White Paper

Defining the Next Generation eBusiness Platform

A Discussion of the Asera eBusiness Operating System™

April 2001

Contributor:

Chandar Venkataraman

Product Management

Asera, Inc.

The Asera eBusiness Operating System™

Businesses need to leverage the Web to gain and maintain competitive advantage, but must do so in a way that is both cost-effective and sustainable over the long run. With the myriad of choices currently available, the challenge for companies is to deliver an integrated, robust eBusiness solution that allows them to leverage existing applications, rapidly adapt to the unique needs of their business, and continually evolve as requirements change over time.

The Asera eBusiness Operating System™ is a breakthrough, integrating architecture for developing and deploying *composite applications*, which are custom business processes that are built from disparate software applications and data sources.

The Asera eBusiness Operating System:

- Provides a complete technology platform designed to address today's needs for integration, adaptability and evolution.
- Delivers a uniquely tailored and unified user experience globally, regardless of device, under a single portal.
- Supports mission-critical applications in a robust, industrial-strength environment.

This paper describes the Asera eBusiness Operating System, the benefits of its unique design, and the Operating System's frameworks and components. It also describes how composite applications can be built and deployed on the Operating System. This paper has been written for business executives who are responsible for their company's eBusiness solution and the underlying technology decisions.

The Need for a Next Generation eBusiness Platform

"Enterprises not focused strategically on application integration invite serious business risks in terms of longer time-to-market and higher cost of new application deployments."

— Gartner Group, "The Risks of Application Integration," January 2001

Companies face a number of challenges in selecting and implementing the software and technology solutions required to support their business endeavors. This has become particularly problematic in recent years as companies attempt to deploy existing business practices, systems and resources across the Web. Critical to the success of these initiatives is the ability to identify and build on a platform that will cost-effectively support both the current and future needs of the business.

Today's business models also dictate that companies tightly link their businesses with those of their trading partners, suppliers and customers. Critical to achieving this integration is a robust eBusiness platform that not only affords real-time connectivity across multiple business constituents, but also automates and integrates complex business processes across the extended value chain.

Moreover, as companies are continually looking to leverage technology to better serve the different needs of their business, it is natural for them to want to pursue best-in-class applications and technologies for each business area or function. However, such a strategy results in the monumental task of integrating a diverse group of applications that can be, at best, difficult to integrate. Particularly problematic is the fact that best-in-class applications are often *not* designed to work with other applications, as they are proprietary in nature.

While there is a need to integrate complex business processes at multiple levels, current integration technologies, such as Enterprise Application Integration (EAI) and portal technologies, tend not to address the integration problem beyond the data or the user interface level. As a result, companies find themselves having to:

a) Relax their requirements—Instead of pursuing best-in-class applications for each business area or function, companies acquire a single vendor solution. The solution provides the "tight" integration they were looking for, but at the expense of providing cutting-edge functionality and technology provided by a combination of best-in-class solutions; *or*

b) Settle for lower levels of integration—Companies pursue best-in-class applications with the desired functionality in each area, but at the cost of not being able to tightly integrate each application across the rest of the company's systems. They settle for lower levels of integration, which do not offer the power and flexibility of a tightly integrated solution; *or*

c) Build it themselves—Companies that do not wish to lose out on best-in-class application functionality or tight levels of integration invest heavily in in-house development efforts to achieve the desired result. However, such projects take a long time to deploy and are costly and resource-intensive. Moreover, the resulting solutions are not easily adapted to the different needs within the business, or to the changing needs of the business over time.

Companies also need their eBusiness solution to have the capability of being rapidly personalized to meet the unique needs of various constituents—company employees, trading partners, suppliers and customers. They need to be able to quickly personalize user interfaces, workflow, content and context for every application used by the company's constituents.

It is critical to establish a systems environment that can support continuous evolution of a company's eBusiness strategy. Today's business climate clearly necessitates the ability to react to constantly evolving business strategies and technologies. While pursuing the right set of application functionalities at the right level of integration, the solution must also be configured so that it is flexible enough to support any future software revisions or changing business requirements.

