

# *Legislative XHTML*

## Integrating ECMA Script & RDF

concerning XHTML annotation of:

- [Council Decision of 10 February 2004 \(OJ 465\)](#)
- [Council Decision of 3 December 2002 \(OJ 331\)](#)
- [Council Decision of 28 February 2000 \(OJ 59\)](#)

## Background

- Semantics Engineer – training adaptive agents through database annotations
- OASIS Legal XML – early active member, eContracts co-chair
- Contributor to W3C Semantic Web Best Practices and Deployment Workgroup document: [A Semantic Web Primer for Object-Oriented Software Developers](#)
- Public domain contributor via SourceForge, Data Consortium, & Hypergrove
  - (a) ECMA/CORBA/C++ toolkit
  - (b) DTD Consolidation of ISO 639, 4217, and SI
  - (c) Namespaces for real property leases
  - (d) Named Value Notation proposal
  - (e) Legal-RDF Script Ontology
  - (f) Washington State Statute resources
  - (g) NAFTA Industrial Classification resources; (h) Calendar resources
- Applications/data architect, XML evangelist, and OMG corporate representative
- IBM developer on SGML Document Editor & OS/2 teams

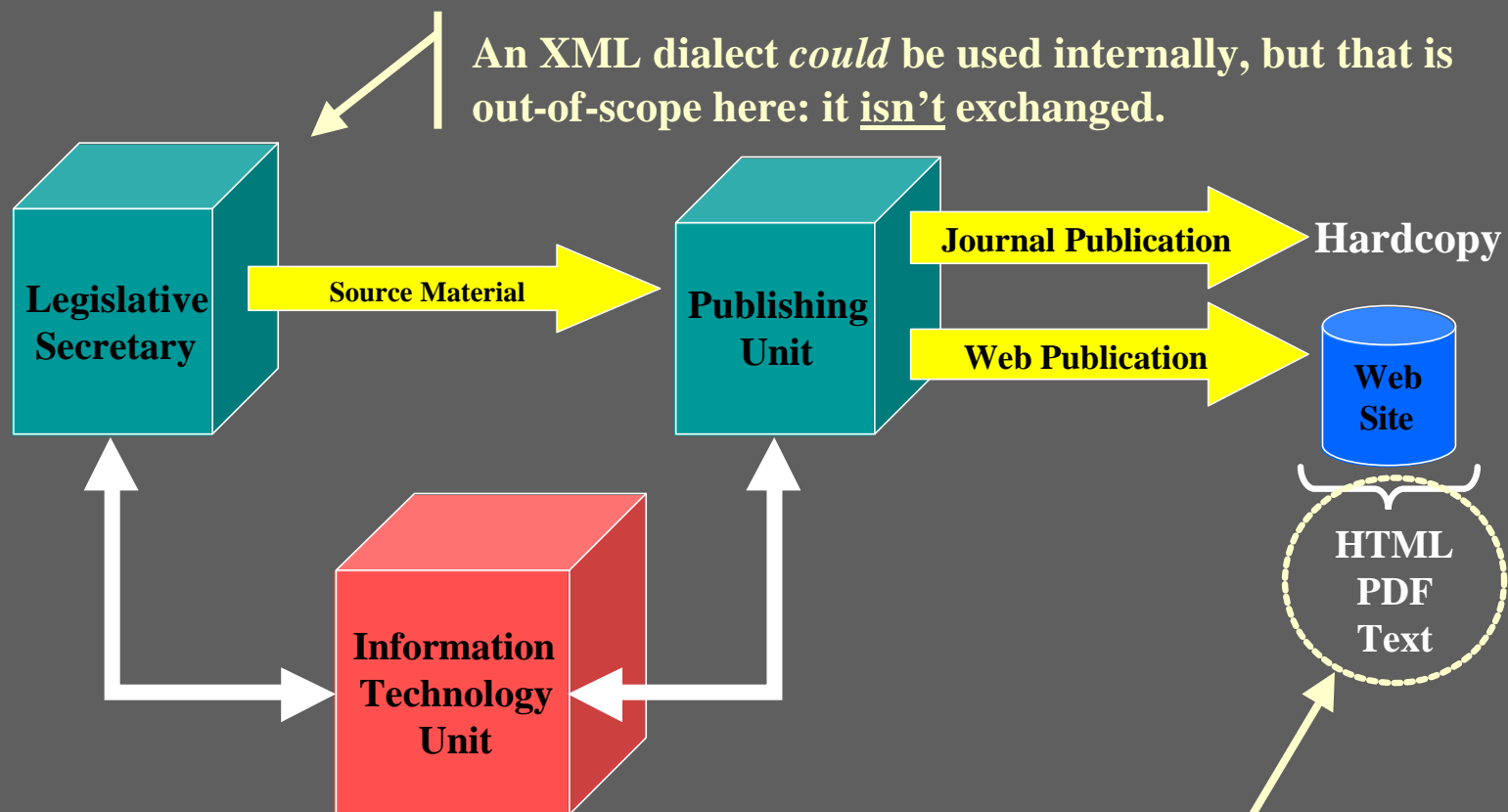
## *Keys to a Legislative XML Standard*

1. Easy adoption by decision-makers
  - promote good government practices
  - be an economically justifiable evolution
  - cater to skill levels of legal aides
2. Easy adoption by technicians
  - build upon already-published statutory material
  - leverage current technical architectures, standards & tools
  - develop a technically “elegant” proposal
3. Easy adoption by public and academia
  - integrate with existing client hardware & software
  - enable significant client-based re-purposing

