

A Task Oriented Approach to Delivery in Mobile Environments

François Paradis, Francis Crimmins, Nadine Ozkan

CSIRO Australia
Mathematical and Information Sciences
{Francois.Paradis, Francis.Crimmins, Nadine.Ozkan}@csiro.au

Abstract. The delivery of Internet content on mobile devices not only poses technological challenges, but also raises usability issues. In this paper, we propose an approach driven by tasks, where a task model is used to dynamically drive the interaction with the user and adapt the content to the device. We argue that such an approach is appropriate given the use of mobile devices for very specific tasks. We present an application to the journalistic domain: we have built a knowledge portal to assist a reporter writing an article. This portal demonstrates how a task model can be used for content adaptation on a mobile device.

Introduction

Recently there have been a number of studies on the user interface and interaction with mobile devices [BFJ01]. A key idea from these is that mobile devices are not simply limited versions of personal computers: they offer different functionalities and are used for different purposes [OT01]. They often are a good choice for “communication” tasks, personalisation [ADW01], or tasks where the context can play an important role [STM00]. In this paper, we propose to push the idea of “task” further, to achieve “task-oriented” delivery. Task models are often used for understanding the end-user and inform or evaluate the design [HSH90, SCS99], but in our case the task model is used to drive the interaction dynamically and to control content delivery. Our approach is based on the assumption that some tasks are better suited than others for mobile devices, and furthermore, that information seeking will be performed differently depending on the task.

In the next section we introduce a task model for the journalistic domain. We then present an implementation of our task-driven approach: a knowledge portal for a reporter.

A Task Model for the Journalistic Domain

Through several interviews with journalists and a literature survey, we defined four major tasks for a reporter writing a story: (we focus on information seeking)

- *Subject definition.* The purpose of this task is to find a theme for the article. The information gathering can include private sources, external sources, key contacts or organisations, etc.
- *Elaboration.* This task consists of collecting data useful to develop the story. Most of the mobility will occur either at this stage or during subject definition.
- *Writing.* At this stage the reporter organises the information and writes an article, usually from the office. Further details may be necessary; either to help put the story in context, or to help its flow.
- *Validation.* The objective of this task is to ensure that all the facts in the story are true and accurate.

These four major tasks are broken down into subtasks, thus defining a task ontology. Each task is defined by the context in which it takes place, its decomposition into subtasks, or the *action* to perform it (e.g. browsing or searching). We have chosen to represent tasks using XML. Below is an example:

```
<task name="archive search">
  <title>Search for an article in the news archives</title>
  <context><device refid="pc"/></context>
  <action><search><service name="P@NOPTIC"/>
    <collection>newsarchive</collection>
    <output>Rank, Title, Summary</output>
  </search></action></task>
```

The task “archive search” is typically done on a “pc”, during elaboration and validation (not shown here). It consists of a text search, which defines how to get parameters from the user (here a default search box to enter keywords), invoke an appropriate “service” or search engine (here, “P@NOPTIC”) and present results to the user (for this task, the article's title, source and summary should be displayed).

Demonstrator

The aim of our knowledge portal is to aid an Australian reporter writing a political story, by providing access to articles, facts and political sources. From the reporter's point of view, the knowledge portal is an access point to services, accessible through a Web browser on her iPAQ or desktop computer. The services are: live news (e.g. AAP MediaNet), document search (e.g. Hansard), World fact (e.g. CIA World Fact Book), image search (Ditto.com), issue search (aka expert search), and people search.

At the heart of the knowledge portal is the task engine, which organises and presents tasks using the task model and other configuration data. We now briefly discuss how the interaction and the content is adapted depending on the task.

Adapting Interaction with Task

As mentioned earlier, interaction with the user is driven by the task engine. This affects the way the user navigates to select a task and how she specifies her information need.

The portal builds menus from the task ontology. The reporter selects a particular task by choosing tasks and subtasks from the menus. On the PC, the task menus appear on a frame on the left of the screen, while forms and contents appear in the central-right section. Also note that the tasks themselves can differ depending on the device, although our aim was to keep the structure similar, so that user wouldn't be lost from one device to another. For the same reason, we have not made the menus context-dependent (e.g. depending on the task done last, the location of the reporter, etc.).

The task model also specifies how to build a form so that the reporter can query a service. For example, the task “archive search” presents a query text box, with hidden parameters “collection” and “output”, and an action URL pointing to the P@NOPTIC service (the actual URL is found in the service definition). Other tasks may use the same service with different parameters. For example, a recent article search also uses P@NOPTIC, but attempts to get articles from the last month.

Entering free text queries is cumbersome with the iPAQ, as with most mobile devices. Whenever possible, we provide the user with an alternate navigation or “query-by-click” based on the task at hand. For example, if the reporter leaves the search field empty in a recent article search, she will be presented keywords from the latest press releases; choosing one will launch a query with that keyword. Similarly, the user may browse a list of politicians for a people search, a list of countries for a World fact, etc.

Adapting Content with Task

Content adaptation in our portal is supported by *wrappers*, which send a query to an internal or external service and reformat the output. We implemented these wrappers using Norfolk [PV98], because it is well-suited to handle semi-structured information, and relatively robust to the changes in structure/contents.

