

AXPERT – THE PROCESS SERVER

Technology White Paper

Sabarish Santhanam

Technical White paper on “PROCESS SERVER” – New concept for building process independent enterprise applications

2008

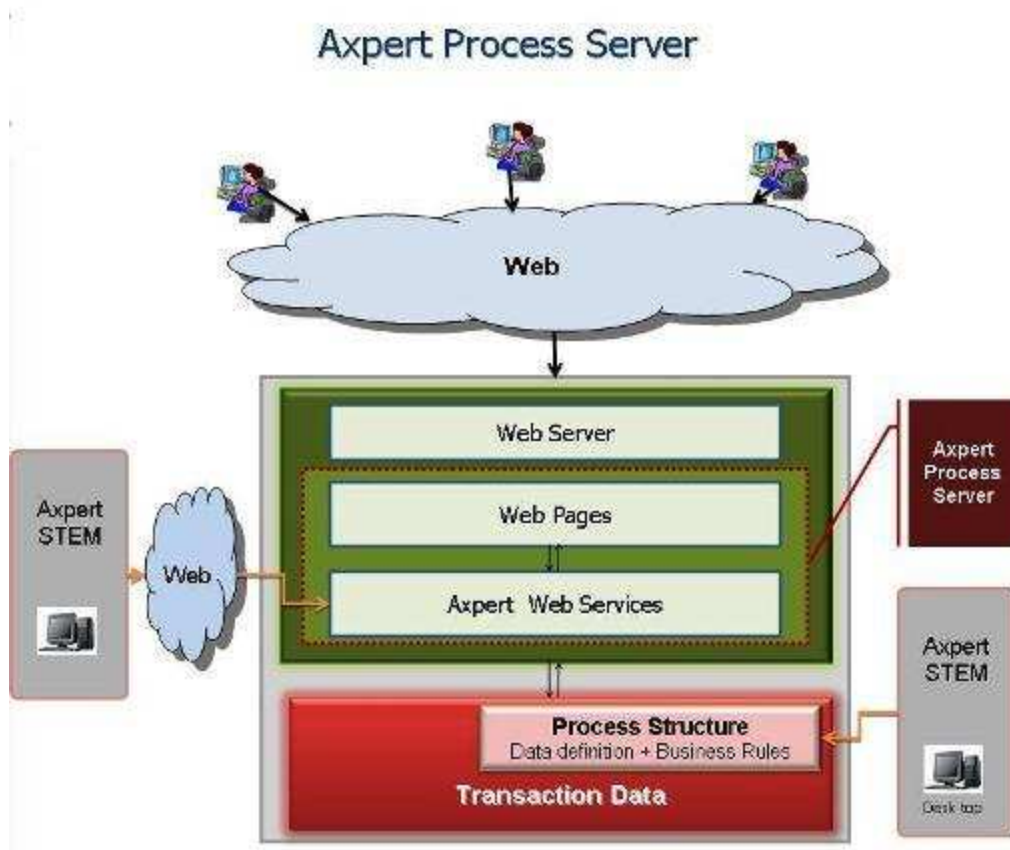
AXPERT – THE PROCESS SERVER

INTRODUCTION

A **Process Server** is a collection of Web services that can create and manage Business Processes and render as a Software Application. Process Server is the **next logical step to the Database Server in Technology Evolution**. A database server manages data that is represented as table structures. Similarly the Process Server manages business processes that are represented as process structures.

A process structure is a set of related database tables along with business rules. Business rules seldom make sense without data. Hence, business rules are tightly bound to a collection of data elements that are persisted into one or more related database tables. Every business entity in a software application can be considered as a process structure.

A business process is created by inter-connecting the different process structures. The connection between process structures are achieved by mapping the elements in one process structure to another. The process structures can be connected with different kinds of maps based on business need.



The basic version of Axpert consists of two major components

- Axpert process server
- Axpert STEM (windows desktop application)

Axpert STEM (Structure Execution and Management) is used to define process structures.

The defined process structures are rendered by the Axpert Process Server as an application.

On creating a process structure, Axpert stem will store the structure definition into database in XML format. Further, it creates the required tables in the database to store user data.

