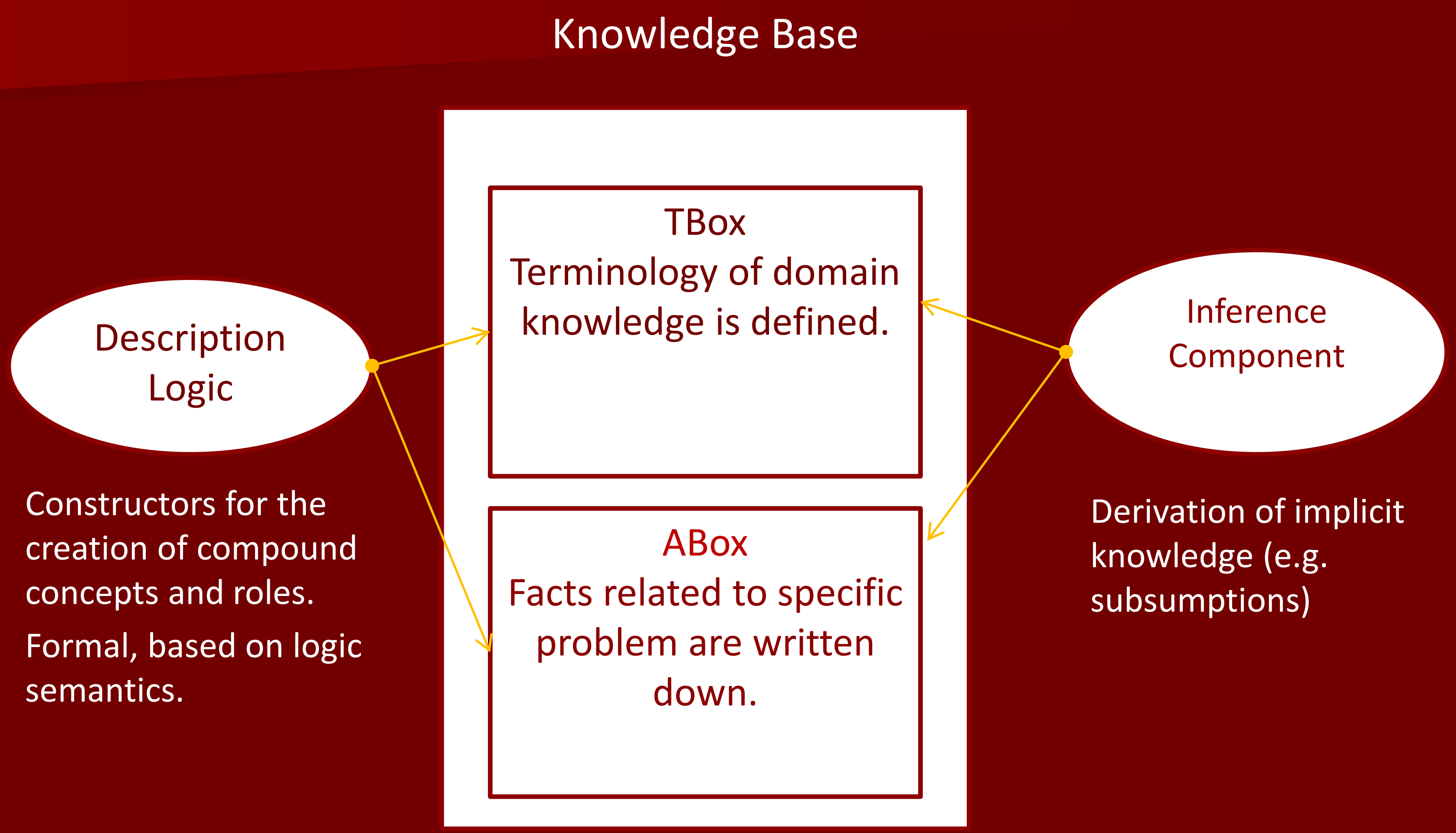
CEID

MSc on DATA DRIVEN COMPUTING AND DECISION MAKING (DDCDM)

Ontology Reasoners

I. Hatzilygeroudis-Isid. Perikos

Description Logic Systems



Knowledge Base

TBox
Terminology of domain knowledge is defined.

