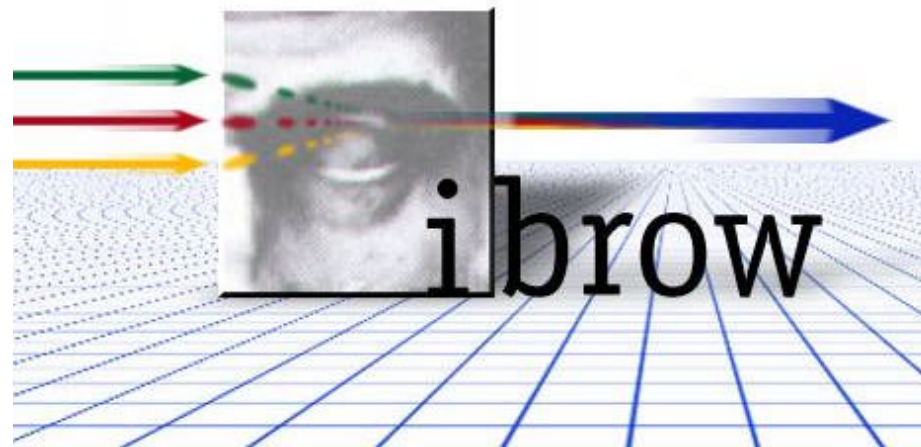


IBROW: An Intelligent Brokering Service for Knowledge- Component Reuse on the World-Wide Web

UPML and the Semantic Web



Dieter Fensel

Vrije Universiteit Amsterdam

Contents

- The Semantic Web
- Ibrov: Making the semantic web *dynamic*
- Languages for the semantic web
- Conclusions

1 The Semantic Web

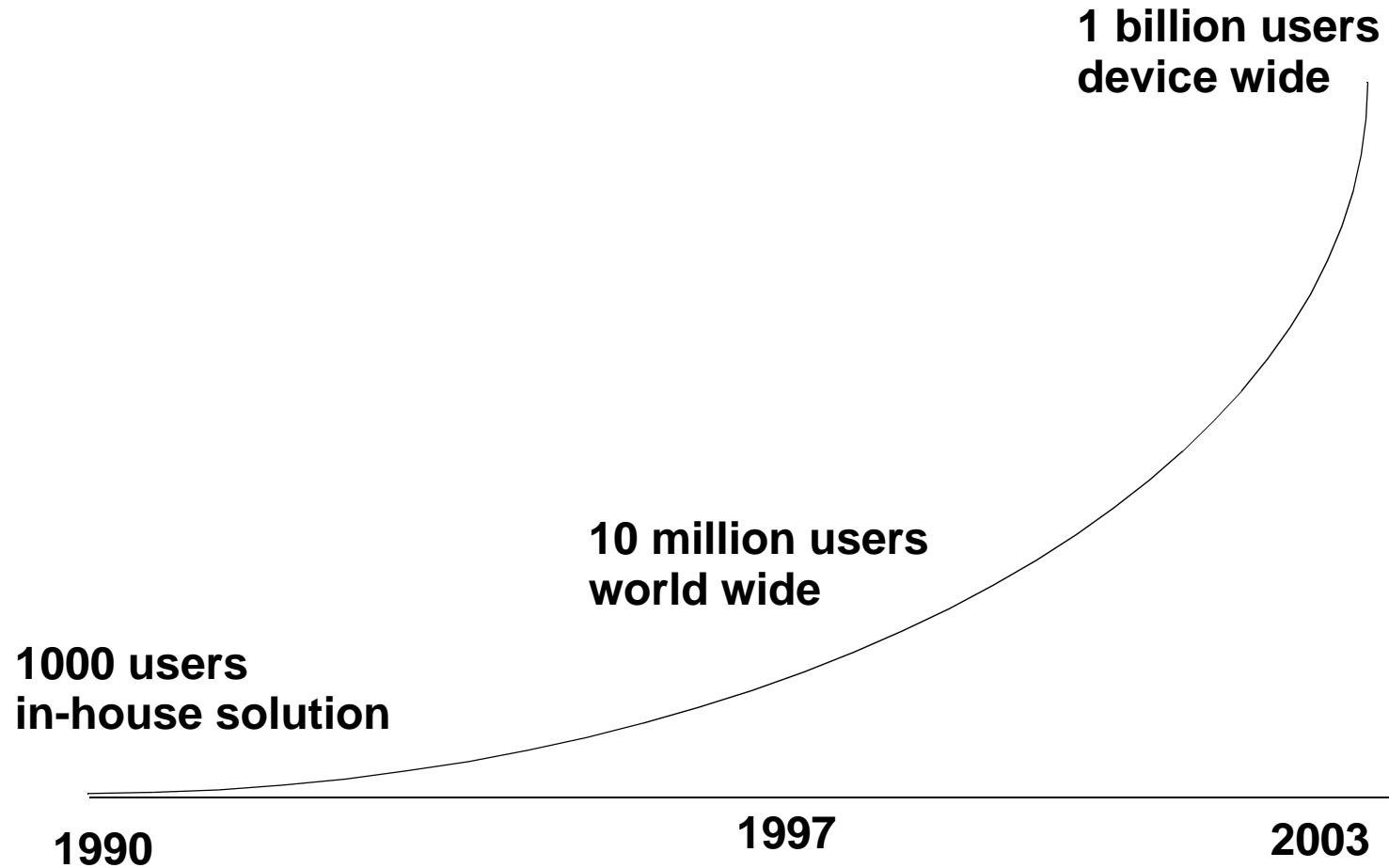
- The World Wide Web is a big and impressive success story, both in terms
 - of the amount of available information and
 - of the growth rate of human users.
 - It starts to penetrate most areas of our daily life and business.
 - This success is based on its simplicity.
- ==> The restrictiveness of HTTP and (early) HTML allowed software developers, information provider, and users to make easy access of to new media helping it to reach a critical mass.

