14

Running: A Mixed Language Software as an e-Learning Solution for the State Budget Management

Guillaume Koum and Innocent Dzoupet
[1]*Computer Sciences, Mathematics and Systems Simulation Laboratory-*
Polytechnic National HighSchool, Yaounde,
[2]*iTech Company, Yaounde,*
Cameroun

1. Introduction

Governments' attitude relevant to financial information systems consists to computerize systems, following sectors of activities which give name to computer applications such that:

- Taxes
- Customs
- Treasury
- Budget
- Etc…

Otherwise, research has shown that, for technology to be effective tool in the learning process, three conditions must be met. (1) there needs to be high-quality content that is relevant to the learning needs of the students and to the education system of the country; (2) teachers must be trained to integrate the new digital resources into their teaching practices; (3) technical support must be readily available to address any problems that arise [1].

In the State budget management a Government in achieving the objectives outlined in the finance law, undertakes at least two key activities which are to [2]:

- Coordinate the State Budget process by the ministry of Finance
- Work with other State Government ministries to monitor and assess financial and non-financial performance against Budget forecasts.

In open source software development, many issues are solved by integrating the users of the software in the development process, or even letting these users build the system themselves. Education requirements in languages training can be derived from software engineering. If the application is fully bilingual, it must provide the ability to receive the data in the two languages without showing a preference for either language. For us, we admit that these languages are French and English. So the interface to the data provider allows for the selection of available data in both French and English. Whenever both French and English are used on the same page, the language of each passage of text is indicated.

Electronic learning (e-learning) is a concept where teaching programmes conceived elsewhere are presented on the Web, under the supervision of a lecturer. Lessons organised in chapters are proposed on the Web. The lecturer must follow the sequential presentation of the material. The domains of application include the learning of a new language [3]. Consequently the user can learn either French or English with the described system.

We consider that the current system is partially bilingual. We assure that there is not a need of standard for the data interface and that the e-Government agreement is not relevant. The mixed language content is visible on buttons as well as in the texts.

An electronic issue management system, alternatively known as a help desk system, refers to a computer application that can be used to electronically automate the process of managing business issues, including problems, defects, tasks, changes or new requests. The difficulties found in using such a system are often from the lack of expertise to resolve the issues that are stored by the system. Ontology and case-based reasoning are suited to better provide structured information and enable the capturing of tacit knowledge of experts of the domain [4], [5].

An informal ontology may be specified by a catalog of types that are either undefined or defined only by statements in a natural language. A formal ontology which can be compared to a terminological ontology is specified by a collection of names for concept and relation types organized in a partial ordering by the type-subtype relation [6], [7]. A formal ontology is further distinguished by the way the subtypes are distinguished from their super-types: an *axiomatic ontology* distinguishes subtypes by axioms and definitions stated in a formal language, such as logic or some computer-oriented notation that can be translated to logic; a *prototype-based ontology* distinguishes subtypes by a comparison with a typical member or *prototype* for each subtype.

We use both terminological and prototype-based ontology to conceptualize translation issues as far as the State management budget is concerned. We implement all this in a case-based reasoning (CBR) system. Several issues in knowledge representation and ontology are closely related to CBR due to the fact that CBR is richer not only in the complexity of case-base design but also in the background of knowledge representation

The chapter is organized as followings. After the introduction, the first section talks about multilingual and bilingual software. The way bilingual software can maintain interface in two languages is underlined. Mind mapping is considered as a translation technique. The second section presents the mechanisms surrounding the preparation and execution of the State budget. The third section is dedicated to different ontology models required for the system. The fourth section presents case-based reasoning (CBR), a technique in artificial intelligence field conducive to learning. CBR is associated with the prototype-based ontology to solve language translation problems. The fifth section describes RUNNING application through variety of functionalities. The sixth section shows some results of the system about State revenue and expenditure. Finally, the chapter is concluded with perspectives expected.

2. Bilingual software

2.1 Multilingual software

Writing multilingual software has never been easy {8]. Translating user interface is only half of the task. It is mandatory for user satisfaction. The screen must not contain text in more than one language. This is why the language choice is an acquired. One has to remember about many different things, such as time/date format, money symbol and a lot of other things. But even translation interface used to be very difficult task because most programming environments lack usable tools for it. For example, in Visual C++ to create multilingual software one has to manually translate *.rc file and then after every change to resources, one has to manually retranslate *.rc file. Even languages that were designed with creating multilingual software in mind, such as Java and .NET, there are no built-in tools for translation.

In this regard, the Multi-Language Add-In for Visual Basic 6 provides a general solution for creating and maintaining Visual-Basic projects with support for multiple languages [9].

This involves the following basic steps:

1. Identifying the strings to be translated.
2. Specifying the languages to be supported.
3. Translating the strings into each language.

The Multi-Language Add-In supports these steps as follows:

Identifying the strings to be translated

- The Add-In scans the controls on all Forms, UserControls and UserDocuments to find the text properties which require translation.
- The relevant properties are defined in a Controls Database. This database already supports most commonly used controls and can easily be extended to support any other controls used in each project.
- The Add-In scans program code for strings which may require translation. The strings are displayed in a table where are selected the strings which require translation.
- When a line is selected in the table, the corresponding line is displayed in the Visual Basic Editor.

Specifying the languages to be supported

- When Multi-Language support is added to a project, the original language is specified.
- Additional languages are added using a simple dialog.

Translating the strings into each language

- The strings from controls and the strings from source code are displayed in two separate tables.
- In each table, there is a row for each string and a column for each language.
- The strings can be edited in the table.
- The strings are stored in a project database.

2.2 User Interface in bilingual software

Bilingual software must support and facilitate the entry of data from users and collection of data from other systems and sources in both French and English for instance. The data held, managed and processed by a software application is usually derived from inputs external to the application [10].

Where a manual user interface is used to enter data, this interface should allow for the entry of all language-sensitive items in both languages. To achieve this, data entry fields can either be placed in parallel or sequentially on the same area of the interface or they can be arranged in different places, but linked to the same interface state.

Placing the data entry fields for each language on the same page is more straightforward for the user, making the requirement to enter both languages more explicit. Where the data entry fields for each language are placed separately, it is important that they occur within the same user 'state' where the user should be expected to enter data for both languages before progressing to the next step in their interaction with the application. One language is given prominence and should be the preferred language for the user.

In general, bilingual software applications ensure equal treatment of all languages and encourage entry of data for all supported languages. Since data isn't always available in both languages, it is important for a software application to allow users to enter data in a single language. However, there is the potential for users to neglect to enter data in the alternate language, even when it is available and the software application discourages this. Where the layout of the interface doesn't make it completely clear and evident that there is the capability to enter both languages, indicators must be located alongside those fields that require bilingual entry. Ideally, this indicator can also provide a link to where the equivalent data entry field in the alternate language is located.

Ideally, all data are held in both languages, with a full bilingual dimension. However the reality is that some data might be available in one language and not the other. The system is then partially bilingual. The design of the data search capability must be such that the user is provided with the maximum flexibility in how to search, what language to use for the search and what data to search.

2.3 Languages translation and mind mapping

On a very early stage of studying a language, students first check the words in a dictionary, they try then to link them together, often consult grammar books. However, once there is advance in studies, the process of getting the words' meanings and the process of re-constructing the sentence into the target language comes automatically and naturally together and this perhaps constitutes the difference between a language learner and a translator.

