THE FIRST TAX RETURN ASSESSMENT EXPERT SYSTEM IN SWITZERLAND

Challenges and Solutions

Marco Bettoni
Fachhochschule beider Basel - 4132 Muttenz, Switzerland
Email: m.bettoni@fhbb.ch

Georges Fuhrer
Steuerverwaltung Baselland - 4410 Liestal, Switzerland
Email: Georges.Fuhrer@fkd.bl.ch

Keywords: knowledge engineering, expert systems, rule-based systems, business rules, Aion, tax return assessment

Abstract: In realising the first Tax Return Assessment Expert System in Switzerland we had first to make a convincing business case for an AI innovation in a traditional governmental environment, secondly to show in the daily business environment that what can be demonstrated to be viable in theory does also work in practice, thirdly to minimize the gap between the *tacit* knowledge in the head of assessment experts and the *explicit* model of expertise that specifies the domain knowledge and finally to minimize the gap between the specified expert knowledge and the system knowledge formalized in the knowledge base. We present the project, our approach to meeting these challenges, the current state of the productive system and a sketch of the final system under development.

1. INTRODUCTION

The Allegro Tax Assistant (ATA) is an advanced, automated tax return assessment rule-based expert system. The system is being developed to increase the assessment capacity of the Internal Revenue Service, quickly put to work changed regulations and gain a better insight into the data, the quality and the knowledge of the assessment process (Bettoni, 2000).

The system loads selected informations constituting a complete case from the existing tax return database. It then decides whether the case complies with regulations or requires further review by a human assessor. If the system accepts the tax return, no further review is required and the assessor can trigger the invoicing process.

An ATA version limited in scope to assess retired individuals without real estate is in operation at the IRS head office of the Ministry of Finance of the Swiss Canton of Baselland since June 2000.

2. MANUAL ASSESSMENT OF A TAX RETURN

Tax return assessment (TRA) is the process of evaluating a tax return to determine whether the declared values comply with tax regulations.

The process (Fig.1) starts with a citizen filling out tax return forms in which items like personal data, employment, income, assets, liabilities and detractions are collected These forms and related documents are then submitted to an IRS office where the complete file of documents receives an identification number and the data are entered in the database. At this point an assessor can begin her work: she requests the complete set of documents of the case, loads the data into the exisiting tax return information system (TRIS), verifies the information against the supporting documents, evaluates if it complies with regulations, enters needed corrections and triggers the process of invoicing the tax payer.

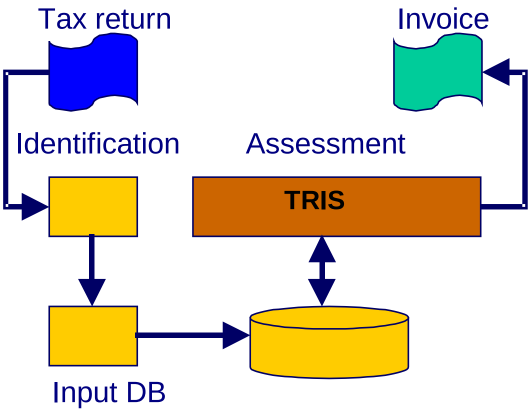

Figure 1: Workflow of manual assessment

3. THE PROBLEM

The Internal Revenue Service of the Ministry of Finance of the swiss confederation state ('canton') of 'Baselland' has 70 assessors distributed in ca. 50 branches. Due to a change in the assessment period which will be reduced from 2 to 1 year (by 2001) the IRS expects an increase in assessment workload of about 30%. Because of functional limits of the currently used tax return information system and due to a strict stop in recruiting more assessors, the IRS began looking at the possibility of creating a new system that would automate the assessment process.

4. ROAD TO AI AT IRS

The idea to automate the assessment of tax returns by means of an expert system was conceived by the second author of this paper since 1997.

As a result of an investigation over the quality of the assessment task a set of error sources had been identified, like interruptions due to phone calls and colleagues, tedious sameness in checking routine cases, time pressure.

The need of minimizing these errors led to the search for a software system that would do the boring figures checking.

During this search, contacts where established with a sales representative of the Aion development environment for expert systems (CA, 2000): without knowing what an expert system is, the second author of this paper recognized that the potential for a rule-based expert system built with Aion to do the boring figures crunching and even to completely assess tax returns was promising: he thus became the main promoter of an AI approach within IRS.

