An Integrated Model for Content Management, Presentation, and Targeting

Hans-Werner Sehring
Namics – A Merkle Company
Hamburg, Germany
e-mail: hans-werner.sehring@namics.com

Abstract—**The World Wide Web is the basis for increasingly many information and interaction services. Personalization provides users with information and services that are adequately tailored to their current needs. Targeting, a form of implicit personalization for groups of users, comes to broader practical use for a growing number of commercial websites. The wider adoption results from the availability of platforms that incorporate targeting. Solutions are usually built on top of content management systems used for the production of websites. The definitions required for targeting are related to content, but they are superimposed in the sense that they are not an integral part of the content model or the content itself. This paper presents an initial model that is used to study the integration of models for content, content visualizations, and content targeting. Potential benefits from an integrated model are manifold. It allows expressing personalization rules along with the content they refer to in a consistent way. This way, personalization is applied by putting content in context rather than through superimposed targeting rules. By expressing personalization rules in the same context-dependent and evolvable way as content, they can also evolve over time and can be adapted to different user contexts. On top of that, they can be defined and maintained by content editors and other users of a content management system.**

Keywords-personalization; targeting; segmentation; context-aware content management; content management.

I. INTRODUCTION

The World Wide Web has undergone a tremendous development. For over two decades now, there is research on *personalization* of contents published on the web and of the presentations used for publication.

There is a wide range of personalization approaches for different purposes and goals [1]. These approaches differ in several aspects [2], e.g., in the way personalizations are derived: explicitly by users stating their preferences or implicitly by deriving them from users behavior and habit. An example for explicit personalization are websites that allow the user to name their interests or that allow to individually rearrange parts of the web site. Implicit personalization is achieved, e.g., by observing interactions of a user with a website [3] or by taking previously visited websites into consideration (customer journeys, at best).

Personalization approaches also differ in the subject of the individual adaptations, e.g., content or content representations (visualizations of content created for publication). Content personalization can be found, e.g., in online shops where users receive individual pricing. Content visualizations are personalized by, e.g., using lists of content entries where these lists are ordered in a user-specific way.

Personalization has already been adopted to a range of specific, innovative websites, in particular those that confront the user with large amounts of content. Such websites use it to filter and prioritize content based on assumed user preferences.

Currently, *targeting* is applied by an increasing number of commercial websites. We consider targeting as implicit personalization of content for user groups. The adaptation of content is limited to predefined points, though. Typically, part of the content is selected from building blocks that are prepared for the different user groups.

A set of tools that has emerged during the past years constitutes the basis that allows configuring websites for personalization. Examples are personalization engines built into content management systems and commerce platforms, as well as external personalization services that allow adjusting websites to specific user groups.

There is a lack of models that would cover multiple kinds of personalization approaches [4] and, therefore, allow different usage scenarios to be integrated in one solution.

Typically, commercial products use means of personalization that are superimposed to a (non individualized) base system. A content management system, e.g., allows defining a content model according to which content will be edited, managed, and published. This content model is defined in a uniform way for all users and application scenarios. On a different layer, personalization is added by other means, typically rules that define how to adjust content representations of specific user groups.

Therefore, there is no coherence between content models, content visualization layouts, and personalization rules in such systems. Instead, content has to be defined with all possible audiences and usage scenarios in mind, visualizations have to provide the variations to be offered as personalizations, and personalization rules may be defined on the basis of these definitions.

Contemporary products typically require fixed content models and visualizations (or at least ones that cannot be changed by content editors). This only leaves such personalization rules at the content editors' disposal that can be defined with respect to the possibilities and constraints raised by content models and content visualizations.

The aim of this paper is providing first studies towards a fully integrated model that combines many aspects of

 ISBN: 978-1-61208-707-8

content and its personalized utilization. For this study, we use the Minimalistic Meta Modeling Language (M3L) [5] as a testbed. This language is well-suited for content models since it covers variations and contexts of content in a direct way. Insights into a variety of personalization options originate from previous research on Concept-oriented Content Management [6]. These insights are transferred to M3L models.

Future research will investigate how to employ such integrated models to cover a wide range of personalization approaches and applications. With the help of such models it will be possible to use more than the set of predefined configuration options that contemporary systems exhibit. Instead, these models are expected to unveil personalization capabilities over all aspects of services, their content, and their appearance, as well as to give the possibility of utilizing the interconnections between these.