Further, if a best-in-class application vendor falls behind in technology and/or functionality, companies are faced with the task of replacing the application with an equivalent one from another vendor and repeating the entire, often difficult, task of rolling out a whole new system.

With the need to continually evolve an eBusiness solution and to integrate a diverse set of applications, companies today are seeking an eBusiness platform that will enable them to:

- *Integrate* complex business processes across the entire value chain, as well as integrate disparate software applications, legacy and third-party systems within an enterprise.
- *Adapt* all application functionality to support specific businesses and users.
- *Evolve* the combined solution as business strategies and technologies evolve.

And most companies need this to happen at Web speed. Unfortunately, most of today's existing eBusiness platforms fall short when it comes to meeting any of these needs.

Asera believes there is a strong need for the next generation eBusiness platform and that the Asera eBusiness Operating System is the basis for a long-term solution that will meet the needs of business—now and in the future.

The Asera eBusiness Operating System

The Asera eBusiness Operating System is a breakthrough system—integrating architecture that fuses applications into composite applications, which are custom business processes that are built from disparate software applications and data sources and can be deployed globally, regardless of device, under a single portal. It enables tight integration of a company's existing systems with their business partners' systems (referred to as "external" systems) as well as best-in-class applications. The Asera eBusiness Operating System delivers a high-performance eBusiness solution that can be configured and personalized to meet the unique needs of the business and the individual users.

Through its integrated, layered approach, the Asera eBusiness Operating System facilitates rapid development and deployment of composite applications. To do this, the Operating System provides the following features:

- *A complete eBusiness development environment*—the Asera Development Workbench™—that is used for:
  - Building and/or deploying new applications.
  - Configuring and personalizing existing applications.
  - Extending existing applications.

The Asera eBusiness Operating System™

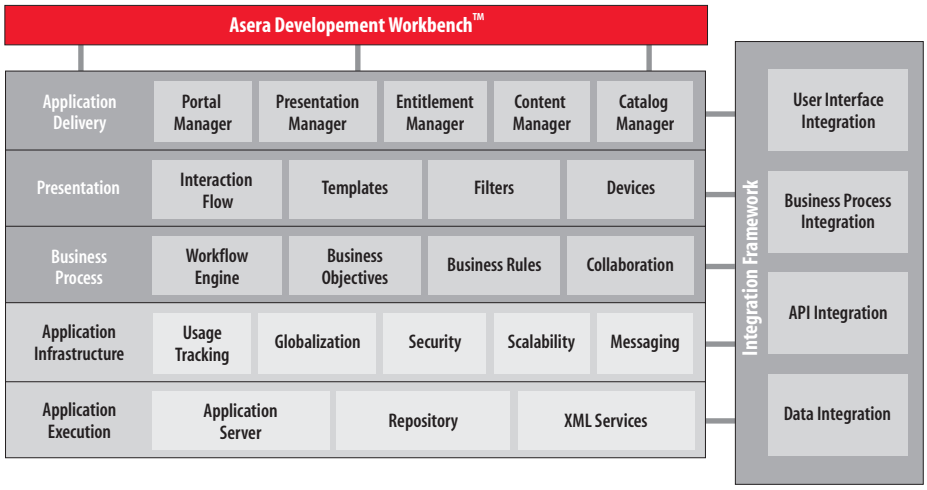

Figure 1. The Asera eBusiness Operating System

- A solid *integration framework* that allows for optimal integration of a company's complex business processes across both external systems and third-party applications.
- A complete *framework for personalized delivery of the user experience*, including portal administration, entitlement, rules-based personalization and content management.
- A unique *framework for presentation of application data*, including screen flow, template management, style sheets and wireless devices.
- A robust and flexible *framework to abstract eBusiness logic and application data.* The framework includes a powerful and configurable workflow engine and a comprehensive set of business objects to support a complete commerce model.
- An *infrastructure framework* that offers a rich set of core application functions such as globalization, security and messaging.
- A robust *execution framework* that provides a runtime execution environment for eBusiness applications. The execution framework includes an application server environment and a repository interface manager.