A process structure will consists of

1) Data elements

The data elements are grouped into data containers. Every container is linked to a database table. Hence, when the structure is created, the tables attached with each data container are created in the database and the fields associated with each container are created as fields in that table. If a structure consists of more than one table then all the tables are related.

2) Computation rules for each element

Each data element can be associated with an expression. The expression can be based on the data elements in the structure. Axpert provides a list of common functions that are needed for the building comprehensive structures. In case more functions are needed, these can be written in dot net as web services and attached to the structure. Alternatively, SQL functions can be written and used in SQL statements that can be attached with the data elements.

3) SQL statements to fetch and populate data into structure

SQL statements can be attached with every data element. The SQL result can be automatically populated into the data element or the user can be allowed to choose from the SQL result.

4) Validation rules

Just as expressions can be set to compute values into data elements, there can be validation expressions that will validate the value in a data element. The result of any validation expression should be 'T' if the value is valid or 'F' in case of invalid.

5) Actions

An action is a sequence of tasks that can be performed within if-else constructs. Process flows are created as actions. An action can be associated

with an event, attached to a button or may be attached to the Axpert scheduler.

6) Maps to other structures

Maps are used to populate data from one structure to another. As part of a map the data elements in the source structure will be mapped to data element in the target. One-to-one, one-to-many or many-to-many maps can be defined between structures. The flow can be controlled based on rules.

The process server will read a process structure (created by the STEM) and renders a page. The page provides a user interface to enter & submit data into the structure. The data that is submitted is validated and stored into associated database tables. The various actions and maps associated with the structure will be executed by the server. The stem consists of a full fledged form designer that can be used create the page design.

The process server can do any or a combination of the following tasks

- Render standard GUI along with navigations
- Manage CRUD operations
- Do Data conversion to various formats
- Provide connections to productivity tools like word, outlook, excel
- Manage Work flow
- Handle reporting
- Do scheduled activities
- Exchange data based on maps

Future versions of the server will be enhanced to provide more such native tasks.

Every task that the process server performs on a process structure is available as web services. For example saving data to any structure is available as service, sending structure data as email is available as web service, converting structure data to PDF is available as another web service. Hence, all functionality in the end application that is rendered by the process server is available as web services. Hence, the level of inter-operability is the highest.

Implementing Business rules

Computations

The simple rules to compute values of data elements are defined as mathematical expressions and attached to them. The expressions are based on the data elements within the structure. If the expression needs data from other structures, those are brought into data elements in this structure using SQL statements.

Axpert provides a set of functions that are mostly required to build enterprise wide applications. However, if there is a need for more functions, SQL functions can be written and called through SQL statements or Web services can be written and attached.

Very complex computations can be achieved by writing a set of expressions that are to be executed in a sequence.

Validations

Data that is submitted to the structures need to be validated. Mathematical expressions that return true or false are attached to data elements. Hence, the value submitted to every data element is validated using the given expression.

Database processes

Many of the very complex and high volume processes can be achieved through a stream of SQL statements inter sparse with mathematical expressions. These are attached to the structure as actions that can be invoked based on events or button clicks.

For example a payroll process, that consists of various rules that are persisted to tables need to be applied to employees or employee groups. These can be achieved by writing SQL streams.

Algorithms

There may be need for doing iterative loops and complex programming logic for achieving some tasks. For example Material Requirement Planning algorithm, traveling salesman algorithm, capacity planning algorithms and the likes are complex piece of programs that cannot be achieved just by a sequence of expressions. Some of the algorithms like MRP are available in the process server. The future versions of the process server will consist of all the common algorithms. However, at any point of time a complex algorithm is needed that is not part of the process server, those can be achieved by writing database SQL functions or attaching them as web services to the process server.

Axpert features

Reporting

Tabular reports are mostly SQL results presented in convenient formats. These are defined as information views (IViews) as part of the process structures. The IViews are read by the process server and presented in a tabular format. The process server will manage control breaks, filtering, sorting and grouping of the SQL result. The IViews are dropped into pages and navigation from the page is provided by way of hyper links.

Hence, reporting becomes a simple on the fly job.

The Axpert STEM provides a PDF form designer that can be used to create any free form report based on one or more SQL results. These are also stored as part of the structures. The process server will read these formats and render free form reports.