“Although Web architecture allows for the deployment of new data formats, the creation and deployment of new formats (and agents able to handle them) is **expensive**. Thus, before inventing a new data format (or "meta" format such as XML), designers should carefully consider re-using one that is already available.” [W3 Architecture]

# Use Case

An XML dialect *could* be used internally, but that is out-of-scope here: it isn't exchanged.



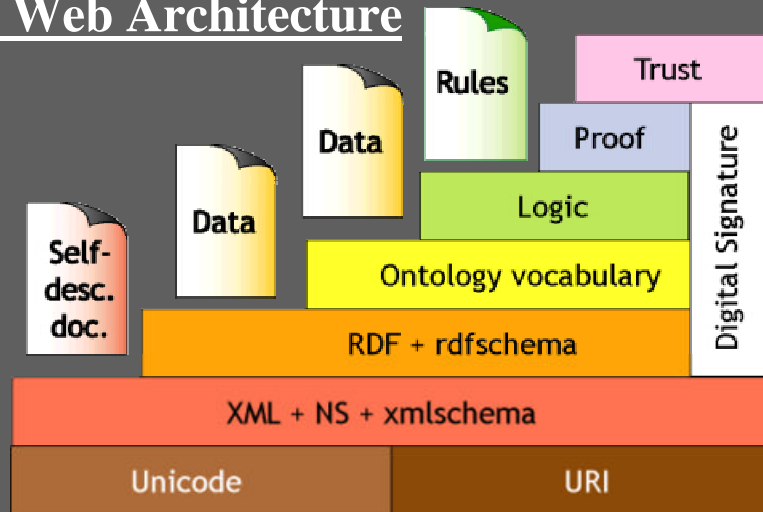
Objective  
Migrate to a maximally exchangeable XML dialect

## ***Problems with Specialized XML for Document Exchange***

- Can an XML dialect be created that is better than XHTML for transmitting flowed documents? *Case study*: OpenOffice having trouble achieving traction
- XML dialects conventionally transport structured data, not flowed documents
- Web Accessibility Initiative conventions relate to XHTML, not XML
- XML presentation stylesheets are still a ‘programming instruction’: XML is not intended now or in the future to be routinely displayed by a browser
- Exposing XML elements & attributes can alienate non-technical users
- Difficulties coordinating deployment/training across jurisdictions with XML tools that are inevitably quite different from Word and other common tools

## Technology Integration

### W3 Web Architecture



### Dynamic HTML

European Computer Manufacturing Association Script (ECMA JavaScript)

+ W3 Document Object Model

+ W3 XHTML with CSS

### Recent Publications

- **XHTML V2 – Proposed W3 XHTML WG Recommendation**
  - standardizes grammatical paragraph model, adding <section>, <h>, <l>, and <nl>
- **RDF/A – Proposed W3 Semantic Web BP&D WG Recommendation**
  - standardizes document annotation (part of XHTML V2)
- **E4XML – Proposed ECMA Recommendation**
  - standardizes dynamic XML access

