



# AIONWORLD DEVELOPERS CONFERENCE

Rule Your Enterprise

February 21-25, 1999  
Orlando, Florida

**PLATINUM**  
TECHNOLOGY



# Implementing Knowledge- Intensive Applications:

With Rule-Based  
Technology or with  
Conventional  
Computing?

*Marco C. Bettoni*

*Basel Institute of  
Technology and  
Management*



# Navigator

---

- Case Study: Tax Return Assessment
- Assessment Task
- With Conventional Computing
- Evolution of SW-Technologies
- With Rule-Based Technology



# Tax Return Assessment: Problem

---

- Canton of „Baselland“ (BL)
- Revenue Service
- Period: reduce from 2- to 1-year return
- Employed & unemployed persons
- 140‘000 returns/period
- Volume/period: increase ca. 30%
- Staff: no recruiting
- Quality: same or better



# Tax Return Assessment: System Requirements

---

- Assess *routine* tax returns
- Approve or refer returns
- Show reasons that caused referral
- Quality of approve decisions
- Justify decisions
- Maintenance by assessors
- Reuse: self-employed persons, etc.
- 70 users in ca. 50 locations



# Tax Return Assessment: Client's Questions

---

- Increase automation degree: possible ?
- Extend existing system ?
- Why are expert systems (ES) the most appropriate technology ?
- When are ES appropriate ?
- Benefits for the Revenue Division ?
- Success stories ?



# Automation: Trends

---

- Improve core (strategic) operations
  - Volumes of core operations: ↑
  - Responsiveness to changes: ↑
  - Front- and back-office involved
  - Productivity of skilled staff: ↑
  - Promote less-skilled employees
- ➡ *knowledge-intensive automation (KIA applications)*



# The Knowledge Society

---

- *Knowledge-intensive* tasks and products
- Company-specific processes
- Shorter innovation cycles
- Decreasing decay rate of knowledge
- Higher quality requirements
- Increasing employee qualifications
- Multimedia technologies (Internet)





# Evolution of IS-Applications

---

- Data processing systems
  - enter, access, display data
- Management information systems
  - access and display informations
- *Knowledge-intensive* applications
  - apply knowledge
- support → execution
- recording → assisting & improving



# *Knowledge-intensive* Tasks

---

- Logic: complex, dynamic, holistic
- Context: core processes
- Dependencies & connections: many
- Changes: frequent & rapid
- R&D staff, R&D budget
- Employee's education level
- Customer specific products
- Products bundled with services



# *Knowledge-intensive Assessment*

---

- "You can't do it differently !"
- expensive, difficult, time consuming
- skilled workers
- Tax regulations (state, confederation)
- Manual, Guide, Forms
- Many key concepts
- Strong interconnection



# Automation: How ?

---

- How to master complexity ?
  - How to cope with dynamics ?
  - Learn from nature
  - How does it the human mind ?
  - 'The knowledge principle' (E. Feigenbaum)
  - Focus on knowledge and its organisation
- ➡ *Method: Knowledge Modeling*



# Knowledge Processing by Conventional SW: *Properties*

---

- "*Flow-based*" systems
- Knowledge cast into procedures
- ✗ Knowledge & Program: dependency
- ✗ Knowledge modeling: complication
- ✗ Knowledge formalisms: cryptic
- ✗ Knowledge organisation: dispersed
- ✗ Knowledge redundant, inconsistent
- ➔ *Knowledge & Program: interwoven*



# Knowledge Processing by Conventional SW: *Problems*

---

- ✗ Expert & developer: communication gap
- ✗ Specification & model: organisation gap
- ✗ Specification & code: language gap
- ✗ Find knowledge
- ✗ Extend knowledge
- ✗ Maintain knowledge
- ✗ Reuse knowledge



# Evolution of SW Technologies

---

- Isolation: Database concept, GUI
  - Deduction: Post (1943), Markov (1956), Feigenbaum (1964)
  - Emulation: OOP, Lisp, Jackson method
  - Abstraction: Compiler concept
- ➔ *human-oriented software (except relational DB)*



