

Reusability in Governmental Electronic Services

R

George Lepouras

University of Peloponnese, Greece

Costas Vassilakis

University of Peloponnese, Greece

INTRODUCTION

Reusability in the everyday life is the capacity of using existing objects or even concepts again in the same or other contexts. When applied to information systems reusability is the capability of using the same parts of an application in other applications or in other contexts. As defined by IEEE (IEEE, 1990), *reusability* is the degree to which a software module or other work product can be used in more than one computing program or software system. Although rapid application development environments such as MS Visual .NET™ (Visual .NET, 2005) and Delphi™ (Borland Delphi, 2005) have to some extent employed reusability of components to aid the fast implementation of software applications, the extent to which existing objects can be used again in the implementation of new software systems is usually limited to basic building blocks of the interface. One problem that hinders reusability of larger building blocks is the fact that once a component which encompasses a number of objects is built it also encapsulates algorithms (sets of well-defined instructions that perform a task) in the form of code that define the functioning of the component. Tight coupling between the program logic and the program code makes portability of the component between applications difficult, when even small modifications in the program logic are required. A second problem that hinders reusability is that even when the same component can be directly used between applications, recognizing that such a component exists and retrieving it, is not an easy task.

Reusability plays an important role in software development industry. If a set of well-defined components is available, valuable resources can be saved by utilizing again these components. As Rich Seely observes (2003) “as part of that cost-saving message, Gartner recommends vendors and consultants focus on reusability of Web services applications and components.”

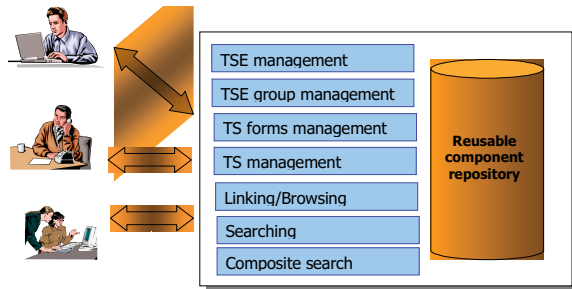
BACKGROUND

Electronic government is an area where a lot of development effort is lately devoted to. Electronic government aims to promote the use of electronic means, mainly electronic services to facilitate communication and interaction between civilians or businesses and the government. According to the European Commission (European Commission, 2000) “transaction services, such as electronic forms, are perceived as the future of electronic government.”

An electronic transaction service is usually the electronic counterpart of an existing service, implementing the business process logic involving the filling and submission of forms containing the necessary data, the processing of these data according to rules derived from laws and regulations and finally the return of a reply to the user.

In order to implement transactional electronic services, the collaboration between a number of experts is required. To this end, reusability can help by minimizing the effort needed for developing online transactional services. Electronic government offers a prominent area for the application of reusability since services offered to citizens from the same or different public authorities have common parts that could be reused between their electronic counterparts. However, in order to have effective reuse of components the two main problems previously described have to be solved. Back in 1995, Dusik and Katwijk (1995) identified the importance of a software development environment in which reuse, in various forms, would be an integrated element. As Gall, Jazayeri, and Klosch (1995) noted, the goal for reusability should be to create a software development process based on the “use” rather than the “reuse” of standard components. The approach used during the SmartGov project (2001) involved the design and implementation of an e-service development environment that would enable developers and domain experts to use components that they or other users had created to create their own transaction services.

Figure 1. Introducing the reusable component repository



SMARTGOV APPROACH

In contrast to simple information services, transaction services allow users to submit their data and in response, the public administration performs a service such as the issuing of a certificate or the tax clearance. Transaction services allow the user to perform common services online, implementing thus one of the main objectives of the electronic government, namely the facilitation of the interaction between civilians and businesses with the public authorities.

To be able to implement reusability effectively, one has to start by decomposing a transaction service to its basic building elements. In the first level, an electronic service consists of a number of forms the user is required to fill in. In the case of short documents, one form may be enough, where for lengthy documents, more than one form may be necessary. A form itself may comprise of several *areas*, and each area commonly contains individual *fields*, which are conceptually interrelated. The term *field* denotes the equivalent of a paper form field, which in the electronic service may be implemented as text input field, selection list, radio button group, etc.