ABox
Facts related to specific problem are written down.

Description Logic

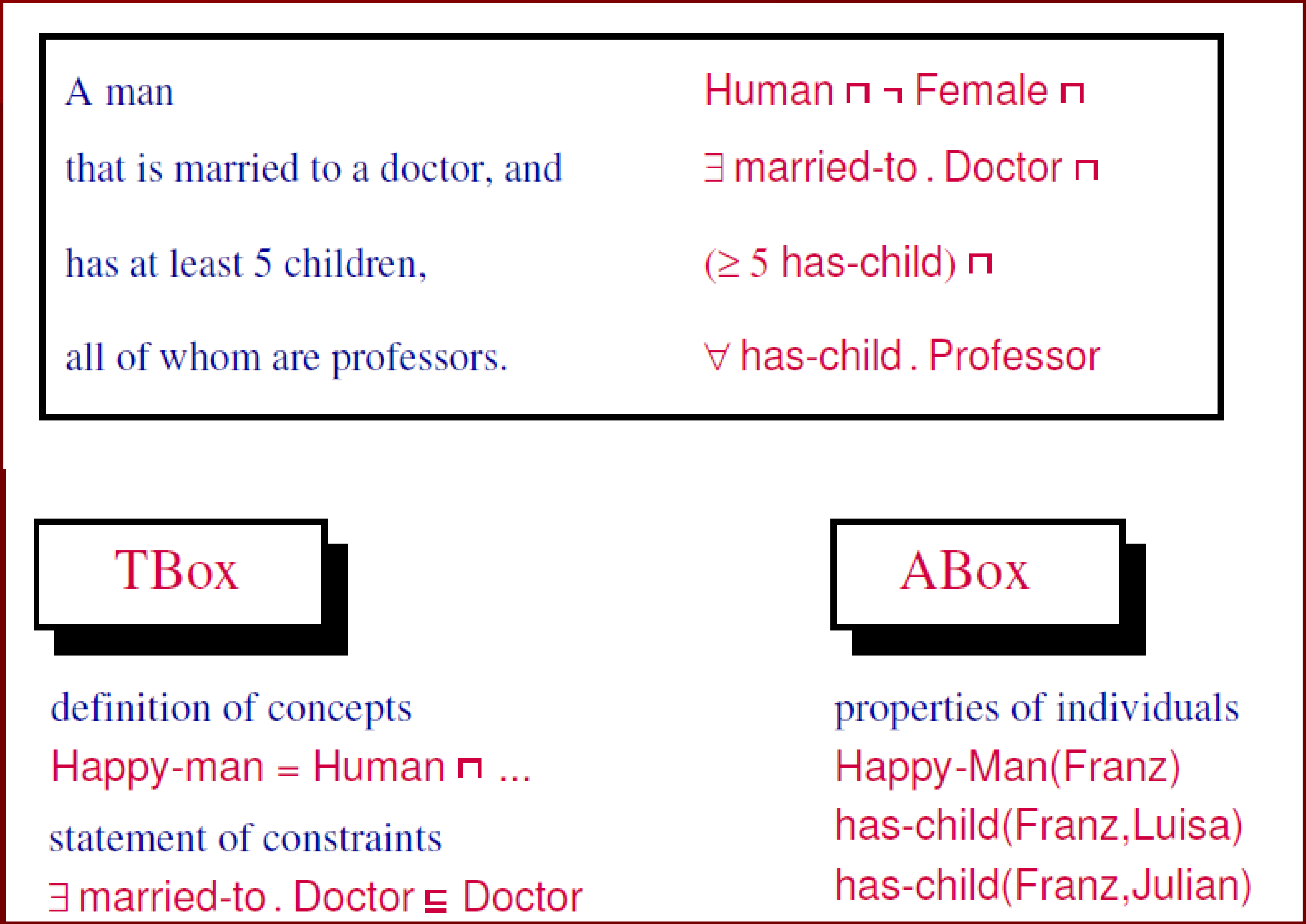
Constructors for the creation of compound concepts and roles.

Formal, based on logic semantics.