The Semantic Web

- However, this simplicity may hamper the further development of the Web.
- Or in other words:

What we see currently is the very first version of the web and the next version will probably even more bigger and much more powerful compared to what we have now.

The Semantic Web



The Semantic Web

The bottlenecks of the current web technology create

- problems in searching information,
- problems in extracting information,
- problems in maintaining information, and
- problems in generating information.

The Semantic Web

Tim Berners-Lee has a vision of a *semantic web* which

- has machine-understandable semantics of information, and
- trillions of small specialized reasoning services that provide support in automated task achievement based on the accessible information.

==> This gives a completely new perspective for the knowledge acquisition, knowledge engineering, and knowledge representation communities.

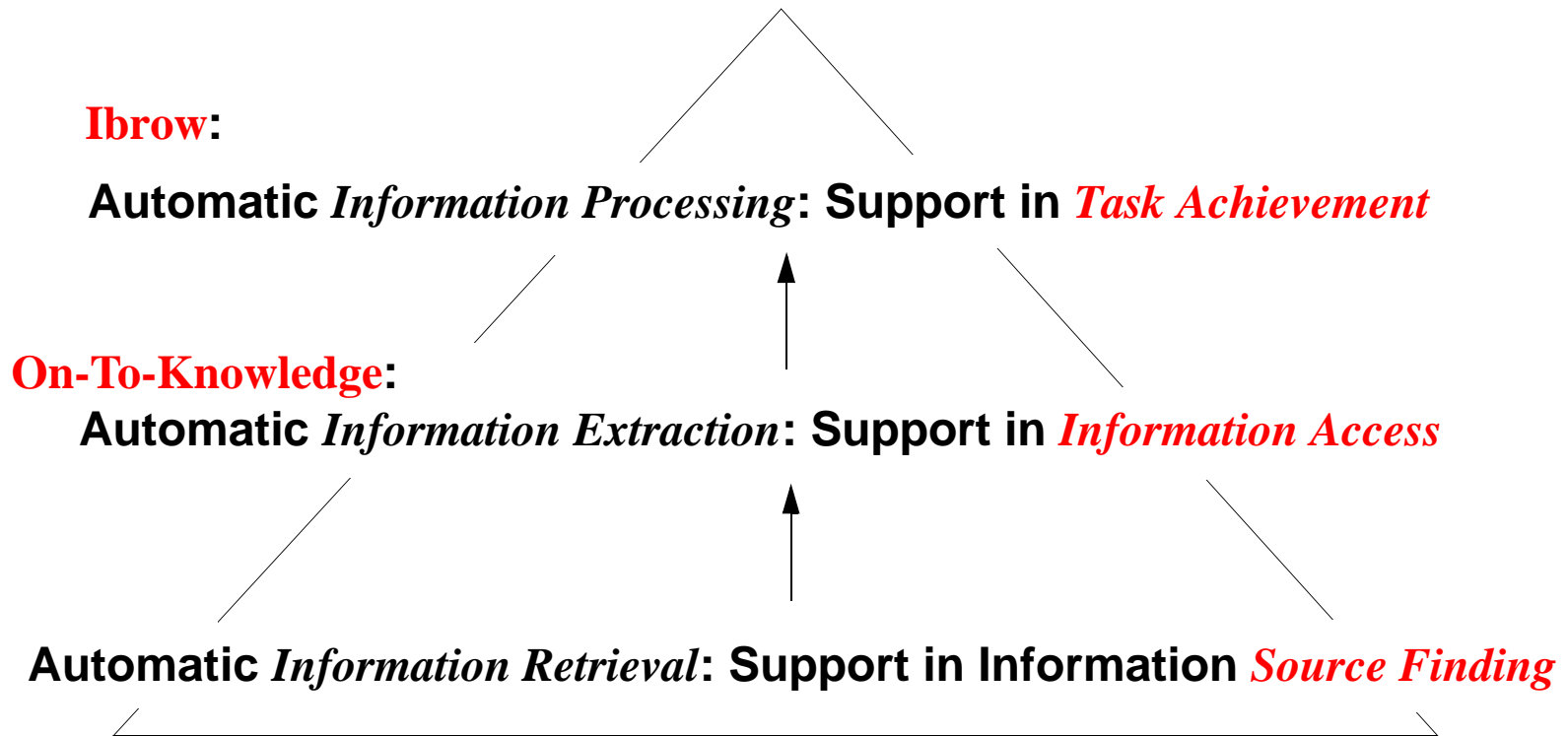
==> It requires language suitable for describing with machine-understandable semantics web-based information sources and reasoning components.

