THE INNER-WORKINGS OF COMACTIVITY BUSINESS PROCESS PLATFORM

Lean Software for Lean Business

ComActivity

comactivity.net

The order of the day for every executive is ROI. New Mobile phones and Office tools are nice but the reality is that in today's economy each and every IT investment better pay off immediately. Therefore the only true "killer application" is that which provides greater value from already existing technology assets. That is the promise of ComActivity Business Process Platform. You can use it for web-enabling existing applications, developing an executable process or integrating several applications into one optimized solution. The bottom line is that ComActivity Business Process Platform (BPP) delivers very real and solid business benefits.

The User Interface of ComActivity BPP

ComActivity's user interface is delivered through an open standard portal .

The dictionary defines a portal as a grand or imposing door or entrance. In the IT world the portal definition also includes a place where people work. The change from the Windows desktop or green screen user interfaces is profound. Instead of looking into a company server the portal, running inside a browser as it does, is designed to also look out toward the entire world. There can be many different types of portals; accommodating different kinds of enterprise work and roles.

In ComActivity's portal milieu all types of classic enterprise activities are supported.

Until recently portals have not made much of a difference for classical enterprise work as the earliest portal products were never designed for transaction work. That is now changed. Master data maintenance, order entry, production reporting… in ComActivity's portal environment all classic enterprise activities can be incorporated *and* leveraged. So if you are focused on enhancing and extending classical enterprise work; you could not ask for a better time to expand your horizons.

ComActivity's user interface are made up of panel windows known as portlets.

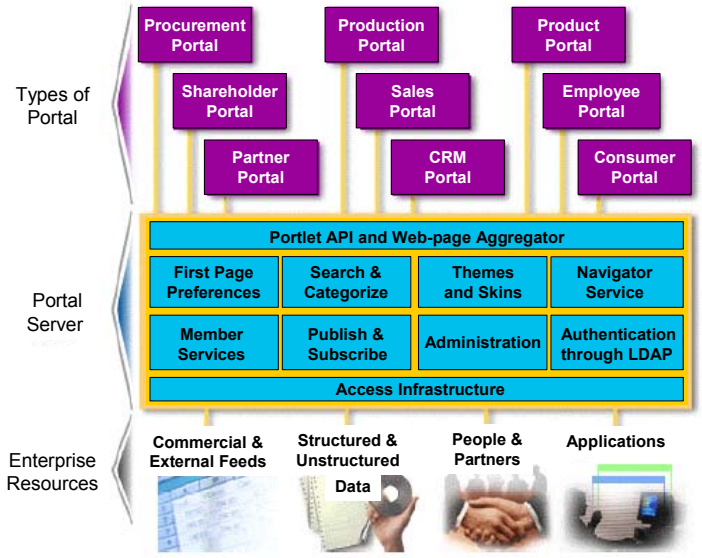
The most visible feature of ComActivity portal is its user-interface which is made up of panel windows known as portlets. A portlet is a discrete component with its own data sources and behavior that gets displayed on the portal's Web page. Each portal view has the individuality from the combination of its portlets and thus *the power of the portlets effectively determines the value of the portal*. To enforce consistency all portlets can be setup to adhere to the same centrally managed style and theme settings.

ComActivity works in Tandem with Open Standard Portals

ComActivity adheres to the Java EE 5 standard.

Standard-based portals offer a huge business advantage in contrast to closed, proprietary alternatives in that it sets your business free instead of locking it in. ComActivity's portal offering is based on the premium industry standard Java EE 5 and its reference implementation Jetspeed which is carried through as an open source project. IBM WebSphere Portal is also based on Java EE 5 so the door is wide open to run ComActivity's applications on IBM WebSphere Portal.

This level of interchangeability is made possible as all portlets share and rely on the same portal infrastructure. The most important is the facility of the Java portlet invocation API (JSR 168). This API is vital as it allows many portlets from different vendors to be invoked in the course of filling one single request. It also helps rendering portlets in parallel so that the portal assembles all the markup fragments once all portlets are ready to go.



The Portal Server which is positioned in the illustration above has many responsibilities such as providing an attractive starting point, user sign on, presentation themes and skins, page aggregation and administration across the portal environment. ComActivity's enterprise portal implementation follows best practice and does not sub-version the portal server. It is a key principle of ComActivity to base all products on open standards.

ComActivity is Built on Open Standards

To attain independence, fairness and freedom you simply have to ally with open standards.

Companies will not tolerate to be dictated or locked-in. They want fairness independence and freedom. ComActivity believes the best way forward is to realize that enterprise software products must not rely on proprietary infrastructure but on standards that are free and open for everybody. Only those application vendors that all-the-way-through base their products on open industry standards will survive.

The winners in the SOA-based on demand era of enterprise computing are companies able to align with open standards and compete with better implementations.

