
We recommend you cite the published version.
The publisher's URL is:

Refereed: Yes

(no note)

Disclaimer

UWE has obtained warranties from all depositors as to their title in the material deposited and as to their right to deposit such material.

UWE makes no representation or warranties of commercial utility, title, or fitness for a particular purpose or any other warranty, express or implied in respect of any material deposited.

UWE makes no representation that the use of the materials will not infringe any patent, copyright, trademark or other property or proprietary rights.

UWE accepts no liability for any infringement of intellectual property rights in any material deposited but will remove such material from public view pending investigation in the event of an allegation of any such infringement.

PLEASE SCROLL DOWN FOR TEXT.
Research Question

‘To what extent is it possible to improve user-driven collaborative software development through interaction with diagrams and without requiring people to learn computer languages?’

Introduction

• Many computer literate people are experts in a particular domain, such as biology
• Enable computer literate scientists to model and visualise problems using software by minimising code writing
• User Driven Programming/Modelling

Problem Statement

• Software development is difficult for users - time, experience, access to programming tools
• Modelling with relationship tree and visualisation - possible to construct visualisation software for non-programmers
• Methodology needed for creation of systems to enable collaborative end-user modelling/visualisation
• Methodology would use visualisation to allow scientists to model, visualise and debate taxonomies/phylogenies

Generally Models are not Requirements

Visualisation - General Case

Methodology

Translation

• Ontology representation is translated into a computer model
• Ontology defines relationships between things, apply this to Semantic Web
• Relationships conveyed to a software model that evaluates them
• Taxonomy visualised and output to web

Requirements

• Requirements can then be integrated into the visualisation development processes
• Visualised Ontology drives visualised interaction
• Step by Step Translation
• This Research arises from work with engineers on their product process taxonomies

Implementation

Stepped translation - Ontology to Modelling and Visualisations

Step 1

Step 2

Step 3

Figure 3 - Stepped Translation and Visualisation

Results

<table>
<thead>
<tr>
<th>Improvement</th>
<th>Achieved By</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maintenance</td>
<td>Structuring and Translation</td>
</tr>
<tr>
<td>Extensibility</td>
<td>Structuring and Visualisation</td>
</tr>
<tr>
<td>Ease of Use</td>
<td>Visualisation, Interaction, and Translation</td>
</tr>
<tr>
<td>Sharing of Information</td>
<td>Shared Ontology and Interoperability</td>
</tr>
</tbody>
</table>

Recommendations

• Enable people to create software visually
• Create design abstractions familiar to domain experts e.g. diagrams for engineers
• Ensure interoperability using open standards
• Automate user to computer translation process

Conclusion

Can Widen programming participation by including computer literate non-programmers -

Enable interactive modelling/visualisation of problem