Professional translators work with more than words: they translate concepts as well [11]. This may mean that the sentence structure will differ greatly from the original text, and the entire vocabulary may differ from "dictionary" definitions, depending upon the register of the text. Sometimes a single sentence may be translated as two sentences, or two sentences may be combined into one. During the course of revision, sentence structure may be changed, and the translator may even come up with more effective ways of expressing concepts contained in the text

Translation techniques depend of what kind of texts to translate. If it is a legal or engineering text then terminology (words) is more important, then perhaps translating literally on first draft is satisfactory, but not desirable either. If it is an economic or a financial text, then this method will not work as it is first necessary to summarize the meaning of the whole text, not just the words or indeed the sentences.

However in general, professional translators are characterized by a lack of mind.

Memory is more than recalling information for exams or trivial games. It's an important work skill that users can develop and improve, because the ability to remember is a major advantage with tips like:

- remembering key statistics during a negotiation;
- quoting a precedent-setting action when making a decision;
- impressing clients with knowledge of their product lines:
- Etc….

Mind maps (also called concept maps or memory maps) are an effective way to link ideas and concepts in human brain, and then "see" the connections firsthand [12].

Mind mapping tends to focus on remembering facts and details. Consequently, memorizing correct responses is more important than understanding the concept. Mind mapping is a note-taking technique that records information in a way that shows how various pieces of information fit together. There's a lot of truth in the saying "A picture speaks a thousand words", and mind maps create an easily-remembered "picture" of the information on is trying to remember.

This technique is very useful to summarize and combine information from a variety of sources. It also allows thinking about complex problems in an organized manner, and then presents findings in a way that shows the details as well as the big picture.

The mind map itself is a useful end product. Nevertheless, the process of creating the map is just as helpful for the memory. Since languages translators does not have mind, if their support tools are associated with mind mapping a new approach can be defined to implement language translation assistants for some domain of activities. In particular remembering terminology is necessary.

3. State budget process

Preparation of the Government's annual budget is the responsibility of the staff of the General Direction of Budget. The execution and implementation of the budget is an on-going annual cycle comprised of the following highlights [13].

3.1 Financial analysis of actual and projected expenditure

At the end of a fiscal year, the Budget team reviews the financial data for each ministry and program. This analysis provides important insights into how State Government raises revenue and spends money. Generally these insights lead to assist the development of the Government's budget proposal for the coming fiscal year.

3.2 Preparation of the budget

Following ministerial responses to credits requirements, the Budget team prepares the ministry's recommended budget proposal. A document filled at this occasion is sent back by the branch ministries to the ministry of Finance.

3.3 Submission of the budget to the National Assembly

The Government presents the budget proposal to the Legislature during an annual budgetary session. According to the fiscal regime of the State, the Government shall propose this budget document to the National Assembly with a delay before the budgetary session.

3.4 Budget enactment

The National Assembly must consider and adopt the final version of the appropriation bill within a fixed time following the beginning of the session. The consideration of the budget proposal starts in the finance and budget Committee[1]. Here budget discussions are conducted by the Committee President to resolve all major budget issues. At the end of the discussions and after all the budget analyses, the Committee then develops its budget recommendations for consideration by the entire house in the plenary session.

The adoption of the final version of the bills takes place in the plenary session in the presence of the entirety of the House and the whole Government team. Presided over by the Speaker of the House, this adoption goes through two steps including the budget hearings known as the parliamentary debate and the passing or the vote on the bills. At the beginning of the parliamentary debate, the Head of Government presents the information on the trends of the national economy, the public finances guidelines and the development of major investment projects. Then follows the questions and answer session during which Cabinet Ministers are called upon by parliamentarians to defend their budget requests.

The passing of the finance law like that of other legislations is conducted by the Speaker of the House at the floor of the entire House. Unlike other legislations, the finance law must be debated and passed in two separate and sequential steps. First, the first part of the finance law is debated and passed. It is only after the adoption of the first part that the second part of the finance law may be debated and passed.

3.5 Budget promulgation

Once the National Assembly considers the budget proposals and adopts the final version of the appropriation text, it forwards it to the Presidency of the Republic. The President of the Republic signs it into law within a fixed period following the receipt.

3.6 Budget implementation

After law promulgation, the implementation of the State budget begins. Ministries can then spend the appropriated funds. At the same time, the General Direction of Budget prepares the financial statements describing the prior fiscal year. At this point, the budget process begins a new exercise for the next fiscal year. During the fiscal year, the legislature may deem it necessary to adjust the current year's budget. This change is implemented via the Budget Adjustment Act.

[1] The finance and budget Committee is made off of a certain number of parliamentarians. Other Committees are set up at the National Assembly.

4. Ontology spot

The subject of *ontology* is the study of the *categories* of things that exist or may exist in some domain [13], [14]. The product of such a study, called *ontology*, is a catalog of the types of things that are assumed to exist in a domain of interest D from the perspective of a person who uses a language L for the purpose of talking about D. The types in the ontology represent the *predicates*, *word senses*, or *concept and relation types* of the language L when used to discuss topics in the domain.

4.1 Terminological ontology

In a terminological ontology categories need not be fully specified by axioms and definitions. An example of a terminological ontology is WordNet [15], whose categories are partially specified by relations such as subtype-supertype or part-whole, which determine the relative positions of the concepts with respect to one another but do not completely define them.

For this purpose; information in WordNet is organized around logical groupings called synsets. Each synset consists of a list of synonymous words or collocations, and pointers that describe the relations between this synset and other synsets. Two kinds of relations are represented by pointers: lexical and semantic. Lexical relations hold between semantically related word forms; semantic relations hold between word meanings.

Formal concept analysis approach is adopted for the terminological ontology construction. Formal background is constructed based on the concept of commitment order linked to other concepts. In the formal background, the object is one document concerning the information for supplier or beneficiary and the properties are the ontology concepts. The concepts and classification relationships between concepts are structured in the form of concept trees and translated into ontology.

Commitment and order are **direct antonyms**, a pair of words between which there is an associative bond resulting from their frequent co-occurrence. Commitment order can also be considered as a **holonym (Y** is a holonym of **X** if **X** is a part of **Y). A meronym** is the name of a constituent part of, the substance of, or a member of something **(X** is a meronym of **Y** if **X** is a part of **Y).** So, all proprieties in this document are meronyms **(**except commitment order which is a holonym**).**

Regarding the State budget management, the commitment order in the information for the supplier or beneficiary document appears every time a nature of voucher is indicated. They are three boxes with the words “Nature of document”.

- The certificate of indebtedness: it is accompanied by a commitment order in two copies the green original and the yellow duplicate.

The supplier shall:

1. Fill the order specified on the commitment order or documents attached thereto.
   Commitment order or any other orders should not be filled unless they are
   accompanied by a certificate of indebtedness.
   Only the Certificate of indebtedness commits the state and makes settlement possible.

2. After filling the order forward to issuing service indicated on the document
   a. The Certificate of indebtedness the following information
      the amount invoiced
      the method of settlement requested
      references of attached invoices.
   b. An original and a duplicate of the invoice(s) for which settlement is being requested.
   c. The yellow duplicate of the commitment order (the supplier should keep as proof the rein original of the commitment order).
   d. For contracts: a registered copy of the contract.

If the order is to be continued, the supplier should receive a new certificate of indebtedness with which to invoice subsequent order, in which case points (a) and (b) above should be observed.

The certificate of indebtedness should accompany the advice of settlement for the previous orders filled.

- The advice of settlement, as its name indicates, is a simple notice sent at the end of the transaction to the beneficiary of commitment order to inform him that the Treasury has effected settlement.