# Expert System: Concept

---

- a software system where
- practical knowledge (experiential)
- from an isolated special domain
- is formalized, managed and applied
- with the goal of obtaining results
- which can support an expert in *knowledge-intensive* routine tasks





# Experiential Knowledge

---

- **Facts, cases**  
(*know-that*)
  - **Consequences**  
(*know-then*)
  - **Context**  
(*know-why*)
  - **Sequences**  
(*know-next*)
- Talk will be given on Monday, Feb. 21, 1999
  - As the talk begins at 3:45 pm, ...
  - Talk: explore, conceive, prepare slides
  - Search address, operate PC, verify data



# Knowledge-Based Expert Systems: Concept

---

- I Separation of knowledge and program  
(*Isolation, unwinding*)
- D Dynamic generation of solution paths  
(*Deduction, inference*)
- E Adaptation of knowledge organisation  
to human thinking (*Emulation*)
- A Adaptation of knowledge formalisms to  
human language (*Abstraction*)



# Rule-Based Expert Systems: Modeling

---

- with concepts
  - with objects (facts, frames)
  - with methods (procedures)
  - with hierarchies
  - with relations
  - with rules
- ➡ *Object-oriented & rule-based ES*



# Rule-Based Expert Systems: Best Technology

---

- Knowledge elements in tax assessment: laws, regulations, procedures, manual, guide, court decisions, etc.
  - Modeling means with rule-based ES: concepts, facts, rules, classes, inference
- ➡ *best correspondence between knowledge elements and modeling means*
- ➡ *successful similar application*



# Work of Domain Experts with ES

---

- Develop knowledge model
- Extend knowledge model
- Improve knowledge model
- Delegate routine cases
- Difficult tasks: more time
- Assuring quality: more time
- Training new employees
- More knowledge sharing



# Expert Systems: Benefits

---

- Quicker reactions to changes
- Knowledge maintenance by users
- Knowledge reuse (knowledge model)
- Easier to extend
- Effective participation of domain experts
- Knowledge sharing and communication
- Useful for quality assurance
- Useful for training



# Expert Systems Application: Selection Criteria (1)

---

- Task *knowledge-intensive* but routine
- Business rules, business policies
- Manually solved in *1 to some* hours
- Expert can describe her knowledge
- Recognised domain experts available
- Different domain experts agree
- Task not too new, not too small, not too big, can be divided into subtasks



# Expert System Application: Selection Criteria (2)

---

- Used in core processes
- Frequent maintenance of knowledge
- Shortage of domain experts, delegate
- Domain expert is on leave
- Want to ease access to knowledge
- Want to improve quality
- Multiple use possible
- Similar successful systems exist





# Expert Systems: Limits

---

- No *context-intensive* tasks
- Rather simple inference methods
- No induction, no analogy
- Maintenance by experts: demanding
- Rather simple explanations
- Expert knowledge & model: gap
- Learning: only in research prototypes



# Expert Systems: Risks

---

- Correct evaluation of selection criteria
- Modeling: more an art than a technique
- Shortage of experienced developers
- Consistency check: no general methods
- Availability of experts
- Integration
- Overall benefits



# Messages

---

- Tax return assessment is a *knowledge-intensive* task which should be automated by means of *knowledge modeling*
- Conventional software displays important weaknesses in knowledge modeling
- *Rule-based expert systems* are a better technology for *knowledge-intensive* applications.



T

---

Implementing  
Knowledge-Intensive  
Applications:

With Rule-Based Technology  
or with Conventional  
Computing ?

*Marco C. Bettoni*

*Basel Institute of Technology and  
Management*



**AIONWORLD  
DEVELOPERS  
CONFERENCE**

**Rule Your Enterprise**

February 21-25, 1999  
Orlando, Florida

**PLATINUM**  
TECHNOLOGY