==> This is the vision of Ibrov and the purpose of UPML.

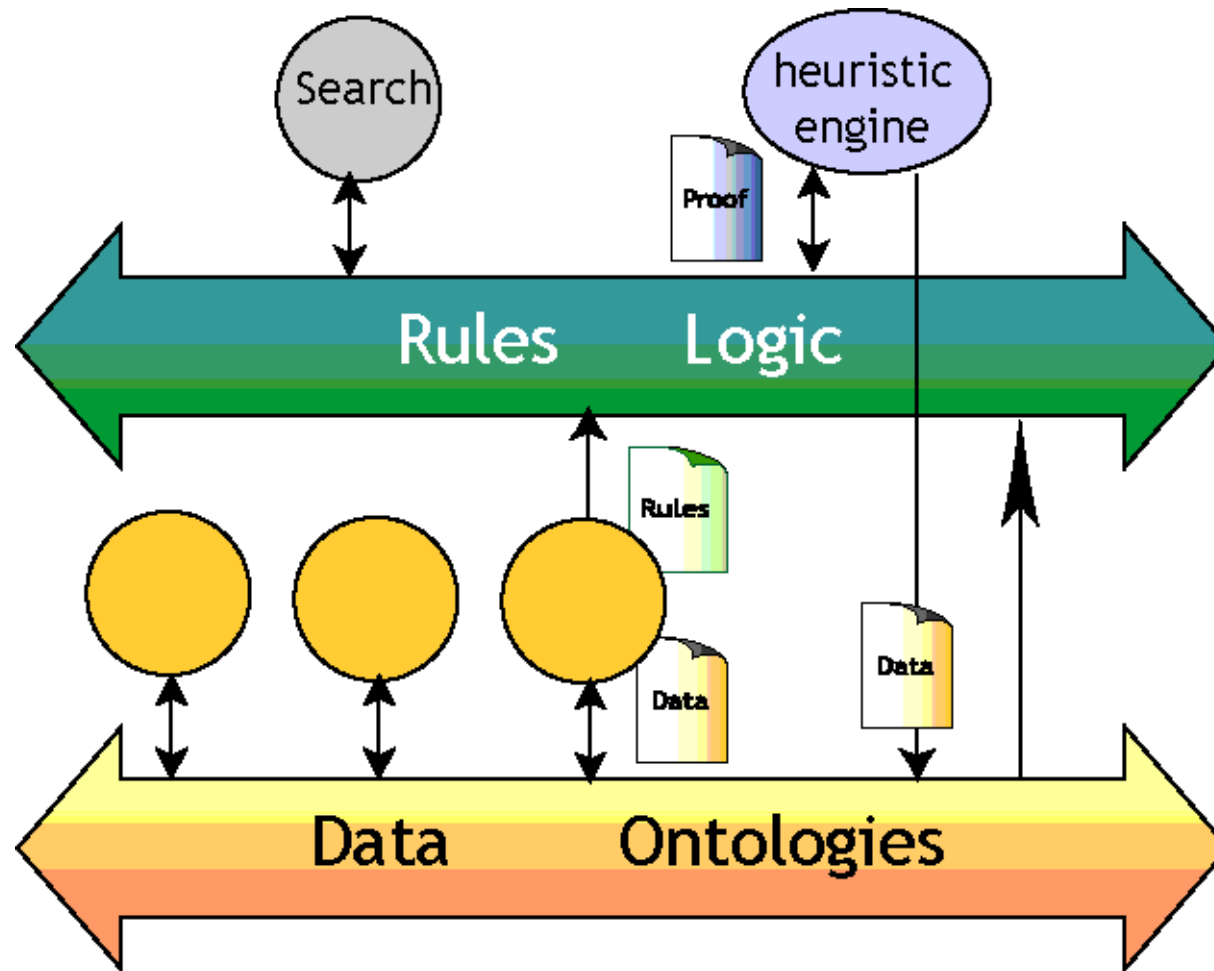
2 Ibrow: Making the semantic web *dynamic*