Inference Component

Derivation of implicit knowledge (e.g. subsumptions)

TBox & ABox



A man	$\text{Human} \sqcap \neg\, \text{Female} \sqcap$
that is married to a doctor, and	$\exists\, \text{married-to} \,.\, \text{Doctor} \sqcap$
has at least 5 children,	$(\geq 5\ \text{has-child}) \sqcap$
all of whom are professors.	$\forall\, \text{has-child} \,.\, \text{Professor}$

TBox

definition of concepts

$\text{Happy-man} = \text{Human} \sqcap \ldots$

statement of constraints

$\exists\, \text{married-to} \,.\, \text{Doctor} \sqsubseteq \text{Doctor}$

ABox

properties of individuals

Happy-Man(Franz)

has-child(Franz,Luisa)

has-child(Franz,Julian)

TBox & ABox

- TBox sentences describe a conceptualization, i.e., a set of concepts and their properties pertaining to a domain.
- An Abox describes named individuals and their relationships with possible reference to Tbox concept descriptions.

Reasoning

- Reasoner is a program that extracts logical consequences from a set of explicitly stated facts or axioms.
- Typically provides automated support for inference functions such as sorting, debugging, and querying.

Reasoning-prerequisites

- Soundness guarantees that any proposition that is provable in a deductive system is also true in all interpretations or syructures of the semantic theory of the language which it is based on.
- Completeness guarantees that every valid (true) proposition is also provable.
- Taken together they ensure that all and only valid (true) propositions are provable.

Reasoning

- A Reasoner must:
  - Handle atoms (Provide reasoning in ABox)
  - Do not support the unique name assumption
  - Support implication checks
  - Answer conjunctive questions in Abox
  - Work with XML schema databases

Protégé

- He is an editor of ontologies and knowledge bases (http://protege.stanford.edu).
- It is also an open source Java tool that provides an extensible architecture for building custom knowledge-based applications.
- OWL Plug-in of Protégé provides support for editing semantic web ontologies

Protégé

- Three reasoners ahave been embedded in protégé:
  - Pellet
  - Hermit
  - Fact++ (since protégé 4.0 alpha)

