Presto: A Pluggable Platform for Supporting User Participation in Smart Workflows

Pau Giner, Carlos Cetina, Joan Fons and Vicente Pelechano
Centro de Investigación en Métodos de Producción de Software
Universidad Politécnica de Valencia
46022 Valencia, Spain
{pginer, ccetina, jjfons, pele}@pros.upv.es

***Abstract*—Mobile devices play a key role in reducing the gap between the physical and the digital world. The use of automatic identification capabilities can improve the user participation in business processes. However, the solutions available today for connecting physical elements with digital services are not suited for the support of smart workflows due to their limited customization capabilities. This makes it difficult to offer users the services that are relevant according to their role in the business process. This work introduces Presto, an extensible platform for supporting the user perspective in smart workflows. Presto is based on a pluggable architecture that can be extended to provide support for new tasks and technologies. In order to facilitate the development of these plug-ins, tools that automate the development process are also provided. Several Presto-based systems have been developed in order to validate the proposal.**

I. Introduction

The capabilities of mobile devices for the Automatic Identification of real-world elements (Auto-ID) are becoming widespread. Some examples are the use of on-board cameras for decoding barcodes or the use of Near Field Communication for reading RFID contactless tags. When leveraging these Auto-ID capabilities, daily activities become more fluid for users. For example, users in a library can borrow a book just by taking a picture of its barcode with their mobile devices.

This work is focused on providing support for Smart Workflows [5], which are business processes that cross the boundary to the physical world. Most of the research in Ubiquitous Computing has been focused on supporting the informal and unstructured activities that are typical of much of our everyday lives [1]. However, the smart workflow application domain is different, since it is focused on well-defined and structured activities. Considering the widespread availability of mobile devices, we propose their use to enable an effective participation in smart workflows.

This work introduces Presto, a customizable platform for mobile devices that supports user participation in smart workflows. On the one hand, Presto provides context-aware task management. In other words, it indicates to users the tasks that they can do according to (1) their role in the process, (2) the physical environment and (3) the process state. On the other hand, Presto allows the use of the real-world as the user interface for completing process tasks. The data demanded by a task is gathered by means of physical mobile interactions [3].

This work has been developed with the support of MEC under the project SESAMO TIN2007-62894 and cofinanced by FEDER.

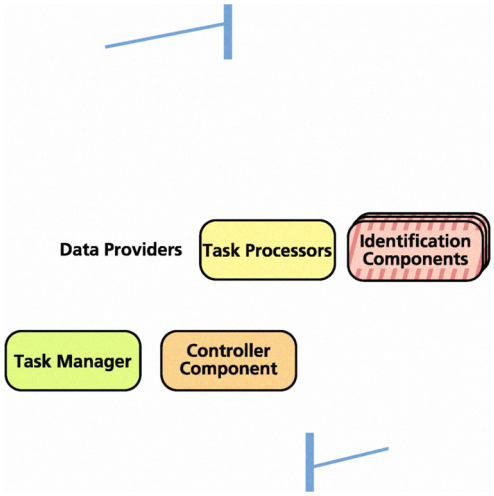

Fig. 1. Architecture component overview

II. A Pluggable Architecture

The architecture of Presto has been defined following an architectural process that decouples the technology-independent concepts from the actual technological solutions [4]. In this way, it is obtained a software architecture that is minimally affected by technological cycles and permits automation in the development. Presto architecture –see Fig. 1– is composed of the following components:

- **Task Manager.** This component acts as a buffer where pieces of pending work are waiting to be completed. It receives messages from external services such as Business Process Management systems and waits for components to process them.
- **Task Processors.** A *Task Processor* is defined to handle a particular type of task –e.g., borrowing a book from the library as in the example. *Task Processors* provide the users with the required information, services and interaction mechanisms for completing the task.
- **Identification Components.** *Identification Components* provide mechanisms for accessing the physical environment and transferring identifiers between physical and digital spaces by means of some Auto-ID technology.

Digital Object Identifier: 10.4108/ICST.MOBIQUITOUS2009.6929
http://dx.doi.org/10.4108/ICST.MOBIQUITOUS2009.6929

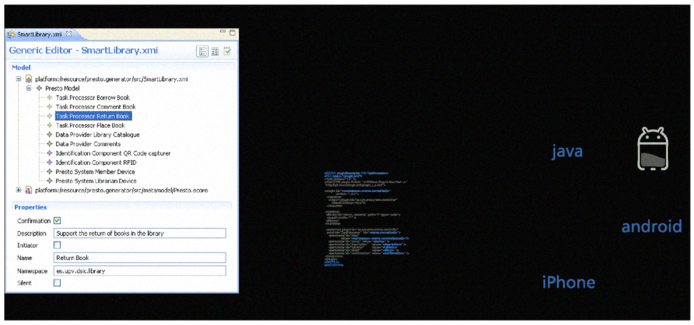

Fig. 2. Tool support for automating the development of Presto-based solutions

- **Data Providers.** *Data Providers* are in charge of transforming the identifiers provided by *Identification Components* into information that is relevant for the user. Each *Data Provider* represents a possible projection of a physical element in the digital world.
- **Controller Component.** The controller is in charge of coordinating the above components and defines the extension points for incorporating new components in a pluggable manner.