The rest of this paper is organized as follows. Section II describes targeting approaches typically found in commercial software products. Section III provides a short introduction into the M3L. Section IV presents a first modeling experiment to utilize the M3L for expressing and integrating the common targeting approach into content models for websites. Conclusions and acknowledgement close the paper.

II. TARGETING IN COMMERCIAL PRODUCTS

There is a wide range of approaches to personalization that can be found in the literature and in prototype implementations. In this paper, we constrain ourselves to targeting which is of particular importance for commercial websites. Targeting is a form of implicit personalization of content assembled for presentation with respect to a customer group. The personalization itself is directed by rules set up by content editors.

A. Segment-based Targeting Rules

For targeting, as it is found in many commercial products, users of a web site are assigned *segments*. Segments are categories describing a user's interest or preferences. These are predefined for a particular website (though there are scientific approaches that include deriving segments by, e.g., means of clustering [7]).

The assignment of segments to users is based on *tracking* (or *analytics*) used during web page delivery. By tracking, accesses to web pages are recorded. Depending on the granularity required, interactions on smaller parts than a whole page may be counted [8].

From the web pages visited by a user, her or his interests are derived by collecting the topics covered by those web pages. The web pages considered in this collection could be, e.g, those web pages that have been visited most often, or the web pages for which the visits exceed a given threshold.

The segments assigned to a user (by that time) are used as a parameter to content selection and production of documents from content. This way, content and its representations are personalized for user groups, namely groups consisting of users with the same segments assigned.

B. Related Work

Targeting is found in diverse systems and services, e.g., in Content Management Systems (CMSs), commerce systems, and marketing suites.

1) Personalization Engines in Content Management Systems. Some CMSs have means of segmentation built in. These systems allow equipping content with rules for the selection of content to be included in published web pages based on user segmentation. Like many others, the CMS products of CoreMedia [9] and Sitecore [10] work this way.

2) Superimposed Personalization. Instead of an integrated personalization engine inside a CMS, personalization can alternatively be applied on the basis of an external service. Adobe Target [11] is a prominent representative of this personalization approach.

3) Consideration of Additional Information on Users. Instead of just considering user behavior in the form of web page access profiles, increasingly many applications are also based on explicit customer data. Such data come from, e.g., a Customer Relationship Management (CRM) system, from the history of transactions in a commerce system, from the history of cases in a support system, or from feedback given by means of ratings. Personalization may additionally be based on context information, e.g., the time of day, the device the visitor uses, or some kind of work mode she or he is in [12]. Such context information is partially considered in commercial personalization engines.

III. THE MINIMALISTIC META MODELING LANGUAGE

The *Minimalistic Meta Modeling Language* (*M3L*, pronounced "mel") is a modeling language that is applicable to a range of modeling tasks. It proved particularly useful for context-aware content modeling [13].

For the purpose of this paper, we only introduce the static aspects of the M3L in this section. Dynamic evaluations that are defined by means of different rules are presented in the subsequent section.

The descriptive power of M3L lies in the fact that the formal semantics is rather abstract. There is no fixed domain semantics connected to M3L definitions. There is also no formal distinction between typical conceptual relationships (specialization, instantiation, entity-attribute, aggregation, materialization, contextualization, etc.).

A. Concept Definitions and References

A M3L definition consists of a series of definitions or references. Each definition starts with a previously unused identifier that is introduced by the definition and may end with a period, e.g.:

```
Person.
```

A reference has the same syntax, but it names an identifier that has already been introduced.

We call the entity named by such an identifier a *concept*.

The keyword is introduces an optional reference to a *base concept*, making the newly defined concept a *refinement* of it.

A specialization relationship as known from object-oriented modeling is established between the base concept and the newly defined derived concept. This relationship leads to the concepts defined in the context (see below) of the base concept to be visible in the derived concept.

The keyword is always has to be followed by either a, an, or the. The keywords `a` and `an` are synonyms for indicating that a classification allows multiple sub-concepts of the base concept:

```
Peter is a Person. John is a Person.
```

There may be more than one base concept. Base concepts can be enumerated in a comma-separated list:

```
PeterTheEmployee is a Person, an Employee.
```

The keyword `the` indicates a closed refinement: there may be only one refinement of the base concept (the currently defined one), e.g.:

```
Peter is the FatherOfJohn.
```

Any further refinement of the base concept(s) leads to the redefinition ("unbinding") of the existing refinements.