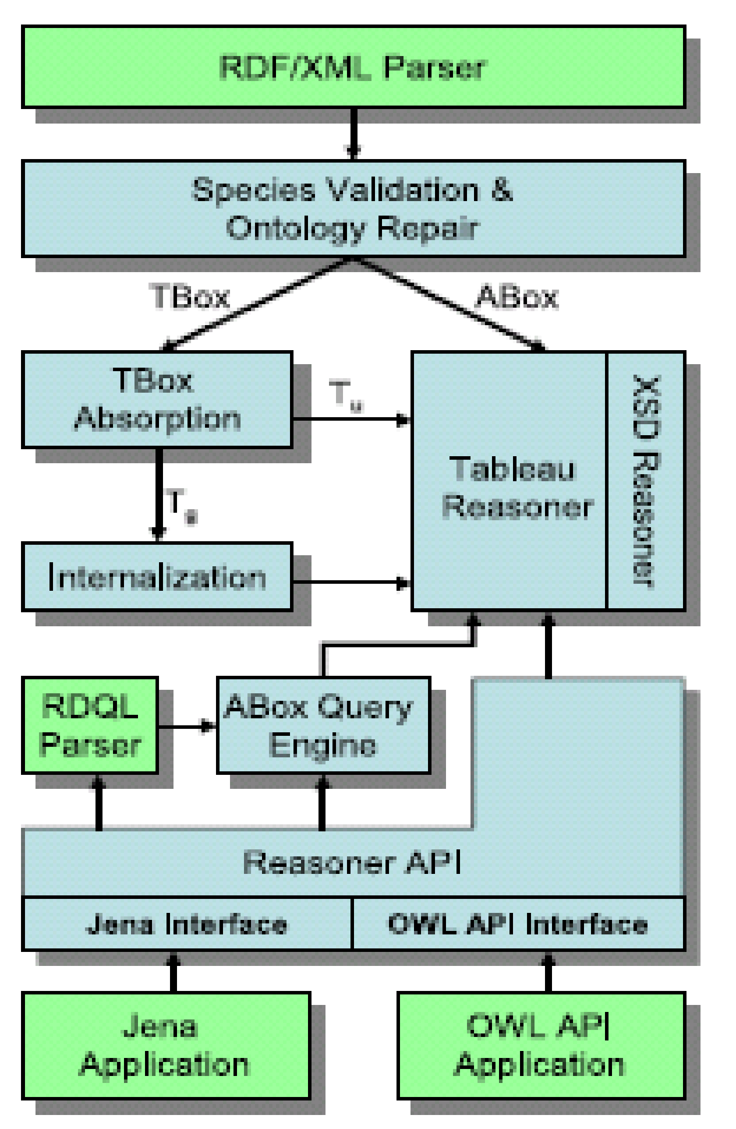
PELLET

- It is based on tableaux algorithms developed for expressive Description Logics (DLs).
- It supports all OWL DL constructors, including owl:oneOf and owl:value
- It is implemented in pure Java and is available under the MIT & DuLi:AGPL license.

PELLET

- It uses a combination of existing sound and complete algorithms.
- It provides inferences that are sound and complete for OWL DL without nominals (ie SHIN(D)) and without inverse properties (ie SHON(D)).

PELLET - Architecture



HERMIT

- It is a DL reasoner that implements a hypertableau calculus, which greatly reduces the number of possible models to consider.
- It incorporates the "anywhere blocking" technique, which limits the sizes of the created models.
- HermiT, given an OWL file, can determine whether the ontology is consistent or not, detect subsumption relationships between classes, and much more.

HERMIT

- It supports reasoning with ontologies containing description graphs.
- Description graphs allow the representation of structured objects, i.e. objects composed of many interconnected parts in arbitrary ways.
- It is available as an open source Java library and includes both a Java API and a simple command-line interface.

HERMIT

- It can process ontologies in any format that is manageable by the OWL API, including RDF/XML, OWL Functional Syntax, KRSS, and OBO
- It supports OWL 2 DL, which corresponds to SROIQ DL.