It is in beneficiary's interest to ensure that the amount and method of settlement shown on the advice of settlement are consistent with the information he supplied on the certificate of indebtedness.

Where the need arises, he should contact the treasury sub-department of expenditure

- The advice of cancellation: it is possible for the issuing service to cancel a commitment order wholly or in part.

In the event of total cancellation, the order should not be filled at all, no settlement can be made.

Where the cancellation is partial, the advice of cancellation should be accompanied by a new certificate of indebtedness showing only the amount corresponding to that part of the initial commitment that has not been cancelled.

In either case, the supplier or beneficiary should be sure to contract the issuing service

If that was not done at the time cancellation procedure were initiated.

A methodology based on Mind Maps captures all and only the knowledge we need for a scope of the commitment order. The borders among proprieties are traced by the tree branches and eliminate synonyms not in the lexical sense but in the semantic sense. The ontology model avoids the management of conflicting knowledge as a terminological issue.

A good Mind Map shows the "shape" of the subject, the relative importance of individual points, and the way in which facts relate to one another. A complete Mind Map may have main topic lines radiating in all directions from the center. Sub-topics and facts will branch off these, like branches and twigs from the trunk of a tree. The title of the subject explored is written in the center of the page (Commitment order in Figure1)

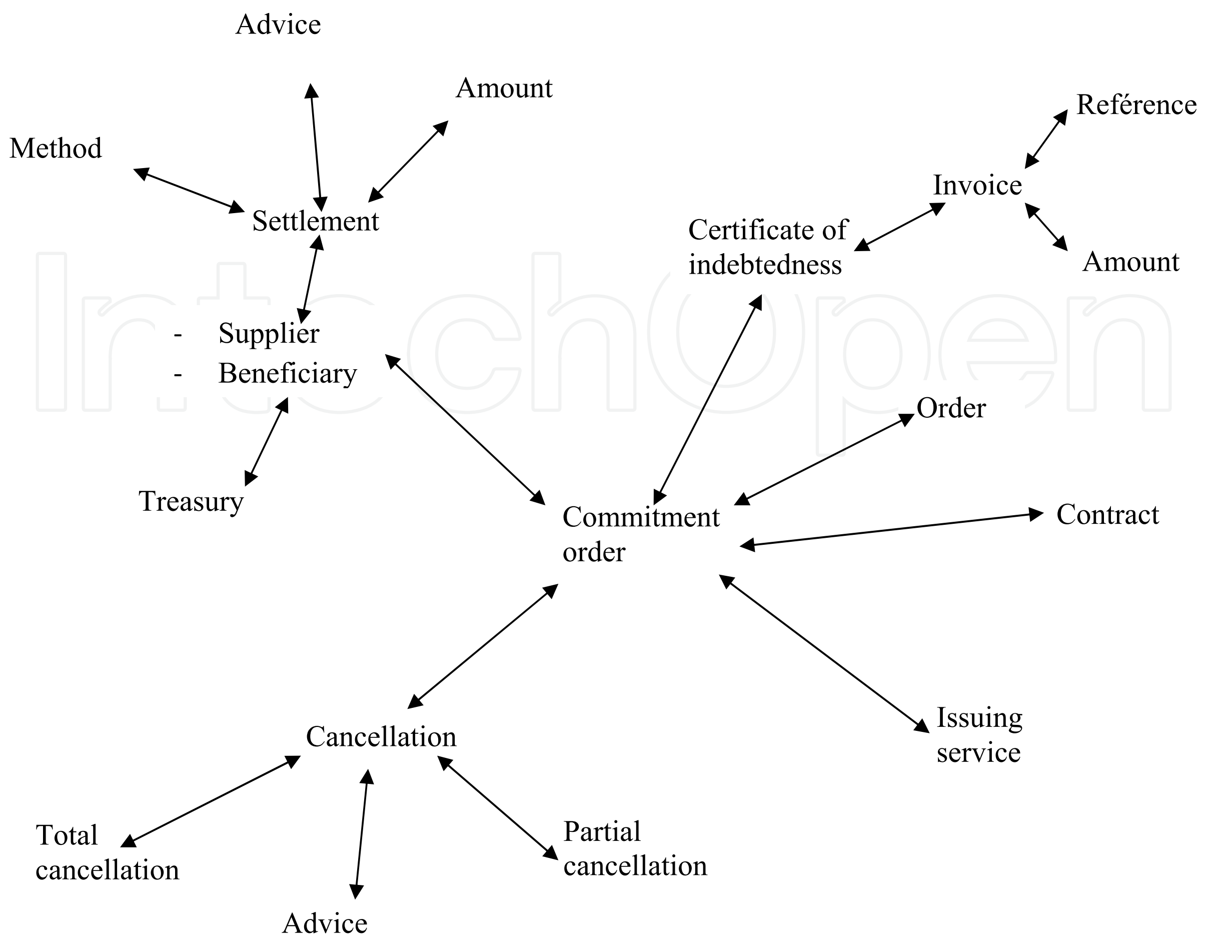


Fig. 1. Terminological ontology

4.2 Prototype-based ontology

A prototype-based ontology is described as a terminological ontology whose categories are distinguished by typical instances or *prototypes* rather than by axioms and definitions in logic [16].

For every category *c* in a prototype-based ontology, there must be a prototype *p* and a measure of *semantic distance* d(*x, y, c*), which computes the dissimilarity between two entities *x* and *y* when they are considered instances of *c*. Then an entity *x* can be classified by the following recursive procedure:

- Suppose that *x* has already been classified as an instance of some category *c*, which has subcategories $s_1,...,s_n$.
- For each subcategory s_i with prototype p_i, measure the semantic distance d(x, p_i, c).
- If d(x, p_i , c) has a unique minimum value for some subcategory s_i, then classify *x* as an instance of s_i, and call the procedure recursively to determine whether *x* can be further classified by some subcategory of s_i.
- If *c* has no subcategories or if d(x, p_i , c) has no unique minimum for any s_i, then the classification procedure stops with *x* as an instance of *c*, since no finer classification is possible with the given selection of prototypes.

Since a prototype-based ontology depends on examples, it is often convenient to derive the semantic distance measure by a method that learns from examples, such as statistics, cluster analysis, or neural networks.

We may allow an individual in our ontology to have multiple prototypes. Therefore we represent this relationship between Individuals with some hasPrototype property. So we need a way to define multiple inheritance mechanism on instances (and that is exactly what the hasPrototype property would do).

For the translation to be performed, some text analysis is required. Keyword counts are a simple analytical technique, but they ignore sentence structure. This is why a prototype-based ontology can be necessary in order to give a normal sense to sentences. While considering a sentence which is a text as an object, we can define prototypes on it. Moreover a sentence drains an idea and can be limited to a single term.

Example 1

There may be an object Regional_HospitalA in region A. Then, we proceed to create object Regional_HospitalB by specifying Regional_ HospitalA as its prototype, as well as specifying that Regional_HospitalB is situated in region B.

We precise that Regional_HospitalB has the same properties as Regional_HospitalA, except that its region is B.

We could choose to describe it like this:

```
 Individuals = {Regional_HospitalA, Regional_HospitalB, A, B}


Regional_HospitalA  hasRegion A

Regional_HospitalB  hasRegion B


Regional_HospitalB  hasPrototype Regional_HospitalA
```

While reasoning, we ignore the inherited property value if the object redefines it itself. In this case, the information that Regional_HospitalB is in region B should have priority over the information that Regional_HospitalB may be situated in region A because it is like Regional_HospitalA.

