Production Management and Process Control, Vol. 36, 2022, 178–184
https://doi.org/10.54941/ahfe1001633

Low-Code Development Platform for Business Process Automation: Aurea BPM

Robert Waszkowski

Military University of Technology, Faculty of Cybernetics, Kaliskiego 2,
00-908 Warszawa, Poland

ABSTRACT

The article introduces overall low-code platform architecture for business process automation. The architecture assumes the division of the platform into subsystems organized in the substantive layer, as well as in the layers of system management and integration. The substantive layer consists of the Design and Development, Metadata Repository, Deployment, and Runtime subsystems. The most important of them are Design and Development and Runtime. The former allows for the development of process models in the form of diagrams and metadata, while the latter is responsible for running applications in the target environment and delivering them to end users. Models developed in the Design and Development subsystem are stored in the Metadata Repository subsystem, which is additionally responsible for providing users with controlled user access to system resources. The article presents the architecture of the solution and the individual steps of implementing an exemplary business process.

Keywords: Human-computer interaction, Process optimization, Smart production, Low-code development platform, Business processes, Aurea BPM

INTRODUCTION

Figure 1 shows the overall Low-code Platform architecture. The architecture assumes the division of the platform into subsystems organized in the development layer, as well as in the system management and integration layers (Sanchis, 2019).

The development layer of the platform consists of the Design and Development, Metadata Repository, Deployment and Runtime subsystems. The most important of these are definitely Design and Development and Runtime (Woo, 2020). The first allows the development of models in the form of diagrams and metadata, while the second one is responsible for launching the application in the end-user environment.

Models developed in the Design and Development subsystem are stored in the Metadata Repository subsystem, which is additionally responsible for providing controlled user access to model and metadata resources (Braude, Bernstein, 2011).

The developed and validated models are prepared and turned into executable form in the Deployment subsystem. The Runtime subsystem uses such prepared models to deliver the ready-to-use application to end users.

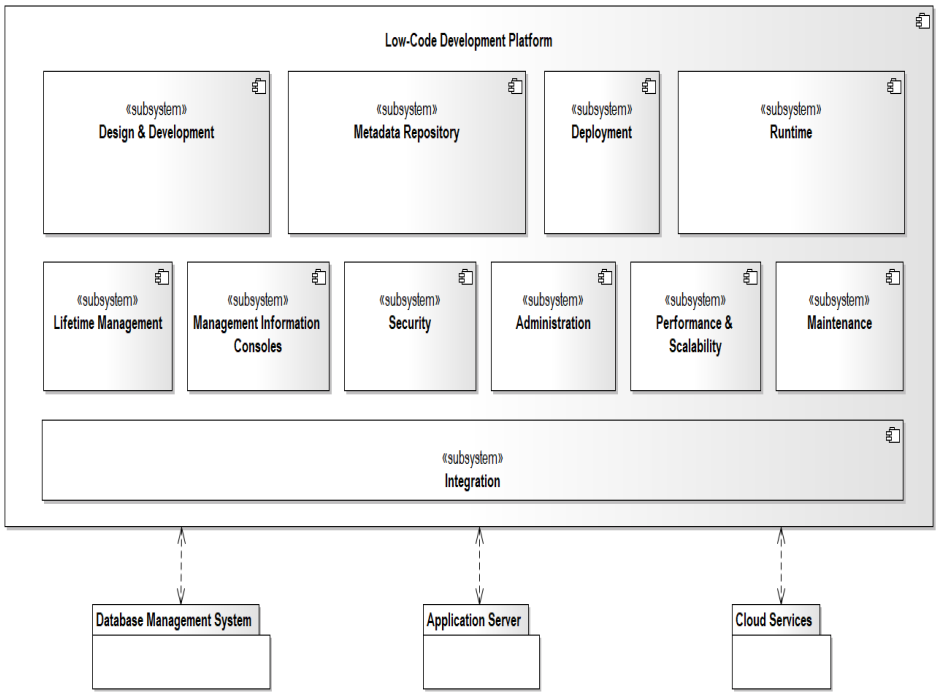

Figure 1: Overall low-code platform architecture. Source: own elaboration.