# Technology Stacks

## Namespaces

ECMA
XHTML

← carriedBy

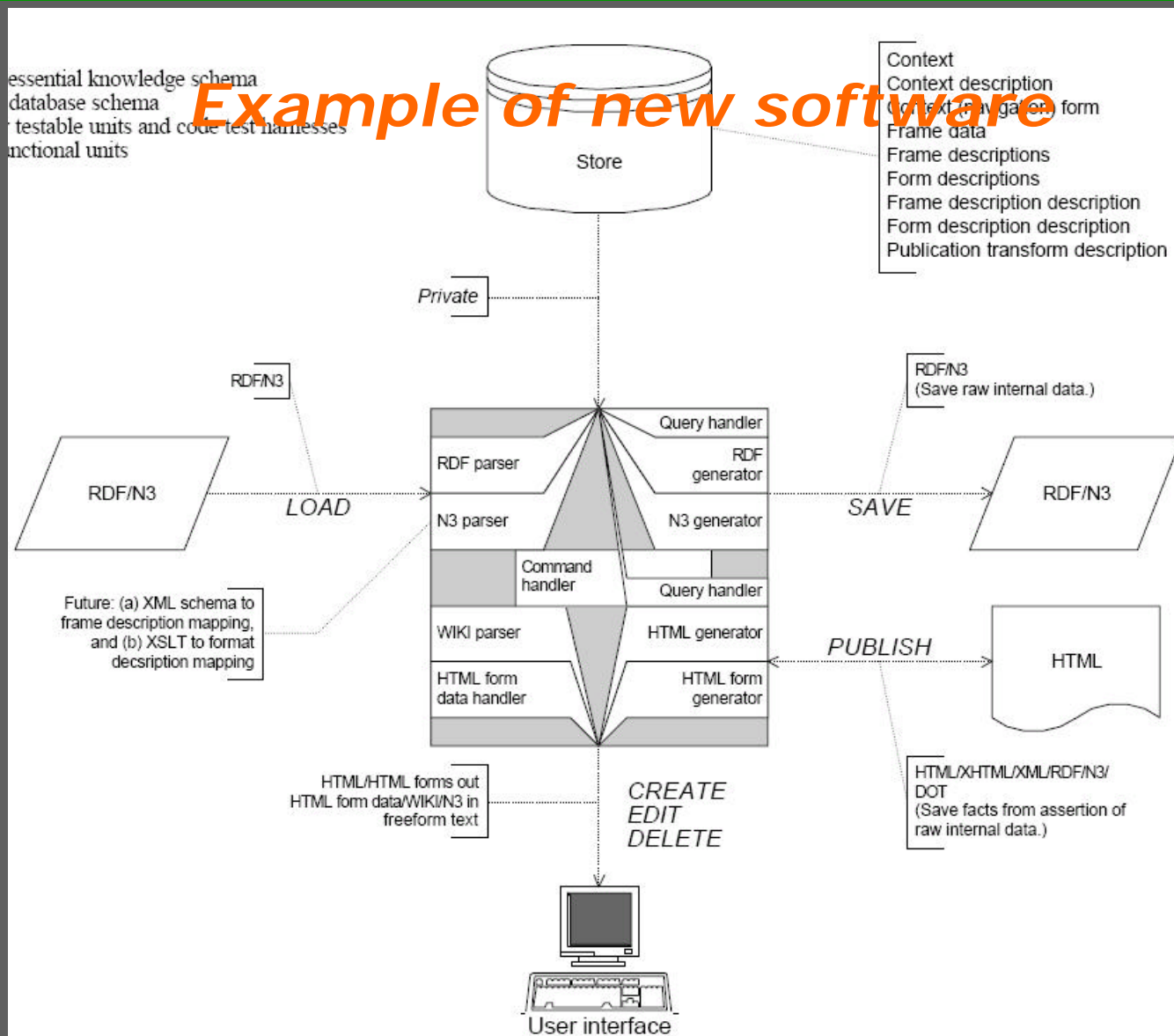
## Models

→ manipulatedBy

## Tools

EU Document Models	EU Domain Models	
Semantic Document Model	Economic/Political Resource Models	
Core Model (Metaclasses, Qualities, Quantities)		
Dublin Core	ISO	Système Intl Units
W3 OWL/RDFS	XML Schema	

