Semantic Web TOOLS

Presented at the
DARPA DAML PI Conference
By
BBN Technologies
www.bbn.com
16 October 2003
Top Level Agenda

- **Background**
  - Cola Atkinson

- **DAML → OWL Conversion**
  - Troy Self

- **Tools Inventory & Assessment**
  - Howard Bender

- **DAML Web Site → Direction**
  - Cola Atkinson

- **Discussion**
  - Feedback Solicited
BACKGROUND
The goal of the DAML program is to create technologies that will enable software agents to dynamically identify and understand information sources, and to provide interoperability between agents in a semantic manner.

1. Create an Agent Mark-Up Language (DAML) built upon XML that allows users to provide machine-readable semantic annotations for specific communities of interest.

2. Create tools that embed DAML markup on to web pages and other information sources in a manner that is transparent and beneficial to the users.

3. Use these tools to build up, instantiate, operate, and test sets of agent-based programs that markup and use DAML.

4. Measure, via empirical experimentation, the productivity improvements provided by these tools.

5. Apply these tools to third party agent development, military-specific problems, and support for the intelligence community so as to evolve DAML technologies towards large-scale use.

6. Transition DAML to the commercial and military markets via partnerships with industrial and defense-related (Command & Control (C2) and intelligence) organizations.
Relevant Events

• DARPA DAML Objective 1 Achieved by the Community
  – Create an Agent Mark-Up Language
  – DAML → DAML+OIL → OWL

• W3C Candidate Recommendation
  – Last Call for documents 31 March 2003
  – Candidate Recommendation 18 August 2003

• OWL Conversion Directive from April 2003 PI Mtg
  – Convert DARPA funded DAML products to OWL within 60 days
  – Includes data, ontologies, tools & other applications

• Follow up message from DARPA PM 8/21/2003
  – “At the October PI meeting I would like all active program software and ontologies to be fully OWL.”

• BBN’s Recent Focus
  – Continue Language, Rules and Services work
  – Owl Conversion for www.daml.org
  – Tools Inventory and Assessment
  – Support planning for the outyears
DAML → OWL
Conversion Effort
Lessons Learned

Presented by Troy Self
tself@bbn.com
16 October 2003
Objective

• Comply with DARPA PMO requirement to convert all .daml files on www.daml.org to .owl files.
  – Convert all BBN cognizant ontologies, instance data, tools & applications
  – Develop and post lessons learned for benefit of daml community
  – Provide conversion support as required to community
Issues

- **Lots of .daml**
  - 172 Ontologies
  - 1027 Instance files
  - 15 Tools
  - 12 Agents
  - SONAT demonstration

- **Conversion tools lag behind evolving OWL Working Drafts**

- **Limited tool support for rdf:datatype**

- **Getting Apache to favor .owl over .daml**
Approach

• Start with the .daml files on or referenced on www.daml.org
• Use MINDSWAP’s owlConverter.pl to convert .daml files to .owl files
• Validate resulting files using the OWL Validator tool
• Identify recurring error types and modify converter to handle
• Iterate conversion process until only unique errors remain
• Manually fix unique errors and validate
.daml → .owl Conversion Process

Ontology Conversion Process

Error Analysis

Manual Fixes

OWL Validator

Instances

Ontologies

owlConverter.pl

Daml tools/services

owl ontologies

data.daml

ontologies.daml

Copy of www.daml.org

Data Instance Conversion Process

Instances

Validated owl files

Manual Fixes

Unique Errors

Recurring Errors

Converter Updates

SONAT Conversion Process

End-to-End SONAT TEST

OWL SONAT

SONAT Server

daml app

daml agents

owl app

owl tools & services

Update/Test Agents

Update/Testing App

Generate App Test Data

UpdateTools/Services

BBN Technologies, LLC
Outcomes

• All 1200 .daml files converted to .owl files
• Improved owlConversion tools
• All www.daml.org content is valid OWL
• Updated Apache’s Content-Negotiation Module
• Lessons learned document on web site
Lessons Learned

• Many errors in data files are due to errors in their respective ontologies.
  – Validate/Fix ontologies first
• The owlConverter cannot handle files with no line-breaks.
  – Ensure line-breaks exist first
• The addition of xml:base to RDF 2003 was very helpful
  – Added to all converted OWL files
• owl:imports wants an owl:Ontology
  – Add the owl:Ontology definition to data files if necessary
• Some errors are out of your control
  – Manually validate the file or instruct the validator to ignore
• Invalid DAML makes the process more difficult
  – Ensure that the DAML content is valid before converting
OBJECTIVES

• Start process of defining & building a user friendly web-based development resource environment to promote education, use & transition of Semantic Web technologies

  – Near-term goals:
    • Identify available OWL-compliant tools
    • Organize available tools into development process categories
    • Conduct initial gap assessment by category
    • Solicit additional input from the semantic web community

  – Long-term goals
    • Support DARPA in developing strategy to close the gaps
    • Implement CougaarForge-like developer resource environment

Conduct an inventory & initial assessment of tools available or referenced on www.daml.org
ISSUES

• Lots of items on the Tools page
• Not all OWL compliant
• Organized into a large number of ad hoc categories
• Quality is highly variable
  – Some tools are full products
  – Some tools are continuing research efforts
  – Some tools are one-time exercises or proofs-of-concept
• Some useful tools are not listed
• There is an obvious lack of easy-to-use tool “sets” that work seamlessly together
APPROACH
APPROACH

• Define a manageable set of tool categories based on development process workflows
• Inventory collection of tools available or identified on www.daml.org
• Sort inventory list into tool/process categories
• Develop simple use cases for comparing tools
• Top level assessment of tools in each category
  – OWL Compliant
  – Basic functionality
  – Usability
• Conduct initial gap assessment
• Present results & solicit feedback
Define Tool Categories