User roles

User roles are created and access rights are set to structures. Access to different structures can be allowed or denied. Further, access to the transaction data in a structure can also be defined. This will help in implementing record level access to records. The access to every component in a form can be controlled for a role based on conditions. This will enable building complex forms that need to be displayed differently for different roles.

User groups are created and one or more roles are assigned. Users are created and attached to user groups. The users from an active directory or from a LDAP server can also be assigned to different user groups created in Axpert. This will enable single sign on. (LDAP and active directory connectivity under development)

Axpert scheduler

Actions that are attached to different structures can be assigned to the scheduler. These actions will be executed based on the time interval. There can also be business rules based on the (structure data elements) to execute actions. This will enable defining exceptional reporting, escalations and automated work flow.

(Feature under development)

Axpert Exchange server

This server will handle data exchange between different applications. The first version of this server will support data exchange in XML format through an FTP server. This can be used to exchange data with external vendors/partners. An XML map is written to map the data in the XML to the data elements in a structure. Similar maps can be created to convert SQL result to a user provided XML file based on a map.

(Feature under development)

Axpert synchronizer

This is a set of services that will handle data exchange between two Axpert process servers. This will enable implementation of distributed data architecture (store and forward smart client). This is achieved by submitting the structures to the synchronizer that need to be synchronized between servers. The synchronizer will automatically detect the new/modified data in the structures and send to the other servers it is intended for. The server identifications of the different server to which the data needs to flow can be made part of the structure in data elements.

Axpert Advantage

Highest level of process independence

All business processes are stored in the database as process structures. This enables quick and timely changes to the software system thereby making it more responsive than a system that is deep coded.

Process relationships as data

The relationships between processes are stored as maps along with the process structures. Hence, the dependency of any data element on subsequent processes can be analyzed as data. So, all the analytical tools that exist for data can be applied to find out process dependencies and flows. This will enhance maintainability of the application.

Lean architecture

The process server is a small set of web services that can parse and render process structures as an application. The size of the web services will not increase based on the number of process structures. The web services do not change based on the business rules or the kind of business process being implemented. This means easier load balancing, lower cost of deployment and better performance.

Application development closer to user

This can be a very significant advantage when building unique business processes that are not standardized. In this case adopting an agile method to develop the process structures with frequent deployments to end users or doing a pair development along with the domain user can provide magical results.

Further, reporting can be done on site along with customer. Creating reports is just writing SQL (select) statements. The bulk of the reporting can be achieved using this.

Five times faster development compared to traditional coding

Application development will be five times faster than developing an application using coding. The saving is because most of the mundane activities in application development are handled by the process server. Some of the mundane application development tasks include database table creation and maintenance, transaction management, CRUD operations, report writing, managing GUI, automatic support for data import & export, OLE integration with Office tools, Email support etc.

Change management made simple

The process independence of the application makes change management simpler. Business rules can be changed by changing process structures. New business processes can be created by defining new process structures and mapping them to the existing ones.

Frequently asked questions

Is the process server scalable?

The process server is a small set of web services. The web services are light weight protocols and hence are very robust.

The web services mainly perform the following operations

- String parsing
- Pass SQL to database

The rules are predominantly executed through optimized string parsing. Hence, these are not long running web services and hence will not affect scaling.

The SQL statements are just passed to the database with right parameter values. Hence, in many cases the process server acts as just a conduit to database.

So, the level of scaling that can be achieved is directly dependent on the capability of the server. When a strong RDMB is used scaling does not become an issue at all.

There may be performance degradation because the process server parses the process structures for every request. Is this true?

There will not be any significant performance degradation because string parsing is the only extra over head over coded applications. The string parsing has been optimized over a period of time and simple caching mechanisms have been put in place. Moreover, the leanness of the web services (just about 3 MB) and the advantages offered by a process independent application significantly scores over this small overhead

Can complex business processes be defined as process structures?

Business processes of any complexity can be defined as process structures. This stems from the fact that a process structure is an extrapolation of table structures that form the basis for a database application. In the past few decades robust, scalable and very successful applications have been built around table structures (database servers). Hence, a process structure based application should be able to cater to any complex business process.

Further, any business process can be visualized as a collection of data input, validation, data flow and actions. All these can be defined as part of a process structure. Hence, any business process can be defined as process structures.