Inference Tools
Resource Aggregators
Semantic DB



## *Design Approach*

- Each document defines its own ‘namespace’
- Annotation is functionally separated from authoring
- Annotation = named chunks of text
- Non-technical naming conventions are fundamental

XML names: `<dc:title xml:lang='EN'>`

RDF names: `<hasTitle>`

ECMA names: `Document.Title`



Most intuitive  
Least technical


## Technical Approach - 1

- Assign script names by a specialized Modular XHTML property attribute  
Create *property* attribute in an ECMA namespace – it implements a managed *notation* corresponding to ECMA object naming practices

### XHTML V2 Example

```
<body ecma:property=' Council Deci si on' >
  <secti on ecma:property=' Deci si onPart. 1' >
    <p ecma:property=' Secti onParagraph. 1' >
      Context. Council Deci si on. Deci si onPart. 1. Secti onParagraph. 1
      contains the text for this paragraph in its "eng" slot.
    </p>
  </secti on>
</body>
```

*Fully qualified  
ECMA object  
name*



- Define script names as 'direct objects' in RDFS/OWL  
Resource classes & properties correspond to ECMA objects & slots  
Implement simple verb model: is, has (default) plus tenses and semantics  
Constraint and rule definition languages can apply

## Technical Approach - 2

Example:

`<span ecma:property='Instance.InstanceProperty.PropertyDate.eng'>end of year</span>`

### Generic Legal-RDF – RDF/XML Syntax

```
<Instance rdf:ID='x' asOf='20051231' >
  <has>
    <InstanceProperty rdf:about='p' >
      <has>
        <PropertyDate rdf:ID='z' eng='end of year'
          rdf:about='day#20041231' />
      </has>
    </InstanceProperty>
  </has>
</Instance>
```

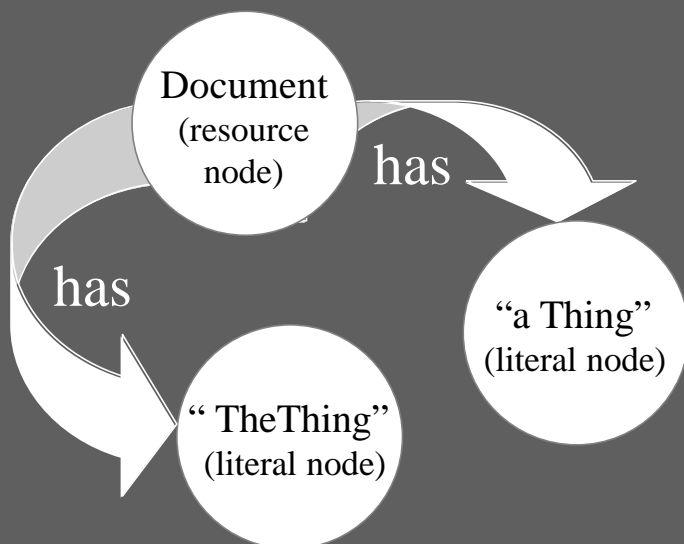
Other possible tenses  
*willHave, had, mustHave, mayHave, willHaveNot, hadNot, mustHaveNot, mayHaveNot*

### Generic Legal-RDF – ‘N8’ RDF Datastore

```
t1: x a Instance          --- -- - 20051231
t2: p a InstanceProperty has t1 F 20051231
   z: day#20041231 a PropertyDate has t2 F 20051231
t3: day#20041231 eng "end of year" --- z F 20051231
```

Records tense of predicate verb, its existential context, a “not” flag, and an ‘asOf’ date

## Document Model Requirements



“A thing” = document is defining its own concept of that “thing”

“The thing” = document is referring to instances external to the document or created by the document.

