

# End-User Development for Enterprise Resource Planning Systems\*

Andreas Roth, Stefan Scheidl

SAP Research  
Bleichstr. 8  
64283 Darmstadt  
{andreas.roth, stefan.scheidl}@sap.com

**Abstract:** Enterprise resource planning (ERP) systems are a good target for end-user development (EUD), as the typical process of implementing and maintaining ERP systems involving external IT experts is too costly and time consuming. Especially in the sub-domains of business intelligence applications and workflow management tools, we identify good opportunities of applying EUD techniques. Several architectural approaches may support the application of EUD in ERP.

## 1 Introduction

In this paper we present an overview of opportunities of end-user development in the context of enterprise resource planning (ERP) systems. ERP systems are software systems which support the planning of resources available in an enterprise (like means of production, human resources, etc.). ERP systems have been identified as a perfect target for end-user development (EUD) in [LPW06]. This paper aims at detailing this.

ERP systems are mainly standard software systems, reducing costs, risk, and delay compared to custom-built software. The typical process of implementing and using an ERP looks roughly as follows, employing the nomenclature of [MT00]:

1. *Chartering phase:* The software vendor ships a standard solution which best fits the needs of the customer (according to a precise requirements analysis). Typically industry specific versions are available.
2. *Project phase and shakedown phase:* Since standard software is delivered as a package designed to meet the general needs of a class of organizations [BHM01], the solution must be adapted more specifically to the customer's and its employees' needs. This is a collaborative effort involving business experts, local IT, and external IT consultants knowing about the standard ERP system.

---

\* This work was supported financially by the German "Federal Ministry of Education and Research" (BMBF, project EUDISMES, number 01 IS E03 C).

3. *Onward and upward phase*: The system is used and maintained. Whenever business requirements change, e.g. by new laws, new technical developments, changing market or customer requirements, the system must be adapted. This involves external IT consultants as well as internal IT services, since the software must manually be changed. According to [NGC03], this step consumes a fourth of the initial ERP implementation costs.

Big efforts are needed for steps 2 and 3. The adaptation, such as adding functionality or tweaking the UI, can become difficult because of the enormous complexity of state-of-the-art ERP systems. Moreover, the customization process is aggravated due to (cultural) misfits [SKT00]. Two general strategies can be applied in this phase [BHM01]: processes are adapted to the ERP system, then only little customizations are needed, or the software is precisely adapted to the process. An example of a company experiencing the high risk in these phases is documented e.g. in [Wo02].

As [MT00] remarks, adaptation during maintenance of an ERP system imposes difficulties: Upgrades and Add-ons to the ERP software override adaptations, or, if the adaptations should be kept, upgrades and add-ons could even be made impossible. User-written add-ons are expensive, often experiencing low performance, and are hard to get rid of [EB05]. It is furthermore a viable hypothesis that large companies with higher management expertise, fine grained and standardized processes can easier adopt processes pre-defined by ERP than small companies (SMEs) with more human centric, less rigid processes. They will be more in need for customization, though having less financial resources for required external IT expertise.

## 2 Opportunities for EUD in ERP

In the following we present a selection of ERP sub-domains which may constitute good examples of an increased potential towards EUD.

### 2.1 Customizing Business Intelligence Applications

Business Intelligence (BI) applications are tools for analyzing data for the purpose of making better and faster business decisions. As [EH05] remarks, in the age of “information democracy” every employee is (potentially) a consumer of BI applications, using business user-oriented easy-to-use user interfaces. Yet building such applications is not easy at all.

Since the details of business decisions are specific to each company, there is no complete analytic application out of the box. Companies spend thus much time on the customization and extension of commercial products. [Ec05] reports that, on the average, a third of the final analytic application were manually customized using e.g. SQL and that adaptation took 7.5 months on average. This seems much too long, if one considers the purpose of BI applications aiming at fast decisions.

EUD techniques [LPW06], in particular visual programming, example based approaches, but also collaborative approaches should be able to drastically improve this adaptation.

## 2.2 Creating and Managing Workflows

Workflows automate business processes which involve the movement of documents, information or tasks between participants. The importance of workflows is widely accepted and there are software systems which allow for visually creating workflows. Though these tools can be classified as end-user friendly, for a typical end-user, workflows appear as an unnaturally rigid schema, out of which it is impossible to escape.

