What's New in
ActiveVOS™ 8.0

ActiveVOS 8.0 delivers new capabilities:

- that help designers and stakeholders gain a shared understanding of the behavior of processes
- that make it easy for developers to integrate data sources
- that provide deployment versioning capabilities that make it easy to keep the old process and services running after deploying a new version, and
- that improve the user experience of knowledge-workers by providing them the information they need to perform their tasks.
ActiveVOS 8.0 at a Glance

- ActiveVOS 8.0 offers developers and subject matter experts new capabilities that significantly improve the ability to express the functions and understand the behavior of a process. Designers now have at their disposal:
  - BPMN’s concept of a pool and lanes make it easy for designers to convey the business functions or roles being carried out by the participants of the model.
  - BPMN 2.0 boundary events provide designers with powerful graphical annotations to convey the interrupting and non-interrupting execution behavior following the arrival of message, timer, error and compensation events.

- ActiveVOS 8.0 improves the ability to control the layout of the process model through advancements in unstructured modeling that complement existing structured control flow whose goal is to reduce the amount of time required to generate the process. With 8.0’s unstructured modeling capabilities, designers now have greater control over the layout and how the process looks which is particularly important when BPMN lanes and boundary events are required.

- ActiveVOS 8.0 offers a new Data Access Service that provides access to data sources without requiring a database to expose its data as a service or require the use of an ActiveVOS Java invoke handler. The 8.0 Data Access Service – a new ActiveVOS System Service - now offers first-class access to any JDBC-accessible data source. Fully integrated with ActiveVOS Designer and Server, 8.0’s Data Access Service provides implementers with the ability to interact with a database to manipulate its structure and content, and execute queries and stored procedures. Built-into
ActiveVOS Designer is a data source explorer and SQL query builder that make it easy for developers to define and test the queries that the Data Access Service will execute at runtime. The ActiveVOS 8.0 Data Access Service saves implementers time and the effort to integrate data sources in business process applications.

- ActiveVOS 8.0 provides a new deployment and versioning mechanism – the deployment contribution – that helps teams develop and deploy multiple versions of a business process application. With contributions, changes deployed by a team of developers no longer impact other developers working on another version of the application. The 8.0 contribution mechanism improves the ability by teams of developers to incrementally build and deploy versions of a business process application and minimize the risk of introducing incompatibilities at runtime.

- Users executing steps of a process need information presented to them that is relevant and well organized. ActiveVOS 8.0 provides designers with the means to define custom task properties and task search filters and columns for use with ActiveVOS Central or custom-built task-based applications. This helps users sort and search for loan applications from a US ZIP code for example and complete their tasks faster.

- One of the biggest sources of help desk requests are password resets. To address this challenge, enterprises deploy Single-Sign On (SSO) environments that allow users to sign-in once using a single identity to automatically access any application integrated into its SSO infrastructure. ActiveVOS 8.0 now integrates with Single-Sign On frameworks, allowing users to access forms and ActiveVOS Central without being prompted for credentials. This reduces the effort of deploying and operating business process applications, and improves user experience.

- While service-orientation has taken hold certain aspects are still problematic for developers. One in particular is that fault generation is too often underspecified by service developers. Faults issued by these services are therefore undeclared and cannot be systematically handled by a process. The presence of such faults must nonetheless be handled. ActiveVOS 8.0 makes access to undeclared faults and SOAP faults possible. Without this capability developers need to resort to arcane techniques
during development, testing and production to understand and handle undeclared and unexpected faults that today is still the norm.

**BPMN Pools and Lanes**

The introduction of pools and lanes in ActiveVOS 8.0 now provides users with the ability to easily depict the functions or roles of activities displayed in the process model. ActiveVOS 8.0 improves the expressiveness of the process canvas by making it easier for process stakeholders to quickly understand who the participants are and their respective roles.

How a process looks and is presented to users matters particularly when lanes are used by the designer. To allow a high degree of control over the process’s layout, advancements in unstructured modeling in ActiveVOS 8.0 make it easy to place activities on the canvas at a specific location, unlike the structured modeling style (for example the use of the fork-join construct found in the Control Flow palette) that helps the user by automating layout. You now have a choice over the modeling style to use for your processes.
BPMN Boundary Events

ActiveVOS 8.0 adds BPMN’s concept of boundary events to convey the behavior of interrupting and non-interrupting of message, timer, error and compensation events.

By attaching a boundary event to an activity or a subprocess (either expanded or collapsed), the designer can now easily convey that a human task will be invoked after the expiration of a timer. An interrupting boundary event will “interrupt” the execution of the activity or subprocess. On the other hand, a non-interrupting timer boundary event shown here on the “Proceed to QA and Deployment” subprocess conveys the meaning that the “Portfolio Manager” is to be notified after the expiration of the timer.

The behavior of these events and associated actions are now more readily discernable by users.
Data Access Service

Fully integrated with ActiveVOS Designer and Server, 8.0’s Data Access Service provides implementers with the ability to interact with a database to manipulate its structure and content, and execute queries and stored procedures.

A single request/response activity is all that is required to execute a single or batched request that take as input statements or a query, and return as XML name-value pairs that are easily manipulated.