However, no known rule-based system had ever been deployed in Switzerland for tax return assessment and no AI application existed within the Ministry of Finance. Fortunately during an 'open doors' event at FHBB (the Basel State University of Applied Sciences) the needed competence for developing a knowledge-based expert system was found to be available within the same administration.

5. OBJECTIVES AND REQUIREMENTS

In a first step toward setting up the 'Allegro' project, a list of key objectives was compiled. The primary ojective was to assess an increased volume of returns/year without needing to hire additional staff and within the limits of the existing budget. Assessor should also be freed from the need to process simple, tedious cases and have more time to spend on difficult ones.

Further the quality of the assessment had to remain at least at the same level or to become better. As assessment guidelines and tax regulations change constantly, another important objective was to quickly put to work in the new system any changes of that kind.

Finally the total investment in external consulting had to be less than $500,000 and the system had to be released before December 31, 2001.

When building the system the developers would need to keep in mind several ambitious requirements. First, the main task and scope of the system was to automatically assess employed & unemployed individuals (State & Confederation taxes). If the case was completely correct, the system would approve it and trigger the invoicing procedure. If the system would refer the case to an assessor, the system should be specific about the key reasons that caused referral, allowing the assessor to concentrate on the error areas.

Another key requirement was that maintenance would have to be done by assessors both in order to accomodate changing assessment guidelines and regulations and in order to fine tune the system so that it would automatically process a larger percentage of cases.

Further the system had to be integrated with the existing tax return information system and had to be able to operate on its database. Batch processing on a server was selected as the most appropriate system run mode with a perspective on a later evolution to an on-line availability at client level.

6. CHALLENGES

Such an ambitious project faced the project team with a set of 4 main challenges. Firstly we had to make a convincing business case for an AI innovation in a traditional governmental environment.

Secondly we had to show in the daily business environment that what can be demonstrated to be viable in theory does also work in practice.

Thirdly we had to minimize the gap between the *tacit* knowledge in the head of assessment experts and the *explicit* model of expertise that specifies the domain knowledge.

Finally we had to minimize the gap between the specified expert knowledge and the system knowledge formalized in the knowledge base.

6.1 Promising Technology

Although the project of automating tax return assessment was widely supported in upper management, the technology of rule-based expert systems had been recognized as suitable only by the previously mentioned promoter.

Thus the first challange was convincing the IRS upper management that the potential of rule-based technology for automating the assessment of tax returns was promising and that a rule-based expert system would meet the requirements much better than a conventional 'flow-based' (procedural) system.

To this aim we organized an 'Expert System Technology Event' at IRS: in September 1998 about 30 members of middle and upper management participated in a set of presentations by FHBB and CA with the goal of increasing their understanding of the technology's potential.

The result was, that the 'Allegro' project was definitely approved, FHBB was appointed as consultant for developing the system and Aion selected as the development environment (December 1998).

6.2 Prototype in Production

Even if they now understood better the technology, most persons in upper management were still (understandably) unsure of deploying a new, untried technology for the core process of tax return assessment.

Thus, the second challange was to convince the IRS management that what we had demonstrated in theory at the mentioned technology event would not only be feasible as a research prototype but also work in practice, in the real daily environment, with real users, their systems and their real data.

For this reason, when we started system development (knowledge engineering) in January 1999 under the acronym ATA (Allegro Tax Assistant), our main milestones became:

a) first to develop an ATA-prototype limited in scope to a simple but still representative subset (retired individuals without real estate) of the full range of tax returns;

b) secondly to integrate this prototype with the existing database;

c) thirdly to adapt the tax return information system (TRIS) by including access to and visualisation of the ATA output so that the assessor could profit by the results of automatic assessment;

d) finally to release both the new ATA system and the adapted TRIS application to the main IRS branch.

In June 2000 the first ATA release with 200 expert rules began to be used in the real daily assessment environment. After this success FHBB and IRS signed an agreement for regulating the exploitation of the project results in other swiss cantons (August 2000).

6.3 Engineering the Expertise

The third challange was conceiving and performing the *knowledge engineering* process in a way, which would minimize the gap between the *tacit knowledge* (problem solving faculty, i.e. 'expertise' in the head of assessors) that should be or actually is applied in the manual assessment process on one side and the *explicit knowledge* (so called 'model of expertise' (Schreiber et al., 1993) that is used as specification for constructing the knowledge-based system on the other side.