- WWW: Current web technology provides human access to unstructured information sources.
- *Static* part of the Semantic Web: Ontology technology will help humans in accessing information based on machine-processable semantics.
- *Dynamic* part of the Semantic Web: Heuristic reasoning services will help humans in task achievement by automatically processing information based on their machine-processable semantics.

Ibrow: Making the semantic web *dynamic*



Ibrow: Making the semantic web *dynamic*

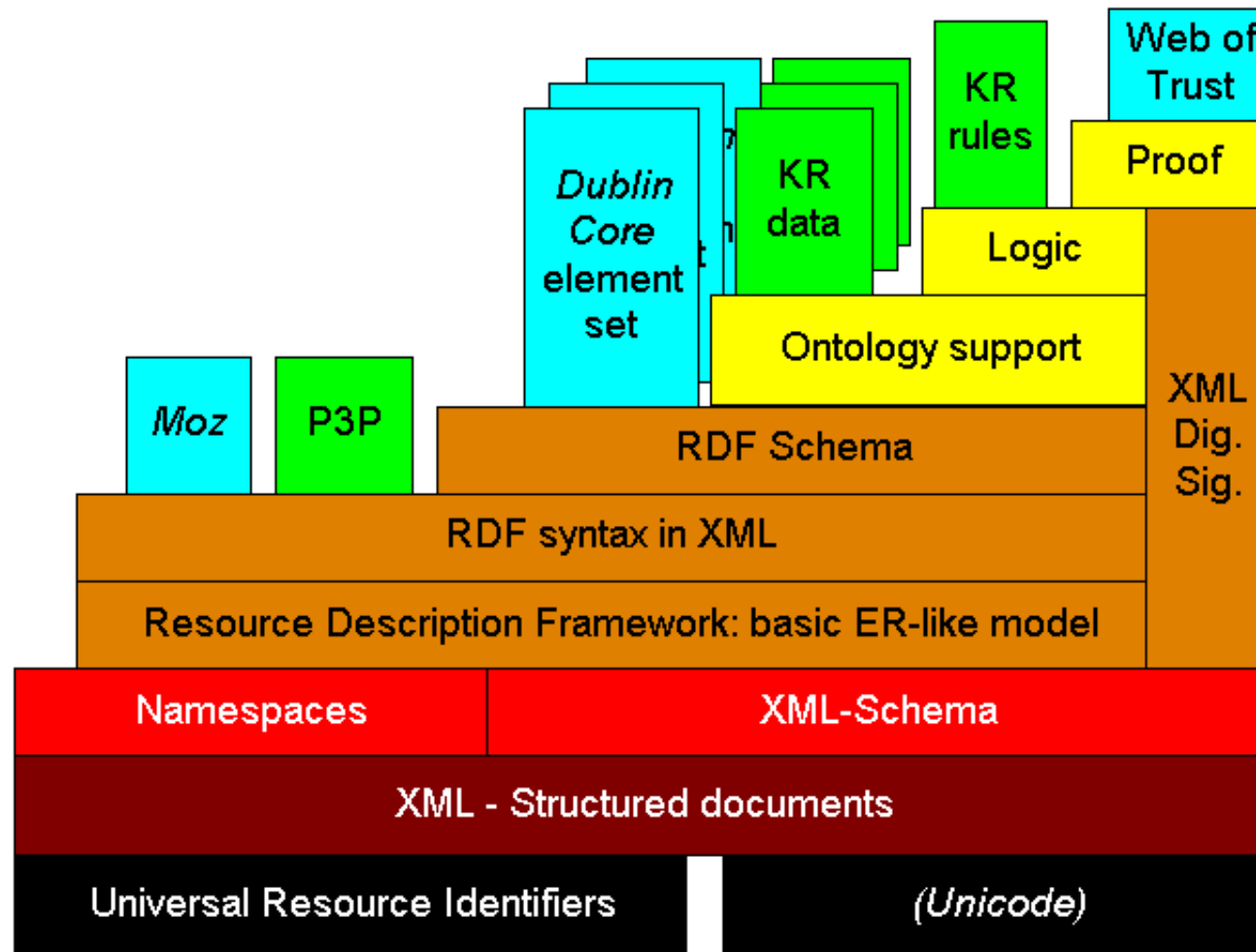


3 Languages for the Semantic Web

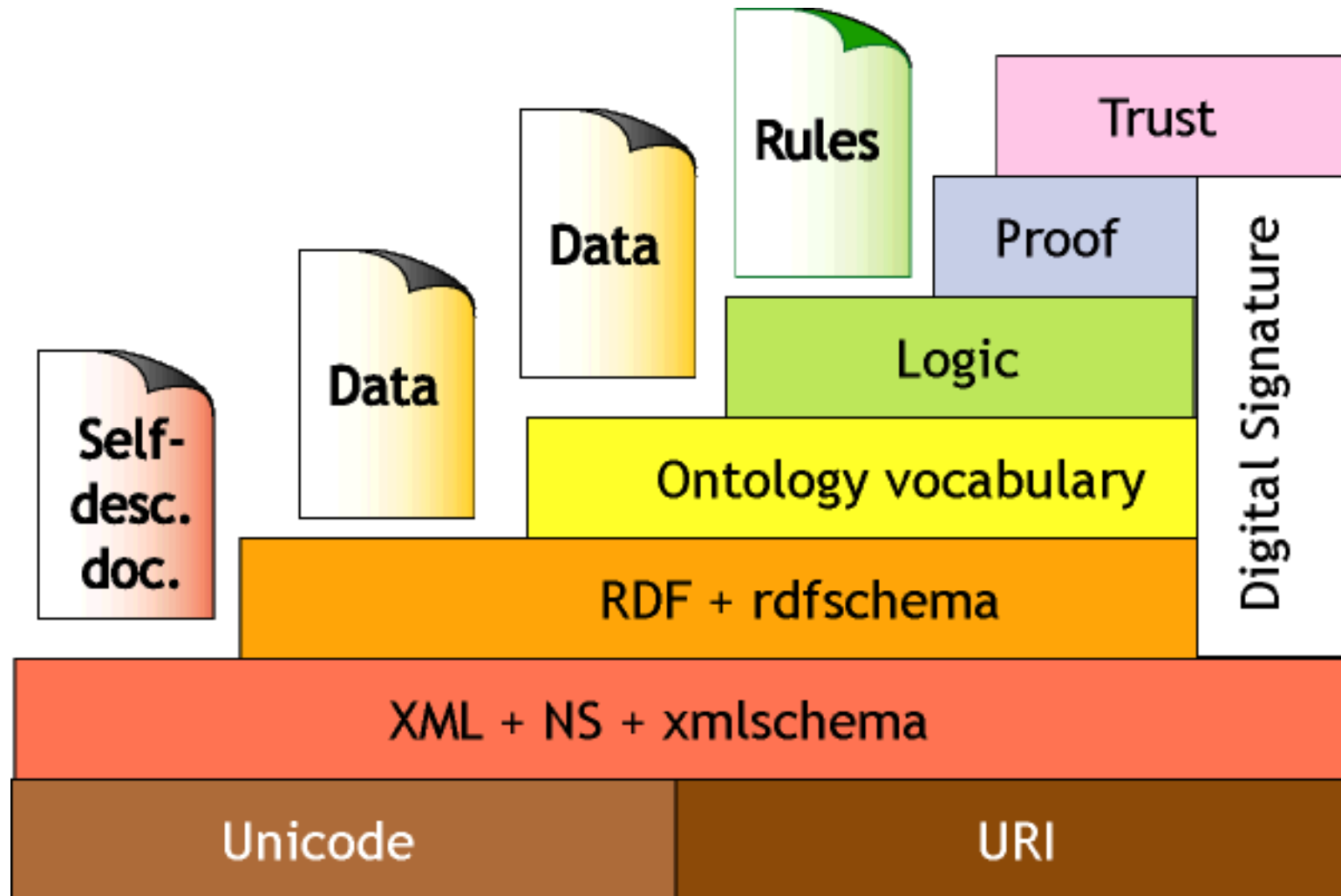
- The web needed a language for describing how to present information to humans.
- The semantic web needs a language for describing the semantics of information in a machine processable way.
- The next generation of the semantic web (i.e., the *dynamic* web) needs a language framework for describing heuristic reasoning services on top of machine-processable data.