If there is no translated information on region B to be recorded by the user during its work, search through input ontology (investments list in the investment budget) and output ontology (projects list in the appropriate program) helps to solve the problem.

Input ontology

We consider the concept All State revenue. It can represent the foundation of the input ontology illustrated in Figure2.

Output ontology

We consider the concept « All State expenditure ». It can constitute the basis of the output ontology presented in Figure3.

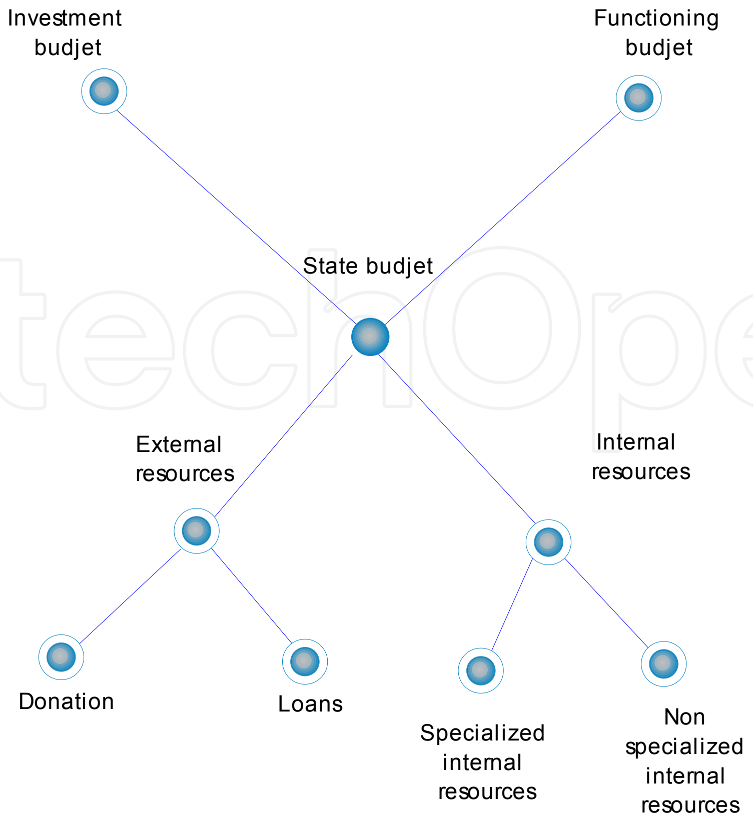


Fig. 2. Input ontology

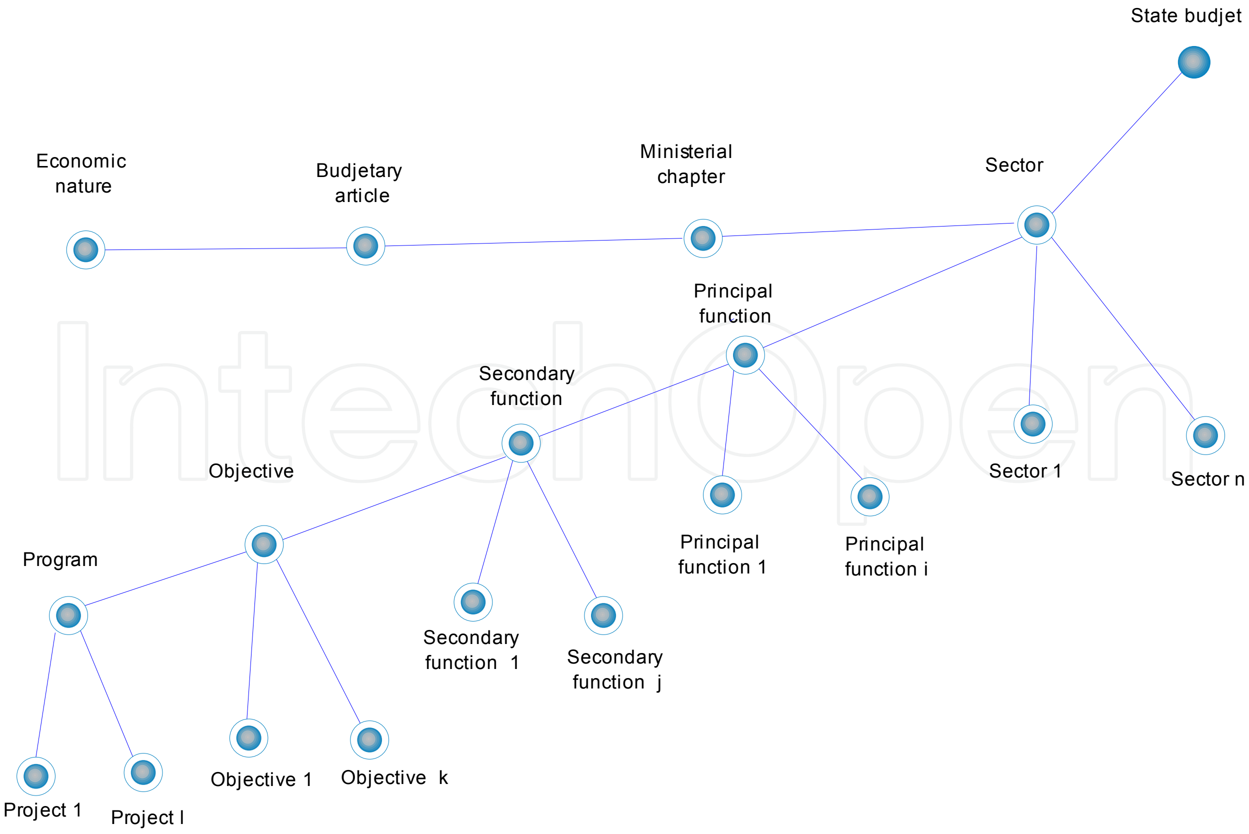


Fig. 3. Output ontology

Budgetary ontology

The finance law is a law of the land and provides for and authorizes all State revenue and expenditure for the upcoming fiscal year. It is founded on a budgetary ontology presented in figure….

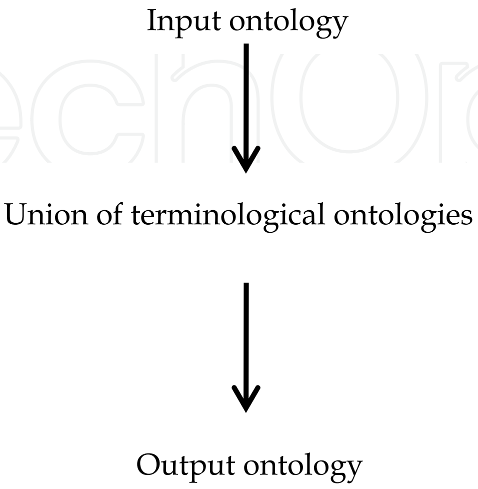


Fig. 4. Budgetary ontology

5. Case-based reasoning

5.1 Principle

Case-based reasoning (CBR) is the process of solving new problems based on the solutions of similar past problems [17], [18]. A new problem is solved by finding a similar past case, and reusing it in the new problem situation. The knowledge and reasoning process used by an expert to solve the problem is not recorded, but is implicit in the solution.

To solve a current problem: the problem is matched against the cases in the case base, and similar cases are retrieved. The retrieved cases are used to suggest a solution which is reused and tested for success. If necessary, the solution is then revised. Finally the current problem and the final solution are retained as part of a new case.

All case-based reasoning methods have in common the following process:

- retrieve the most similar case (or cases) comparing the case to the library of past cases;
- reuse the retrieved case to try to solve the current problem;
- revise and adapt the proposed solution if necessary;
- retain the final solution as part of a new case.

