA, 2003, pp. 16–27.
[107] B. Boehm, A. Brown, R. Madachy, Y. Yang, A software product line life cycle cost estimation model, in: Proceedings of the International Symposium on Empirical Software Engineering, Redondo Beach, CA, USA, 2004, pp. 156–164.
[108] C. Gacek, M. Anastasopoules, Implementing product line variabilities, in: Proceedings of the Symposium on Software Reusability: Putting Software Reuse in Context, New York, NY, USA, 2001.
[109] M. Matinlassi, Comparison of software product line architecture design methods: COPS, FAST, FORM, KobrA and QADA, in: Proceedings of the International Conference on Software Engineering, Edinburgh, Scotland, UK, 2004, pp. 127–136.
[110] P. Sochos, I. Philippow, M. Riebisch, Feature-oriented development of software product lines: mapping feature models to the architecture, in: Proceedings of the International Conference on Object-Oriented and Internet-based Technologies, Erfurt, Germany, 2004, pp. 138–152.
[111] M.L. Griss, Implementing product-line features by composing aspects, in: Proceedings of the Software Product Line Conference, Denver, Colorado, 2000, pp. 271–288.
[112] A. Hein, M. Schlick, R. Vinga-Martins, Applying feature models in industrial settings, in: P. Donohoe (Ed.), Software Product Lines, The Springer International Series in Engineering and Computer Science, 2000, pp. 47–70.
[113] S. Deelstra, M. Sinnema, J. Bosch, Experiences in software product families: Problems and issues during product derivation, in: Proceedings of the Software Product Line Conference, Boston, MA, USA, 2004, pp. 165–182.
[114] V. Cechticky, A. Pasetti, O. Rohlik, W. Schaufelberger, XML-based feature modelling, in: Proceedings of the International Conference on Software Reuse: Methods, Techniques and Tools, Madrid, Spain, 2004, pp. 101–114.
[115] D. Faust, C. Verhoef, Software product line migration and deployment, Softw. Pract. Exp. 33 (10) (2003) 933–955.
[116] J. Liu, J. Dehlinger, R. Lutz, Safety analysis of software product lines using state-based modeling, J. Syst. Softw. 80 (11) (2007) 1879–1892.
[117] I. Reinhartz-Berger, A. Sturm, Utilizing domain models for application design and validation, Inf. Softw. Technol. 51 (8) (2009) 1275–1289.
[118] M.-O. Reiser, M. Weber, Multi-level feature trees: a pragmatic approach to managing highly complex product families, Requir. Eng. 12 (2) (2007) 57–75.
[119] M. Kim, S. Park, V. Sugumaran, H. Yang, Managing requirements conflicts in software product lines: a goal and scenario based approach, Data Knowl. Eng. 61 (3) (2007) 417–432.
[120] F. Ahmed, L.F. Capretz, The software product line architecture: an empirical investigation of key process activities, Inf. Softw. Technol. 50 (11) (2008) 1098–1113.
[121] M. Khurum, T. Gorschek, A systematic review of domain analysis solutions for product lines, J. Syst. Softw. 82 (12) (2009) 1982–2003.
[122] S. Buhne, K. Lauenroth, K. Pohl, Modelling requirements variability across product lines, in: Proceedings of the International Conference on Requirements Engineering, Paris, France, 2005, pp. 41–50.
[123] J. White, J.H. Hill, J. Gray, S. Tambe, A. Gokhale, D. Schmidt, Improving domain-specific language reuse with software product line techniques, IEEE Softw. 26 (4) (2009) 47–53.
[124] P. Padmanabhan, R.R. Lutz, Tool-supported verification of product line requirements, Autom. Softw. Eng. 12 (4) (2005) 447–465.
[125] K. Chen, W. Zhang, H. Zhao, H. Mei, An approach to constructing feature models based on requirements clustering, in: Proceedings of the International Conference on Requirements Engineering, Paris, France, 2005, pp. 31–40.
[126] P. Trinidad, D. Benavides, A. Durán, A. Ruiz-Cortés, M. Toro, Automated error analysis for the agilization of feature modeling, J. Syst. Softw. 81 (6) (2008) 883–896.
[127] S. Trujillo, D. Batory, O. Diaz, Feature oriented model driven development: a case study for portlets, in: Proceedings of the International Conference on Software Engineering, Minneapolis, MN, USA, 2007, pp. 44–53.
[128] R.E. Lopez-Herrejon, D. Batory, W. Cook, Evaluating support for features in advanced modularization technologies, in: Proceedings of the European Conference on Object-Oriented Programming, Glasgow, UK, 2005, pp. 169–194.
[129] A. Classen, Q. Boucher, P. Heymans, A text-based approach to feature modelling: syntax and semantics of TVL, Sci. Comput. Program. 76 (12) (2011) 1130–1143.
[130] S. Segura, R.M. Hierons, D. Benavides, A. Ruiz-Cortés, Automated metamorphic testing on the analyses of feature models, Inf. Softw. Technol. 53 (3) (2011) 245–258.
