Common Features of Killer Apps: A Comparison with Protégé

Harith Alani, Kieron O’Hara, and Nigel Shadbolt

The 8th Int. Protégé Conference, Madrid, 2005
Killer Apps!

• What are they?
  – Highly transformative technologies that create new markets and wide spread patterns of behaviour

• The term “Killer App” was first used in the mid-1980s to describe Lotus 1-2-3, once demand for it become the major driver for buying IBM PCs
Semantic Web Killer App?

• A very common question:
  – *Where is the Killer App for the SW?*

• Many suggestions have been made:
  - *The semantic web IS the killer app!*
  - *Winners of Semantic Web Challenge must be KApps right?*
  - *Noo! It's Haystack*
  - *What about Adobe that supports RDF?!*
  - *Integration, integration, integration*
  - *No it's the integration idiot!*
  - *I think SW Services are the SW KApps!*
  - *NoofCORP is neat!*
Understanding Killer Apps!

- Killer apps don’t need advertising!
- Not any application can qualify as a killer!
- Applications must fulfil some requirements or possess some features to have the chance of becoming a Killer App
- Understanding those requirements and features might help building more successful applications
- A peek in the worlds of business and economy might help finding out what those features are
Features of Killer Apps

• Most of the features we found are pretty obvious! But it’s surprising how most applications ignore them!

• Protégé is used as an example of a successful application

• We compare between some of the general features of KillerApps, and those of Protégé

• Protégé is not a KillerApp for the Semantic Web, but it’s certainly a KillerApp for ontology editing
Superiority

• Must provide higher service quality (e.g., email vs snail mail, broadband vs dial-up)
  – What will your semantic web application give me that I cannot get elsewhere?
  – How is this better?

• Must show clear advantage over competitor products
  – Can I get the same functionality using other, cheaper, technology?
  – Can you demonstrate how difficult, if not impossible, it is to build this service using more traditional technologies?
  – Is the cost of migrating to this technology well justified?

• Protégé
  – Competitors include OntoEdit, Ontolingua, WebOnto, OilEd, KAON, etc
  – Comparison reported in Ontological Engineering, Springer 2004, showed many superior features of Protégé
Cost vs Benefit

- Cost-benefit analysis is essential
  - Cost of construction, conversion, maintenance, etc.

- KApps tend to be cheaper than alternative products. The more affordable it is, the more users it will attract
  - How costly it is to use this technology in the short and longer term?

- Many examples of free KApps; eg web browsers, search engines, chat software. They rely on their large user communities to generate value (eg from online ads, subscriptions to advanced services)
  - How can you generate value from your service/application?

- Protégé
  - Absolutely free!
  - For users, it helps to bring down costs of ontology editing and maintenance
  - For developers, apparently not much income has been generated
Community of Practice

- Metcalfe’s law: utility of a network equals approximately the square number of its users
  - Explains value of networked applications such as telephone, email, chat software
  - Core to the SW

- Must have potential to create a community of users
  - How can our application encourage community building?
  - How do you support, interact with, and listen to your users?

- Protégé
  - Over 27k registered users so far
  - Well attended conferences and busy mailing lists
  - Very good technical support for its user community
  - Users can build and share plugins
Open System

• A system draws additional value from other systems when its open to direct interaction with them
  – Reduces cost of data conversion and technology transfer
  – Propose supporting technology, rather than alternatives!

• Openness is at the heart of the SW
  – Will your application help to bring more RDF to the SW?