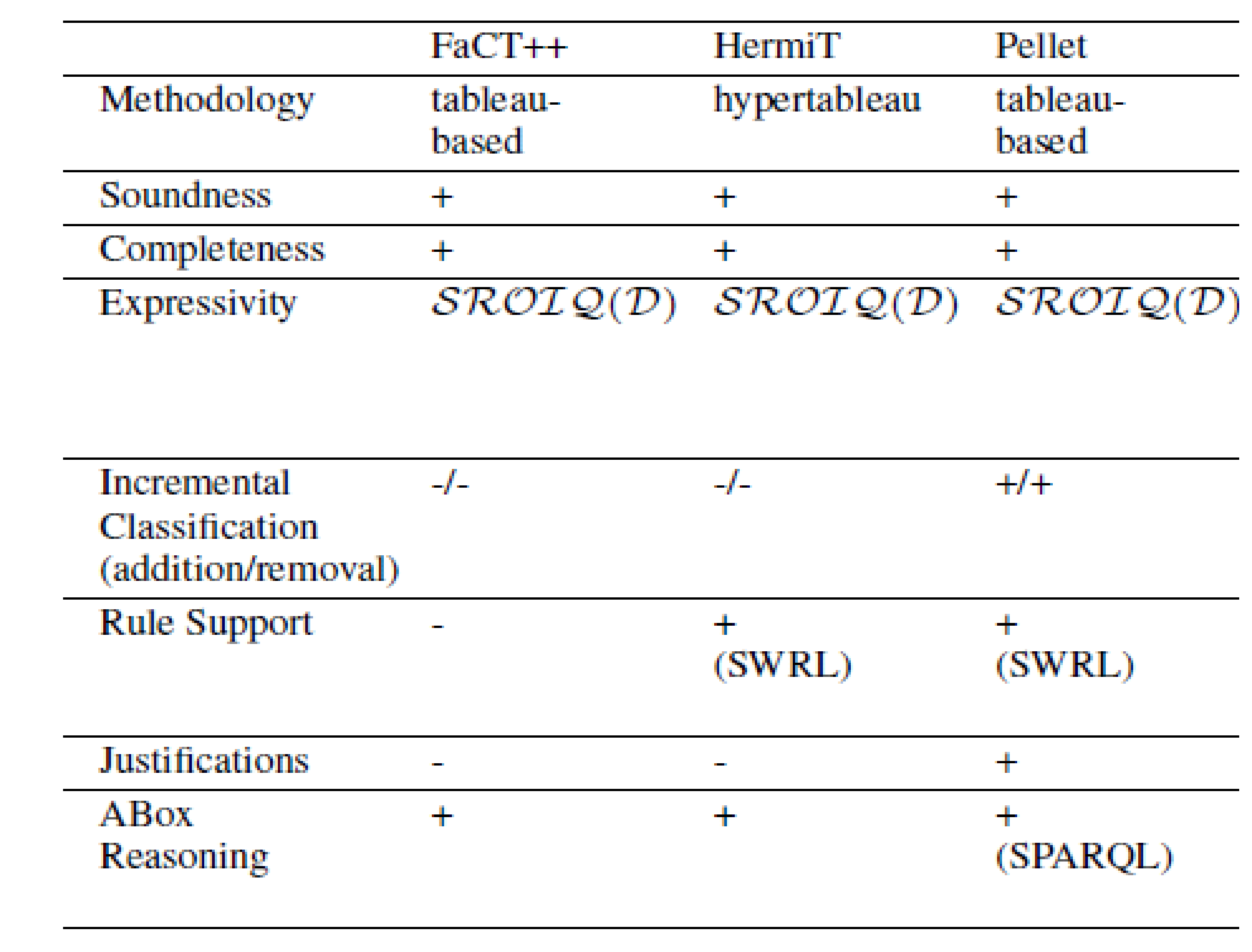
FACT ++

- It is a reasoner based on the tableaux method for expressive DLs.
- Covers OWL and OWL 2 (lacks support for key constraints and some data types) based on DL ontologies.
- Open source software distributed under the LGPL license.