[131] J.-M. Davril, E. Delfosse, N. Hariri, M. Acher, J. Cleland-Huang, P. Heymans, Feature model extraction from large collections of informal product descriptions, in: Proceedings of the Joint Meeting on Foundations of Software Engineering, New York, NY, USA, 2013, pp. 290–300.
[132] M. Mendonca, D. Cowan, Decision-making coordination and efficient reasoning techniques for feature-based configuration, Sci. Comput. Program. 75 (5) (2010) 311–332.
[133] M. Acher, P. Collet, P. Lahire, R.B. France, Composing feature models, in: Proceedings of the Software Language Engineering Conference, Denver, CO, USA, 2009, pp. 62–81.
[134] M. Acher, P. Collet, P. Lahire, R.B. France, Familiar: a domain-specific language for large scale management of feature models, Sci. Comput. Program. 78 (6) (2013) 657–681.
[135] J. Liu, S. Basu, R.R. Lutz, Compositional model checking of software product lines using variation point obligations, Autom. Softw. Eng. 18 (1) (2011) 39–76.
[136] J. Kienzle, N. Guelfi, S. Mustafiz, Crisis management systems: a case study for aspect-oriented modeling, in: S. Katz, M. Mezini, J. Kienzle (Eds.), Transactions on Aspect-Oriented Software Development VII, Lecture Notes in Computer Science, 2010, pp. 1–22.
[137] D. Dhungana, P. Grünbacher, R. Rabiser, T. Neumayer, Structuring the modeling space and supporting evolution in software product line engineering, J. Syst. Softw. 83 (7) (2010) 1108–1122.
[138] I. Schaefer, R. Rabiser, D. Clarke, L. Bettini, D. Benavides, G. Botterweck, A. Pathak, S. Trujillo, K. Villela, Software diversity: state of the art and perspectives, Int. J. Softw. Tools Technol. Transf. 14 (5) (2012) 477–495.
[139] L. Neves, L. Teixeira, D. Sena, V. Alves, U. Kulezsa, P. Borba, Investigating the safe evolution of software product lines, in: Proceedings of the International Conference on Generative Programming and Component Engineering, New York, NY, USA, 2011, pp. 33–42.
[140] L.P. Tizzei, M. Dias, C.M. Rubira, A. Garcia, J. Lee, Components meet aspects: assessing design stability of a software product line, Inf. Softw. Technol. 53 (2) (2011) 121–136.
[141] M.I. Ullah, G. Ruhe, V. Garousi, Decision support for moving from a single product to a product portfolio in evolving software systems, J. Syst. Softw. 83 (12) (2010) 2496–2512.
[142] I. Reinhartz-Berger, Towards automatization of domain modeling, Data Knowl. Eng. 69 (5) (2010) 491–515.
[143] J. Lee, D. Muthig, M. Naab, A feature-oriented approach for developing reusable product line assets of service-based systems, J. Syst. Softw. 83 (7) (2010) 1123–1136.
[144] S. Zschaler, P. Sánchez, J.a. Santos, M. Alférez, A. Rashid, L. Fuentes, A. Moreira, J.a. Araújo, U. Kulesza, VML∗ – a family of languages for variability management in software product lines, in: Proceedings of the International Conference on Software Language Engineering, Eindhoven, The Netherlands, 2010, pp. 82–102.
[145] M. Galster, D. Weyns, D. Tofan, B. Michalik, P. Avgeriou, Variability in software systems – a systematic literature review, IEEE Trans. Softw. Eng. 40 (3) (2014) 282–306.
[146] E. Bagheri, D. Gasevic, Assessing the maintainability of software product line feature models using structural metrics, Softw. Qual. J. 19 (3) (2011) 579–612.
[147] D. Mellado, E. Fernández-Medina, M. Piattini, Security requirements engineering framework for software product lines, Inf. Softw. Technol. 52 (10) (2010) 1094–1117.
[148] F. Roos-Frantz, D. Benavides, A. Ruiz-Cortés, A. Heuer, K. Lauenroth, Quality-aware analysis in product line engineering with the orthogonal variability model, Softw. Qual. Control 20 (3–4) (2012) 519–565.
[149] S. Montagud, S. Abrahão, E. Insfran, A systematic review of quality attributes and measures for software product lines, Softw. Qual. Control 20 (3) (2012) 425–486.
[150] B. Mohabbati, D. Gašević, M. Hatala, M. Asadi, E. Bagheri, M. Bošković, A quality aggregation model for service-oriented software product lines based on variability and composition patterns, in: Proceedings of the International Conference on Service-Oriented Computing, Paphos, Cyprus, 2011, pp. 436–451.
[151] R. Rabiser, P. Grunbacher, D. Dhungana, Requirements for product derivation support: results from a systematic literature review and an expert survey, Inf. Softw. Technol. 52 (3) (2010) 324–346.