Benefits of the Asera eBusiness Operating System Design

As a result of its layered approach, design philosophy and robust functionality, the Asera eBusiness Operating System is scalable, flexible and extensible. Each of the Operating System components combines to form an environment that offers the following design features and benefits:

The Asera eBusiness Operating System Is Designed for Rapid Integration and Continual Change.

The Asera eBusiness Operating System is designed to deliver an environment that supports rapid and dynamic integration of data and business processes across an enterprise. As a result, the Operating System provides the following benefits and functionality:

- *Rapid Integration*—Businesses can offer a complete commerce model that delivers all the key components of the commerce value chain by rapidly integrating business processes and any number of related applications across the extended enterprise. They can quickly integrate external systems, best-in-class applications and existing enterprise software into a unified eBusiness environment.

- *Multiple Levels of Integration*—Businesses can choose to integrate with external systems and third-party applications at the proper levels, taking into consideration both the specific technology of the application as well as any unique requirements of the business. Third-party applications and external systems can be integrated to the Asera Platform™ at multiple levels, ranging from loose integration (providing single sign-on access to the relevant applications) to the tightest level of integration, in which application engines are "plugged" into the Platform (providing complete integration of business data and application workflow).
- *Transparent Extension and Replacement of Applications*—As new technology innovations or business requirements emerge, integrated third-party application components can easily be extended or replaced without any disruption to existing systems.
- *Data Aggregation From Disparate Sources*—Data can be aggregated from any number of disparate sources such as the systems of external partners, internal databases, Enterprise Resource Planning (ERP), Customer Relationship Management (CRM), Sales Force Automation (SFA) and legacy systems, external Web sites, news feeds, corporate collateral and user-generated content. The aggregation of data can be achieved either in batch mode or in real time, depending on how frequently the information needs to be updated.
- *Multiple Protocols for Messaging and Integration*—The Platform supports a solid messaging and integration framework that supports multiple protocols such as HTTP, JDBC, RMI, COM, FTP, SMTP and SOAP.
- *Numerous Formats for Business Document Exchange*—The Platform also supports a comprehensive framework for business document exchange by supporting various B2B protocols and standards such as RosettaNet, cXML, CBL, OAG, OBI, FpML, BizTalk, ebXML and EDI.

The Asera eBusiness Operating System Is Designed to Deliver a Unified User Experience.

The Asera eBusiness Operating System is designed to deliver a unified user experience across any eBusiness application functionality, both within an enterprise and to the enterprise's trading partners. As a result, this Platform is able to deliver the following benefits:

- *User Authentication*—The Platform provides the tools to authenticate users when they first sign on. Specific access privileges dictate what applications users have access to, as well as what they can accomplish within each application.
- *Single Sign-on and Navigation Transparency*—Users need to sign on only once to access any eBusiness application functionality. Users navigate among multiple applications in a seamless, continuous manner and are completely unaware when they move from one application to another.
- *Ability to Tailor Unique User Experience*—Each user is presented a unique user experience that is defined by a unique "user business process." User interface, workflow, content and context can each be tailored to a user's unique role within the enterprise.
- *User Experience Adapts to Changes*—Every user experience is completely adaptive. When new applications are added, the exist-

ing user experience (in terms of workflow, look and feel, and user entitlements) can be adapted to include the newly integrated application functionality—and can be done seamlessly, rapidly, and with tight levels of integration.

The Asera eBusiness Operating System Is Designed for Industrial Strength.