We see a high potential for applying EUD techniques to offer greater customizability to workflow management, such as the inclusion of collaborative, ad-hoc activities. Suitable candidate EUD techniques must be highly adaptive [LPW06], that is, in this case, workflow models are created just while the process is in progress.

## 2.3 Further Opportunities

It is a viable part of ERP systems to provide a structured interface via the web, usually called *enterprise portal*. EUD applies here by allowing users to personalize workspace, locality, language, layout, etc., providing role-based content access, and by allowing for easier provision of content. Much has been realised in recent portal software, but there is still potential for more adaptive techniques for tailoring portals.

When basic conditions of business change, even the core business logic should be faster adaptable than the traditional customization process suggests. Business experts knowing well about required changes, but not about the needed technical realisation, should thus be enabled to perform the changes as much as possible on their own. Business rules [Ro03], capturing business logic in a separate and understandable way, are a promising approach to achieve this goal. In the large ERP suites business rules have, to a large extent, not arrived at their ultimate potential yet.

## 3 Promising Architectural Approaches

The (quite general) remedy to enable EUD is to provide appropriate interfaces for end-users which they can well enough handle. Interfaces suited to end-users should necessarily not provide the full range of handles available to a normal developer. Instead they should, on the one hand side, be less powerful by offering reduced, simplified possibilities of modifications, thus reducing the cognitive burden on the end-user. On the other hand, they should be more powerful, consisting of large, high-level building blocks, which reflect business specific patterns. The better the needed simplifications and patterns are predicted, the better will an end-user be able to tailor her or his system. Doing this prediction for the ERP domain remains future work. In the following, we rather investigate recent architectural approaches which seem to be promising instruments to realise them.

Identifying large building blocks corresponding to regularly occurring business specific patterns is greatly advised by *component-based*, and more recently by *service-oriented architectures*. The latter paradigm relies on loosely coupled services being the founding components of business processes. After having established a service-oriented structure, services can be orchestrated to business processes, or an orchestration can be modified according to changed business needs. Because developers now deal with large building blocks which correspond to a certain business domain, even business experts with little technical expertise can, with suitable tool support, be expected to compose services.

Other relevant approaches to EUD in ERP systems are *model driven* and *model based development*. The goal is to let end-users design models, preferably in a graphical notation they are familiar with, and to automatically generate code from this model. A prerequisite is that these models can easily be captured by domain experts. Thus specifically adapted languages are needed to accomplish this, a goal which can be reached by employing *domain specific languages*.

## 4 Conclusions

We have presented an overview of the field of ERP software with respect to EUD. We have identified domains in which EUD seems a promising approach to decrease development and maintenance costs and time. These are in particular business intelligence and workflow management systems. Several recent architectural approaches seem to be promising candidates for realising these opportunities. Working out the details and building prototypes is subject of ongoing work.

## Literature

- [BHM01] Brehm L.; Heinzl, A.; Markus, M.L.: Tailoring ERP Systems: A Spectrum of Choices and their Implications. HICSS, 2001.
- [EB05] Experton Group; Bohr Consulting: Wirtschaftlichkeitsanalyse von SAP Systemen, 2005.
- [Ec05] Eckerson, W.: Development Techniques for Creating Analytic Applications, TDWI Report Series, March 2005.
- [EH05] Eckerson, W.; Howson, C.: Moving BI to the Enterprise. DM Review, September 2005.
- [LPW06] Henry Lieberman; Fabio Paternó; Volker Wulf (Eds.): End User Development, Springer, Dordrecht 2006.
- [MT00] Markus, M.L.; Tanis, C.: The enterprise system experience: from adoption to success, in Zmud, R.W. (Ed.): Framing the Domains of IT Management: Projecting the Future Through the Past. Pinnaflex Educational Resources, Inc., Cincinnati, 2000; pp. 173-207.
- [NGC03] Ng, C. S. P.; Gable, G.G.; Chan, T.: An ERP Maintenance Model. HICSS, 2003.
- [Ro03] Ross, R. G.: Principles of the Business Rule Approach. Addison-Wesley Longman Publishing Co. Inc., 2003.
- [SH00] Scheer, A.W.; Habermann, F.: Making ERP a Success. Commun. ACM, 4/2000; pp. 57-61.
- [SKT00] Soh, C; Kien, S. S.; Tay-Yap, J.: Enterprise resource planning: cultural fits and misfits: is ERP a universal solution? Commun. ACM. 4/2000; S. 47-51.
- [Wo02] Worthen, B.: Nestlé's ERP Odyssey. CIO Magazine, May 2002.