The invocation of the Data Access Service uses a request/response style of interaction where the request message defines the statement or query along with execution options, and the response returns the result set or number of rows affected by operation. A fault is returned if an error is encountered executing the SQL request.

ActiveVOS provides the ability to execute a single request or multiple requests with a single operation. When using single data access request, various parameters can be provided including: the SQL Statement to be executed; optional batch parameters; and optional attributes such as the maximum number of rows to return and the maximum time to wait.
among others. A multi-data access request is similar but differs in that multiple requests are issued.

The response to the query is returned within an XML document that either contains the results of the query as sub-elements identified by column names or an indication of the rows affected.

Built-into ActiveVOS Designer is a data source explorer and SQL query builder shown here that make it easy for developers to define and test the queries that the Data Access Service will execute at runtime.

**Deployment Contribution**

ActiveVOS 8.0 offers a new deployment mechanism – the deployment contribution – that isolates development teams deploying multiple revisions of WSDL, XML Schema and POJO artifacts as they develop iterations of business process applications. Prior to this release it would be possible for a developer to deploy for example a schema document that was incompatible with prior versions of the document used by other process definitions. This would result in avoidable runtime errors.

To provide better isolation for multi-version processes and their artifacts, 8.0 now makes use of contributions that package as a deployment unit a
contribution’s resources (i.e. WSDL, XSD, HTML) and process definitions (i.e. BPEL, PDD).

Contributions make it easy to do the following tasks on the server:

- Automatically manage versioning of the contribution and its artifacts
- Delete all old process instances and old resources by deleting the contribution
- Maintain your own resources such that they do not collide with those of other developers
- Rollback the current contribution to an earlier version and vice versa

To support this, the ActiveVOS Resource Catalog is now versioned implying that multiple versions of WSDL, XSD, and POJO can be deployed at the same time. At runtime this ensures that the artifacts deployed by a contribution are the only one accessed by it.

A contribution is a deployed business process archive, managed on the server as a unit of files. Rather than deploy and replace individual processes and resources, you now deploy both current and updated files as a unit to the ActiveVOS Server. The contribution’s deployment log and detail page shown here make it easy to understand dependencies and the artifacts that make up the contribution.
Custom Task Properties and Advanced Search

The ability to quickly gain access to information on a user’s task list or to search for information is important to help them carry out tasks in a timely manner. An example is the ability to process loan requests between $4,000 and $8,000, or mortgage requests by loan type as depicted here using additional columns of information.

ActiveVOS 8.0 now offers the means by which to add custom task properties, and organize, filter and search for tasks. ActiveVOS 8.0’s Central and WS-HumanTask APIs can now make use of declarative and user-defined task properties to satisfy these needs.

ActiveVOS 8.0’s WS- HumanTask API property or a user-defined property can now serve as a selection filter in ActiveVOS Central and other task clients. These can be used to display additional columns of information in ActiveVOS Central’s task view or other task clients, and be used to filter a user’s tasks.

This capability provides developers with the means to organize task lists allowing users to sort and organize their tasks as they need, thus
significantly enhancing a developer’s ability to improve business outcome.

**Single Sign-On**

Business process applications need to integrate seamlessly within an enterprise’s application framework. The ability to provide a single sign-on user experience is important if the application is to gain quick acceptance by its end users.

ActiveVOS 8.0 introduces SSO capabilities to ActiveVOS Central, ActiveVOS Consoles and its services making it possible to integrate with the SSO authentication and authorization framework in place such as Jasig’s CAS, Sun’s OpenSSO, and the SSO frameworks built into applications servers.

The incorporation of SSO enables the ability to:

- Configure ActiveVOS Central and ActiveVOS Console to use the same SSO environment used elsewhere within the enterprise
- Integrate ActiveVOS Central forms embedded in a user’s web-based application with SSO capabilities
- Integrate ActiveVOS Central as an integrated part of a portal-based application.

**Access to Undeclared and SOAP Faults**

ActiveVOS 8.0 now makes it possible to gain access to process undeclared faults and SOAP faults.

Developers can now access fault information from undeclared faults using custom functions that provide detailed fault information within a catch or catch-all. These include:

- `getFaultCode()` - returns the Fault Code QName value
- `getFaultString()` - returns the Fault String value
- `getFaultDetail()` - returns the Fault Detail element - note that this is the same element assigned to the fault variable

When a fault is caught, access to `soap:Fault` element (if the fault was due to a SOAP invoke) are now available via the `getSOAPFault()` custom function.
These capabilities will greatly assist developers during development, testing and production to understand and handle undeclared and unexpected faults that today are the norm.

About Active Endpoints

Active Endpoints’ (www.activevos.com) ActiveVOS is the business process management system (BPMS) that development teams will love. ActiveVOS empowers project teams to create business process management (BPM) applications using services, making their businesses more agile and effective. ActiveVOS promotes mass adoption of SOA-enabled BPM applications by focusing on accelerating project delivery time with a complete, affordable and easy-to-use system. Active Endpoints is headquartered in Waltham, MA with development facilities in Shelton, CT.