The Asera eBusiness Operating System is designed for industrial-strength deployment, scalability and performance. As a result, the Operating System provides the following benefits for a company's mission-critical applications:

- *Fault Tolerance*—Applications deployed on the Platform are fault tolerant. Since the Platform uses and supports a clustered environment for fault tolerance along with an enterprise-strength infrastructure, user sessions can transparently fail-over to a different node if one node crashes.
- *Change and Upgrade Transparency*—The Platform is optimized for isolating changes or upgrades in order to prevent any disruption of service. Application changes, platform upgrades, functionality releases and regular maintenance events do not require any application reprogramming or disruption of service.
- *Granular Delivery of Functionality*—Application functionality can be delivered on a granular basis, much like the cable television model. Users select and pay for only those features that they want to use.
- *Incremental Enhancements*—The Platform is optimized to support incremental enhancements. Users can easily and immediately take advantage of small increases in functionality, rather than waiting and/or paying for major releases of which they might be interested only in a subset of the newly deployed functionality.
- *Dedicated Service Environment*—Each customer can have a dedicated service environment delivered, depending on specific Service Level Agreements (SLAs).
- *Device Independence*— Asera applications can be delivered over a device-independent, standards-based network infrastructure that supports the Web as well as mobile and handheld wireless devices.

The Frameworks and Components of the Asera Platform

The Asera Platform is comprised of a number of distinct, interrelated layers referred to as frameworks. Each of the frameworks has multiple components that provide the services or functionality of each layer. The Asera Platform's integrated, layered approach is what enables the Platform to offer its robust set of services for eBusiness applications development, deployment and management.

In the rest of this section, the frameworks within the Asera Platform, as well as the components or services that make up the frameworks, are discussed in more detail.

Application Delivery Framework

The application delivery framework provides services to deliver a unique experience to each user. The framework allows for personalization of the user experience by combining system-level entities, such as rules and entitlement, and user level entities, such as portal pages. To accomplish this, the application delivery framework includes the following managers:

- *Portal Manager*—The portal manager offers a portal framework that allows users to build and manage their own portals. The portal manager provides the following portal functionality:
  - *Portal Objects*—Provide windows to the Asera eBusiness applications. The objects provide

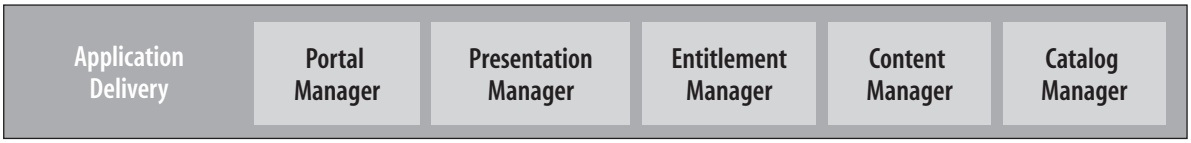

Figure 2. Application Delivery Framework

a view into each application module, which users can then further customize based on content and structure.

- *Portal Content*—Allows users to select specific modules (portal objects) for their individual portal page. It also allows users to specify criteria for the initial content of each application module. For example, it would allow users to specify relevant search criteria for the content.
- *Portal Layout*—Allows users to select the layout of their portal page from a list of options predefined by the administrator.
- *Portal Style*—Allows users to personalize the style of their portal page.

· *Presentation Manager*—The personalization manager offers a rules-based personalization engine that enables applications to bepersonalized for specific users. It allows for the definition of rules that result in certain actions being executed based on predefined user profiles or usage conditions. It also provides for the tracking of user profiles and usage patterns over time. This allows for dynamic personalization of content for each user, resulting in rich up-sell and cross-sell opportunities.

· *Entitlement Manager*—The entitlement manager controls user access to applications, activities, tasks and underlying data, based on the roles, profiles and relationships defined for each user. It supports the following entities:

- *Resources*—Identify the underlying components that can be accessed. This includes applications, activities, tasks and data.
- *Roles*—Identify which group has access to a logical set of resources.
- *Relationships*—Allow for the specification of relationships between partners. Examples of such specific relationships would be enterprises creating business relationships with their customers, or Net Market Makers (NMMs) creating business relationships between multiple suppliers and buyers.
- *Profiles*—Specify the data and content that a trading partner and its members are entitled to see within each application.