Statements about already existing concepts lead to their redefinition. For example, the following expressions define the concept Peter in a way equivalent to the above variant:

```
Peter is a Person.
Peter is an Employee.
```

B. Content and Context Definitions

Concept definitions as introduced in the preceding section are valid in a context. Definitions like the ones seen so far add concepts to the top of a tree of contexts. Curly brackets open a new context, e.g.:

```
Person { Name is a String. }
Peter is a Person{"Peter Smith" is the Name.}
Employee { Salary is a Number. }
Programmer is an Employee.
PeterTheEmployee is a Peter, a Programmer {
  30000 is the Salary.
}
```

We call the outer concepts the *context* of the inner, and we call the set of inner concepts the *content* of the outer.

In this example, we assume that concepts **String** and **Number** are already defined. The sub-concepts created in context are unique specializations in that context only.

As indicated above, concepts from the context of a concept are inherited by refinements. For example, Peter inherits the concept **Name** from **Person**.

M3L has visibility rules that correlate to both contexts and refinements. Each context defines a scope in which defined identifiers are valid. Concepts from outer contexts are visible in inner scopes. For example, in the above example the concept **String** is visible in **Person** because it is defined in the topmost scope. **Salary** is visible in **PeterTheEmployee** because it is defined in **Employee** and the context is inherited. **Salary** is not valid in the topmost context and in **Peter**.

C. Contextual Amendments

Concepts can be redefined in contexts. This happens by definitions as those shown above. For example, in the context of **Peter**, the concept **Name** receives a new refinement.

Different aspects of concepts can explicitly be redefined in a context, e.g.:

```
AlternateWorld {
  Peter is a Musician {
    "Peter Miller" is the Name.
  }
}
```

We call a redefinition performed in a context different from that of the original definition a *conceptual amendment*.

In the above example, the contextual variant of Peter in the context of **AlternateWorld** is both a **Person** (initial definition) and a **Musician** (additionally defined). The **Name** of the contextual **Peter** has a different refinement.

A redefinition is valid in the context it is defined in, in sub-contexts, and in the context of refinements of the context (since the redefinition is inherited as part of the content).

D. Concept Narrowing

There are three important relationships between concepts in M3L.

M3L concept definitions are passed along two axes: through visibility along the nested contexts, and through inheritance along the refinement relationships.

A third form of concept relationship, called *narrowing*, is established by dynamic analysis rather than by static definitions like content and refinement.

For a concept c_1 to be a narrowing of a concept c_2, c_1 and c_2 need to have a common ancestor, and they have to have equal content. Equality in this case means that for each content concept of c_2 there needs to be a concept in c_1's content that has an equal name and the same base classes.

For an example, assume definitions like:

```
Person { Sex. Status. }
MarriedFemalePerson is a Person {
  Female is the Sex.
  Married is the Status.
}
MarriedMalePerson is a Person {
  Male is the Sex.
  Married is the Status.
}
```

With these definitions, a concept

```
Mary is a Person {
  Female is the Sex.
  Married is the Status.
}
```

is a narrowing of `MarriedFemalePerson`, even though it is not a refinement of that concept, and though it introduces separate nested concepts `Female` and `Married`.

E. Semantic Rule Definitions

For each concept, one *semantic rule* may be defined.

The syntax for semantic rule definitions is a double turnstile ("|=") followed by a concept definition. A semantic rule follows the content part of a concept definition, if such exists.

A rule's concept definition is not made effective directly, but is used as a prototype for a concept to be created later.

The following example redefines concepts **MarriedFemalePerson** and **MarriedMalePerson**:

```
MarriedFemalePerson is a Person {
  Female is the Sex. Married is the Status.
} |= Wife.
MarriedMalePerson is a Person {
  Male is the Sex. Married is the Status.
} |= Husband.
```

The concepts **Wife** and **Husband** are not added directly, but at the time when the parent concept is evaluated. Evaluation is covered by the subsequent section.

Concepts from semantic rules are created and evaluated in different contexts. The concept is instantiated in the same context in which the concept carrying the rule is defined. The context for the evaluation of a rule (evaluation of the newly instantiated concept, that is) is that of the concept for which the rule was defined.