Our approach (Bettoni 2001) is based on constructivist theories of knowledge that support viewing knowledge engineering as a constructive modeling activity (Ford et al., 1993; von Glasersfeld, 1996; Schreiber et. al., 1993; Bettoni, 1997). This modeling perspective implies in our approach the need for active cooperative participation by both domain experts (DE) and software engineers (SE) in the process of knowledge-based system (KBS) construction.

In engineering the ATA system we have distinguished two sub-processes:

a) Engineering the expertise: building a knowledge-based *conceptual* model of the tax return assessment knowledge (expertise, competence)

b) Engineering the system knowledge: building a knowledge-based *executable* model, i.e. a software system that applies tax return assessment knowledge (rule-based expert system).

Process (a) is usually called 'knowledge acquisition' and process (b) is mostly referred to as 'design and implementation'.

Traditionally both activities were mostly performed by 'knowledge engineers' who are basically software engineers with a specialization in building knowledge-based systems.

In engineering expertise the problem for the software engineers (who should observe and interview 'experts') is that they have no experince in the specific domain: so they begin with learning about the principles and guidelines of the domain, try to become familiar with the terminology etc. But even after a long period of (theoretical) study software engineers will still be at the level of a 'beginner'.

In engineering ATA we have *no person who works exclusively in the role of knowledge engineer*: the tasks of knowledge engineering are distributed among two differently focused knowledge engineers.

An experienced domain expert (the second author of this paper) has the responsibility of developing the model of expertise: he is not a software engineer but a professional of the tax domain with a univeristy degree in business administration.

He coordinates the work of other experts who describe their problem solving behavior and knowledge, he analyses the needed or applied knowledge and develops a first version of the expertise model.

This model receives then a tight feedback from a software engineer experienced in engineering system knowledge (the first author of this paper).

In this way the conceptual model can first be created as free as possible form technological influences and as near as possible to the original *tacit knowledge* (Polanyi, 1966; Nonaka and Takeuchi, 1995); but then the model is adapted to be compatible with the restrictions and constraints that unavoidably come from technology.

We think that firstly this way of co-evolution between expertise model and system model, secondly the distribution of knowledge engineering over two differently focused professionals (i.e. having also the domain expert work as knowledge engineer), thirdly a well defined cooperation between the two different knowledge engineers and last but not least our basis in constructivist theories of knowledge have been the four main keys to successfully engineering the expertise of tax return assessment.

6.4 Engineering the Knowledge Base

Finally the fourth main challange was formalizing (conceptualizing, structuring and implementing) the explicit expert knowledge by means of software structures in a way that would minimize the gap between the given *conceptual* model of expertise and the resulting *executable* system knowledge.

Our approach is strongly influenced by the KEE (IntelliCorp, 1991) and Lisp (Steele, 1990) paradigms although we use products like Aion which have left behind them the world of Lisp and moved to the Object-Oriented (OO) camp, where the industry is.

One fundamental problem with the industry OO approach compared with KEE regards the treatment of *business rules* (Wilson, 1998). OO methodologists treat business rules as constraints, as modeling elements, as procedural code, as values in business object variables or as full objects but *never* combine business rules with a treatment of inferencing.

Aion on the other side has succeeded in integrating inferencing into a pure OO model (Garone and Buck, 2000) but has added rules as a *new*, discrete programming construct (a rule method), instead of using one of the exisiting standard OO elements, like for instance objects.

KEE on the contrary provided both rules as objects (missing in Aion) and inferencing (missing in standard, industry-wide OO paradigms): from our experience in developing with KEE rule-based expert systems like the PROFI elevator configurator (Businger, 1993) and the MASTER Simscript code generator (Bettoni & Bernhard, 1994) we are convinced that this approach was one of KEE's main strengths because it allowed the knowledge engineer to treat *rule logic* and *fact logic* in a more consistent and coherent way.

Inspired from the advantages offered by the KEE solution we have conceived rules as objects (classes) at the level of our design model and have correspondently implemented in Aion for each rule a specific rule-object collecting not only one or more rule-methods for specifying the rule's logic (connected facts) but also attributes like tolerances to be used when evaluating a rule formula or procedures for computing deviations from a rule specific reference value.

This makes the logic of the ATA knowledge base consistent, coherent and transparent.

Moreover, assessors tend to think of assesment regulations and checking procedures in terms of attributes and behaviors collected in a unit and associated with a 'position', i.e. a specific entry field in the tax return form.