- Repository
- Editor
- Validation Tool
- Crawler
- Translation Tool
- Query Tool
- Annotation Tool
- API
- Mapping Tool
- Reasoning Tool
- Browser
- Visualization Tool
## Inventory Tools on DAML Web Site

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>AeroDAML</td>
<td>30</td>
</tr>
<tr>
<td>2</td>
<td>Algae</td>
<td>31</td>
</tr>
<tr>
<td>3</td>
<td>Articulation Service</td>
<td>32</td>
</tr>
<tr>
<td>4</td>
<td>BrownSauce</td>
<td>33</td>
</tr>
<tr>
<td>5</td>
<td>Chimaera</td>
<td>34</td>
</tr>
<tr>
<td>6</td>
<td>ConsVISor</td>
<td>35</td>
</tr>
<tr>
<td>7</td>
<td>cwm</td>
<td>36</td>
</tr>
<tr>
<td>8</td>
<td>D2RMAP</td>
<td>37</td>
</tr>
<tr>
<td>9</td>
<td>DAML API</td>
<td>38</td>
</tr>
<tr>
<td>10</td>
<td>DAML Crawler</td>
<td>39</td>
</tr>
<tr>
<td>11</td>
<td>DAML DB</td>
<td>40</td>
</tr>
<tr>
<td>12</td>
<td>DAML dotnetAPI</td>
<td>41</td>
</tr>
<tr>
<td>13</td>
<td>DAML Emacs Mode</td>
<td>42</td>
</tr>
<tr>
<td>14</td>
<td>DAML Markup Tool</td>
<td>43</td>
</tr>
<tr>
<td>15</td>
<td>DAML Semantic Search Service (SQIRE)</td>
<td>44</td>
</tr>
<tr>
<td>16</td>
<td>DAML Sidebar</td>
<td>45</td>
</tr>
<tr>
<td>17</td>
<td>DAML UML Enhanced Tool (DUET)</td>
<td>46</td>
</tr>
<tr>
<td>18</td>
<td>DAML Validator</td>
<td>47</td>
</tr>
<tr>
<td>19</td>
<td>DAML Viewer</td>
<td>48</td>
</tr>
<tr>
<td>20</td>
<td>DAML VisuaLinks</td>
<td>49</td>
</tr>
<tr>
<td>21</td>
<td>DAML XSLT Adapter</td>
<td>50</td>
</tr>
<tr>
<td>22</td>
<td>DAML+OIL Ontology Checker</td>
<td>51</td>
</tr>
<tr>
<td>23</td>
<td>DAML+OIL Plugin for Protege 2000</td>
<td>52</td>
</tr>
<tr>
<td>24</td>
<td>DAMLJessKB</td>
<td>53</td>
</tr>
<tr>
<td>25</td>
<td>dbview</td>
<td>54</td>
</tr>
<tr>
<td>26</td>
<td>Drive</td>
<td>55</td>
</tr>
<tr>
<td>27</td>
<td>dumpont</td>
<td>56</td>
</tr>
<tr>
<td>28</td>
<td>Euler proof mechanism</td>
<td>57</td>
</tr>
<tr>
<td>29</td>
<td>Excel to RDF converter</td>
<td>58</td>
</tr>
<tr>
<td>30</td>
<td></td>
<td>59</td>
</tr>
<tr>
<td>31</td>
<td></td>
<td>60</td>
</tr>
<tr>
<td>32</td>
<td></td>
<td>61</td>
</tr>
<tr>
<td>33</td>
<td></td>
<td>62</td>
</tr>
<tr>
<td>34</td>
<td></td>
<td>63</td>
</tr>
<tr>
<td>35</td>
<td></td>
<td>64</td>
</tr>
<tr>
<td>36</td>
<td></td>
<td>65</td>
</tr>
<tr>
<td>37</td>
<td></td>
<td>66</td>
</tr>
<tr>
<td>38</td>
<td></td>
<td>67</td>
</tr>
<tr>
<td>39</td>
<td></td>
<td>68</td>
</tr>
<tr>
<td>40</td>
<td></td>
<td>69</td>
</tr>
<tr>
<td>41</td>
<td></td>
<td>70</td>
</tr>
<tr>
<td>42</td>
<td></td>
<td>71</td>
</tr>
<tr>
<td>43</td>
<td></td>
<td>72</td>
</tr>
<tr>
<td>44</td>
<td></td>
<td>73</td>
</tr>
<tr>
<td>45</td>
<td></td>
<td>74</td>
</tr>
<tr>
<td>46</td>
<td></td>
<td>75</td>
</tr>
<tr>
<td>47</td>
<td></td>
<td>76</td>
</tr>
<tr>
<td>48</td>
<td></td>
<td>77</td>
</tr>
<tr>
<td>49</td>
<td></td>
<td>78</td>
</tr>
<tr>
<td>50</td>
<td></td>
<td>79</td>
</tr>
<tr>
<td>51</td>
<td></td>
<td>80</td>
</tr>
<tr>
<td>52</td>
<td></td>
<td>81</td>
</tr>
<tr>
<td>53</td>
<td></td>
<td>82</td>
</tr>
<tr>
<td>54</td>
<td></td>
<td>83</td>
</tr>
<tr>
<td>55</td>
<td></td>
<td>84</td>
</tr>
<tr>
<td>56</td>
<td></td>
<td>85</td>
</tr>
<tr>
<td>57</td>
<td></td>
<td>86</td>
</tr>
<tr>
<td>58</td>
<td></td>
<td>87</td>
</tr>
</tbody>
</table>
## Sort Available Tools into Categories