DESIGN AND DEVELOPMENT

Figure 2 presents all modules of the Design and Development subsystem. These modules can be divided into two groups: mandatory and optional. The required modules are BPMN Process Modeler, Process Data Modeler, Business Rules, Role Management, UI Designer. Optional modules that occur only in some Low-code Development Platforms (McKendrick, 2017) are Visibility Matrix, Multi-faceted Modeler or Mathematical Model Integrator.

BPMN Process Modeler is a required component for all LCDPs designed to build applications based on business process models (Waszkowski, 2018). It is used for developing and storing business process models. Process Data Modeler is used to model the data used by the application to support business processes (Nowicki, 2016). Business Rules define aspects of a business activity by introducing certain constraints into the system, usually resolved as true or false (Waszkowski, Kiedrowicz, 2015). They are used to ensure the correctness of business structures or to control or influence business behaviour.

Role Management module is an important element that complements the process model designed with the use of the process modelling notation. It allows defining the necessary mappings between the pools and swimlanes available in the BPMN notation and the company's organizational structure.

UI Designer, as the module responsible for user interface modelling, is responsible for preparing screen forms for handling business process tasks (Guerrero, Lula, 2002). Each human task must be handled by an appropriate employee. To handle such tasks, screen forms are used. Thank to them

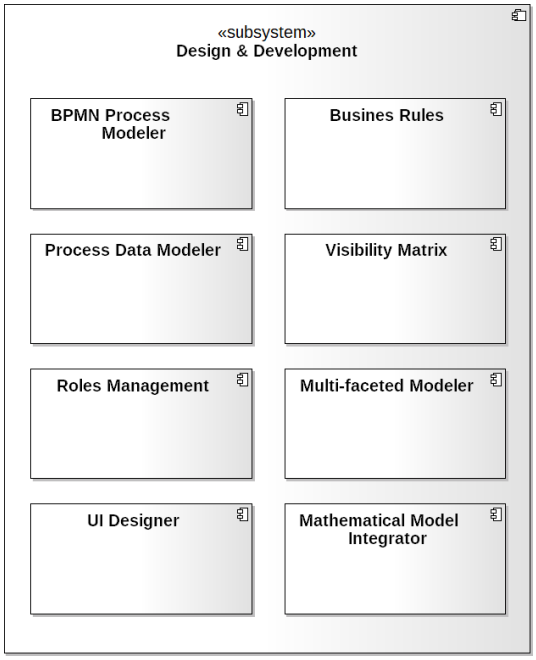

Figure 2: Design and development subsystem.

the user has access to the data of the handled instance of the business process, can modify this data and documents (Kiedrowicz, 2016), and has access to a number of actions related to the handling of tasks. The UI Designer is responsible for preparing the end-user application.

METADATA REPOSITORY

Figure 3 presents all modules of the Metadata Repository subsystem. It includes such modules as: Processes, Tasks, Lists (or Registers), Scripts, Notifications, Dictionaries, Document Templates, and Filters.

The process model management module (Processes) is responsible for:

- Storing process definitions modelled in the BPMN Process Modeler,
- Managing access to process definitions in accordance with user privileges,
- Managing process versions,
- Storing documentation of processes.

The Tasks module is responsible for storing and editing task definitions in processes, data definitions, roles, and information about data visibility in the user interface.

Data registers are freely definable named sets of records which are stored in the system in the form of tables. The Scripts management module (Scripts) allows to add, update, and delete script definitions, as well as execute the scripts.

The Notifications module is responsible for configuring notifications depending on the occurrence of specific events, while the Dictionaries module is responsible for storing internal and external dictionary definitions.