The purpose of standards is to dually support conformity and increase productivity of the whole – not for this day only but also for tomorrow. Sure the progression of infrastructures will continue but basically most of the standards are already here – and here to stay for a forseeable future. The winners in the SOA-based on demand era of enterprise computing are companies able to align with open standards and compete with better implementations. ComActivity is determined to be a winner and know what standards it takes to get there. In the following some of these standards will be introduced and briefly explained:

Java Enterprise Edition 5 (Java EE 5) defines the standard for developing true multi-tier enterprise applications. It simplifies enterprise applications by basing them on standardized, modular components; by providing a whole set of services to those components and by handling many details of application behavior automatically i.e., without complex programming.

Ajax and Flex are web development technique for creating interactive web applications. Ajax and Flex increases the speed and usability.

AJAX (Asynchronous JAva script with Xml) is a tool standard for creating interactive Web-clients. When people speak of Web 2.0, it is often AJAX they refer to. ComActivity is using AJAX and/or Flex/Flash to accomplish dynamic responses and minimization of page shifts. It should be noted that Flex (Flex in runtime is Flash) is used for nearly all graphics.

The Standard General Markup Language (SGML) developed by IBM and approved as an ISO standard 1986 is the mother of all the X-standards that abound these days. XML (eXtended Markup Language), XSL (eXtended Stylesheet Langauge) and its transformation cousine XSLT as well as HTML, DHTML and XHTML are all key ingredients in the emerging World Wide Web. Add TCP/IP (Internet is TCP/IP), HTTP and its build-

upon SOAP (Simple Object Access Protocol) and you have the baseline of Web Services, which is destined to become the dominant integration glue.

ComActivity can interrelate with other organizations of data than the relational model. But of course is the relational model superior and thus as well what all ComActivity developed applications are based on. Closely related to security and the relational model is the definition of a logical unit of work (commit control). If no logical unit of work is in place; neither can a fail-safe chain of activities and data consistency be realized.

ComActivity supports all Web standards as well as all SQL-compliant databases.

For a BPP to be powerful and practical it must incorporate an Integrated Development Environment (IDE). Eclipse is an open-source IDE offering, commercial-quality, and full-features for developing highly integrated tools with a consistent interface, managed code and powerful testing. ComActivity's Visual Studio is based on Eclipse. As a result of relying on Eclipse – which also IBM does – we can focus on developing high level tools for achieving business benefits instead of developing infrastructure.

ComActivity's IDE is based on open-source Eclipse.

ComActivity BPP *is* a Service Oriented Architecture

Application architecture is absolute crucial for long term viability. In this regard there are three principals that are more important than others:

Separation of Concerns – is the architectural principle that limits the scope of changes as the application evolves. The rule of order is thus to isolate the components so that a change in one component of the application will not require a change in another. This is often called loose coupling.
Inversion of Control – is the architectural principle that any logical component is always managed by itself. Everything a component needs in terms of contexts, configurations and loggers is given to the component. That is, every stage in the life of a component is controlled by the code or instruction that created the component.
Conceptual Integrity – is the architectural principle that all components of an arrangement are known by the arrangement in that every component reflects the same philosophies and balancing of desiderata. Simplicity and elegance come from conceptual integrity.

ComActivity does not use empty affarmatives about its SOA compliance. We actually can tell how we implement SOA in the real world.

There are different ways to describe a Service Oriented Architecture. In our opinion the 3 principals of above are right on the spot and are the key principles of ComActivity's SOA architecture.

ComActivity BPP is a Powerful Solution to Advance Your Enterprise Work

Building upon open and elastic standards and very solid experience from application development ComActivity has invested all its energy to bring about a powerful solution to truly advance enterprise work. The platform is used to display and maintain product catalogs, organizing photos and work instructions. As well it works fine to host Intranets and Extranets. However, the key objective is to safely support transactions.

ComActivity BPP is a modern and mature product.

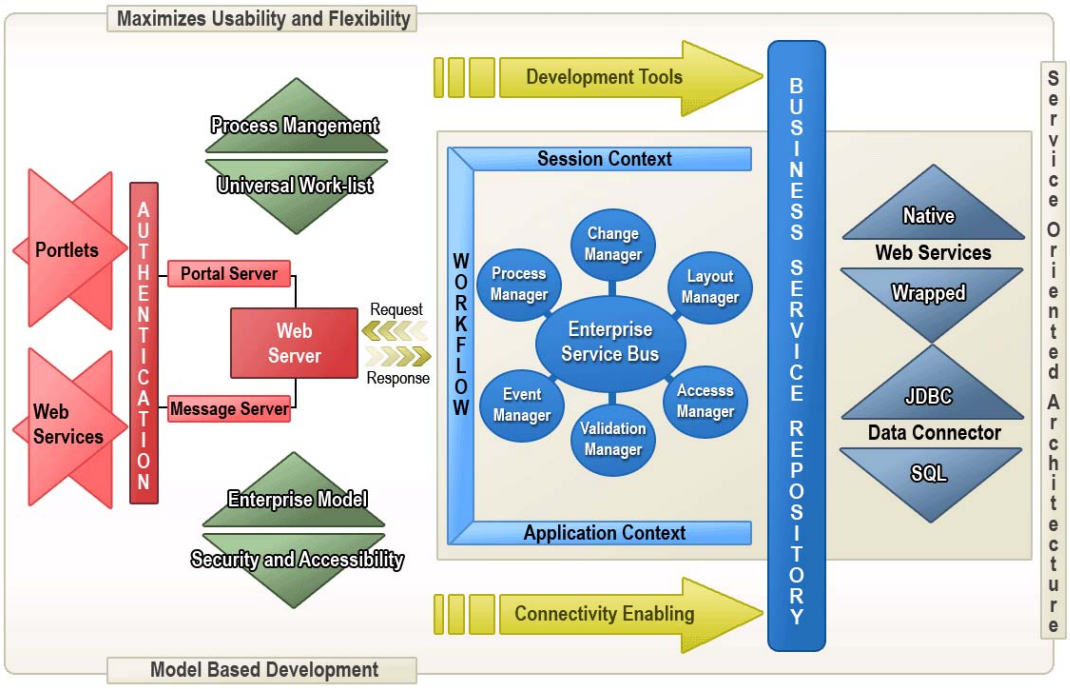
To accelerate every-day business transactions – to attain optimized, secure and collaborative business processes, better business analytics, ad hoc reporting, properly designed documents with or without bar-coding and highly attractive user interfaces – ComActivity is the answer. Indeed the value proposition of ComActivity is broader and deeper, more risk-free and faster to payback than any other alternative.