<table>
<thead>
<tr>
<th>APIs</th>
<th>Editors &amp; Annotation Tools</th>
<th>Repositories</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 Algae</td>
<td>13 DAML Emacs Mode</td>
<td>11 DAML DB</td>
</tr>
<tr>
<td>9 DAML API</td>
<td>17 DAML UML Enhanced Tool (DUET)</td>
<td>49 OntoMap.org</td>
</tr>
<tr>
<td>12 DAML dotnetAPI</td>
<td>23 DAML+OIL Enhanced Tool for Protege 2000</td>
<td>60 Parka-SW</td>
</tr>
<tr>
<td>26 Drive</td>
<td>40 Medius Visual Ontology Modeler</td>
<td>74 Sesame</td>
</tr>
<tr>
<td>37 Jena</td>
<td>41 MnM</td>
<td>82 Unicorn System</td>
</tr>
<tr>
<td>38 Jena Location Modification</td>
<td>44 OilEd</td>
<td></td>
</tr>
<tr>
<td>58 OWLP</td>
<td>46 OntoEdit</td>
<td></td>
</tr>
<tr>
<td>64 RDF API</td>
<td>47 Ontolingua</td>
<td></td>
</tr>
<tr>
<td>75 Source &amp; Dynamic Loading Exts for Jena</td>
<td>50 Ontomat</td>
<td></td>
</tr>
<tr>
<td>85 Wilbur RDF Toolkit</td>
<td>56 OWL Plugin for Protege 2000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>63 Protégé</td>
<td></td>
</tr>
<tr>
<td></td>
<td>66 RDF Instance Creator (RIC)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>69 RDFedt</td>
<td></td>
</tr>
<tr>
<td></td>
<td>72 SMORE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>73 SemTalk</td>
<td></td>
</tr>
</tbody>
</table>

## Browsers & Visualization Tools

<table>
<thead>
<tr>
<th>BrownSauce</th>
<th>Safari DAML</th>
<th>Source &amp; Dynamic Loading Exts for Jena</th>
</tr>
</thead>
<tbody>
<tr>
<td>4 BrownSauce</td>
<td>37 Jena Location Modification</td>
<td>85 Wilbur RDF Toolkit</td>
</tr>
<tr>
<td>5 Chimaera</td>
<td>44 OilEd</td>
<td></td>
</tr>
<tr>
<td>19 DAML Viewer</td>
<td>46 OntoEdit</td>
<td></td>
</tr>
<tr>
<td>27 dumpont</td>
<td>47 Ontolingua</td>
<td></td>
</tr>
<tr>
<td>30 Fenfire Loom</td>
<td>50 Ontomat</td>
<td></td>
</tr>
<tr>
<td>32 HyperDAML</td>
<td>56 OWL Plugin for Protege 2000</td>
<td></td>
</tr>
<tr>
<td>34 IsaViz</td>
<td>63 Protégé</td>
<td></td>
</tr>
<tr>
<td>42 nstordf</td>
<td>66 RDF Instance Creator (RIC)</td>
<td></td>
</tr>
<tr>
<td>43 Object Viewer</td>
<td>69 RDFedt</td>
<td></td>
</tr>
<tr>
<td>45 OntoDoc</td>
<td>72 SMORE</td>
<td></td>
</tr>
<tr>
<td>52 OntoRama</td>
<td>73 SemTalk</td>
<td></td>
</tr>
</tbody>
</table>

## Query & Reasoning Tools

<table>
<thead>
<tr>
<th>DAML Semantic Search Service (SQIRE)</th>
<th>DAMLJessKB</th>
<th>Euler proof mechanism</th>
</tr>
</thead>
<tbody>
<tr>
<td>7 cwm</td>
<td>24 DAMLJessKB</td>
<td>28 Euler proof mechanism</td>
</tr>
<tr>
<td>15 DAML Semantic Search Service (SQIRE)</td>
<td>28 Euler proof mechanism</td>
<td>31 F-OWL</td>
</tr>
<tr>
<td>24 DAMLJessKB</td>
<td>28 Euler proof mechanism</td>
<td>31 F-OWL</td>
</tr>
<tr>
<td>35 Java Theorem Prover (JTP)</td>
<td>48 Ontology Mapping</td>
<td>53 OpenCyc</td>
</tr>
<tr>
<td>48 Ontology Mapping</td>
<td>53 OpenCyc</td>
<td>62 Pellet</td>
</tr>
<tr>
<td>70 RACER</td>
<td>70 RACER</td>
<td>77 Surnia</td>
</tr>
<tr>
<td>78 SweetJess</td>
<td>78 SweetJess</td>
<td>80 TRIPLE</td>
</tr>
</tbody>
</table>

## Crawlers

<table>
<thead>
<tr>
<th>RDF Model Browser</th>
<th>VisioDAML</th>
<th>WebScripter</th>
</tr>
</thead>
<tbody>
<tr>
<td>67 RDF Model Browser</td>
<td>83 VisioDAML</td>
<td>84 WebScripter</td>
</tr>
<tr>
<td>81 Uml2Daml Converter</td>
<td>86 WSDL to DAML-S Converter</td>
<td>87 XML Schema to DAML Translator</td>
</tr>
</tbody>
</table>

## Repositories

<table>
<thead>
<tr>
<th>Aerodaml</th>
<th>Articulation Service</th>
<th>D2RMAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Aerodaml</td>
<td>3 Articulation Service</td>
<td>8 D2RMAP</td>
</tr>
<tr>
<td>14 DAML Markup Tool</td>
<td>20 DAML VisuaLinks</td>
<td>21 DAML XSLT Adapter</td>
</tr>
<tr>
<td>25 dbview</td>
<td>29 Excel to RDF converter</td>
<td>33 idl2owl</td>
</tr>
<tr>
<td>36 java2owl</td>
<td>39 KAON-REVERSE</td>
<td>51 OntoMerge</td>
</tr>
<tr>
<td>54 OWL Converter</td>
<td>55 OWL Genie</td>
<td>61 PDDL to DAML Translator</td>
</tr>
<tr>
<td>68 RDF Web Scraper</td>
<td>76 Spectacle:Server</td>
<td>79 Trellis</td>
</tr>
<tr>
<td>81 Uml2Daml Converter</td>
<td>86 WSDL to DAML-S Converter</td>
<td>87 XML Schema to DAML Translator</td>
</tr>
</tbody>
</table>