Retrieving a case starts with a (possibly partial) problem description and ends when a best matching case has been found. The subtasks comprise:

- identifying a set of relevant problem descriptors;
- matching the case and returning a set of sufficiently similar cases (given a similarity threshold of some kind); and
- selecting the best case from the set of cases returned.

Some systems retrieve cases based largely on superficial syntactic similarities among problem descriptors, while advanced systems use semantic similarities.

Reusing the retrieved case solution in the context of the new case focuses on: identifying the differences between the retrieved and the current case; and identifying the part of a retrieved case which can be transferred to the new case. Generally the solution of the retrieved case is transferred to the new case directly as its solution case.

Revising the case solution generated by the reuse process is necessary when the solution proves incorrect. This provides an opportunity to learn from failure.

Retaining the case is the process of incorporating whatever is useful from the new case into the case library. This involves deciding what information to retain and in what form to retain it; how to index the case for future retrieval; and integrating the new case into the case library.

5.2 Learning in Case-Based Reasoning

A very important feature of case-based reasoning is its coupling to learning. CBR is an approach to incremental, sustained learning, since a new experience is retained each time a problem has been solved, making it immediately available for future problems. Learning in CBR occurs as a natural by-product of problem solving. When a problem is successfully solved, the experience is retained in order to solve similar problems in the future. When an attempt to solve a problem fails, the reason for the failure is identified and remembered in order to avoid the same mistake in the future.

Case-based reasoning allows learning from experience, since it is usually easier to learn by retaining a concrete problem solving experience than to generalize from it.

5.3 Tradeoff between prototype-based ontology and case-based reasoning

The establishment of prototype-based ontology lays the foundation for case knowledge sharing. The tradeoff associates prototype-based ontology and case based reasoning technology in State budget management. The system structure of translation process is designed on the basis of prototype-based ontology and case-based reasoning theory and its application. The system not only considers the full use of bilingual domain experts' experiences and knowledge, but also can supports sharing and reuse of case knowledge in budgetary case bases. So, it solves the problem of knowledge reuse generated for those learning one or the other language with focus on budgetary domain. Also a simple semantic similarity algorithm can be admitted and used to compute similarity between new case and a case from case bases. So the real meaning of each term or sentence in different case expression is discovered and they are recorded in case bases with their mapping relation.

Example 2

We suppose that there exists an object Regional_RoadA1A2 in region A. Then, we need to create object Regional_RoadB1B2 in region B by specifying Regional_ RoadA1A2 as its prototype.

We indicate that Regional_RoadB1B2 has the same properties as Regional_RoadA1A2, except that its region is B.

As in example1, we could choose to describe it like this:

```
 Individuals = {Regional_RoadA1A2, Regional_RoadB1B2, A,B}

Regional_RoadA1A2 hasRegion A
Regional_RoadB1B2 hasRegion B

Regional_RoadB1B2 hasPrototype Regional_RoadA1A2
```

If there is no translated information on region B at the disposal of the user, we have to refer first to object Regional_HospitalB which already is known (example1).

Regional_HospitalB is linked to Regional_RoadB1B2 through the adjective Regional, in one of the terminological ontologies. The noun issuing from it is RegionB.

Two lessons can be observed from this situation.

- Regional_RoadB1B2 inherits from Regional_RoadA1A2 and from Regional_HospitalB because of hasPrototype propriety.
- The translation of RegionB's characteristics already done constitutes the most similar case retrieved, that must be reused, revised (a region is presented differently whether it is roads or hospitals) and retained, according to CBR schema (Figure5).

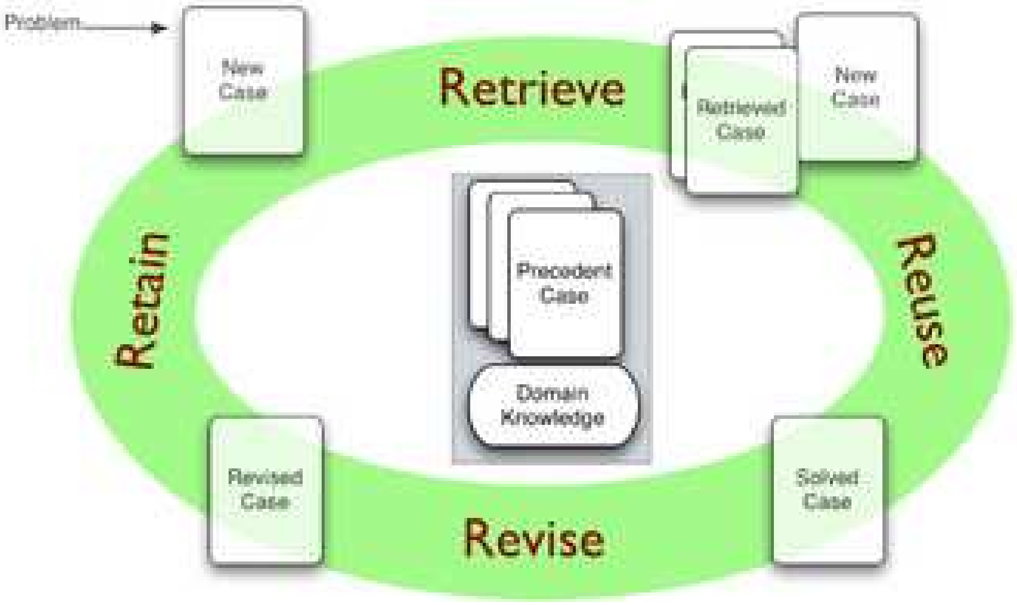


Fig. 5. CBR schema

The leaning cycle is then schematized in Figure6.

6. Running application

6.1 Presentation

RUNNING is a partially bilingual application. It is written in open source engineering through Symfony [19], a web application framework for PHP projects. Symfony aims to

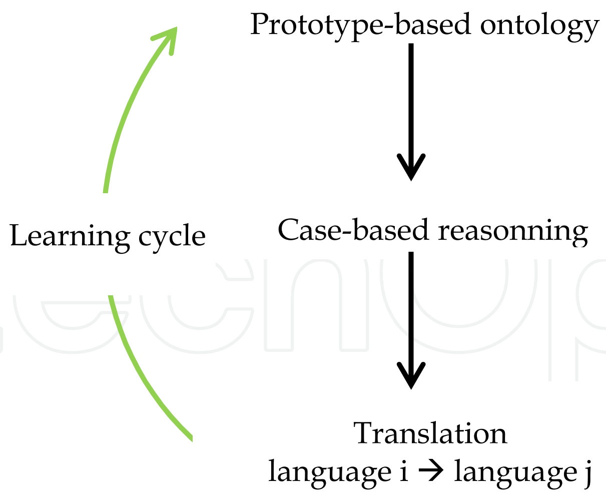


Fig. 6. Translation process

speed up the creation and maintenance of web applications, and to replace the repetitive coding tasks by power, control and pleasure. Symfony is a full-stack framework, a library of cohesive classes written in PHP. It provides an architecture, components and tools for developers to build complex web applications faster.