• Protégé
  – One of Protégé’s main advantages is its extendibility
  – Open source
  – Great value is added to Protégé from external, free, contributions (plugins)
Ease of Use

• Easy to use, non complex apps gets used more than others
  – No steep learning curves (imagine if you can’t use the Web before learning HTML!)
  – Don’t expect users to know RDF or anything about ontologies

• Protégé
  – Ease of use is one of the main focuses of Protégé
  – Graphical interface
  – Not much knowledge of RDF or OWL syntax is required
  – Important to facilitate OWL editing even further (eg ezOWL)
Personalisation

• Users are more royal to customisable services
  – But it has to be done properly!
  – Many of today’s killer apps have some level of personalisation (eg Amazon, AutoTrader, rightmove, eBay, pogo)

• Protégé
  – Customisable data entry forms
  – Some personalised settings are stored
  – What more can be offered?
Protégé: Further Issues
Scalability

• We are starting to see systems with small ontologies, but with a large number of instances
  – Eg CSAktiveSpace, winner of 2003 SWC, around 80 concepts, 25M triples
  – Flink, 2004 SWC winner, FOAF-like ontology, 35M triples

• Protégé
  – Main design goals were interoperability and ease of use
  – Some triple-stores are designed for scale; eg 3store, Sesame, and Kowari
  – We often see users building their ontologies in Protégé, then migrating them to another triple store for deployment
  – Could we have the best of both worlds in one system? Or get a better integration of Protégé with such stores?
Language Support

• Support for Semantic Web languages, such as RDF and OWL is crucial

• Protégé
  – Has always been amongst the first to provide support for such languages
  – Some Protégé-specific RDF syntax has been added for more detailed representations
  – As for OWL, some parsing incompatibilities can be spotted against Jena and SWOOP
• Online access to knowledge is essential for the Semantic Web

• Sesame, 3Store, and many other triple stores are designed for online querying and access using latest SW query languages such as RDQL and SPARQL

• Protégé
  – No direct support to these querying languages
  – No easy method for online access to knowledge base
  … that I know of!
Semantic Web Challenge

A Semantic Web Application has to meet the following minimal requirements.

1. First, the information sources used
   - should be geographically distributed,
   - should have diverse ownerships (i.e. there is no control of evolution),
   - should be heterogeneous (syntactically, structurally, and semantically), and
   - should contain real world data, i.e. are more than toy examples.

2. Second, it is required that all applications assume an open world, i.e. assume that the information is never complete.

3. Finally, the applications should use some formal description of the meaning of the data.

Besides the minimal criteria, a number of desires are formulated. The more desires are met, the higher an application can score. The desires are:

- The application uses data sources for other purposes or in another way than originally intended
- Using the contents of multi-media documents
- Accessibility in multiple languages
- Accessibility via devices other than the PC
- Other applications than pure information retrieval
- Combination of static and dynamic knowledge (e.g. combination of static ontologies and dynamic work-flows)
- The results should be as accurate as possible (e.g. use a ranking of results according to validity)
- The application should be scalable (in terms of the amount of data used and in terms of distributed components working together)

• Currently mainly focussing on the use of core SW characteristics

• Future calls might wish to include some of the KApp features discussed here
In Summary

• It’s difficult to predict where new killers will come from

• However, the history of killer apps makes it likely that any SW killers will have to provide:
  – a service that is not possible or practical under more traditional technologies
  – some clear benefit to developers, data providers, and end users with minimum extra costs
  – an application that becomes indispensable to a user-base much wider than the SW researchers community
El Fin!
Leave you with some funding ideas ....
advertising with Protégé ....
Member of eBay. Could be an individual person or a group (e.g., organisation, company).
Match Found:
You know Stuart who knows Linda who works with Susan who is a Protégé user. Then: you should date Susan!

Stuff about people!
I have noticed that you have a very small ontology. Even though size does not matter, but larger ontologies look more attractive. **not to worry!**

Please check our automatically selected ads to solve your problem.

Note: clicking on our ads will not affect your Protege project.
bring back the Nerd!

and create the Nerd’s Mini Mall ....
Or a mini mall for ontologies ....
Special Offers

1. We have a buy 1 get 2 free offer that lasts forever, so hurry before it's too late.

2. Use the AKT Ref Onto and you could win up to 200 EUR!
   Note: AKT is not responsible for any damages that might happen as a result of using this ontology.

3. Buy the Voodoo ontology and get 2 grams of lion hair absolutely free.

Thank you for using our Ontology Mini Mall