[152] J. White, D. Benavides, D. Schmidt, P. Trinidad, B. Dougherty, A. Ruiz-Cortes, Automated diagnosis of feature model configurations, J. Syst. Softw. 83 (7) (2010) 1094–1107.
[153] K. Lee, K.C. Kang, Usage context as key driver for feature selection, in: Proceedings of the Software Product Line Conference, in: Lecture Notes in Computer Science, Jeju Island, South Korea, 2010, pp. 32–46.
[154] J. Guo, J. White, G. Wang, J. Li, Y. Wang, A genetic algorithm for optimized feature selection with resource constraints in software product lines, J. Syst. Softw. 84 (12) (2011) 2208–2221.
[155] M. Dong, D. Yang, L. Su, Ontology-based service product configuration system modeling and development, Expert Syst. Appl. 38 (9) (2011) 11770–11786.

[156] M. Boskovic, E. Bagheri, D. Gasevic, B. Mohabbati, N. Kaviani, M. Hatala, Automated staged configuration with semantic web technologies, Int. J. Softw. Eng. Knowl. Eng. 20 (4) (2010) 459–484.
[157] A. Sayyad, J. Ingram, T. Menzies, H. Ammar, Scalable product line configuration: A straw to break the camel's back, in: Proceedings of the International Conference on Automated Software Engineering, Silicon Valley, CA, USA, 2013, pp. 465–474.
[158] V. Alves, N. Niu, C. Alves, G. Valença, Requirements engineering for software product lines: a systematic literature review, Inf. Softw. Technol. 52 (8) (2010) 806–820.
[159] S. Oster, F. Markert, P. Ritter, Automated incremental pairwise testing of software product lines, in: Proceedings of the Software Product Line Conference, Jeju Island, South Korea, 2010, pp. 196–210.
[160] A. Hervieu, B. Baudry, A. Gotlieb, PACOGEN: automatic generation of pairwise test configurations from feature models, in: Proceedings of the International Symposium on Software Reliability Engineering, Hiroshima, Japan, 2011, pp. 120–129.
[161] G. Perrouin, S. Sen, J. Klein, B. Baudry, Y.L. Traon, Automated and scalable t-wise test case generation strategies for software product lines, in: Proceedings of the International Conference on Software Testing, Verification and Validation, Paris, France, 2010, pp. 459–468.
[162] C.H.P. Kim, D.S. Batory, S. Khurshid, Reducing combinatorics in testing product lines, in: Proceedings of the International Conference on Aspect-oriented Software Development, New York, NY, USA, 2011, pp. 57–68.
[163] G. Perrouin, S. Oster, S. Sen, J. Klein, B. Baudry, Y. le Traon, Pairwise testing for software product lines: comparison of two approaches, Softw. Qual. J. 20 (3–4) (2012) 605–643.
[164] E. Uzuncaova, S. Khurshid, D. Batory, Incremental test generation for software product lines, IEEE Trans. Soft. Eng. 36 (3) (2010) 309–322.
[165] M. Franceschet, A comparison of bibliometric indicators for computer science scholars and journals on web of science and Google scholar, Scientometrics 83 (1) (2010) 243–258.
[166] J.C. de Winter, A.A. Zadpoor, D. Dodou, The expansion of Google scholar versus web of science: a longitudinal study, Scientometrics 98 (2) (2014) 1547–1565.
[167] P. Jacso, Google scholar revisited, Online Inf. Rev. 32 (1) (2008) 102–114.
[168] R. Cathcart, A. Roberts, Evaluating Google scholar as a tool for information literacy, Internet Ref. Serv. Q. 10 (3–4) (2005) 167–176.
[169] R. Vine, Google scholar, J. Med. Libr. Assoc. 98 (4) (2006) 97–99.
[170] P. Jacso, As we may search. comparison of major features of the web of science, Scopus, and Google scholar citation-based and citation-enhanced databases, Curr. Sci. 89 (9) (2005) 1537–1547.
[171] J.P.A. Ioannidis, A. Tatsioni, F.B. Karassa, Who is afraid of reviewers' comments? Or, why anything can be published and anything can be cited, Eur. J. Clin. Investig. 40 (4) (2010) 285–287.
[172] J. Beall, "Predatory" open-access scholarly publishers, Charlest. Advis. 12 (4) (2010) 10–17.
[173] M.A. Martinez, M.J. Cobo, M. Herrera, E. Herrera-Viedma, Analyzing the scientific evolution of social work discipline using science mapping, Res. Soc. Work Pract. 25 (2) (2015) 257–277.
[174] A.-W.K. Harzing, R. van der Wal, Google scholar as a new source for citation analysis, Ethics Sci. Environ. Polit. 8 (1) (2008) 61–73.
[175] A. Nederhof, E. Van Wijk, Mapping the social and behavioral sciences worldwide: use of maps in portfolio analysis of national research efforts, Scientometrics 40 (2) (1997) 237–276.