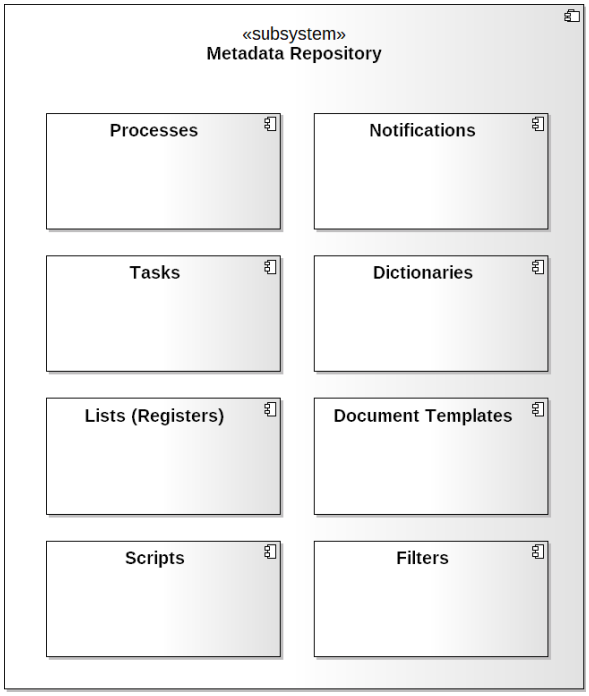

Figure 3: Metadata repository subsystem.

DEPLOYMENT

Figure 4 presents all modules of the Deployment subsystem. It combines the following modules to support end-user application generation: Application Generators, Data Model Generator, System Configuration Generator.

The Application Generators module groups together programs for generating end-user applications to handle tasks. These include such programs as user interface generator, business logic generator, forms generator, registry generator, report generator, etc.

Data Model Generator is the module responsible for transforming data models into database objects. The System Configuration Generator is responsible for generating the system configuration for deployment in a specific customer environment. It allows to configure system roles, system functionalities, menus, dashboards, access rights.

RUNTIME

Figure 5 presents all modules of the Runtime subsystem. It includes such modules as: Process Engine, Business Rule Engine, Forms Runtime, Lists and Registers Runtime, Runtime Database, Scheduler Service, Optimization Problem Solvers, Simulation Runtime Interface.

The business process engine (Process Engine) is responsible for running process instances on the basis of BPMN models stored in the repository of process models (Metadata Repository, Processes).

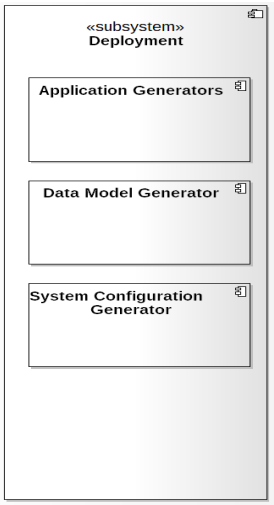

Figure 4: Deployment subsystem.

Business Rule Engine, on the basis of business process data, determines the output values of business rules defined in Business Rules module (Design and Development) and stored in Tasks module (Meta-data Repository).

The Forms Runtime environment is used to prepare screen forms for display in the user interface (Brandl, 2002). Form definitions are taken from the UI Designer and Process Data Modeler modules (Design and Development subsystem), the data of a particular process instance are read and written in the Tasks module (Metadata Repository subsystem). In addition to data, forms also contain Task Actions, standard and custom.

The aim of the Scheduler Service is to determine the optimal task execution order in the business process and to allocate appropriate resources to perform these tasks. The Optimization Problem Solvers module is an interface to the Problem-Solving Environment (PSE).

CONCLUSION

This article presents the four most important modules that make up the Aurea BPM Low-code Development Platform. They constitute the substantive layer of the platform and are responsible for creation, installation and launching of applications in the end-user environment. In addition to the described subsystems: Design and Development, Metadata Repository, Deployment, and Runtime, the subsystems of the system management layer are also used in the Aurea-BPM Low-code Development Platform as support for the process of system development and maintenance (Aurea, 2021). The layer consists of the following subsystems: Lifetime Management, Management Information Consoles, Security, Administration, Performance and Scalability, and Maintenance (Jasiulewicz-Kaczmarek, 2018).