## Validation Tools

<table>
<thead>
<tr>
<th>ConsVISor</th>
<th>DAML Validator</th>
<th>DAML+OIL Ontology Checker</th>
</tr>
</thead>
<tbody>
<tr>
<td>6 ConsVISor</td>
<td>18 DAML Validator</td>
<td>22 DAML+OIL Ontology Checker</td>
</tr>
<tr>
<td>57 OWL Validator</td>
<td>16 DAML Sidebar</td>
<td></td>
</tr>
</tbody>
</table>
Develop Use Cases

• **Create an Ontology**
• **Create Instance Data**
  – Manually
  – Automatically (from a database)
  – From XML
  – Other
• **Query Over Distributed Data**
• **Create an Application that Employs Reasoning**
CREATE AN ONTOLOGY

Saved Ontologies

Repository

Editor

Validation Tool

Translation Tool

Annotation Tool

Crawler

API

 Ontology Info

Mapping Tool

Query Tool

Reasoning Tool

Browser

Visualization Tool

Ontology

Ontologies

Ontology Info
CREATE INSTANCE DATA AUTOMATICALLY

Unstructured Data ➔ Repository ➔ Translation Tool ➔ Mapping Tool
                      Editor ➔ Query Tool ➔ Reasoning Tool
                      Validation Tool ➔ Annotation Tool ➔ Browser
                                      Crawler ➔ API ➔ Visualization Tool
                                                     OWL Content
CREATE INSTANCE DATA FROM A DATABASE

Structured Data

Repository
Translation Tool
Editor
Query Tool
Validation Tool
Annotation Tool
Crawler
API
Mapping Tool
Reasoning Tool
Browser
Visualization Tool

OWL Content
CREATE INSTANCE DATA FROM XML
CREATE INSTANCE DATA FROM OTHER DATA
Question: Find all the organizations with grant money available for OWL research
Supposition: Granting organization data is in OWL form
CREATE AN APPLICATION THAT EMPLOYS REASONING

OWL Ontology & Data

- Repository
- Translation Tool
- Mapping Tool
- Editor
- Query Tool
- Reasoning Tool
- Validation Tool
- Annotation Tool
- Browser
- Crawler
- API
- Visualization Tool
TOOL ASSESSMENT APPROACH

- Categorize Tools
- Identify needed capabilities
- Define Test Criteria

Categories

Scenarios

Assess Tools

Assess Categories

Make Recommendations

Test Cases

Test Tools

Assess Tools
TOOL ASSESSMENT APPROACH

- Categorize Tools
- Identify needed capabilities
- Define Test Criteria

Categories

Scenarios

Assess Tools

Assess Categories

Make Recommendations

Test Cases

Test Tools

Categories

TOOL ASSESSMENT APPROACH

- Categorize Tools
- Identify needed capabilities
- Define Test Criteria

Categories

Scenarios

Assess Tools

Assess Categories

Make Recommendations

Test Cases

Test Tools
TOOL ASSESSMENT APPROACH

- Categorize Tools
- Identify needed capabilities
- Define Test Criteria

Categories

Scenarios

Test Cases

Assess Tools

Assess Categories

Make Recommendations

Test Tools
TEST CRITERIA

• **Editors & Annotation Tools**
  ‒ Can the editing tool automatically generate Class, Property, Restriction, and Instance markup tags in documents?
  ‒ Can the editing tool perform the following editing functions: run, quit, cut text and objects, paste text and objects, copy text and objects, insert text and objects, delete text and objects, undo, highlight syntax/visualization, rename a field, set/change a document namespace?
  ‒ Can the editing tool validate and write OWL data both from a pre-loaded file and from a newly created file?
  ‒ Can the editing tool accept both short-hand and non short-hand OWL notation for both ontology and data files?
  ‒ Can the editing tool read both large and small documents in both OWL and non-OWL formats?
TEST CRITERIA

• Browsers & Visualization Tools
  – Can the browser perform normal browsing functions such as Open, Close, Save As, Print?
  – Does the browser offer editing operations such as Copy, Select All, Find?
  – Can the browser load and visualize OWL ontologies and data?
  – Can the browser merge sources from different files that use the same ontology into a single browser window?
  – Can the browser navigate between URIs in the same directory and across namespaces?
  – Can the browser point to Class or Instance fields?
  – Can the browser show reified statements produced by an ontology?
  – Can the visualization tool perform refreshing and the operations advertised by the tool?
  – Can the visualization tool gracefully handle invalid data, inform the user, and request alternative actions?
  – Can the visualization tool view multiple files concurrently?
TEST CRITERIA

- **APIs**
  - Can the API read, parse and write OWL files?
  - Can the API modify (add, delete, change) OWL data?

- **Crawlers**
  - Can the crawler automatically search and find OWL information on the Web?

- **Query & Reasoning Tools**
  - Can the query tool search for information based on user-specified requests?
  - Can the reasoning tool accept an OWL file as a knowledge base?
  - Can the reasoning tool successfully use rules with the tool and information from the knowledge base?
  - Can the reasoning tool add new rules?
TEST CRITERIA

• Repositories
  – Can the repository initialize the data store upon first use?
  – Can the repository handle changes to the ontology?
  – Can the repository read data, write data, update data, and delete data?
  – Can the repository resolve concurrent access by multiple clients?
  – Can the repository perform the following reasoning: subClassOf transitivity, sameAs, transitiveProperty?
  – Can the repository provide for secure transactions?
  – Can the repository return the results of a simple query within 1 second?
  – Can the repository store data with greater than 1 million statements?
TEST CRITERIA

• Translation & Mapping Tools
  – Can the translation tool translate non-OWL to OWL for both data and ontologies?
  – Can the mapping tool indicate OWL-to-OWL equivalences and mapped relationships?