==> This is UPML

Languages for the Semantic Web: The layered languages



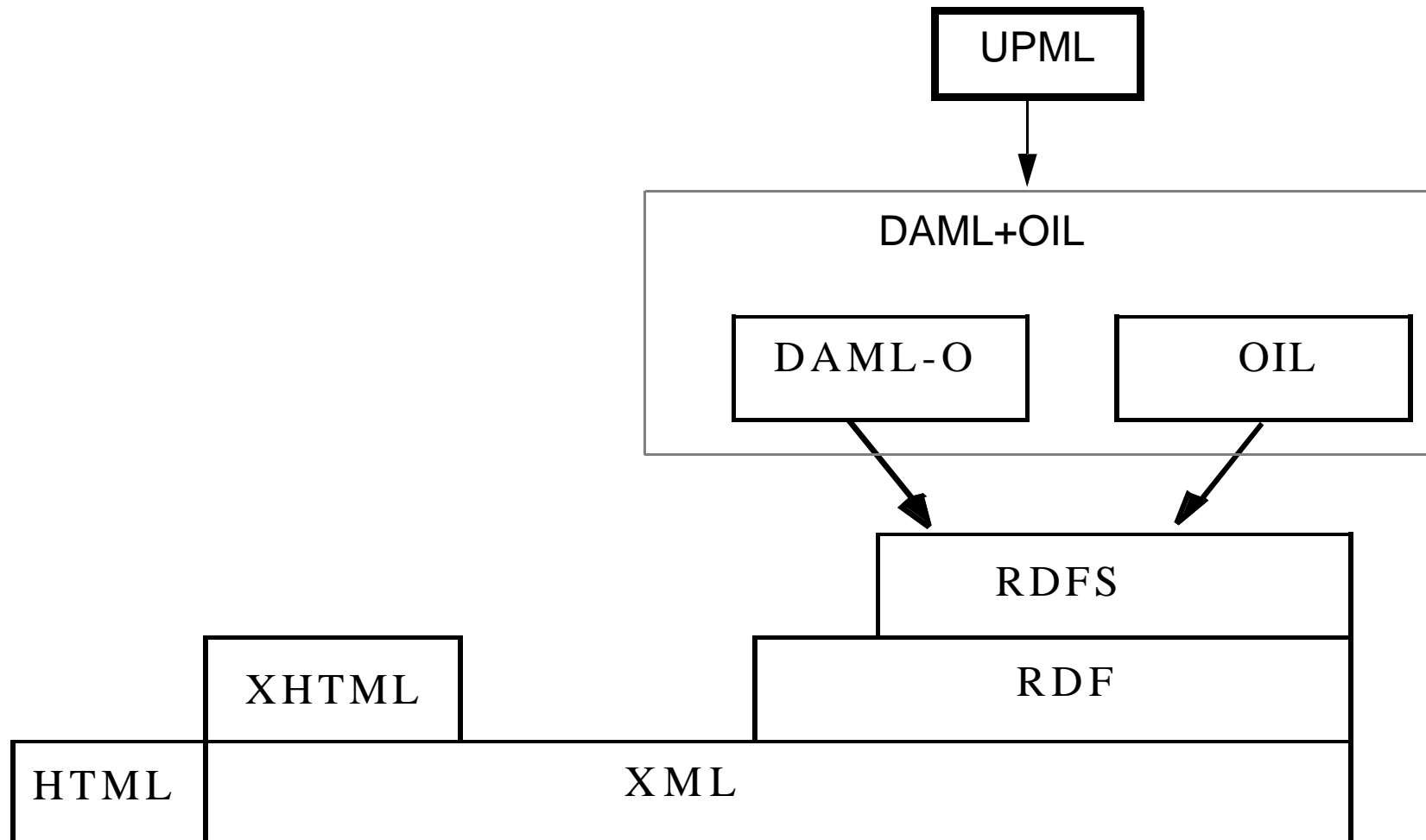
Languages for the Semantic Web: The layered languages



Languages for the Semantic Web: The layered languages

- XML provides a serialized syntax for tree structures.
- RDF defines a data model on top of XML.
- RDF schema (RDFS) defines basic ontology primitives in RDF.
- OIL (developed in Ontoknowledge) extends RDFS to provide a full-fledged web-based ontology language.
- UPML defines an *architecture* on top of these languages for describing reasoning services on the web.