The use of the architecture described in this article allows to separate the system development subsystems from the runtime subsystems while maintaining a close relationship between the models of the designed application and

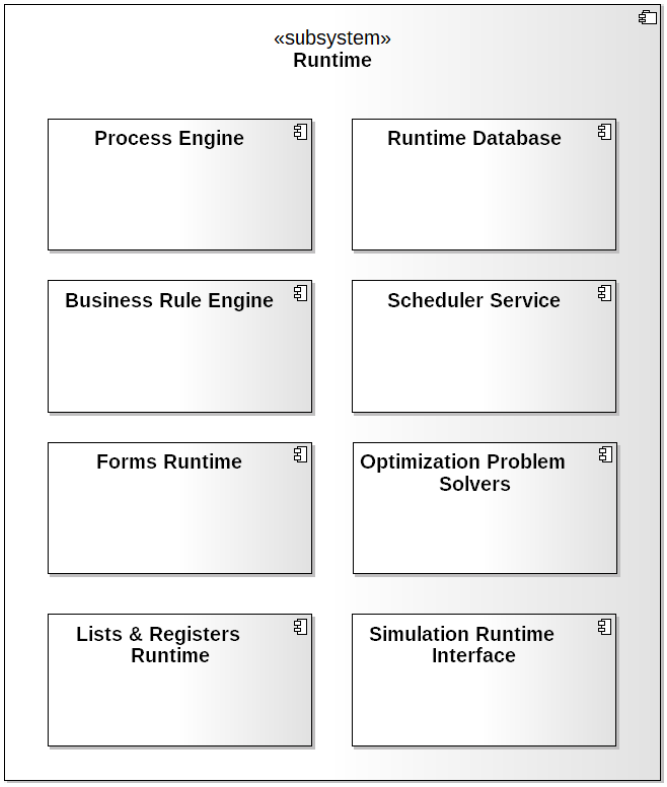

Figure 5: Runtime subsystem.

its runtime form. This approach significantly speeds up the introduction of changes to the software and gives the opportunity to manage changes to the implemented and used system on an ongoing basis.

This type of human-computer interaction makes it possible to increase the productivity of the processes of producing ready-to-use software products by using low-code tools, as an element which, in the field of software production and customization, fits in with the idea of Industry 4.0.

The methods described above have been used in the projects partially supported by EU and Norway Fund.

REFERENCES

Aurea BPM system documentation (2021) aurea-bpm.com

Brandl A. (2002) Concepts for Generating Multi-User Interfaces Including Graphical Editors, Computer-Aided Design of User Interfaces III, DOI: 10.1007/978-94-010-0421-3_15.

Braude E.J., Bernstein, M.E. (2011) Software engineering: modern approaches, J. Wiley & Sons (2011)

Guerrero C.V.S. and Lula B. (2002) A Model-Guided and Task-Based Approach to User Interface Design Centered in a Unified Interaction and Architectural Model, Computer-Aided Design of User Interfaces III, DOI: 10.1007/978-94-010-0421-3_11.

Jasiulewicz-Kaczmarek, M et al. (2018) Implementing BPMN in Maintenance Process Modeling, Advances in Intelligent Systems and Computing, Volume 656, DOI: 10.1007/978-3-319-67229-8_27

Kiedrowicz M. et al. (2016) Optimization of the Document Placement in the RFID Cabinet, MATEC Web of Conferences, Volume 76, DOI: 10.1051/matecconf/20167602001

McKendrick J. (2017) The rise of the empowered citizen developer, Unisphere Res. Div. Inf. Today Inc, vol. 2017

Nowicki T. et al. (2016) Data flow between RFID devices in a modern restricted access administrative office, MATEC Web of Conferences, Volume 76, DOI: DOI10.1051/matecconf/20167604004

Sanchis R. et al. (2019) Low-Code as Enabler of Digital Transformation in Manufacturing Industry, Appl. Sci., vol. 10, DOI: 10.3390/app10010012.

Waszkowski R. (2018) Multidimensional modeling and analysis of business processes, IOP Conference Series-Materials Science and Engineering, Volume 400, DOI: 10.1088/1757-899X/400/6/062031

Waszkowski R., Kiedrowicz M. (2015) Business rules automation standards in business process management systems, in book: Information Management in Practice

Woo M (2020) The Rise of No/Low Code Software Development - No Experience Needed?, Engineering, vol. 6, DOI: 10.1016/j.eng.2020.07.007.