### Informational Requirements

- Nameable strings of numeric text
- Nameable blocks of hierarchical text
- Nameable strings of flowing text

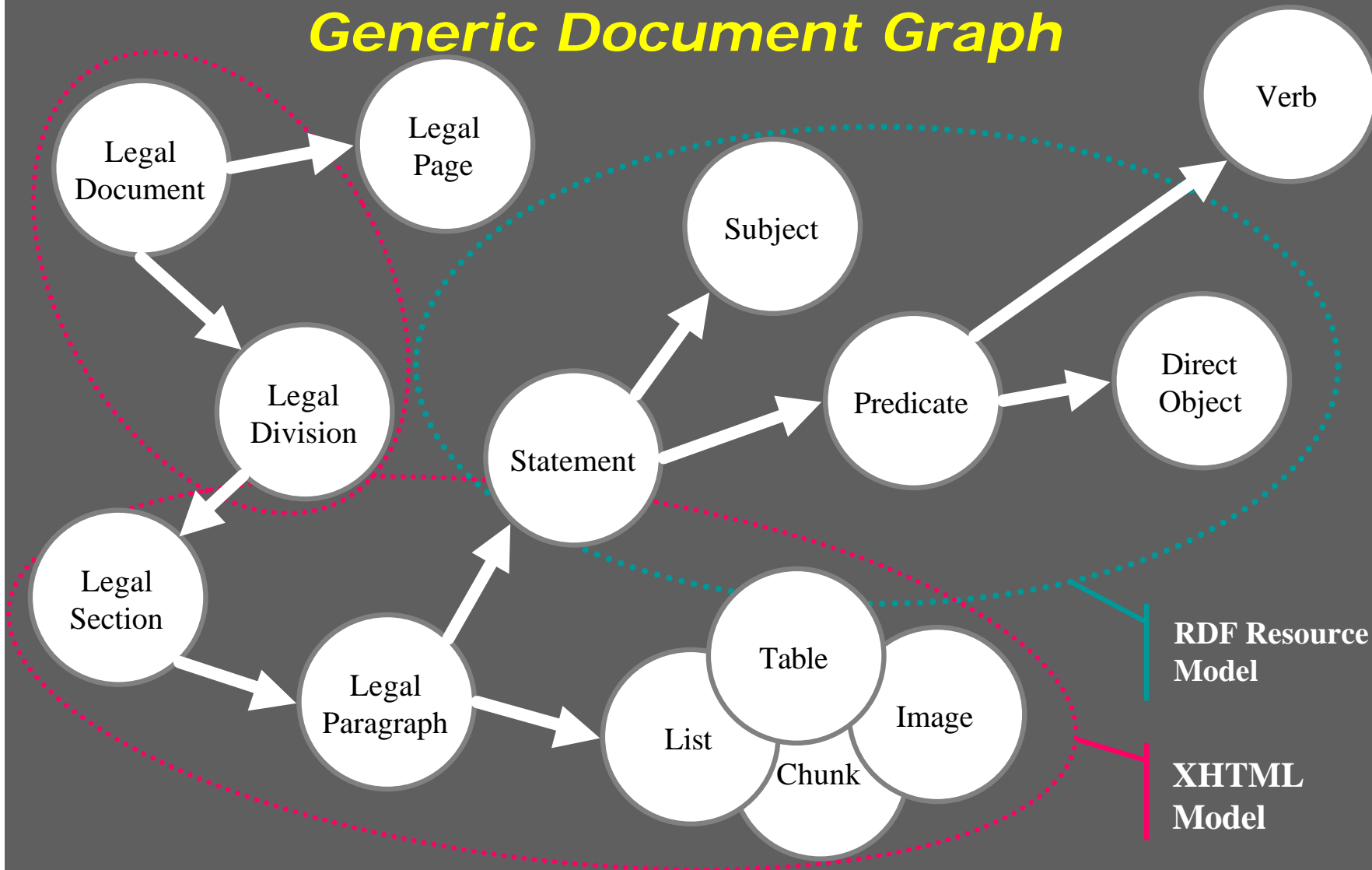
### Presentational Requirements

- Closely match hardcopy layout styling
- Document navigation semantics
- Concurrent models of document display

### Grammatical Requirements

- Sentence structure
- Basic text indexing
- Integration with Wordnet-type resources

# Generic Document Graph



## ***Core Ontology Models***

### Metamodels – types of resource classes

- Kellog Grammar Model
- Topic Model

### Facet Models – properties of resources

- Objective Quantities
  - numerics captured by ISO/SI Unit defined names
- Subjective Qualities
  - compatible with Aspect Programming techniques