In the example above, the concept **Wife** is created in the root context and is then further evaluated in the context of **MarriedFemalePerson**.

Rules are passed from one concept to another by means of inheritance. They are passed to a concept from (1) concepts the concept is a narrowing of, and (2) from base classes. Inheritance happens in this order: Only if the concept is not a narrowing of a concept with a semantic rule then rules are passed from base concepts.

For example, **Mary** as defined above evaluates to **Wife**.

F. Syntactic Rule Definitions

Additionally, for each concept one *syntactic rule* may be defined.

Such a rule, like a grammar definition, can be used in two ways: to produce a textual representation from a concept, or to recognize a concept from a textual representation.

A semantic rule consists of a sequence of string literals, concept references, and **the name** expressions that evaluate to the current concept's name.

During evaluation of a syntactic rule, rules of referenced concepts are applied recursively. Concepts without a defined syntactic rule are evaluated to/recognized from their name.

For example, from definitions

```
WordList {
  Word. Remainder is a WordList.
} |- Word §" " Remainder.
OneWordWordList is a WordList |- Word.
Sentence { WordList. } |- WordList "."
HelloWorld is a Sentence {
  Words is the WordList {
    Hello is the Word.
    OneWordWordList is the Remainder {
      World is the Word.
} } }
```

the textual representation `Hello World.` is produced.

Syntactic rule evaluation is not covered in this article.

IV. A Model of Content Personalization

This section provides a first simple M3L model of content, its visualization on web pages, website users, web page accesses, and the targeting of the web pages to the users based on past web page accesses.

```
WebPage.
SegmentingWebPage is a WebPage {
  Topic is a Segment. }
User.
Visit {
  Visitor is a User.
  ViewedPage is a WebPage. }
Segment.
```

Figure 1. Base model for targetable websites.

A. A Web Page and User Behavior Model

Figure 1 shows the essence of a M3L model for a web page, its users, the web page accesses, and segments in which to classify users.

Actual web pages are defined as refinements of the **WebPage** concept. Such concepts contain content as needed and they evaluate syntactically to HTML code for the presentation of that page. Figure 2 shows an example.

A **SegmentingWebPage** has a **Topic** assigned. The topic is represented by a **Segment** (see below).

The **User** concept serves as the identity of a web page visitor. It may contain user data.

A **Visit** records the access of a user to a web page. In real-world applications, typically a tracking tool is used for this purpose.

Segments are used in a twofold manner: On web page accesses, they name the topic of a web page in order to derive the area of interest of a visitor. When delivering the web page in a personalized way, a user's segment is used to select and evaluate personalization rules.

Segments might be managed in a structure like shown in the example. Only the segments themselves are significant.

B. Tracking Web Page Visits

Targeting is based on the users' behavior. Behavior is analyzed by tracking web page accesses. In the example of the M3L model we do so by creating (or finding) a matching **Visit** instance for a web page and user.

If the user is unknown, we create a **User** concept instance at the time of the first request.

```
Teaser.
RessortPage is a SegmentingWebPage {
 Title is a String.
 MainContent is a String.
 NewsTeaser is a Teaser.
} |- §"<html>"...§"</html>"
SoccerOverviewPage is a RessortPage {
 Soccer is the Title.
 "On this page..." is the MainContent.
 Segments{Ressorts{Sports.}} is the Topic.
}
Segments {
 Ressorts {
  Politics is a Segment.
  Sports is a Segment. } }
```

Figure 2. Example of a targetable website.

```
Tracking {
 Score {
  SegmentedUser is a User.
  AssignedSegment is a Segment.
  Value. }
 Visit
 |= Score {
  Visitor is the SegmentedUser.
  ViewedPage { Topic. }
   is the AssignedSegment.
 }
 ScoreUpdate is a Score
 |= Score { 1 is the Value. }
 ScoreIncrement is a ScoreUpdate {
  Value is an Integer. }
 |= ScoreIncrement {
  Integer {
   Value is the Pred.
  } is the Value.
 }
}
```

Figure 3. Base model for tracking.

The assignment of segments to a user is based on the *score* a segment got for a user. This score is the number of visits of a user to web pages with a topic that equals that segment.