Languages for the Semantic Web: The layered languages



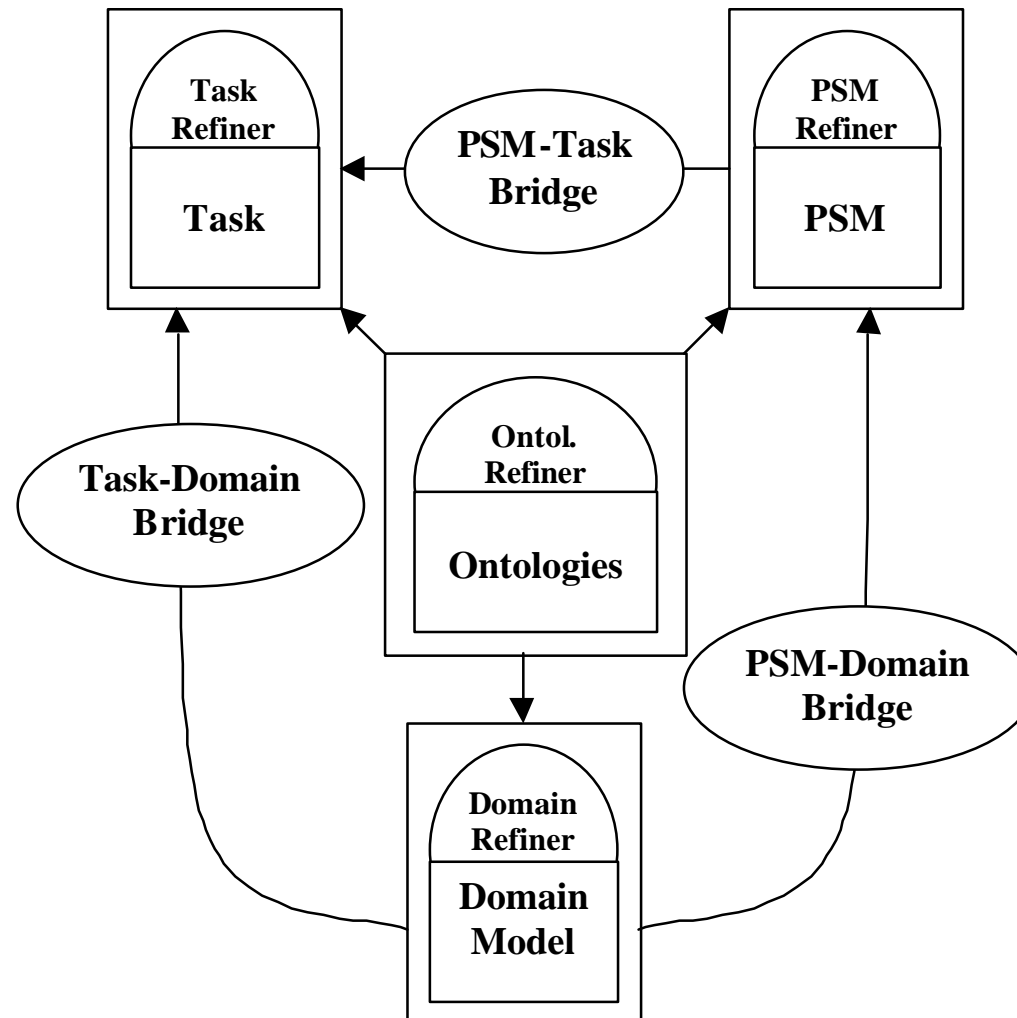
Languages for the Semantic Web: UPML is an Architecture

- UPML is not yet another language but an architecture that can be used to annotate elements of the semantic web.
- Which specification language is used for specifying the elementary slots of UPML is kept open.
- In case a final standard for describing the static aspects of the semantic web will be defined by the W3C (i.e., DAML+OIL⁺⁺) it will be used by UPML for this purpose.
- Currently, UPML is language neutral and different vendors may use different languages (i.e., it is too early for us to standardize this aspect).

Languages for the Semantic Web: UPML and RDF/XML Schema

- UPML defines a set of templates that can be used to annotate elements of the semantic web.
- We defined an XML schema syntax for these templates allowing to use XML editors, parsers, and browsers to work with UPML.
- We defined an RDF schema syntax for these templates allowing to use RDF editors, parsers, and browsers to work with UPML.
This syntax is also used by Protégé-2000 (see later).

Languages for the Semantic Web: The Architecture of UPML



Languages for the Semantic Web: Domain Model

- The description of the domain model introduces domain knowledge as it is required by the problem-solving method and the task definition.
- It consists of three elements:
 - characterization of properties,
 - assumptions,
 - and the domain knowledge itself.
- It is an specific element of describing knowledge-intensive reasoning services and link dynamic reasoning resources with static knowledge of the semantic web.
- It is the element that links UPML with current *static part* of the semantic web technology.