ComActivity we insist that the future of legacy extension products is dim without being able to support development of new full size applications.

A key differentiator of ComActivity BPP in comparison with previous generations of enterprise application environments is that it is at least a 10 times faster while producing much more transparent and thereby easy to understand applications. This is thanks to what is known as Model Based Development which eliminates 90 % of all programmatic code. Through the model-based approach the activities of *designing* and *developing* a business application become one and the same.

Model Based Development is a key capability for reducing the cost and time to develop and maintain applications.

The Inner-workings of ComActivity Business Process Platform



- **Portlets** are discrete application components having their own content sources and functional behavior while operating within a portal framework. Typically you will combine several portlets to create a portal-page that support a specific task. It is also possible to let "same-time" portlets occur in a parallel portal-page.
- **Web Services** are close to portlets. The simplest way to explain the difference is to say it is a portlet-function without a user interface. That is how Web Services are implemented in ComActivity BPP. XML and SOAP are basics while the other components (WSDL and UDDI) are easy to establish when needed.
- **Authentication** is all about making it safe and convenient to identify and authorize every user or service request. ComActivity BPP uses standard LDAP (Lightweight Directory Access Protocol) to do this and is thus capable to support single sign on.
- The **Web Server** has many responsibilities. ComActivity is certified to run on Tomcat and IBM WebSphere Application Server. Further inside and supported by ComActivity's runtime we offer a complete Job Scheduler to ensure timely automation of batch jobs.
- The **Enterprise Service Bus** is the place of the real action. And even more so in the world of ComActivity as it closes in to fulfill the dream of "no specific code applied". The heart of the ESB is its collaborative processors which count many loosely coupled services tied together through a centralized feeder and results broker. A Rules Engine is one of the inherit services and is used to encode and execute declarative specifications or plain Java code.
- Grand and glorious business processes are fine but they are always made up of smaller pieces of workflows. Every work-task that runs in ComActivity is a **Workflow** based on the Workflow Management Coalition standard (XPDL).
- The **Business Service Repository** represents the intelligence; the place where all application resources are defined and safely stored in

appropriate formats. Accordingly it is also the point of integration such as defining all the paths to connectivity enable your legacy data such as tables, help texts, language dependent terminologies and etc.

- The **Data Manager** is about managing the reading and updating of data which takes place via SQL/JDBC and always with full support of two-phase commit control and referential integrity. Additionally it supports any kind of API (Application Program Interface) and Web Services integration or specific protocols such as OS/400 PCML.
- ComActivity **Development Tools** are based on the Eclipse platform: an open-source environment for creating, integrating, and deploying application development tools for use across a broad range of computing technology. The strategy of building productivity tools is long-term and as of today ComActivity offers Eclipse plug-ins for designing executable processes, workspaces, portlets and documents. In addition we provide tools for Web Service setup, Data modeling and smart field definitions and much more.
- The **Enterprise Model** provides a powerful framework – including all constituencies of operations – to support consistency and speed up the development of next generation applications. The **Universal Work-list** is a common application to capture work-items from each process and thus making collaboration easy and transparent.
- ComActivity's **Security and Accessibility** service provide a single setup of columns for which every change of data should be logged. It contains what functions that are free or restricted for each user, group of users or role. It allows to restrict row access based on content such as only allow a user to work with orders tied to a specific customer.

This is a short explanation of what ComActivity BPP *is* and *offers* to do. We are ready to give details about every component artifact but above all we insist there is no better way to understand the power and beauty of ComActivity BPP than a live demonstration. Creating a process with associated workspaces and portlets; change them in real-time and you will see that **ComActivity offers the shortest time from idea to action**.

~ Seeing is Believing ~