• Validation Tools
  – Can the validation tool recognize errors and point to the specific line in error?
  – Can the validation tool load ontologies created at another site?
  – Can the validation tool validate cardinality and property semantic values?
  – Can the validation tool process ontologies in either OWL-Lite, OWL-DL, or OWL-Full, depending on the specifications of the tool?
  – Can the validation tool validate ontologies written in either long form or short-hand form?
TOOL ASSESSMENT APPROACH

- Categorize Tools
- Identify needed capabilities
- Define Test Criteria

Categories

Scenarios

Assess Categories

Make Recommendations

Assess Tools

Test Cases

Test Tools
• Tool Category
• Tool Description
• OWL Compliance
• Include Ontologies, Data, or Both
• Requirements (operating environment)
• Installation Instructions
• Restrictions on Use
• Address of Web Site
• Open Source Status
• Location of Documentation
• Point-of-Contact
TOOL ASSESSMENT APPROACH

- Categorize Tools
- Identify needed capabilities
- Define Test Criteria

Categories

Scenarios

Test Cases

Assess Tools

Assess Categories

Make Recommendations

Test Tools
TOOL ASSESSMENT APPROACH

- Categorize Tools
- Identify needed capabilities
- Define Test Criteria

Diagram:

1. Categories
2. Scenarios
3. Assess Tools
4. Assess Categories
5. Test Cases
6. Test Tools
7. Make Recommendations
RESULTS
OWL Compliance Survey

APIs
- 2 Algae
- 9 DAML API
- 12 DAML dotnetAPI
- 26 Drive
- 37 Jena
- 38 Jena Location Modification
- 58 OWLP
- 64 RDF API
- 75 Source & Dynamic Loading Exts for Jena
- 85 Wilbur RDF Toolkit

Editors & Annotation Tools
- 13 DAML Emacs Mode
- 17 DAML UML Enhanced Tool (DUET)
- 23 DAML+OIL Plugin for Protege 2000
- 40 Medius Visual Ontology Modeler
- 41 MnM
- 44 OilEd
- 46 OntoEdit
- 47 Ontolingua
- 50 Ontomat
- 56 OWL Plugin for Protege 2000
- 63 Protégé
- 66 RDF Instance Creator (RIC)
- 69 RDFedt
- 72 SMORE
- 73 SemTalk

Browsers & Visualization Tools
- 4 BrownSauce
- 5 Chimaera
- 19 DAML Viewer
- 27 dumpont
- 30 Fenfire Loom
- 32 HyperDAML
- 34 IsaViz
- 42 n3tordf
- 43 Object Viewer
- 45 OntoDoc
- 52 OntoRama
- 59 PalmDAML
- 67 RDF Model Browser
- 83 VisioDAML
- 84 WebScripter

Query & Reasoning Tools
- 7 cwm
- 15 DAML Semantic Search Service (SQIRE)
- 28 Euler proof mechanism
- 31 F-OWL
- 35 Java Theorem Prover (JTP)
- 48 Ontology Mapping
- 53 OpenCyc
- 62 Pellet
- 70 RACER
- 77 Surnia
- 78 SweetJess
- 80 TRIPLE

Crawlers
- 10 DAML Crawler
- 65 RDF Crawler
- 71 scutter

Repositories
- 11 DAML DB
- 49 OntoMap.org
- 60 Parka-SW
- 74 Sesame
- 82 Unicorn System

Translation & Mapping Tools
- 1 AeroDAML
- 3 Articulation Service
- 8 D2RMAP
- 14 DAML Markup Tool
- 20 DAML VisuaLinks
- 21 DAML XSLT Adapter
- 25 dbview
- 29 Excel to RDF converter
- 33 idl2owl
- 36 java2owl
- 39 KAON-REVERSE
- 51 OntoMerge
- 54 OWL Converter
- 55 OWL Genie
- 61 PDDL to DAML Translator
- 68 RDF Web Scraper
- 76 Spectacle:Server
- 79 Trellis
- 81 Uml2Daml Converter
- 86 WSDL to DAML-S Converter
- 87 XML Schema to DAML Translator

Validation Tools
- 6 ConsViSor
- 18 DAML Validator
- 22 DAML+OIL Ontology Checker
- 57 OWL Validator
- 16 DAML Sidebar