· *Content Manager*—The content manager offers generic content management functionality for all applications accessing content. This is done through the following functionality:

- *Content Repository Management*—Provides the ability to manage content by storing relevant metadata, such as keywords, categories and attributes, along with content data in a database management system. It also provides the interfaces required to insert, update or delete content and its associated metadata in the repository.
- *Content Matching and Rendering*—Provides the ability to extract relevant content based on user-specified matching criteria, and to format the extracted data based on predetermined templates before it is rendered to the user.

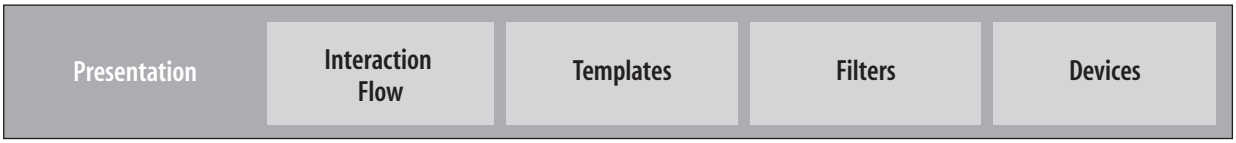

Figure 3. Presentation Framework

- *Content Delivery*—Provides the ability to deliver content to a set of entitled users, via email or the Web.

· *Catalog Manager*—The catalog manager aggregates catalog content from multiple sources, creates a taxonomy of the content, and presents the catalog in a customized format. It allows users to search product offerings by product ID, by keyword, or by simply browsing through the product list on a category-by-category basis. It also defines a comprehensive catalog schema to accommodate a wide variety of catalog information, including URLs, images and pointers to product documents. It uses an extensive data-caching mechanism, provided by the platform execution services, that results in a significant reduction in search time. The catalog manager also allows for an advanced text search engine to be plugged in, such as Oracle Context server.

Presentation Framework

The presentation framework provides components to manage the presentation of user interfaces across multiple devices. The framework allows for the separation of data and logic from their presentation, thereby bringing a great deal of flexibility to the look and feel of applications. It provides declarative definitions for the specific *content* that is to be displayed, its *structure* (layout), its *style* (font, color and so forth), its *format* (actual format of the data—for example, currency or percentage), its *behavior* (actions allowed on the data) and *control* (radio buttons, lists).

The presentation framework comprises the following components:

· *Interaction Flow*—Interaction flow defines the user interface (UI) of an application as specified by a set of Web pages and connections to pages of other applications. The concept behind the interaction flow service provides unique capabilities to personalize the user interfaces of applications.

· *Templates*—Templates are used in generating HTML data to make a Web page. A typical Web page will contain static data as well as tags for displaying dynamic content, all of which can be defined using the following types of templates:

- *Master Template*—Provides the overall page layout common to all applications.

- *Wireframe*—Is referenced in the master templates and provides a more granular layout for a given application.

- *Micro Template*—Is referenced in the wireframes and provides dynamic application content with specific structure and style.

· *Filters*—Filters specify how business data should be displayed and what actions a user can perform on the displayed data. Filters provide a rich set of elements to support the following functionality:

- *Behavior Filtering*—Specifies the behavior of the HTML that is generated and presented, as in a single-select or multi-select menu. The behavior of the UI can be further personalized based on user entitlement and other application semantics. For example, a behavior filter can

Figure 4. Business Process Framework

specify certain fields be rendered as a text-entry element or made inaccessible to certain users.

- *Display Filtering*—Specifies the UI control and structure for each data element that is displayed. This includes such things as radio buttons and checkboxes for controls as well as blocks and tables for structure. Separating the behavioral aspects of data from their display characteristics allows great flexibility in presenting different user interfaces to different users and device types.