From this basic observation we concluded that a *rule-as-object* approach would also give us the best foundation for making both the model of expertise and the executable system knowledge come as closest as possible to each other as well as to the tacit assesment knowledge of tax return assessors.

7. PRODUCTIVE AND DEVELOPMENT SYSTEM

There are at present (April 2001) two versions of the ATA system: a deployed version released in June 2000 ('productive system') with about 200 expert rules and the system under development with about 700 expert rules. Both systems are implemented with Aion 8.1 and run as C++ compiled executables on Windows NT servers.

The main difference between the two systems lies in their architecture (Fig.2): the productive system has a kernel which includes into one library the complete knowledge base (facts and rules).

For the much larger development system, which is expected to implement about 800 expert rules by its release in June 2001, we have introduced a modular architecture of the knowledge base where all the facts are collected into one library and the rules distributed over a variable number of modules (roughly 1 module for 100 rule objects).

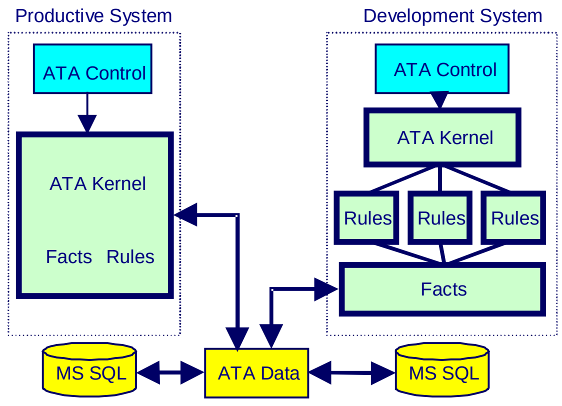

Figure 2: Architecture of 2 ATA versions

The introduction of ATA in the core process of tax return assessment has left nearly unchanged the workflow for the assessors.

This simplicity is a major characteristic of our integration design. As shown in Fig. 3, the ATA system reads and writes to the same database that was and is still used in manual assessment (compare with Fig.1).

What is new for the assessor is, that she can now select cases which have already been automatically assessed.

When data from such cases are loaded into the tax return information system (TRIS), the assessor sees a list of detected errors indicating clearly which field in the tax return is affected. From this list she can directly go to the error areas or, if no error is listed trigger the invoicing procedure without even looking at the tax return details.

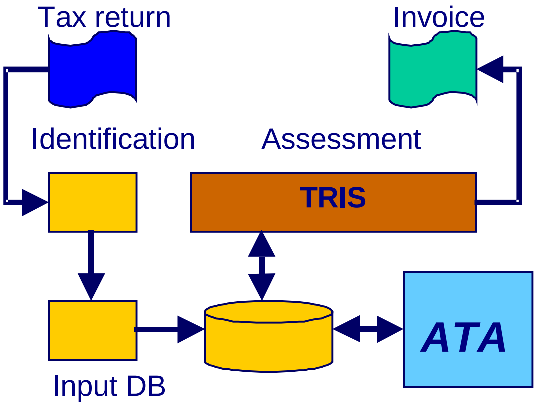

Figure 3: Workflow with ATA

The complete ATA system will be deployed in January 2002. From that moment all tax returns of employed an unemployed individuals will be processed automatically. In the introductory phase a special attention will be devoted to *user* acceptance and correct use. After that, increasing the *assessment* acceptance rate of tax returns handled by ATA will be one major goal.

8. DISCUSSION

At a first, rapid sight, the ideas presented here may appear only weakly new. In fact, the challenges mentioned in this paper are of course well know, nearly every ES project has to face these or similar challenges.

The newness of our work lies primarily in the underlying 'CoKE' methodology (Bettoni 1990, 1995, 1997, 2000, 2001) that has been the basis for successfully developing and implementing the ATA system.

First, it was thank to CoKE that we were able to compare the 'flow based' with the 'rule based' technology in a way that led the IRS management to recognise an ES approach as promising (see 6.1).

Secondly, it was mainly due to CoKE if we were able to design and implement a strategy and a system which demonstrated to the IRS management that what was promising in theory did also work in practice (see 6.2).

Thirdly, it was by applying CoKE that we were able to engineer the expertise of Tax Return Assessment in a way that has minimized the gap between the *tacit knowledge* that should be or actually is applied in the manual assessment process on one side and the *explicit knowledge* that is used as specification for constructing the knowledge-based system on the other side.