In order to measure scores, we introduce the base concept **`Integer`** with just enough conception in order to have the ability to count. To this end, **`Integer`**s have a reference **`Pred`** to their predecessor. Using this reference, the order of integers is defined. The numerical value of an **`Integer`** is thus the length of the chain of its predecessors. In M3L:

```
Integer { Pred is an Integer. }
0 is an Integer.
1 is an Integer { 0 is the Pred. }
```

The concepts defined in Figure 3 are used to manage scores. The **`Value`** of a **`Score`** that a segment has for a user is assigned an **`Integer`** concept as a refinement. **`Visit`**s have assigned the user and the visited page.

On every request of a user *u* for a web page *p*, the web server issues a

```
Tracking {
  Visit {
    u is the Visitor. p is the ViewedPage.
  } |= Score {
    Visitor is the SegmentedUser.
    p { Topic. } is the AssignedSegment. } }
```

`Visit` is here extended by a semantic rule in order to represent a function that updates the score of a segment for a user. The concept **`Tracking`** provides a scope for individual function invocations.

If such a score already exists with the given user and the web page's topic assigned (recognized by **`Value`** being an **`Integer`**), then it will be narrowed to the matching **`ScoreIncrement`**. That concept in return will increase the value by one. This addition is done by setting value to the successor of the current value.

Else, the semantic rule will initialize the score by setting the **`Value`** to the **`Integer 1`**.

```
CulturePage17 is a WebPage {
 "Museums and Exhibitions" is the Title.
 ReportOnNewExhibition is the MainContent.
}
Segments { Ressorts {
 Politics {
  CulturePage17 {
  LatestPollResults is a NewsTeaser. } }
 Sports {
  CulturePage17 {
   SoccerExhibition is a NewsTeaser.
   RunningGameScore is a NewsTeaser. } }
} }
```

Figure 4. Example of targeting definitions.

```
SegmentDetermination {
  InitialThreshold is an Integer.
  SegmentsOfUser {
    UTS is a User.
  }
  |= Score { UTS is the SegmentedUser. }
  Score_rec is a Score {
    Value is an Integer.
  } |= Score {
    Value { Pred. } is the Value.
    Threshold { Pred. } is the Threshold.
  }
  IncludedScore is a Score_rec {
    0 is the Threshold.
  } |= AssignedSegment.
  ExcludedScore is a Score_rec {
    0 is the Value.
  } |= Segments.
}
```

Figure 5. Base model for segmentation.

C. *Applying Targeting Rules*

When users are segmented, the segmentation can be used to create personalized web pages for users.

Figure 4 shows a simple example of a personalizable web page. The **`CulturePage17`** has a static title and static textual content. It also may contain a list of news teasers that is filled in the context of a user's segment(s). To target web pages to users, each request of a user *u* for a page *p* will lead to an evaluation of *p* in the context of *u*'s segment(s).

The segment(s) of a user typically are derived from the scores they have for that user. In the case of selecting the segments with a certain threshold, the definitions from Figure 5 are used in the selection process.

The highest ranked segments of a user are evaluated inside the concept **`SegmentDetermination`**, that serves as a scope for executions. The concept **`SegmentsOfUser`** acts as a function from **`User`**s to segments with scores above the threshold. That function is invoked in the scope.

The evaluation is based on an **InitialThreshold** that is set inside **SegmentDetermination**. Is is set to the value that has to be reached by scored segments.

The first "invocation" of **SegmentsOfUser** for a user collects all **Score**s of the given user. These scores are then narrowed down during function evaluation. Each iteration of the evaluation starts through the concept **Score_rec** that decreases both **Value** and **Threshold** by one.

If the **Threshold** reaches **0**, then the score is narrowed down to **IncludedScore**. In that case, the value was greater than the threshold. The score is replaced with the segment in this case, thus terminating the recursion.

If the **Value** reaches **0**, however, then the value was less than the threshold. In this case the recursion ends without a specific result by replacing it with **Segments**.

By using the results of the evaluation for the segment contexts used in Figure 4, requests to the sample page **CulturePage17** from a user *u* are targeted to this user:

```
SegmentDetermination {
  SegmentsOfUser { u is the UTS. }
    { CulturePage17. } }
```

At the same time as the targeted web page is derived, a request for a web page will also increment the matching score as defined in the previous subsection. This concludes the circle of segmenting and targeting.