- *Style Sheets*—Specify the style (fonts, colors) of the displayed data and help Web authors separate style from content. By using style sheets, Web authors can place semantic meaning on data (for example, red would signify a problem issue) and apply multiple styles to the same document.

· *Devices*—The Platform supports a wide variety of devices through which users can engage in eBusiness activities. For example, users can conduct business through either an online terminal or a mobile agent such as a PDA or a cell phone. The interaction flow can be uniquely tailored for each device based on the device's specific characteristics.

Business Process Framework

The business process framework provides a robust and flexible model to abstract enterprise logic and application data.

The framework consists of the following components:

· *Workflow Engine*—The workflow engine defines and executes the underlying business logic of all eBusiness applications. The engine supports the following two types of workflow:

- *Functional*—Describes the business logic that is part of an application.

- *Interaction*—Describes the various user interfaces that are part of an application.

Each workflow type consists of a sequence of simple or composite steps that perform functional tasks such as "calculate product price" or interactive tasks such as "display product list." By separating these workflows, the Platform allows for true personalization of each user experience, as well as reusability of the underlying business steps. This approach facilitates rapid adaptation of the user experience to new functionality.

· *Business Objects*—Business objects provide an object-oriented data model to abstract enterprise application data as logical entities. The abstraction allows applications to be transparent as to where and how the application data is physically stored. A business object is defined by a set of attributes (specification of the data), relations (specification of the relations to other objects), and methods (specification of the behavior of the object regarding its creation, inheritance and maintenance).

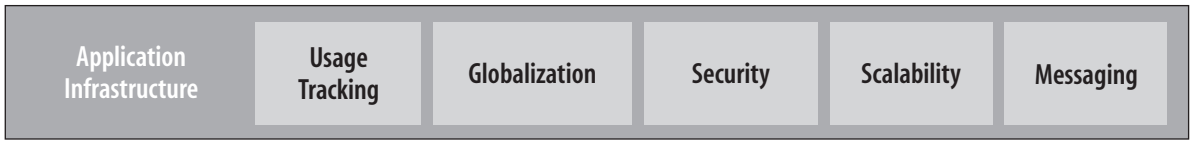

Figure 5. Application Infrastructure Framework

- *Business Rules*—The business rules defines and executes business rules that apply to business data. An example of a business rule would be to allow a specific price discount for orders placed over a certain amount. The business rules models a business rule as both a set of conditions and a set of actions that are executed when the conditions are met.

  The business rules engine allows business rules to be explicitly executed through named rules that can be invoked by an application. Rules can also be implicitly executed through triggers. Triggers specify the actions that are carried out when a business object is fetched, created, modified or deleted.

- *Persistence*—Persistence allows applications to work on business objects with a view that is totally independent of the underlying data sources. It allows a business object to be mapped to physical data sources, with the capability of mapping attributes of a single business object to different data sources. Persistence is achieved by two components:

  - *Connectors*—Interface directly with external data sources. For each data source, there is one associated connector that understands the data in that particular data source.

  - *Adapters*—Perform the actual build of the business objects using data coming from one or more connectors.

Application Infrastructure Framework

The application infrastructure framework includes services that provide infrastructure functionality. Applications using the services provided by this framework are enriched with additional functionality such as globalization and usage tracking.

Examples of infrastructure services include:

- *Usage Tracking*—Provides a configurable mechanism that allows application developers to specify the nature of transactional data that is to be tracked within each application. This information can then be used for a range of functions, from simple reporting to advanced business intelligence.

- *Globalization*—Globalization provides a foundation for application developers to write applications that can later be localized to the specific languages, currencies, addresses and time stamps of different geographies, taking into account the different government regulations or corporate policies governing those geographies. The infrastructure services provide capabilities for parsing, formatting, normalizing and de-normalizing business data. This is done by allowing the data to be normalized in Asera's Universal System Format (USF), and later transforming the normalized data to a format that is appropriate to the specific locale of each user.