RUNNING can motivate users (either of French or of English) in the following categories:

- Drill-and-practice (vocabulary training with the state budget concepts)
- Tutorials (basic knowledge and facts required in the state budget operations)
- Etc…

6.2 System

RUNNING comprises the following modules:

- Management of service providers: It allows service providers identification.
- Management of beneficiaries: It identifies and treats beneficiaries.
- Management of executing agents: It identifies and treats credits managers.
- Opening of accounts: It concerns the opening of credits.
- Management of commitments: It allows provisions and cancels credits reservation.
- Management of settlements: It settles the expenditure or the canceling by the credits manager.
- Data restitutions: It prints reports on the state budget following up.
- Resources identification: It proceeds at resources identification, as loans and legs.
- Projects and programs identification: It establishes various projects and programs.
- Treatment of reception unit of the expenditure: The budgetary line is implicated.
- Allocation of credits by nature to the reception unit: The projection by nature is performed.
- Treatment of tenders: Calls of tenders are treated.
- Implementation of amendments: Amendments are implemented.
- Followings up of the state budget: Followings up activities are concentrated in this restitution.

- Secure access by level of confidentiality: It covers security issues.
- Documentation and online tutorials. The teaching component is recorded.

RUNNING offers a subsidiary mechanism of improving bilingual skills of users regarding the State budget management.

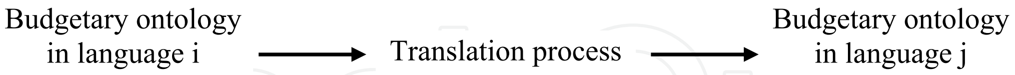

Budgetary ontology in language i
Translation process
Budgetary ontology in language j

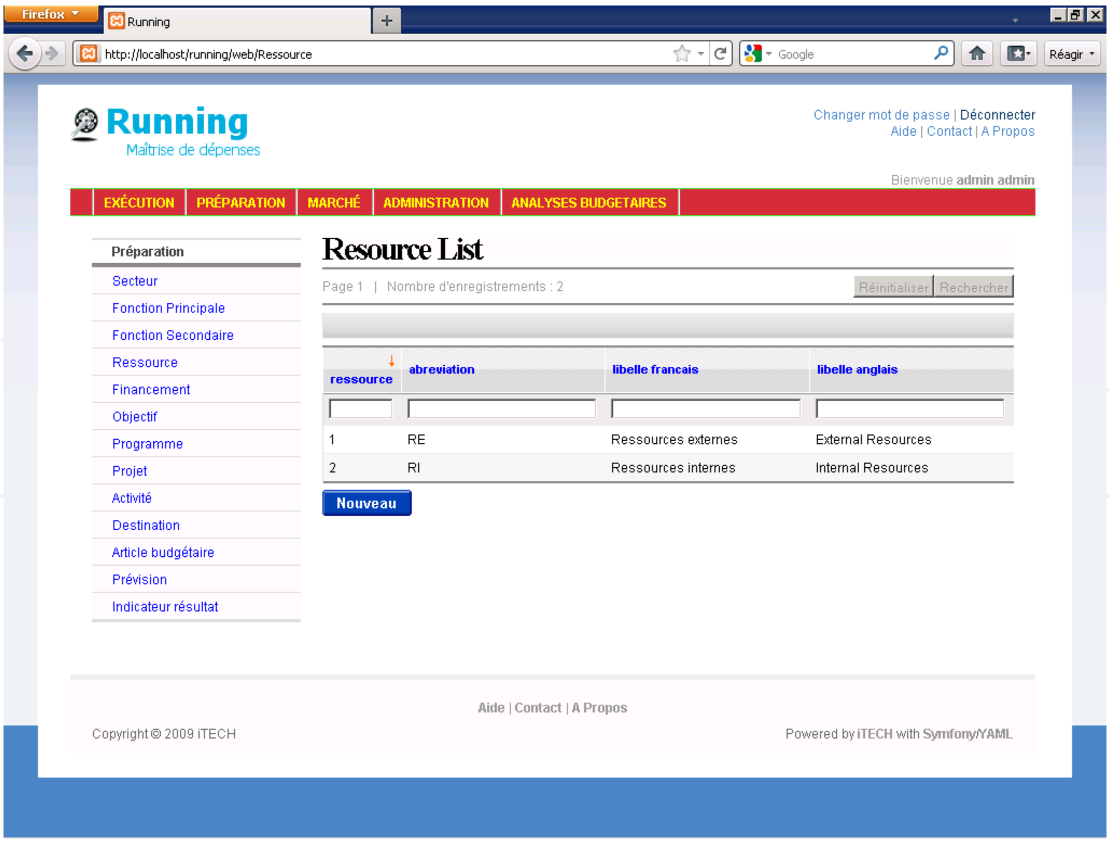
Fig. 7. Budget translation

7. Results

7.1 Results about state revenue

Screens are presented that express State revenue in terms of resources and new financing. RUNNING exploits the input ontology to help users who need to learn how to translate what contributes to budget revenue.

Screen1 displays the resource list constituted of internal and external resources. Internal resources comprise mainly variety of taxes and custom fees. An example of an external resource is the public debt. It can be a bilateral or a multilateral external resource. Resources are listed in French and in English.



Resource List

Page 1 | Nombre d'enregistrements : 2

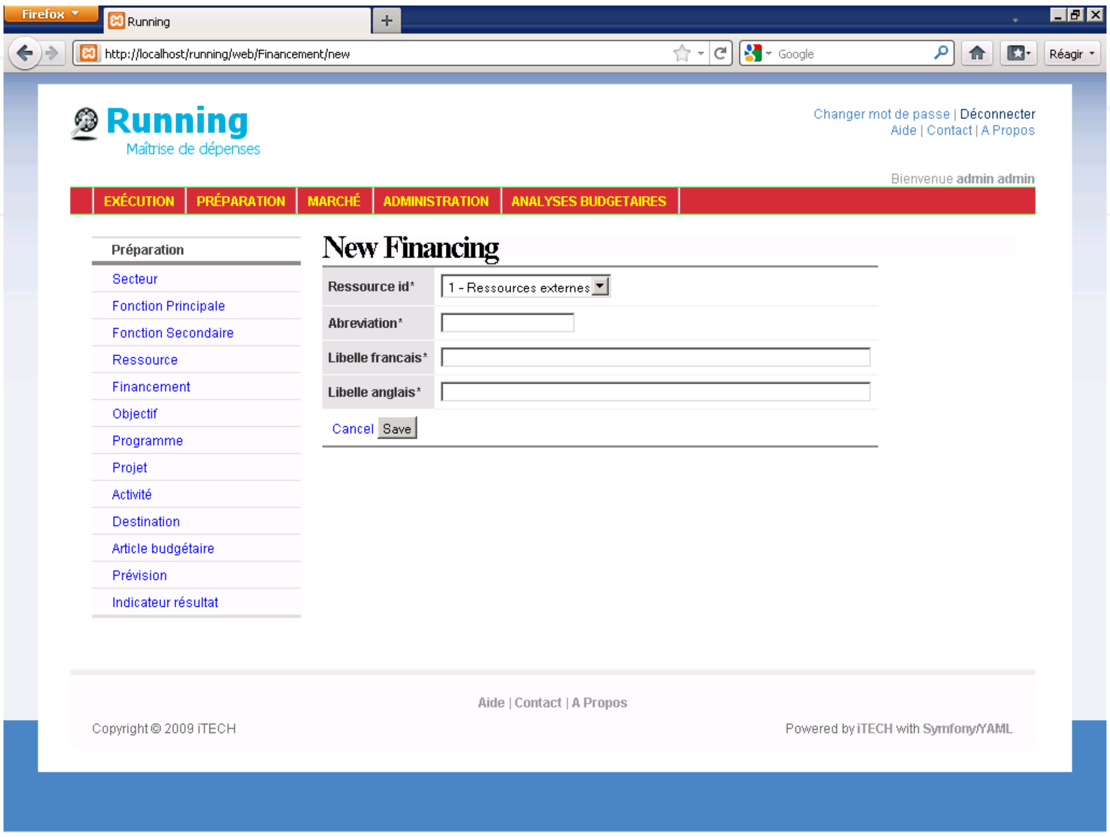
ressource	abreviation	libelle francais	libelle anglais
1	RE	Ressources externes	External Resources
2	RI	Ressources internes	Internal Resources

Copyright © 2009 iTECH

Powered by iTECH with Symfony/YAML

Screen 1. Resource List

Screen2 concerns new financings. They can be provided by financial markets through public titles. In the level of enterprises, securities are strengthened in a financial market for the compartment of the titles of capital and for the compartment of the titles of credit. The wording is introduced in French and in English.



