

## OKBC A Protocol For Knowledge Base Interoperation

Adam Farquhar  
Richard Fikes  
James Rice  
Vinay Chaudhri  
Peter Karp

### Exchanging Knowledge - Problems

- Knowledge Representation Systems (KRS) vary:
  - » Different levels of expressiveness
  - » Different degrees of reification
  - » Different inference capabilities
- Knowledge represented in one KRS has not been usable in another
- Tools that work for one KRS will not work for others
- Effort is wasted!

## Exchanging Knowledge - Approaches

- A Knowledge Interchange Format
  - » KIF specifies a declarative method for exchange
  - » KIF does not address the procedural aspects
    - Open a KB, Save a KB, Create a class, Delete a class, ...
- An open API
  - » OKBC specifies a protocol for KRS interoperation
  - » OKBC supports a client-server model for interaction
  - » OKBC provides transparent network access
  - » OKBC provides an object-oriented view of a KRS

## KRS Variation - Expressiveness

- |  |  |
|--|--|
| <ul style="list-style-type: none"><li>● Ocelot<ul style="list-style-type: none"><li>» A simple frame system with classes, individuals, slots, a limited number of facets</li></ul></li><li>● CLIPS<ul style="list-style-type: none"><li>» An object system with production rules</li></ul></li><li>● Classic<ul style="list-style-type: none"><li>» An elegant classifier system with a substantial set of constraints (e.g., value-type, cardinality)</li></ul></li></ul> | <ul style="list-style-type: none"><li>● Loom<ul style="list-style-type: none"><li>» An expressive classifier system with a full assertion language</li></ul></li><li>● ATP<ul style="list-style-type: none"><li>» A full FOL theorem prover with limited axiom schema</li></ul></li><li>● Ontolingua<ul style="list-style-type: none"><li>» A full KIF representation system</li></ul></li></ul> |
|--|--|

## Knowledge Model

---

- Universe of discourse
- Partition into
  - » Classes
  - » Individuals

Classes	Individuals
Relation	Value-type
Thing	42
Number	Emp27    Pi
Employee	Bob
String	Age-of    4.27
{x x<10}	Inverse-of
Class	"hello"
{1 2 3}	Subclass-of

## Knowledge Model

---

- Relationships
  - » Slots
  - » Facets

Classes	Individuals
Relation	Value-type
Thing	42
Number	Emp27    Pi
Employee	Bob
String	Age-of    4.27
{x x<10}	Inverse-of
Class	"hello"
{1 2 3}	Subclass-of

## KRS Design

- KRS distinguish a set of frames
  - » Objects about which assertions may be made
- What constitutes a frame varies widely!
- Restrictions are arbitrary

Classes		Individuals	
Relation	Facets	Value-type	42
Thing			Pi
Number	Employee	Emp27	4.27
String	Person	Bob	"hello"
{x x<10}	Frames	Age-of	
{1 2 3}	Class	Inverse-of	
	Slots	Subclass-of	

## Dealing with Variation

- What should these OKBC operations return?
  - » Get-kb-frames
  - » Get-kb-classes
  - » Get-kb-individuals
  - » Get-kb-slots
  - » Get-kb-facets
- What guarantees can OKBC provide?
  - » Frames \_ classes
  - » Frames \_ slots
- Under what conditions could guarantees be stronger?

## Dealing with Variation - Names

- KRS use different names for concepts
  - » The most general class: thing, object, all, any, ...
  - » Value restriction: value-type, slot-value-type, type, ...
- Applications must portably refer to these common objects
- OKBC defines standard names
  - » :thing, :value-type, :inverse, :cardinality, ...
  - » If `coerce-to-frame(:thing)` returns a frame, then it must have the correct meaning
  - » A KRS need not provide any standard names

## Dealing with Variation - Inference

- |  |  |
|--|--|
| <ul style="list-style-type: none"> <li>● KRS vary widely in inference ability           <ul style="list-style-type: none"> <li>» Simple lookup</li> <li>» Taxonomic reasoning</li> <li>» Full FOL theorem proving</li> <li>» Forward chaining</li> </ul> </li> <li>● Control of inference           <ul style="list-style-type: none"> <li>» Specify inference level: direct, taxonomic, all-inferable</li> <li>» KRSs must return <i>at least</i> those values</li> </ul> </li> </ul> | <ul style="list-style-type: none"> <li>● Understanding results           <ul style="list-style-type: none"> <li>» Operations return additional information</li> <li>» If the answer is complete               <ul style="list-style-type: none"> <li>– All Universities</li> <li>– All Universities I had time to find</li> <li>– All Universities I can find</li> </ul> </li> <li>» If exactly the requested inferences were performed</li> </ul> </li> </ul> |
|--|--|

## Dealing with Variation - Types

- Procedural tests allow clients to be more robust
  - » If age-of is a slot on human, then slot-p(age-of) returns True
    - Facets on age-of for human can be asserted
  - » If frame-p(age-of) returns True, then age-of is also a frame
    - Properties of age-of can be asserted
- OKBC cannot legislate that all slots are frames