<table>
<thead>
<tr>
<th>Tool Type</th>
<th>Name</th>
<th>No.</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>OWL Compliant</td>
<td></td>
<td>42</td>
<td>48</td>
</tr>
<tr>
<td>Not OWL Compliant</td>
<td></td>
<td>34</td>
<td>39</td>
</tr>
<tr>
<td>In Progress</td>
<td></td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>Not Applicable</td>
<td></td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>APIs</td>
<td>Editors &amp; Annotation Tools</td>
<td>Repositories</td>
<td></td>
</tr>
<tr>
<td>----------------</td>
<td>-----------------------------------------------</td>
<td>-----------------------</td>
<td></td>
</tr>
<tr>
<td>2 Algae</td>
<td>13 DAML Emacs Mode</td>
<td>11 DAML DB</td>
<td></td>
</tr>
<tr>
<td>37 Jena</td>
<td>47 Ontolingua</td>
<td>60 Parka-SW</td>
<td></td>
</tr>
<tr>
<td>58 OWLP</td>
<td>56 OWL Plugin for Protege 2000</td>
<td>74 Sesame</td>
<td></td>
</tr>
<tr>
<td></td>
<td>63 Protégé</td>
<td>82 Unicorn System</td>
<td></td>
</tr>
<tr>
<td></td>
<td>72 SMORE</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Translation &amp; Mapping Tools</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>8 D2RMAP</td>
<td></td>
</tr>
<tr>
<td>33 idl2owl</td>
<td>36 java2owl</td>
</tr>
<tr>
<td>51 OntoMerge</td>
<td>54 OWL Converter</td>
</tr>
<tr>
<td>55 OWL Genie</td>
<td>61 PDDL to DAML Translator</td>
</tr>
<tr>
<td>76 Spectacle:Server</td>
<td>79 Trellis</td>
</tr>
<tr>
<td>1 AeroDAML</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Browsers &amp; Visualization Tools</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>5 Chimaera</td>
<td>19 DAML Viewer</td>
</tr>
<tr>
<td>27 dumpont</td>
<td>51 OntoMerge</td>
</tr>
<tr>
<td>30 Fenfire Loom</td>
<td>54 OWL Converter</td>
</tr>
<tr>
<td>32 HyperDAML</td>
<td>55 OWL Genie</td>
</tr>
<tr>
<td>34 IsaViz</td>
<td>61 PDDL to DAML Translator</td>
</tr>
<tr>
<td>43 Object Viewer</td>
<td>76 Spectacle:Server</td>
</tr>
<tr>
<td>52 OntoRama</td>
<td>79 Trellis</td>
</tr>
<tr>
<td>59 PalmDAML</td>
<td>1 AeroDAML</td>
</tr>
<tr>
<td>67 RDF Model Browser</td>
<td></td>
</tr>
<tr>
<td>84 WebScripter</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Query &amp; Reasoning Tools</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>7 cwm</td>
<td>15 DAML Semantic Search Service (SQIRE)</td>
</tr>
<tr>
<td>15 DAML Semantic Search Service (SQIRE)</td>
<td>28 Euler proof mechanism</td>
</tr>
<tr>
<td>28 Euler proof mechanism</td>
<td>31 F-OWL</td>
</tr>
<tr>
<td>31 F-OWL</td>
<td>48 Ontology Mapping</td>
</tr>
<tr>
<td>48 Ontology Mapping</td>
<td>62 Pellet</td>
</tr>
<tr>
<td>62 Pellet</td>
<td>70 RACER</td>
</tr>
<tr>
<td>70 RACER</td>
<td>77 Sumia</td>
</tr>
<tr>
<td>77 Sumia</td>
<td>35 Java Theorem Prover (JTP)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Crawlers</th>
<th>Validation Tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>10 DAML Crawler</td>
<td>57 OWL Validator</td>
</tr>
<tr>
<td></td>
<td>6 ConsVISor</td>
</tr>
</tbody>
</table>

Available
Available Soon
SOME LESSONS LEARNED
LESSONS LEARNED

- Many tools require a substantial technical software background to operate
- Tool quality is widely variable
- Many tools have little documentation
- Many tools have too many dependencies
- Tools were not designed to work together

Current collection of tools does not provide the quality or breadth of support that will be required to meet DARPA’s objectives.
RESOURCES
RESOURCES

• **Web Sites**
    • Maintained by Mike Dean ([mdean@bbn.com](mailto:mdean@bbn.com))
  – OWL Implementation Page
    [http://www.w3.org/2001/sw/WebOnt/impls](http://www.w3.org/2001/sw/WebOnt/impls)

• **Tools Tutorial**
  – Semantic Web Tools Tutorial by Mike Dean & Jim Hendler

• **Papers**
  – Semantic Web Lifecycle Support
  – Semantic Web Tools for Enhanced Authoring
    by Robert McCool, Richard Fikes, Deborah McGuinness
www.daml.org
Site Direction

Presented by Cola Atkinson
catkinson@bbn.com
16 October 2003

SemWeb Central
Objective

• “Create a new and more professional open source site for program-developed software, including support for download, email lists, bug tracking, versioning, compilation, documentation, license management, sample application and so forth.”

“... make it easy for people who are not on the DARPA/EU payroll to test drive the semantic web ...”
Issues

• **Current web site** is not very user friendly
• Current web site does not conform to emerging practices of the open source community
• Current web site lacks a tools strategy that provides:
  
  “. . . *a solid, easy to use, and straightforward collection of tools that support typical OWL workflows.*”

• Current web site lacks a useful ontology registry
• Site administration is a currently a collateral duty with predictable results

“. . . Current web site is not a place for the uninitiated.”
Approach

- Create a centralized web resource for the semantic web community (with an appropriate site name)
- Set up central online repository for tool source code & community documentation.
- **GForge Web-based Solution**
  - Low barrier of entry for contributors
  - Commercial grade software life cycle management tools
  - Open-sourced (with trusted users having committer rights)
  - Proven collaboration tools
  - Project owners administer their own mailing lists, access control, Wikis, etc.
- Potentially use **CougaarForge** as a model
GForge Solution Would Include:

- **Project management**
  - Task/timeline control, user/group access privileges
- **Source code management & version control**
  - CVS, ViewCVS, Security (SSH)
- **Defect & enhancement request management**
- **Project web site for DAML funded development efforts**
- **Document (version) hosting/release**
- **Collaboration**
  - Mailman (mailing lists)
  - Wiki
  - Jabber (instant messaging)
- **Code sharing**
Additional Potential Capability
(InfoEther Service)

• **Project Dashboard**
  – Automated project build support
  – Automated execution of regression testing
• **Project/source quality analysis with PMD**
  – Custom rules for semweb community
• **SemWeb enhancements to GForge**
  – OWL representations of GForge data (People, Projects, Source code, Bugs, etc.)
  – Commit back into GForge project for widespread use
    • Sharp
    • Phillips
    • UltraLog
    • NASA
Near Term Task for Implementation

- Procure server & Internet domain names
- Configure OS/Security
- Install & configure GForge, Apache, Jabber, Mailman, Wiki, Postgres, CVS, etc
- Import existing tools of proper quality
- Build projects for documentation consolidation
- Move mailing lists from DAML.org
- Migrate relevant HTML content from DAML.org to appropriate projects under new service
- Reach out to the community to introduce them to the new resource capabilities
What Can You Do to Help

- Subscribe to and participate in related discussions at owl-forge@daml.org
- Help identify additional site functionality unique to this community
- Propose projects that will help fill the holes
- Plan to contribute tools and ontologies to the open source site
- Help promote site awareness
Discussion?
Error Analysis Statistics

File Count By Error Type

Number of Files

PARSE ERRORS  ERROR  WARN  INFO

Original  Modified  Final

3  2  0  12
18  11  11  62
39  30  79  62
6  6  3  6
The DARPA Agent Markup Language (DAML) Program officially began in August 2000. The goal of the DAML effort is to develop a language and tools to facilitate the concept of the Semantic Web. Mark Greaves is the DARPA Program Manager for DAML.

