Lessons Learned From Ontology Design

Jean André BENVENUTI*, Laure BERTI-ÉQUILLE**, Éric JACOPIN*

* CREC Saint-Cyr, laboratoire MACCLIA -- GUER, France
** IRISA, Université de Rennes 1 -- RENNES, France
THE SOLDIER’S CODE - 11 Articles - French Land Army, 1999

1) The soldier is entirely devoted to serving France anywhere and anytime.
2) The soldier achieves his mission with the will to win and vanquish, possibly at the risk of his life.

4) The soldier obeys orders with respect of the laws of war and international conventions.

5) The soldier improvises and adapts in all circumstances.
6) As a professional, the soldier keeps fit, physically and intellectually and develops his abilities and his moral strength.
7) Member of a brotherhood of arm, the soldier acts with honor, frankness and loyalty.

5) The soldier improvises and adapts in all circumstances.

of armies in philosophical, political and religious matters.
11) Proud of his commitment, the soldier represents, always and everywhere, his garrison, the land army and France.
The SOLDIER ’S CODE sounds marvelous, but how are we going to teach it?

1) Ensure that the teaching process is the same for every soldier

Objective

The soldier has the appropriate behaviour when facing a practical setting

2) => A Database of Concrete Cases

3) Nicely link the real-life situations to the soldier ’s code

4) Automate the learning process

Why?

=> Distant education (operational context)
=> (Anytime) Self-training and self-evaluation
=> Saving time and money…
Outline

- Description of the Tutoring Process
- Ontology in the Tutoring Process (SABRE)
- Managing Ambiguities (Eigenvalues)
- Discussion (Reuse?)
The Tutoring Process

1) Picks the article to learn
2) … and one corresponding behaviour
3) Chooses an appropriate concrete case

Must find the appropriate behaviour

Teacher

Learner

Concrete Case

Green Book

Practical Setting

International & French Legal Texts

Theoretical Training

Feed-back Experience

Instructor's guide

Select Behaviour

Appropriate Behaviour

Operational or garrison events
Extract of our ontology

1) Let’s pick Art. 4
2) ... and one corresponding **behaviour**?

**behaviours**

- b09: courage
- b13: sense of responsibilities
- b14: dignity
- b16: self-control
- b17: respect for the law
- b18: respect for the regulations
- b19: obedience
- b20: sense of initiative
- b24: competence
- b25: have-judgment

**Achieving the mission**

- Improvising
- Keeping Fit
- Obeying orders
- Force-control

**Teacher**
1) Let’s pick **Art. 4**
2) ... and one corresponding behaviour? Is there one **behaviour** only corresponding to article **Art. 4**?

**behaviours**

- **b09**: courage
- **b13**: sense of responsibilities
- **b14**: dignity
- **b16**: self-control
- **b17**: respect for the law
- **b18**: respect for the regulations
- **b19**: obedience
- **b20**: sense of initiative
- **b24**: competence
- **b25**: have-judgment

**Ontology with Eigenvalues**

**Improvising**
- **Art.5**
  - b25
  - b24

**Keeping Fit**
- **Art.6**
  - b19
  - b16

**Obeying orders**
- **Art.4**
  - b09
  - b20

**Force-control**
- **Art.2**
  - b19
  - b16

**Achieving the mission**

1) Let’s pick **Art. 4**
2) ... and one corresponding behaviour? Is there one **behaviour** only corresponding to article **Art. 4**?
1) Let’s pick Art. 4
2) ... and one corresponding behaviour?
   • behaviour 19 is the eigenvalue of Art. 4
Other behaviours entails Art. 4 as their context

behaviours

b09: courage
b13: sense of responsibilities
b14: dignity
b16: self-control
b17: respect for the law
b18: respect for the regulations
b19: obedience
b20: sense of initiative
b24: competence
b25: have-judgment
Definitions

**Eigenvalue**
- Value of a slot of which uniquely represents an instance of a class

**Eigenslot**
- Slot of an instance of a class with at least one Eigenvalue

**Eigeninstance**
- Instance of a class with at least one Eigenslot

**Eigenclass**
- Art. 5: All instances of that class are Eigeninstances

Examples

- b19: obedience
  - uniquely represents an article

- Art. 5
  - behaviour uniquely represents an article
Reusing eigenvalues?

• **Keywords** for Papers / Talks / Posters (classification)

• **Instances** for classes
  (object-oriented design - **singleton** Design Pattern)

• **Flights** for Airlines / **Theaters** for Films
  (cost reduction: e.g. merging Air-France & KLM)
  (market flooding - university film archives)

• **Logos** for Trademarks / **Flags** for Countries / **Symbols** for Cities (identification - recognition)

• **Wines** for courses
  (best association of objects)
Lessons as questions?

Thank You!

• When must we write rules?
  Good old „Keep it simple” (not simplistic :-) 

• Can ’t we rewrite our classes?
  Good old code review, object-oriented refactoring, reverse engineering…

• Can we use the instances to evaluate the ontology?
  Good old execution tests: run tests to know about the software design