### Resource Models – types of resources

- next slide

## *Resource Models*

### Base Class

- **Actor**
- **Role**
- **Scene**
- **Prop**
- **Drama**
- **Theme**

### Subclasses

Legal entities, groups, and organizations

Legal, commercial, public, and private roles

Location and time expressions

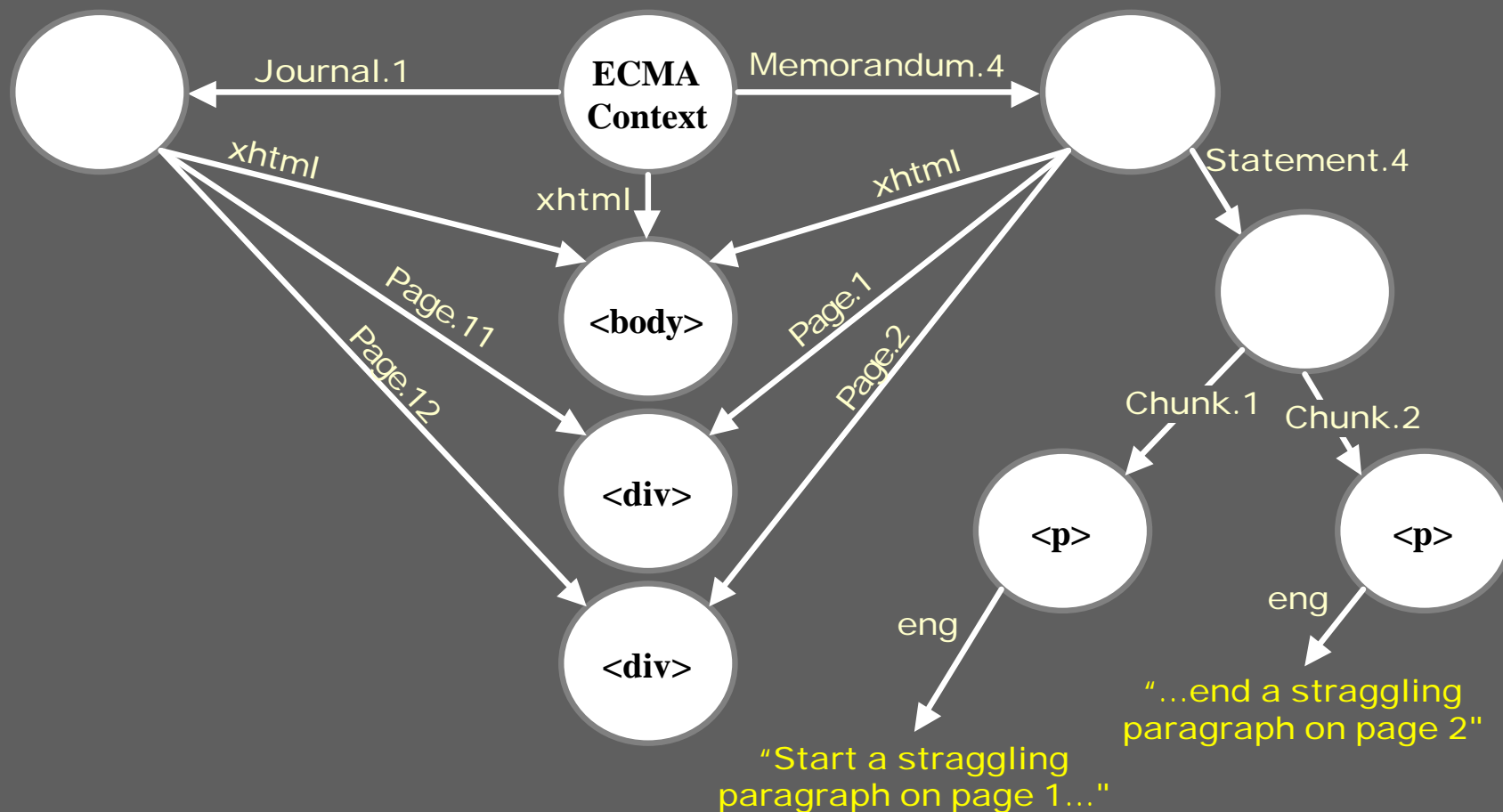
Legal intellectual and commercial products

Legal and illegal acts and activities (events)

Economy, justice, politics, etc.