For example, in a tax return form, distinct areas may be dedicated to collecting data regarding the taxpayer's personal details, income, and expenditures. Form fields are the individual elements that citizens need to fill in, either by direct typing of data in the area pertaining to the field (e.g., typing 13765 in the input area of the *Zip code* field) or by selecting one of the available field options (e.g., *Yes* or *No* for the *Do you own the house you live in?* field). Fields usually come complete with *labels* (i.e., descriptions of their purpose on the form). In some cases, the number of fields needed for some purpose cannot be predetermined. As Shaw pointed out (1995), 90% of most applications code goes into system or administrative code, like user interface code and back-end processing. Thus, reusability of objects combining the visual part of

the field and the inherent processing logic is crucial. Objects greatly increase software reusability and simplify the software development process (Fan, Stallaert, & Whinston, 2000).

As noted earlier for a reusability approach to be effective two issues have to be tackled: the tight coupling between the logic and the program code (i.e., between what we aim to achieve and the code that implements it) and the implementation of suitable mechanisms for retrieving components. The first issue can be solved by providing facilities to customize components without the need for completely rewriting the program code while the second can be solved by offering mechanisms for locating components pertinent to the tasks at hand and mechanisms for publicizing components to other user.

To facilitate these tasks, a *reusable component repository* is introduced, complemented with tools enabling users to browse, query, populate, and customize its contents. The repository approach is illustrated in *Figure 1*.

In the proposed approach, the idea of a basic building block is introduced. A *transaction service element (TSE)* is the equivalent of paper based form field. However, in contrast to a simple field, the TSE has more into it. A TSE can have a multilingual label, the field for inputting data, validation checks for checking the conformity of data to rules, instructions, documentation, or even legislation that applies on the field. The component repository holds *templates of TSEs* and *of groups of TSEs*. The transaction element management (TSE management) facility enables users to create templates of reusable TSEs. A reusable TSE template contains exactly the same information as an individual transaction service element, but is not directly used in transaction services. Instead, users create *instances* of this template and customize it to suit the needs of particular circumstances, since a TSE need not appear identical in all its occurrences. For instance, a TSE representing a person's VAT number may appear in a tax return form as "Taxpayer's VAT number" in the area for personal details, as "Landlord's VAT number" in the section in which housing expenses are declared and as "Employer's VAT number" in the incomes section. Besides the changes in labels, the validation checks associated with each occurrence may need to be customized (e.g., the Taxpayer's VAT number is always mandatory while the landlord's VAT number is mandatory only if housing expenses are declared; the employer's VAT number may need to be verified to correspond to an enterprise, rather than an individual). Once a TSE template has been instantiated and (possibly) customized, it can be used within a form of a transactional service. Note that customization is still possible after the establishment of the link between the instantiated TSE and the transactional service. A similar approach is used for TSE groups (i.e., users create in-

3 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/reusability-governmental-electronic-services/11690

Related Content

A Stakeholder Analysis of Business-to-Government Information Sharing: The Governance of a Public-Private Platform

Bram Klievink, Marijn Janssen and Yao-Hua Tan (2012). *International Journal of Electronic Government Research* (pp. 54-64).

www.irma-international.org/article/stakeholder-analysis-business-government-information/74814

E-Participation and Canadian Parliamentarians

M. Francoli (2007). *Encyclopedia of Digital Government* (pp. 697-702).

www.irma-international.org/chapter/participation-canadian-parliamentarians/11579

Modeling the Ranking of Evaluation Criteria for Cloud Services: The Government Organization Perspective in India

Kshitij Kushagra and Sanjay Dhingra (2018). *International Journal of Electronic Government Research* (pp. 64-82).

www.irma-international.org/article/modeling-the-ranking-of-evaluation-criteria-for-cloud-services/211203

Citizens' Adoption of Pay-to-Use E-Government Services: An Empirical Study

Amitabh Ojha, G. P. Sahu and M. P. Gupta (2013). *E-Government Services Design, Adoption, and Evaluation* (pp. 120-140).

www.irma-international.org/chapter/citizens-adoption-pay-use-government/73038

Visual Process Support to Assist Users in Policy Making

Dirk Burkhardt, Kawa Nazemi and Jörn Kohlhammer (2014). *Handbook of Research on Advanced ICT Integration for Governance and Policy Modeling* (pp. 149-162).

www.irma-international.org/chapter/visual-process-support-to-assist-users-in-policy-making/116661