Presto is implemented using different Java-based technologies such as JavaFX, Java Plug-in Framework (JPF) and Service Data Objects (SDO). JavaFX is used for the definition of rich user interfaces, JPF supports the extensibility of the platform and SDO offers the data abstraction capabilities to integrate different data sources. Furthermore, the architecture can be easily translated to a different technology since the concepts defined are independent from the technology.

III. Tool Support

Tools have been defined in order to (1) specify Presto-based solutions making use of technology-independent concepts, and to (2) automate the development for the platform. The overall strategy of the approach is shown in Figure 2. Developers first specify Presto systems in a technology-independent manner. Then, this specification is taken as input for mapping rules that are suited to specific technologies.

In order to specify Presto models, the technology-independent architecture has been formalized using a metamodel. A metamodel captures the constructs that can be used to describe systems and the ways in which these constructs can be combined. Eclipse Modeling Framework[1] (EMF) facilities for the development of editors have been used to support the definition of Presto specifications. Figure 2 –left– shows an EMF-based editor for Presto-based systems.

We have defined mapping rules for the generation of the software artifacts that are required to support a Presto-based system for the target technology selected in this work. We have developed these mapping rules by means of the openArchitectureWare[2] (oAW) toolset. Although a fully functional system is not obtained by the mapping rules, the produced assets are useful to guide the development. In this way, developers can focus on the business logic avoiding to deal with infrastructure aspects.

[1]http://www.eclipse.org/modeling/emf/
[2]http://www.openarchitectureware.org/

IV. Applying the Proposal

In order to validate the architecture concepts we have developed Presto-based solutions to support user participation in two case studies that were defined in [2]. These case studies describe business process related to the monitoring of tools used for aircraft maintenance; and the protocol followed by a company in order to manage infrastructure incidences.

We also applied Presto to support business processes in a library. The *Smart Library* case study consists in the support for the book loan process in a library. Users by means of their mobile devices can manage their loans themselves. For returning books they only have to drop the books in an RFID-enabled "return box" without requiring the presence of a librarian. The book lending process is complemented by services enabling members to share book-related comments.

The **extensibility of the platform** enabled the creation of custom applications for the different business process participants just by combining plug-ins. The mobile client for the library members includes a *Task Processor* plug-in for each of the tasks they perform in the process -e.g., borrowing a book and comment on books– as well as an *Identification Component* plug-in to support QR Code-based identification. Librarians make use of the *Identification Component* that supports RFID technology for supporting a different set of tasks –e.g., place books to their shelf.

V. Conclusions

This work defines a platform for supporting user participation in smart workflows. This is a solution that can be adapted to the specific needs of each business process. The **extensibility of the platform** and the **flexibility of the physical-virtual linkage** make it possible to offer users the functionality that they require according to their role in the process.

The architectural process followed in this work decouples the architectural concepts from the particular technological choices, improving the evolution capabilities of the resulting systems. The architecture has been used in several case studies and has proven to be suitable for the automation of the development. For more detail on the Presto platform visit http://www.pros.upv.es/labs/projects/presto/

References

[1] G. D. Abowd and E. D. Mynatt, "Charting past, present, and future research in ubiquitous computing," *ACM Trans. Comput.-Hum. Interact.*, vol. 7, no. 1, pp. 29–58, 2000.
[2] P. Giner and V. Pelechano, "An architecture to automate ambient business system development," in *European conference on Ambient Intelligence (AmI 08)*, E. Aarts, Ed. Springer LNCS, 2008, pp. 240–257.
[3] E. Rukzio, K. Leichtenstern, and V. Callaghan, "An experimental comparison of physical mobile interaction techniques: Touching, pointing and scanning," in *8th International Conference on Ubiquitous Computing, UbiComp 2006*, Orange County, California, September 2006.
[4] M. Völter, "Software architecture patterns – a pattern language for building sustainable software architectures," March 2005. [Online]. Available: http://www.voelter.de/publications/index/detail3607290014.html
[5] M. Wieland, P. Kaczmarczyk, and D. Nicklas, "Context integration for smart workflows," in *PerCom*, 2008, pp. 239–242.

Digital Object Identifier: 10.4108/ICST.MOBIQUITOUS2009.6929
http://dx.doi.org/10.4108/ICST.MOBIQUITOUS2009.6929