Languages for the Semantic Web: Task Model

- The task specifies the problem that should be solved by the knowledge-based system.
- Contrary to most approaches in software engineering this problem definition is kept domain independent, which enables the reuse of generic problem definitions for different applications.
- In terms of electronic commerce, a task describes the requirements of a client.

Languages for the Semantic Web: Problem-Solving Methods

- Problem-solving methods describe which reasoning steps and which types of knowledge are needed to perform a task.
- UPML distinguishes two different types of problem-solving methods:
 - Complex problem-solving methods decompose a task into subtasks.
 - Primitive problem-solving methods make assumptions about domain knowledge to perform a reasoning step. They do not have an internal structure.
- Problem-solving methods (i.e., heuristic reasoning components) are the *dynamic* aspect that Ibrow adds to other research on the semantic web.
- In terms of electronic commerce, they describe the service provided by a vendor.

Languages for the Semantic Web: Ontologies

- An ontology provides an explicit specification of a conceptualization which can be shared by multiple reasoning components.
- Ontologies provide the definition of signatures and axioms that are used by the other parts of the architecture.
- Ontology components provide the reusable terminology used by all other components to describe their service.
- It is the key element for interchange.

Languages for the Semantic Web: Bridges

- Bridges explicitly model the relationships between two distinguished parts of an architecture, e.g. between domain and task or task and problem-solving method.
- The can also be found as a design pattern in software architectures.



Languages for the Semantic Web: Refiners

- Refiners can be used to express the stepwise adaptation of elements of a specification, e.g. a task is refined or a problem-solving method is refined
- They can to some extent be found as design patterns in software engineering.
- They enable a structured approach for developing reasoning services, see D. Fensel and E. Motta: Structured Development of Problem Solving Methods, to appear in *IEEE Transactions on Knowledge and Data Engineering*.

Languages for the Semantic Web: UPML_{Protégé-2000}

- Protégé-2000 is the standard editor for ontologies and problem-solving methods.
- Stanford developed an adaptation of Protégé-2000 for UPML.
- It provides a full-fledged editor environment for writing and browsing UPML specifications.
- It automatically generates RDF(S) syntax of UPML specifications.
- It is the merit of Monica Crubézy.

Languages for the Semantic Web: UPML_{Protégé-2000}

The screenshot displays the Protégé-2000 interface with three overlapping windows showing class definitions and their template slots.

- Pragmatics (instance of :STANDARD-CLASS)**
 - Name: Pragmatics
 - Documentation: Pragmatics collect pragmatic information on specification elements.
 - Role: Concrete
 - Template Slots:

Slot Name	Type	Cardinality	Default	Other Facets
author	Instance	Single		classes={Author_}
explanation	String	Single		
last-date-of-modification	String	Single		
reference	String	Single		
URL	String	Single		
usage_history	Instance	Single		classes={Usage_History}
- Competence-Complex-PSM (instance of :STANDARD-CLASS)**
 - Name: Competence-Complex-PSM
 - Documentation: Specs- specifie functionality of t DK fulfils requir definition). Goa subtask decom
 - Role: Concrete
 - Template Slots:

Slot Name	Type	Ca		
postconditions-goal	Instance	Multi		
preconditions	Instance	Multi		
roles	Instance	Multi		
tasks	Instance	Multi		
- Operational-Description (instance of :STANDARD-CLASS)**
 - Name: Operational-Description
 - Documentation: Specs- specification of complex PSM's actual dynamic reasoning process. Explains how the desired competence can be achieved. Defines data flow and execution
 - Role: Concrete
 - Template Slots:

Slot Name	Type	Cardinality	Default	Other Fa
default_implemented_tasks	Instance	Multiple		classes={Task}
intermediate-roles	Instance	Multiple		classes={Intermed
mcl-programs	Instance	Multiple		classes={MCL-Pro
procedures	Instance	Multiple		classes={MCL-Pro
required_tasks	Instance	Multiple		classes={Task}

4 Conclusions

- After three years of work on UPML it has a rather mature state and it is well published in the scientific library.
- Currently, the standardization effort for the *static* aspect semantic web is starting.
 - We participate in an ad-hoc EU/US joint working group on language standardization.
 - We participate in the a working group on standardization issues such as RDF, RDF(S), and ontology languages (i.e, DAML+OIL) announced by the W3C.
- UPML is much ahead of these standardization initiatives because it is about *dynamic* reasoning service.

Conclusions

- Currently we will evaluate the use of UPML within the Ibrow consortium and we will start a standardization effort in 2003.
- We will have a close eye on standardization initiatives on:
 - the static aspects of the semantic web and
 - of service and product specifications in electronic commerce.