Finally it was thank to CoKE that we were able to formalize the explicit expert knowledge by means of software structures in a way that has minimized the gap between the given *conceptual* model of expertise and the resulting *executable* system knowledge.

The basic principle which distinguish CoKE from other methodologies considers tacit knowledge both as a dynamic result of acting and as dynamic feedback, which means that what is feeded back becomes a functional component of the mechanism which generated it (Fig.4). This model is inspired by the pioneering work of Humberto Maturana (Maturana and Varela, 1980).

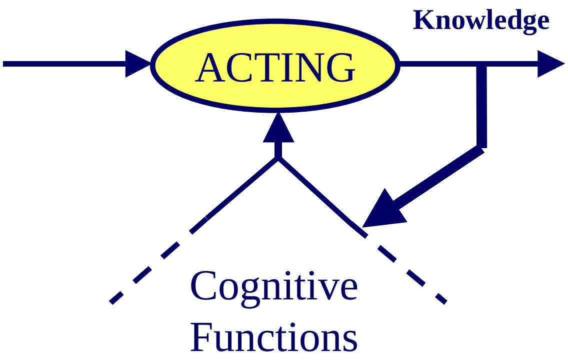

Figure 4: Autopoietic Model of Knowledge

This paper is primarily a presentation of the results of applying our methodology, which itself appears only indirectly within the description of our solutions. CoKE will be presented next year at a coming AI event, hopefully for instance at the next ICEIS.

9. REFERENCES

Bettoni, M., 1990, "Cognition, Semantics and Computers", in: R.A. Zwaan, D.Meutsch (eds.) Computer Models and Technology in Media Research, P.65-98, Elsevier Science Publ., Amsterdam, 1990.

Bettoni, M. and Bernhard, W., 1994, 'MASTER: A Knowledge-based Assistant for Speeding Up Simulation Projects', in: J. Liebowitz (ed.), *Moving Towards Expert Systems Globally in the 21st Century*, CD, MacMillan New Media, Cambridge.

Bettoni, M., 1995, 'Kant and the Software Crisis: Suggestions for the construction of human-centred software systems', AI & Society **9**, p. 396-401

Bettoni, M., 1997, 'Constructivist Foundations of Modeling. A Kantian Perspective', *International Journal of Intelligent Systems*, Vol.12, Nr.8, S. 577-595.

Bettoni, M., 2000, ' Meeting the Challenge of Knowledge-Intensive Automation', Presentation at *CAWorld 2000, Neugents Session*, April 10, New Orleans. URL: http://www.caworld.com/proceedings/2000/neugents/

Bettoni, M., 2001. 'CoKE: A Constructivist Knowledge Engineering Methodology', Internal Report Nr. WM0101, CZM, January 2001.

Businger, A., 1993, *Expertensysteme für die Konfigura-tion: Architektur und Implementierung.*, PhD Dissertation, University of Zürich, Switzerland.

CA 2000, Aion Development Environment, Dec. 2000, www.ca.com/products/aion/aion_pd.htm

Ford K.M., et.al. , 1993, 'Knowledge Acquisition as a Constructive Modeling Activity,' *International Journal of Intelligent Systems*, **8**, 9-32.

Garone, S. and Buck N., 2000, 'Capturing, Reusing and Applying Knowledge for Competitive Advantage. IDC White Paper.

KEE, 1991, Technical Manuals, Vols. 1-4, Mountain View: IntelliCorp Inc.

Maturana, H. and Varela, F., 1980, *Autopoiesis and Cognition*, Reidel, Dordrecht/Boston.

Nonaka, I. and Takeuchi, H., 1995, *The Knowledge Creating Company*, Oxford University Press, New York.

Polanyi, M., 1966, *The Tacit Dimension*, Gloucester, Peter Smith, (reprint 1983).

Schreiber, G. et al., 1993, *KADS. A Principled Approach to Knowledge-Based System Development*, London, Academic Press.

Steele, G.L. jr., 1990, Common Lisp. The Language. Digital Press, DEC.

Talebzadeh, H. et al., 1995, 'Countrywide Loan-Underwriting Expert System', AI Magazine, Spring.

von Glasersfeld, E., 1996, *Radical Constructivism. A Way of Knowing and Learning*, Falmer, London.

Wilson, K., 1998, 'Implementing Business Rules via Inferencing in Objects: The Aion Solution', www.ca.com/products/aion/aion_pd.htm