Why Use DAML?

A set of Roadmaps have been tailored for different user communities to help them access and use the contents of this site more efficiently. The Roadmaps should be particularly useful for new visitors to the DAML web site.

Site Links

- About DAML
- New User Roadmap
- daml-help mailing list
- Announcements
- RoDAML Newsletters
- DAML in the News
- Upcoming Events
- Downloads
- DAML Briefings
- DAML Publications
- Related Links
- Site Map
- Search the Site
- Past DAML Meetings
- DAML Researchers
- DAML Language (DAML–OIL)
- DAML Services (DAML–S)
- DAML Query (DQL)
- DAML Rules
- DAML Temporal Ontology
- DAML Ontology Library
- DAML Crawler
- DAML Tools
- DAML logos
- DAML Applications
- DAML Data Sets
- DAML Use Cases
- Homework Assignments

Recent Announcements

2003-08-19: The OWL Web Ontology Language is now a W3C Candidate Recommendation. See here for details.

2003-05-08: Mark Greaves has been named to succeed Murray Burke as DAML Program Manager, effective 6 June 2003.

2003-05-07: A beta release of DAML–S 0.9 (and OWL–S 0.9), a new release of DAML Services, is now available.

2003-04-21: Due to facility moves and network service changes at the host site, www.daml.org will be down briefly on 22 April between 5:30 pm to 7:30 pm EST and then again for an undetermined time between 8:00 AM and 7:00 PM EST on the 23rd of April. In addition, you may experience slower response times due to network sharing during the period between these two service times. If you experience any www.daml.org access or performance problems after the service periods above, please contact the daml webmaster. Thank you for your patience during this transition period. Brandon Amundson DAML Webmaster.

2003-04-07: An updated version of the DAML Query Language (DQL) is now available here.

2003-03-31: Roger Costello and David Jacobs have published an OWL Tutorial.

... all announcements
DAML Tools

Tools used by various participants in the DAML program.

Summaries

- DAML-specific tools
- DAML-specific tools by category
- All tools
  - All tools by category

Send additions/updates/corrections to tools@daml.org.

You may also view this list in XML, DAML and OWL formats or using Spectacle.

A wishlist of desired tools is also being maintained.

DAML-specific Tools

1. AeroDAML
2. Alag
3. Articulation Service
4. Chinaera
5. ContVisor
6. cnu
7. DIP MAP
8. DAML/XSLT Adapter
9. DAML+OIL Ontology Checker
10. DAML+OIL Plugin for Protégé 2000
11. DAML API
12. Damlae Translator Engine
13. DAML Crawler
14. DAML DB
15. DAML.dotnetAPI
16. DAML Rank Mode
17. DAML-senKB
18. DAML Mark-up Tool
19. DAML Semantic Search Service
20. DAML Sidebar
21. DAML UML Enhanced Tool (DUET)
22. DAML Validator
23. DAML Viewer
24. DAML Visual Links
25. dbview
26. Drive
27. dumpnet

<table>
<thead>
<tr>
<th>Category</th>
<th>Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>Annotation</td>
<td>MmM</td>
</tr>
<tr>
<td>Annotation</td>
<td>Oemat</td>
</tr>
<tr>
<td>Annotation</td>
<td>Trolit</td>
</tr>
<tr>
<td>Application Server</td>
<td>Spectacle Server</td>
</tr>
<tr>
<td>DAML Annotation</td>
<td>AeroDAML</td>
</tr>
<tr>
<td>DAML API</td>
<td>DAML API</td>
</tr>
<tr>
<td>DAML API</td>
<td>DAML-LessKB</td>
</tr>
<tr>
<td>DAML API</td>
<td>Java</td>
</tr>
<tr>
<td>DAML Browser</td>
<td>DAML Markup Tool</td>
</tr>
<tr>
<td>DAML Browser</td>
<td>DAML Sidebar</td>
</tr>
<tr>
<td>DAML Browser</td>
<td>HyperDAML</td>
</tr>
<tr>
<td>DAML Browser</td>
<td>OntoDoc</td>
</tr>
<tr>
<td>DAML Browser</td>
<td>Spectacle Server</td>
</tr>
<tr>
<td>DAML Crawler</td>
<td>DAML Crawler</td>
</tr>
<tr>
<td>DAML Crawler</td>
<td>RDF Crawler</td>
</tr>
<tr>
<td>DAML Editor</td>
<td>DAML Emacs Mode</td>
</tr>
<tr>
<td>DAML Graph Visualization</td>
<td>DAML Visual Links</td>
</tr>
<tr>
<td>DAML Graph Visualization</td>
<td>InaVir</td>
</tr>
<tr>
<td>DAML Graph Visualization</td>
<td>Object Viewer</td>
</tr>
<tr>
<td>DAML Graph Visualization</td>
<td>SemTalk</td>
</tr>
<tr>
<td>DAML Graph Visualization</td>
<td>VisioDAML</td>
</tr>
<tr>
<td>DAML Markup tool</td>
<td>Semantic Markup, Ontology and RDF Editor (SMORE)</td>
</tr>
<tr>
<td>DAML Transformation</td>
<td>DAML XSLT Adapter</td>
</tr>
<tr>
<td>DAML Validator</td>
<td>DAML Validator</td>
</tr>
<tr>
<td>DAML Viewer</td>
<td>DAML Viewer</td>
</tr>
<tr>
<td>DAML Viewer</td>
<td>PaleoDAML</td>
</tr>
<tr>
<td>DAML Viewer</td>
<td>Spectacle Server</td>
</tr>
<tr>
<td>Database Interface</td>
<td>dSview</td>
</tr>
<tr>
<td>Export</td>
<td>OWL Converter</td>
</tr>
<tr>
<td>Import</td>
<td>Excel to RDF converter</td>
</tr>
<tr>
<td>Import</td>
<td>PDDL to DAML Translator</td>
</tr>
<tr>
<td>Import</td>
<td>RDF Web Scraper</td>
</tr>
<tr>
<td>Import</td>
<td>Uni2DAML Converter</td>
</tr>
<tr>
<td>Import</td>
<td>Unicorn System</td>
</tr>
<tr>
<td>Import</td>
<td>XML2DAML Converter</td>
</tr>
<tr>
<td>Inference Engine</td>
<td>cwm</td>
</tr>
<tr>
<td>Inference Engine</td>
<td>Euler proof mechanism</td>
</tr>
</tbody>
</table>
A Potential Direction for SemWeb-Central