- *Administration*—Administration provides a powerful publish-and-subscribe mechanism for applications to receive administrative commands and to send responses. Administration tools can use these services to administer specific applications.

- *Security Management*—Security provides mechanisms for authentication, single sign-on and access control.

Figure 6. Application Execution Framework

- *Authentication*—Performs verification of the identity of any user who attempts to access a protected resource and is highly configurable based on security needs. The mechanism also supports data encryption to achieve data privacy and data integrity.
- *Single Sign-on*—Allows applications to automatically authenticate users without requiring them to repeatedly sign on as they dynamically navigate across individual applications.
- *Access Control Services*—Provide runtime authorization to users who access the underlying resources.

· *Scalability*—The Platform scales to very large installations. The design allows for:

- *Scaling*—Scaling up on the same machine using symmetric multiprocessing technology or scaling out to clusters of machines.
- *Availability*—Transparently failing over user sessions on a given cluster node to a backup node in the cluster.
- *Manageability*—Adding and deleting clusters and/or applications without having to shut down the service to users.

· *Messaging*—Messaging provides a comprehensive messaging framework to integrate with internal and external back-end systems and third-party applications through various protocols such as HTTP, JMS, SMTP and FTP. Messaging also supports multiple modes of integration such as batch, publish and subscribe (EAI middleware), and real-time synchronization (EAI middleware). The framework also allows multiple EAI and B2B integration software to be plugged in as transport and translation engines.

Application Execution Framework

The application execution framework provides a runtime execution environment for eBusiness applications.

The framework includes the following features and functionality:

· *Application Server*—The framework offers core runtime services required for executing high-level applications. Examples of core runtime services include session and context management services, queuing services, caching services and thread execution services. The environment also supports a dynamic load balancing capability.

· *Standards Compliance*—In addition, the application server environment is compliant with the latest Internet technology standards such as Java, JSPs, Java Servlets and EJB. It also supports standard Internet protocols such as HTTP, JDBC, and CORBA for communication, and HTML and XML for information exchange.

· *Repository Interfaces*—The framework offers core services that provide interfaces to repositories such as databases and directory services. Examples of repository interfaces are JDBC and JNDI.

Integration Framework

The integration framework provides the foundation for companies to integrate both complex business processes as

Figure 7. Integration Framework

well as diverse application functionality across the extended enterprise.

The framework offers the following features:

- *Business Document Exchange*—The framework offers a foundation for trading partners to automate their business transactions over the Web. It offers a real-time infrastructure to manage the process of routing and translating business documents across different trading partners, and supports various standards such as XML, EDI, RosettaNet and Biztalk to achieve this.
- *Application Integration*—The framework offers a wide range of mechanisms for third-party application functionality to be integrated into the Asera Platform. The framework supports integration of applications at the following levels:
  - *Data Integration*—Aggregation and exchange of data among internal enterprise systems such as ERP, CRM and legacy systems.
  - *Application Program Interface (API) Integration*—API-level integration to application functionality using interfaces such as Java, XML, JDBC and SOAP.
  - *Business Process Integration*—An even tighter integration of application functionality using the business process framework. This feature offers an integrated process flow across the entire extended enterprise.
  - *User Interface Integration*—Integration of applications at the user interface level using features such as single sign-on, portal framework and the presentation framework.

Building and Deploying New Applications on the Asera eBusiness Operating System

The Asera eBusiness Operating System's integrated, layered approach enables rapid eBusiness applications development, deployment and management. Applications that run on the Asera eBusiness software are user business processes that are dynamically assembled, personalized and presented by the layers of the Operating System.

Specifically, the user business processes are serviced by the Operating System as follows:

- *Application Delivery Framework*—Personalizes and delivers the individual user experience.
- *Presentation Framework*—Dynamically assembles and presents the user interface.
- *Business Process Framework*—Executes the user business process.
- *Application Infrastructure Framework*—Provides infrastructure services that enrich the user business process.
- *Application Execution Framework*— Provides a runtime environment.