RacerPro

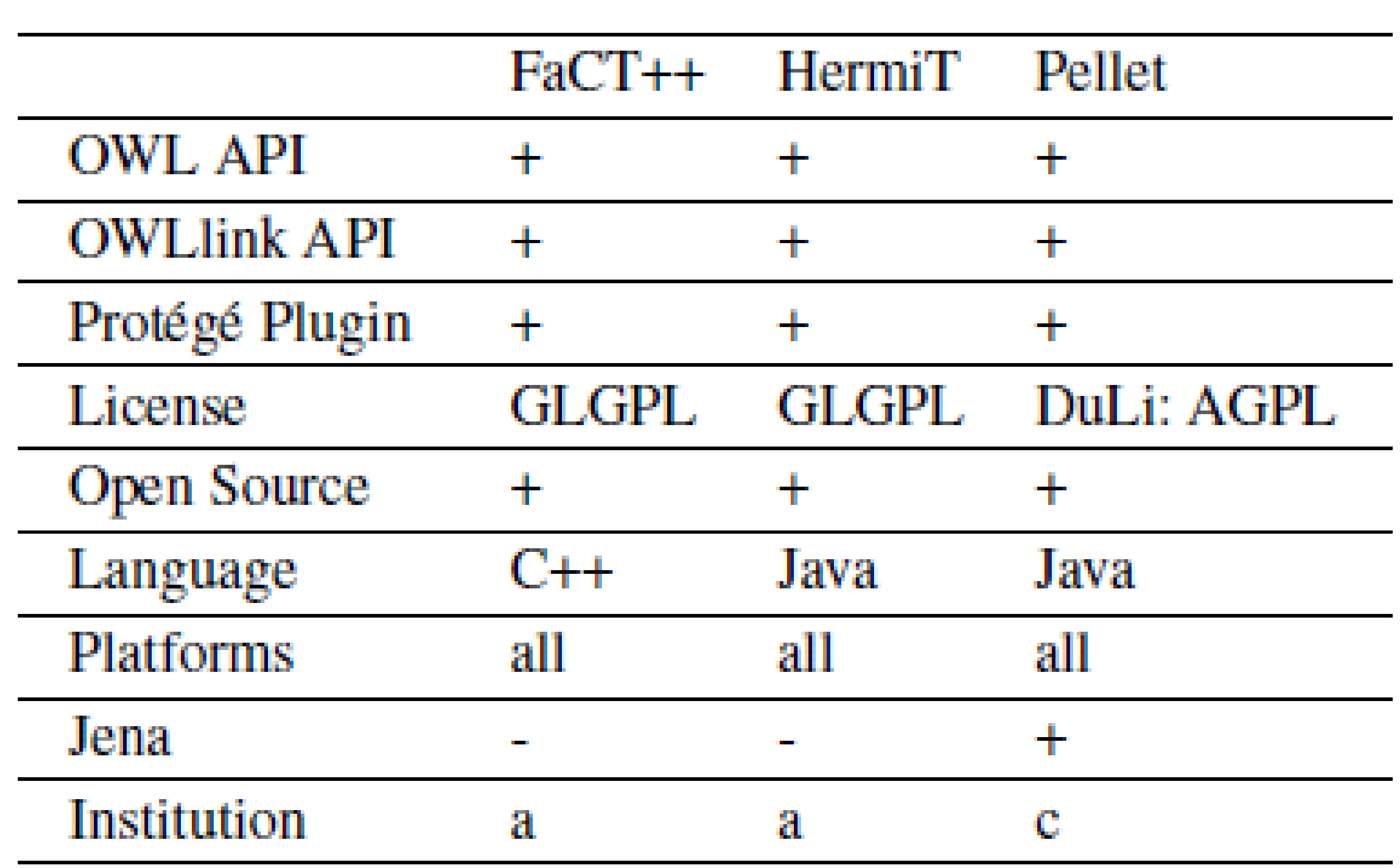
- The first OWL Reasoner on the Market.
- It appeared in 2002
- One of the fastest reasoning systems.
- Based on the tableau method.
- It is used as the back-end inference system with Protégé
- Supports OWL DL.