New Financing

Ressource id*	1 - Ressources externes
Abreviation*	
Libelle francais*	
Libelle anglais*	

Screen 2. New Financing

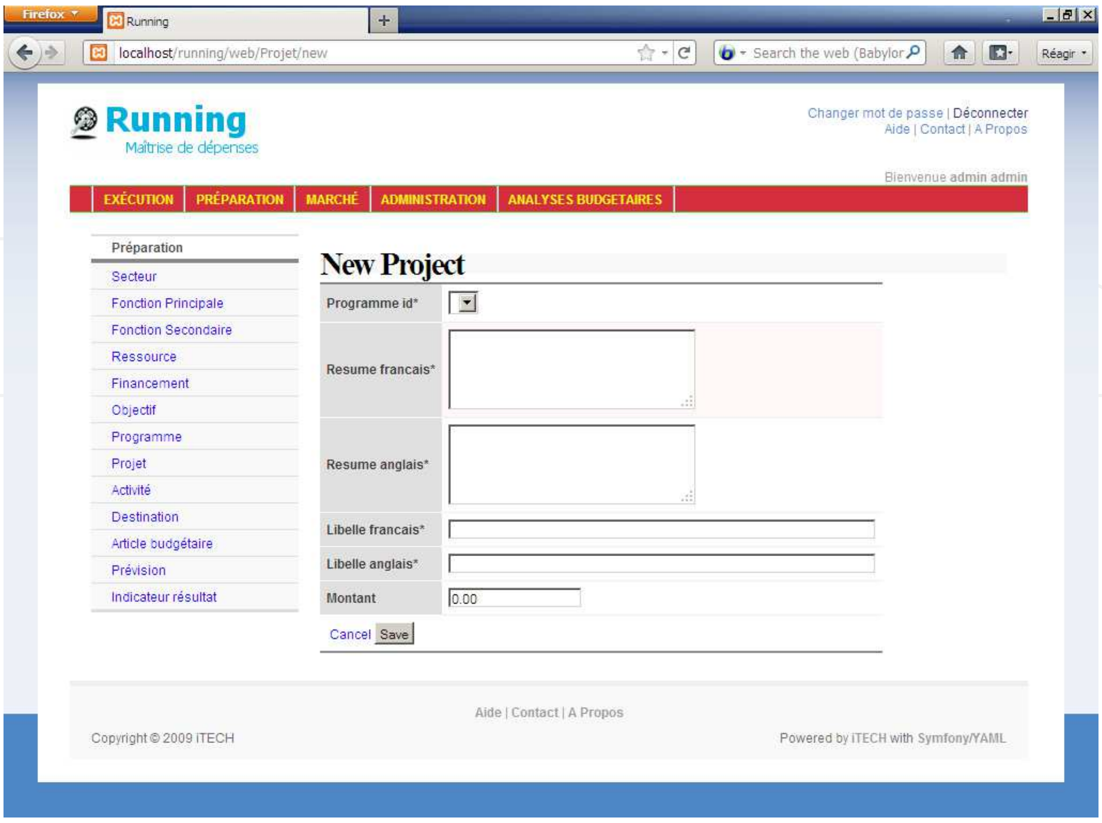
7.2 Results about state expenditure

A project is a one-time effort that produces a specific result. A project is a part of a program. Here are some projects whose translation is studied in example1 and example2:

- Roads construction
- Hospitals building;
- Etc…

Screen3 offers a summary field which contain the project scope. The scope is the description of the boundaries of the project. It defines what the project will deliver. It is the view all stakeholders have from the project, a definition of the limits of the project. In the summary field, requirements must be specified. The scope must be written in French and English.

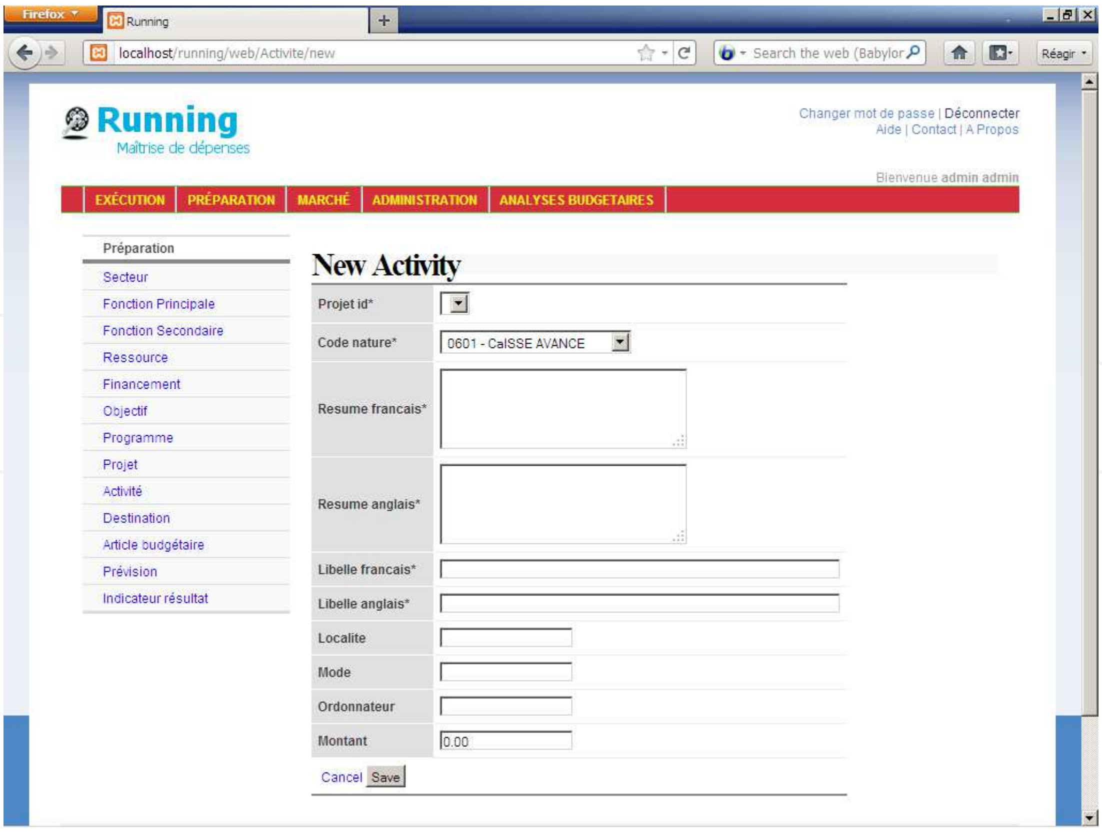
Screen4 allows the description of an activity through the summary field. An activity is a component of a project. It is constituted of sequences and dependencies. At this level durations are determined and the network diagram for the project schedule is developed. The summary field must contain these activity details in French and in English except the network diagram.