This example just demonstrates the selection of content to display at a given position in a web page, as it is also possible with commercial products. With the approach demonstrated here, however, it will also be possible to personalize other aspects of a web page.

V. SUMMARY AND OUTLOOK

The paper concludes with a summary and an outlook.

A. Summary

Many forms of personalization are discussed in literature for quite some time now. Still, integrated models covering most or all aspects of personalization are missing in practice.

This paper presents a study on such an integrated model, that combines content modeling with personalization, and that allows expressing various forms of personalization.

The initial modeling approach achieves to integrate content, content representation, users, page visits, segments, user segmentation, and targeting "rules". This integration allows coherent definitions of targeted web sites.

B. Outlook.

This paper concentrates on implicit personalization of presentations for groups of users, in practice called targeting.

A next step would be to extend the model to other forms of personalization in order to investigate whether these fit in equally well and can be combined within one model.

Content delivery and consumption depends on the context of the user. The utilization of context information for personalization should fit the models well using the M3L. Still, this needs to be studied.

This paper covers an analysis based on a hypothetical model only. It now needs to be connected to a working web server in order to gain practical results.

To increase practical relevance, further information on users should be integrated into the segmentation process. Such information may come from a Customer Relationship Management (CRM) system, from transaction processing systems like shop solutions, and from customer journeys.

ACKNOWLEDGMENT

Targeting is one building block in many digitization projects. Valuable insights have, therefore, been gained thanks to colleagues, partners, and customers.

Though the model presented in this paper is not related to the work at Namics, the author is thankful to his employer for letting him follow his research ambitions based on experience made in customer projects.

REFERENCES

[1] K. Riemer and C. Totz, "The Many Faces of Personalization: An integrative economic overview of mass customization and personalization," in The Customer Centric Enterprise, M. M. Tseng and F. T. Piller, Eds. Berlin, Heidelberg: Springer, pp. 35-50, 2003.

[2] J. Blom, "Personalization: a Taxonomy," Proc. CHI EA '00 Extended Abstracts on Human Factors in Computing Systems, Apr. 2000, pp. 313-314.

[3] J. Zhang, "The Perils of Behavior-Based Personalization," Marketing Science, vol. 30, pp. 170-186, Dec. 2011.

[4] A. L. Montgomery and M. D. Smith, "Prospects for Personalization on the Internet," Journal of Interactive Marketing, vol. 23, pp. 130-137, Jul. 2008.

[5] H.-W. Sehring, "Content Modeling Based on Concepts in Contexts," Proc. Third International Conference on Creative Content Technologies, Sep. 2011, pp. 18-23.

[6] J. W. Schmidt and H.-W. Sehring, "Conceptual Content Modeling and Management," Perspectives of System Informatics, Lecture Notes in Computer Science, vol. 2890, pp. 469-493, 2003.

[7] T. Jiang and A. Tuzhilin, "Improving Personalization Solutions through Optimal Segmentation of Customer Bases," IEEE Transactions on Knowledge and Data Engineering, vol. 21, pp. 305-320, Mar. 2009.

[8] K. S. Kuppusamy and G. Aghila, "A Model for Web Page Usage Mining Based on Segmentation," International Journal of Computer Science and Information Technologies, vol. 2, pp. 1144-1148, 2011.

[9] Website of the CoreMedia AG. [Online]. Available from: https://www.coremedia.com/ [retrieved: March, 2019]

[10] Sitecore. *Behavioral Targeting and A/B Testing: What's Appropriate for Your Business?* [Online]. Available from: https://www.sitecore.com/de-de/company/blog/416/behavioral-targeting-and-a-b-testing-what-39-s-appropriate-for-your-business-4063 [retrieved: March, 2019]

[11] Adobe Target. *Behavioral targeting*. [Online]. Available from: https://www.adobe.com/marketing/target/behavioral-targeting.html [retrieved: March, 2019]

[12] M. Gorgoglione, C. Palmisano, and A. Tuzhilin, "Personalization in Context: Does Context Matter When Building Personalized Customer Models?" Proc. 6th IEEE International Conference on Data Mining (ICDM 2006), Dec. 2006, pp. 222-231.

[13] H.-W. Sehring, "Context-aware Storage and Retrieval of Digital Content: Database Model and Schema Considerations for Content Persistence," International Journal on Advances in Software, vol. 11, pp. 311-322, Dec. 2018.