# EC Journal Model

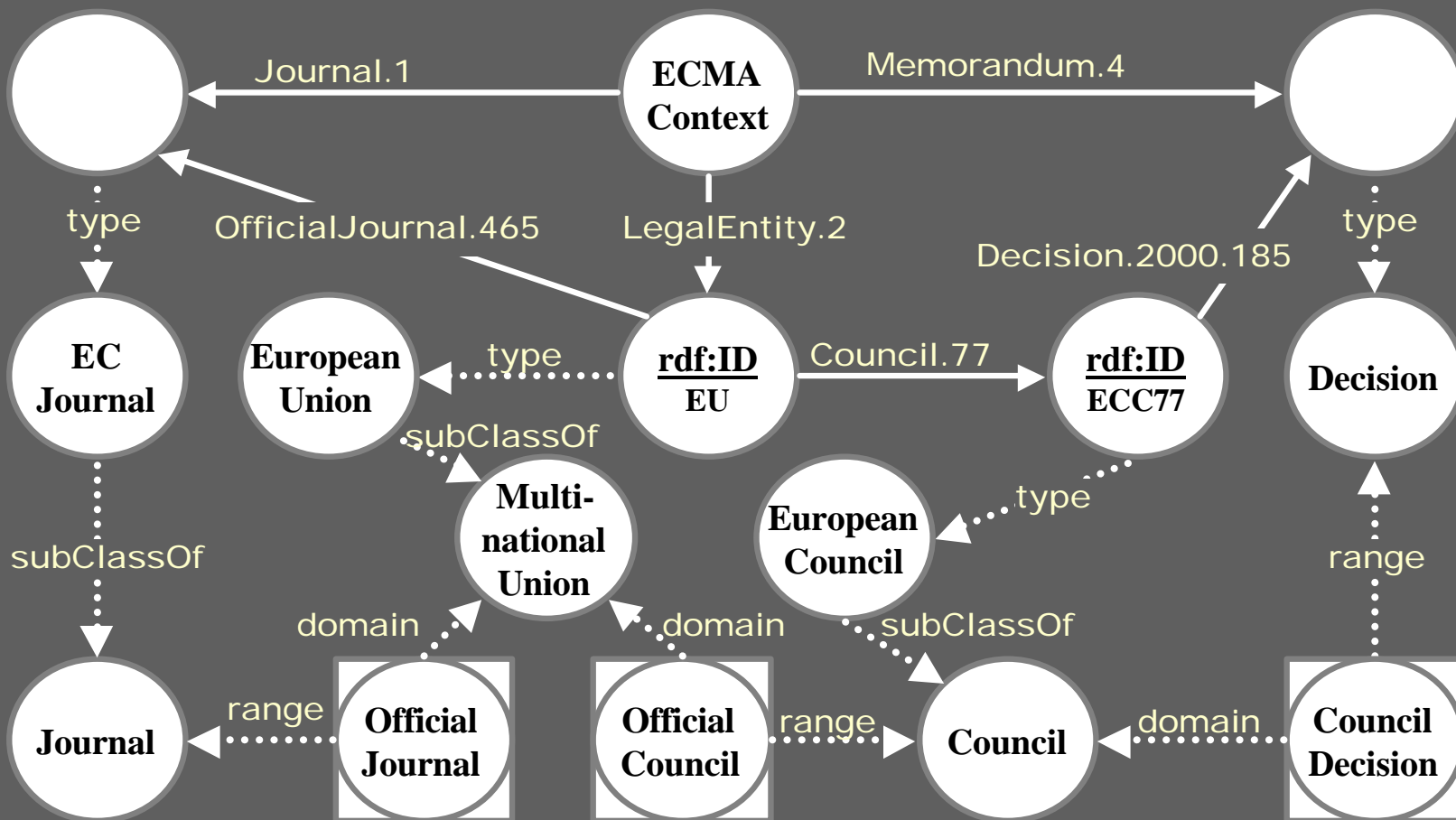
per: [Council Decision of 10 February 2004 \(OJ 465\)](#)







# EU Domain Metamodel



## *Recommendations*

- Let the contents of official documents define official domain models
- Design each document as a self-contained programming environment
- Distinguish between internal systems' schemas and exchange protocols
- Standardize for semantic tools, but build on current technology/systems
- Don't overload xml-protocol validation with semantic validation
- Support W3 Web Architecture's principles and good practices
- Market-test the ultimate solution and selected object and property names
- Prioritize web access to official documents by the entire population