New Project

Programme id*	
Resume francais*	
Resume anglais*	
Libelle francais*	
Libelle anglais*	
Montant	0.00

Screen 3. New Project



New Activity

Projet id*	
Code nature*	0601 - CaISSE AVANCE
Resume francais*	
Resume anglais*	
Libelle francais*	
Libelle anglais*	
Localite	
Mode	
Ordonnateur	
Montant	0.00

Screen 4. New activity

8. Conclusion

For each year, the financial law provides for and authorizes all State revenue and expenditure and presents all the programs aimed at attaining economic, social and cultural objectives. The financial law is applied in the Budget Domain which requires a strong knowledge management. In presence of two languages (French and English in our case) bilingual interfaces need hard efforts of language translation.

When organizations are facing language translation issues, here are some mistakes to avoid:

- Asking users to specify details about the translation rather than using a conform idea to record that information
- Allowing service staff to select or influence potential users, for instance, by transferring some works but not others to particular persons
- Not sharing translation results with service staff

In this chapter, we specify a system allowing users to communicate with RUNNING application and to improve their bilingual skills by learning the vocabulary related to State budget management. An ontological system is developed to bring adequate terminology and knowledge to non-expert users in bilingualism. The term ontology corresponds to a formal structure expressed in different models and different degrees. The budgetary ontology represented has three stages;

- Input ontology
- Union of terminological ontologies
- Output ontology

This decomposition is aimed to conceptualize the domain in such a way that data semantics is well described and understood. Reasoning mechanisms are also précised for users' experience.

The public sector area is characterized by a wide range of task and work arrangements. Some processes in financial applications are still limited to simple processes of registering, accounting and calculating. RUNNING application is provided with value-added features facilitating users to be accommodated with budgetary concepts in French and in English simultaneously.

Besides, decision-making in public administration occurs at organizational or policy level but it is also characteristic of its operative work. Thereby, public agents must be able to access information and knowledge to help their tasks, so that decision-making can take advantage of this evolution.

RUNNING is an application in which the user finds material of Computer-Aided Learning to be more bilingual by understanding complex terms, as far as the State budget management is concerned. Web and open source development characters give it the ability to be migrated in other technical domains where the language translation can represent a difficult problem to be solved by users. This is an interesting way of learning words and notions tied to a specific field of activities.

9. Acknowledgements

Authors like to thank Dr. Mathias Ariel Ngnitedem, Nelson Mandela School of Public Policy and Urban Affairs, Southern University and A&M College (Baton Rouge, Louisiana, USA) for his helpful contribution and knowledge of budgetary domain.

10. References

[1] Jerome Fonin. (2002). Logical Framework and Risk Analysis for a National Distance Education (NDE) Project. Cameroon case Study. Proc. First National Seminar on Distance Education- Yaounde, pp 40-46.

[2] Malcolm Holmes and al. (1998). Handbook of public expenditure management. World Bank. Washington. ISBN 0-8213-4752-7.

[3] Kolyang. (2002). Research and Education Quality Improvement through Information and Communication Technologies and Distance Learning. Proc. First National Seminar on National Distance Education- Yaounde, pp 26-32.

[4] Schank Roger C., & Christopher K. Riesbeck, eds. (1981) Inside Computer Understanding, Lawrence Erlbaum Associates, Hillsdale, NJ.

[5] Schank Roger C., Alex Kass, & Christopher K. Riesbeck (1994) Inside Case-Based Explanation, Lawrence Erlbaum Associates, Hillsdale, NJ

[6] Sowa, John F. (1999c) "Ontological categories," in L. Albertazzi, ed., Shapes of Forms: From Gestalt Psychology and Phenomenology to Ontology and Mathematics, Kluwer Academic Publishers, Dordrecht, 1999, pp. 307-340.

[7] Sowa John F. (2000b) "*Ontology, metadata, and semiotics,*" in B. Ganter & G. W. Mineau, eds., Conceptual Structures: Logical, Linguistic, and Computational Issues, LNAI 1867, Springer-Verlag, Berlin, 2000, pp. 55-81.

[8] V. Prasanna Venkatesan (2010). Armms - An Architectural Reference Model for Multilingual Software (Chapter 6). VDM Verlag Publisher.

[9] Francesco Balena (1999). Programming Microsoft Visual Basic 6.0 Microsoft Press

[10] Dr International. (2002). Developing International Software. Microsoft Press, ISBN 0-7356-1583-7.

[11] Andrew Chesterman (1997). Memes of Translation: The Spread of Ideas in Translation Theory. Benjamins Translation Library Series, No. 22.

[12] Tony Buzan, Barry Buzan (1996). The Mind Map Book: How to Use Radiant Thinking to Maximize Your Brain's Untapped Potential. Plume Publisher.

[13] Helen Steward. (1997).The Ontology of Mind: Events, Processes, and States. Clarendon Press-Oxford.

[14] John Heil. (2003). From an Ontological Point of View. Clarendon Press-Oxford. Christiane

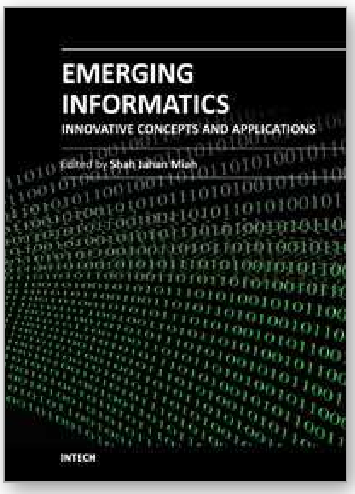
[15] Fellbaum (1998). WordNet: An Electronic Lexical Database. MIT Press.

[16] Eduardo Mena, Arantza Illarramendi (2001). *Ontology-based* query processing for global information system . Business & Economics

[17] Janet Kolodner (1993).Case-based reasoning. Morgan Kaufmann Publishers

[18] Ian Watson (1997). Applying Case-Based Reasoning Techniques for Enterprise Systems. Morgan Kaufman Publishers

[19] Fabien Potencier, Hugo Hamon (2009). Symfony. Eyrolles Collection.



Emerging Informatics - Innovative Concepts and Applications
Edited by Prof. Shah Jahan Miah

ISBN 978-953-51-0514-5
Hard cover, 274 pages
Publisher InTech
Published online 20, April, 2012
Published in print edition April, 2012

The book on emerging informatics brings together the new concepts and applications that will help define and outline problem solving methods and features in designing business and human systems. It covers international aspects of information systems design in which many relevant technologies are introduced for the welfare of human and business systems. This initiative can be viewed as an emergent area of informatics that helps better conceptualise and design new world-class solutions. The book provides four flexible sections that accommodate total of fourteen chapters. The section specifies learning contexts in emerging fields. Each chapter presents a clear basis through the problem conception and its applicable technological solutions. I hope this will help further exploration of knowledge in the informatics discipline.

How to reference
In order to correctly reference this scholarly work, feel free to copy and paste the following:

Guillaume Koum and Innocent Dzoupet (2012). Running: A Mixed Language Software as an e-Learning Solution for the State Budget Management, Emerging Informatics - Innovative Concepts and Applications, Prof. Shah Jahan Miah (Ed.), ISBN: 978-953-51-0514-5, InTech, Available from: http://www.intechopen.com/books/emerging-informatics-innovative-concepts-and-applications/running-a-mixed-language-software-as-an-e-learning-solution-for-the-state-budget-management

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821