• The software development process is a potential first application area for the Semantic Web
• Open-source projects have an incredible amount of content that if transformed into OWL would be a major knowledge base for building high-impact SemWeb applications.
• Using OWL to represent open-source software project entities
  – People, Projects, Source files, Bugs, Tasks, etc
• Build SemWeb software project analysis tools that use these OWL documents
• Let the open-source contributors use SemWeb tools to facilitate their development efforts increasing the interest by software developers in SemWeb applications.
Welcome to CougaarForge!

Welcome to the Cognitive Agent Architecture (Cougaar) Open Source Project site. Cougaar has a Java-based architecture for the construction of large scale distributed agent-based applications. It is the product of a multi-year DARPA research project into large scale agent systems and includes not only the core architecture but also a variety of demonstration, visualization and management components to simplify the development of complex, distributed applications.

Here is the Cougaar FAQ, latest release, JavaDoc, and Wiki.

Latest News

Conference Papers are now online
- Todd Wright - 2003-10-02 12:25 - Cougaar
- Cougaar slides from JavaOne 2002 are now online in http://docs.cougaar.org
(0 Comments) [Read More/Comment]

Bug Fix Release 1.0.1
- Matt Abrams - 2003-09-03 16:27 - Cougaar IDE
A new minor release, 1.0.1, is available that fixes a bug related to launching cougaar nodes on Linux and Mac machines.
(0 Comments) [Read More/Comment]

Release 1.1
The Publisher functionality is now fully implemented and the BOL Guide is now included in the release.
(0 Comments) [Read More/Comment]

- 1.0 Alpha Release 2003-08-15 17:13
- Scheduled downtime August 8 2003 2003-08-06 15:34
- Initial Release 2003-07-31 15:15
- Initial Import 2003-07-29 14:10
- CougaarForge domain name change 2003-07-15 17:24
- Cougaar 10.2.1 is on CougaarForge 2003-06-24 15:57
- Scheduled downtime 5/20/2003 from 0700 to 0710 2003-06-19 16:08

[News archive]
Projects Dashboard

Software Build & Continuous Integration Dashboard
PMD… a Case Study

• NIST study:
  – $59.5B lost due to software defects annually
  – Software market $180B/year
  – Therefore 1/3 of value is lost
  – Increasing code quality would have major monetary impact

• Needed a source code quality analyzer
• Commercial tools did not meet requirements
• InfoEther built “PMD” for UltraLog program
• Open-Sourced on SourceForge.net
http://pmd.sourceforge.net/integrations.html

## Overview
- Installation
- Command line usage
- Ant task usage
- IDE plugin usage
- The PMD GUI
- PMD in the press
- Best practices
- Finding copied and pasted code
- Similar projects
- Credits
- What does "PMD" mean?

## Customizing PMD
- Compiling PMD
- How to write a rule
- How to make a rule set
- How it works

## For example
- Run PMD on a Sourceforge project
- PMD results for Apache projects
- Rule Sets

## Integrations with IDEs

### Summary

<table>
<thead>
<tr>
<th>IDE</th>
<th>Package</th>
<th>Maintainers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eclipse</td>
<td>pmd-eclipse</td>
<td>Tom Burke, Philippe Herlin</td>
</tr>
<tr>
<td>Emacs</td>
<td>pmd-emacs</td>
<td>Nascif Aousalh-Neto</td>
</tr>
<tr>
<td>JBuilder</td>
<td>pmd-jbuilder</td>
<td>Dave Craine</td>
</tr>
<tr>
<td>NetBeans/Sun ONE Studio</td>
<td>pmd-netbeans</td>
<td>Ole-Martin Mork, Gunnlaugur Thor Briem</td>
</tr>
<tr>
<td>JEdit</td>
<td>pmd-jedit</td>
<td>Jiger Patel</td>
</tr>
<tr>
<td>jDeveloper</td>
<td>pmd-idevelop</td>
<td>Tom Copeland</td>
</tr>
<tr>
<td>Maven</td>
<td>pmd-maven</td>
<td>Vincent Massol</td>
</tr>
<tr>
<td>Gel</td>
<td>pmd-gel</td>
<td>Tom Copeland</td>
</tr>
<tr>
<td>TextPad</td>
<td>N/A</td>
<td>Jeff Epstein</td>
</tr>
<tr>
<td>IntelliJ IDEA</td>
<td>N/A</td>
<td>Tom Copeland</td>
</tr>
<tr>
<td>Ant</td>
<td>N/A</td>
<td>Tom Copeland</td>
</tr>
</tbody>
</table>

### Eclipse

Integration into Eclipse is accomplished through the extensive Plugin architecture provided by Eclipse. It is written in Java, and has some nice features to it.

Eclipse Integration will fill the TODO list of the project with each Rule Violation that PMD detects. These violations are marked in the code with the Task Icon on the left hand side of the editor.