Comparison (1)

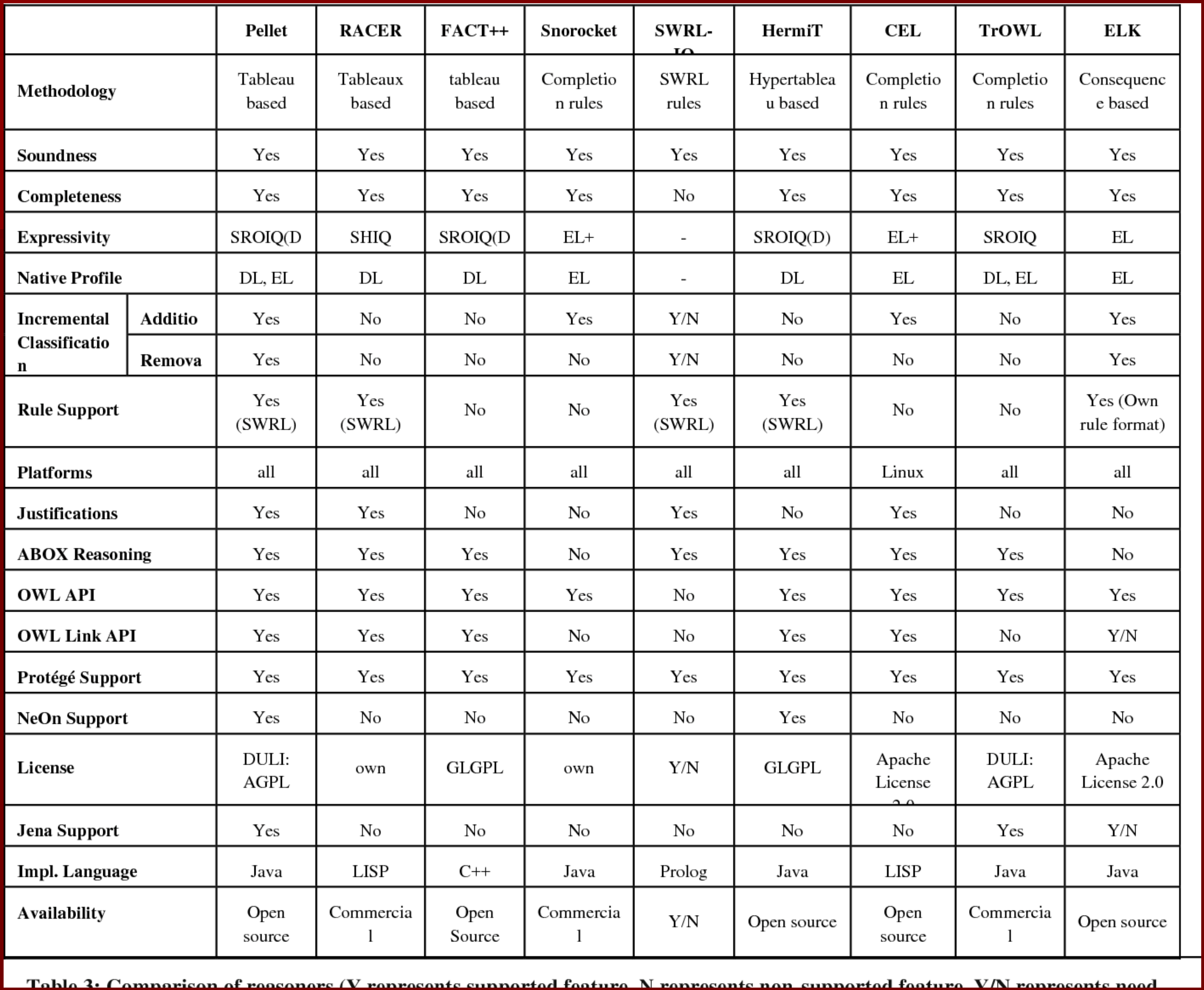


	FaCT++	HermiT	Pellet
Methodology	tableau-based	hypertableau	tableau-based
Soundness	+	+	+
Completeness	+	+	+
Expressivity	$\mathcal{SROIQ}(\mathcal{D})$	$\mathcal{SROIQ}(\mathcal{D})$	$\mathcal{SROIQ}(\mathcal{D})$
Incremental Classification (addition/removal)	-/-	-/-	+/+
Rule Support	-	+ (SWRL)	+ (SWRL)
Justifications	-	-	+
ABox Reasoning	+	+	+ (SPARQL)

Comparison (2)



	FaCT++	HermiT	Pellet
OWL API	+	+	+
OWLlink API	+	+	+
Protégé Plugin	+	+	+
License	GLGPL	GLGPL	DuLi: AGPL
Open Source	+	+	+
Language	C++	Java	Java
Platforms	all	all	all
Jena	-	-	+
Institution	a	a	c



		Pellet	RACER	FACT++	Snorocket	SWRL-IQ	HermiT	CEL	TrOWL	ELK
Methodology		Tableau based	Tableaux based	tableau based	Completio n rules	SWRL rules	Hypertablea u based	Completio n rules	Completio n rules	Consequenc e based
Soundness		Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Completeness		Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes
Expressivity		SROIQ(D	SHIQ	SROIQ(D	EL+	-	SROIQ(D)	EL+	SROIQ	EL
Native Profile		DL, EL	DL	DL	EL	-	DL	EL	DL, EL	EL
Incremental Classificatio n	Additio	Yes	No	No	Yes	Y/N	No	Yes	No	Yes
	Remova	Yes	No	No	No	Y/N	No	No	No	Yes
Rule Support		Yes (SWRL)	Yes (SWRL)	No	No	Yes (SWRL)	Yes (SWRL)	No	No	Yes (Own rule format)
Platforms		all	all	all	all	all	all	Linux	all	all
Justifications		Yes	Yes	No	No	Yes	No	Yes	No	No
ABOX Reasoning		Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	No
OWL API		Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes
OWL Link API		Yes	Yes	Yes	No	No	Yes	Yes	No	Y/N
Protégé Support		Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
NeOn Support		Yes	No	No	No	No	Yes	No	No	No
License		DULI: AGPL	own	GLGPL	own	Y/N	GLGPL	Apache License 2.0	DULI: AGPL	Apache License 2.0
Jena Support		Yes	No	No	No	No	No	No	Yes	Y/N
Impl. Language		Java	LISP	C++	Java	Prolog	Java	LISP	Java	Java
Availability		Open source	Commercia l	Open Source	Commercia l	Y/N	Open source	Open source	Commercia l	Open source

Table 3: Comparison of reasoners (Y represents supported feature, N represents non-supported feature, Y/N represents need