Building Applications for the Asera eBusiness Operating System

The Asera environment facilitates building new applications on the Asera eBusiness Operating System through the following features:

The Asera Development Workbench

Applications can be built entirely on the Asera eBusiness Operating System through the *Asera Development Workbench*. The workbench provides tools and interfaces that allow complex applications to be built and deployed using a declarative approach.

The Asera Development Workbench:

- Enables application developers to rapidly create and manipulate all components needed to create and deliver a user business process. The components include *presentation logic*, *functional logic*, *logical data representation* and *physical data models*.
- Includes a rich set of published interfaces (SOAP and Java APIs) to all core platform and pre-integrated application functionality, thereby providing capabilities for application developers to reuse existing workflow components as Web services in their applications. For example, a new Human Resources application could use an existing "browse catalog" Web service to add a link for displaying the standard list of healthcare providers. By being able to efficiently use existing workflow components in the form of Web services, application developers can build new business processes without having to recode existing logic and screen flow.

Leveraging Domain Application Components

The Asera eBusiness Operating System enables business logic to be leveraged from existing *domain application components*. Domain application components are third-party application engines that offer specific functionality needed to develop domain applications. Examples of domain application components that are currently leveraged for pre-integrated Asera Applications are:

- *Moai's Negotiated Commerce Engine (NCE)*—A flexible enterprise-class application engine designed specifically to allow businesses to host online auctions. The engine supports multiple pricing formats (English, Dutch, Sealed-bid, Reverse and Fixed Price), "incremental" and "decremental" auction bids, automatic time extensions, activity tracking and bidding history. Moai's Open API and Moai's Negotiated Commerce Markup Language (NCML), an XML implementation for dynamic commerce markets, provide access to the objects and functions such as auction objects, user objects and bid objects within the engine. These APIs can be accessed directly from the server's template pages or from other applications such as legacy systems or other eCommerce applications. The APIs support parameter validation, exception handling, security, connection and session management.
- *Selectica's Product Configuration Engine*—Provides a basic rules engine required for configuring complex products. The engine offers preset rules that run in the background to ensure that only viable products are configured. For example, rules could be used to ensure that inaccurate orders or other downstream production problems do not occur.

Adapting Applications With the Asera Platform

Applications that are part of the Asera Platform can easily be adapted to specific types of deployments and constituents. One can use the Asera Development

Workbench to configure the full set of existing business objects to company-specific business rules, modify the existing workflow to accommodate company-specific or industry-specific business processes, or change the look and feel of specific applications to address specific situations.

Evolving Applications With the Asera Platform

Applications that are part of the Asera Platform can constantly evolve as technologies and business needs change. The layered design and approach to application deployment facilitates selectively replacing or extending certain application components. As an example, a domain application component from a vendor can easily be replaced by another vendor solution that is more appropriate for a specific industry.

The Next Generation eBusiness Platform

The Asera eBusiness Operating System provides an integrating architecture for composite applications, thereby allowing for rapid integration of complex business processes and software solutions across the extended enterprise to provide a company with a unified and powerful eBusiness environment. Its flexible architecture affords companies the luxury of using tightly integrated best-in-class applications in a way that can continually adapt and evolve to meet the changing needs of the business.

The Operating System provides the ability to adapt eBusiness solutions to present a unique user experience to each user. The user interface, workflow, content and context can each be tailored to deliver a unique "user business process" for each individual user. At the same time, the Operating System is optimized for industrial-strength scalability, reliability and availability, and is designed to support continual enhancements, change isolation, and granular delivery of application functionality.

The Asera eBusiness Operating System *is* the next generation eBusiness platform—it is the foundation for a company's successful eBusiness strategy, now and for the future.

eBusiness Operating System

600 Clipper Drive
Belmont, CA 94002
Phone: 650·769·1234
Fax: 650·769·1200
www.asera.com

WP-NGE-501