A typical wrapper service is the P@NOPTIC service, which builds a query from the parameters it receives, sends a query to the P@NOPTIC search engine, parses the results, adapts them to the current task and device, and delivers an HTML page to the reporter. Parsing the results is necessary because the search engine results are returned with no structure, whereas the wrapper requires a rich structure with article titles, authors, URLs, dates, etc. The parameters to the search engine and the attributes shown in the results depend on the task. Consider the “archive task” task. The search engine will be invoked with parameter “collection” set to “newsarchive” (i.e. restrict the search to the archive of news articles) and the output will consist of the article titles and query-biased summaries (as long as the screen size allows). For the same kind of search, but in another context or task, different attributes might be displayed. For example, when looking for a recent article, showing the date will allow the reporter to better assess the relevance of the article (fig. 1).

In the case of *question-answering* tasks, another wrapper may be invoked on the first(s) answer(s) to attempt to get a factual result rather than just pointers to documents. For example, using the World Fact service to get geographical information about New Zealand, a query “New Zealand” is first sent to the P@NOPTIC service (on the CIA World Fact Book collection) to return a list of documents. A

wrapper is then invoked on the first document that matches a country description, and a summary containing population, area, etc. is returned along with the document pointers. The World Fact can produce different summaries depending on the task: for example, if one is looking for information regarding the economy rather than the geography, then the summary includes items such as the GDP and unemployment rate.

In our portal, when the user selects a document for viewing, rather than directly viewing the actual document, a summary page is brought up. This is quite valuable on a mobile device where it might not be possible to view the entire document. The summary page contains a document summary and possible actions on this document. For news articles, the summary is the first(s) paragraph(s) of the article, or, if available, the abstract. As we have seen above for the World Fact service, the document summary might depend on the task. Another example of a document summary that varies depending on the task is a query-biased summary [TS98], which excerpts relevant passages of the document based on the query. Common actions on a document include viewing the full document and printing. Once again, actions can vary depending on the task. For example, when viewing an article after a search, the reporter can search for similar articles.

Conclusion

A major hurdle in the delivery of Internet content to mobile devices is the lack of structure in the data and the lack of information about the user's needs (e.g. simple keyword searches). Our task-driven approach provides more *context* about the user's needs, and therefore can come up with better heuristics to “understand” the information and present it on a limited screen. Task modeling obviously adds to the cost of application design, but we feel that some of this effort would have to go into application design anyway (albeit in a more informal manner).

Although we have presented here an application to the journalistic domain, we

Rank	Document	Document Date
1.	The Age: High density residential plan rejected	15 March 2001
2.	The Age: Hulls' illness delays rights bill	20 March 2001
3.	The Age: She's our 19th commissioner and she's the first	17 March 2001

a)

Rank	Document
1.	Bracks angry at reform agenda block ... Premier Steve Bracks yesterday accused the opposition of crippling the government's reform agenda by blocking the upper house. By ADRIAN ROLLINS Tuesday 20 Marc ... 2001 Premier Steve Bracks yesterday accused the oppo the government's reform agenda by blocking three key ... McPhee, to advise on reform of the Legislative Council. The commission will report to Mr Bracks with its restructuring plans. ...

b)

Fig. 1. Document search varies depending on task. a) recent article search b) archive search.

have designed the portal to be extensible and flexible, so it could easily be used for other applications. In this phase of the project, we spent little effort on the user interface, our focus being on the task model. This would probably need to be improved before the approach could be tested with users.

Acknowledgments: We wish to thank Ken Yap (Canon CISRA) and Ross Wilkinson (CSIRO Mathematical and Information Sciences) for their guidance. The ideas discussed in this paper were developed by the KP team, which included (besides the authors): Stewart Baillie, Rowan Mckenzie, Anthony Bennett, Peter Wilkins, and Marilyn Specht (all from CSIRO Mathematical and Information Sciences).

References

- [ADW01] C.R. Anderson, P. Domingos and D.S. Weld, "Personalizing Web Sites for Mobile Users", in *Poster Proceedings of WWW10*, Hong Kong, May 1-5, 2001.
- [BFJ01] G. Buchanan, S. Farrant, M. Jones, G. Marsden, M. Pazzani and H. Thimbleby, "Improving Mobile Internet Usability", in *Proceedings of WWW10*, Hong Kong, May 1-5, 2001.
- [CHV01] Nick Craswell, David Hawking, Anne-Marie Vercoustre and Peter Wilkins, "P@NOPTIC Expert: Searching for Experts not just for Documents", in *Ausweb 2001*, 21-25 April, 2001.
- [HSH90] H.R. Hartson, Antonio C. Siochi and Deborah Hix. "The UAN: A User-Oriented Representation for Direct Manipulation Interface Designs", in *ACM Transactions on Information Systems*, vol. 8, n. 3, pp. 181-203, 1990.
- [OT01] Nadine Ozkan, Robert Tot. *Anticipating the Use of Mobile Video Communication*, CSIRO Mathematical and Information Sciences Technical Report nb. 01/124, July 2001.
- [PV98] F. Paradis, A.-M. Vercoustre and B. Hills, "A Virtual Document Interpreter for Reuse of Information", in *Proceedings of Electronic Publishing '98 (Lecture Notes in Computer Science 1375)*, Saint-Malo, France, pp487-498, 1-3 April, 1998.
- [SCS99] J.M.C. Schraagen, S.E. Chipman, V. Shalin (Eds.), *Cognitive Task Analysis*, Laurence Erlbaum Associates, 1999.
- [STM00] A. Schmidt, A. Takaluoma and J. Mäntyjärvi. "Context-Aware Telephony over WAP", *Personal Technologies*, Volume 4(4), pp225-229, September 2000.
- [TS98] A. Tombros and M. Sanderson, "Advantages of query-biased summaries in IR", in *Proceedings of the 21st ACM SIGIR conference*, pp2-10, 1998.