Visual LANSA version 14 is a game changer for web application development

Richard Lancaster
LANSA Product Center

Developing web applications can be a challenge because of a number of factors. Developers can select from a long list of development tools, programming languages and JavaScript frameworks. How do they choose the most appropriate combination? Will the application have to run in multiple browsers? Will the user interface provide a user experience that engages customers and staff? Where will the business logic run – in the browser, on the server or distributed across both? Then there is integration with line-of-business systems to consider.

What if you could remove or reduce the complexity and frustration by using a single development tool that addresses the problems and changes the nature of the web development game? Visual LANSA version 14 is the all-encompassing development tool that meets these challenges. You can use it for developing and maintaining applications that support multiple client device types, across multiple browsers, while the server platform can be Windows, Cloud (AWS or Azure), Linux or IBM i.

Why do you need a Game Changer?

Application development is more complex today than in the past, and compared to legacy applications, today's modern applications are expected to deliver so much more. The skills required to maintain legacy applications are not the same skills required to develop modern web applications. Many developers struggle with multiple programming languages and complicated development tools.

The tools and frameworks that developers use to build web applications seem to be selected from a "flavour of the month" list that quickly becomes superseded by newer ones. Developing applications using multiple programming and scripting languages requires mastering these tools and languages – making the development task harder and more complex.

There has to be a better tool for developing web apps than the @#% we are using now!*

When applications, built using many different tools and languages, need to be changed and modernized in the future, the risk of not having the required developer skills on hand is increased.

You need tools that can simplify the development task, reduce effort and preserve skills.

Why is Visual LANSA Version 14 a Game Changer?

Visual LANSA simplifies application development, automates much of the development effort, and produces web applications that look and behave like desktop applications.

You use one programming language for the whole application, including the server and client components. No HTML, CSS, JavaScript, C#, COBOL or RPG code required. Optionally, you can augment the code with your choice of JavaScript frameworks and libraries.

With only one language to learn (RDML, Visual LANSA's Rapid Development and Maintenance Language, a high level language that has evolved over nearly 30 years), as opposed to alternative web development methods that require multiple tools and languages, using Visual LANSA you can build web applications faster and deploy them in less time.

Visual LANSA web applications have a desktop quality look-and-feel and run in any browser. There is no need to test the applications in multiple browsers as the code generated by Visual LANSA is cross-browser compatible. You can build applications that support multiple device types and sizes, including smartphones, tablets, desktops and laptops from the same code.

Visual LANSA allows you to distribute the business logic between servers and browsers to serve the requirements of the user interface and integration with line-of-business systems.

Visual LANSA generates much of the application code for you, allowing you to build

web applications in a fraction of the time taken using other tools.

When building desktop or server applications, Visual LANSA developers write the RDML code. The "Create LANSA Application" process then builds the application, combining it with the LANSA runtime (Figure 1).

Using Visual LANSA version 14, the process for building web applications is the same as that for desktop and server applications. Developers write the RDML code for a web application. The "Create LANSA Application" process then builds the dynamic HTML, CSS and JavaScript for the application and combines it with the JavaScript version of the LANSA runtime (Figure 2).

Version 14 of Visual LANSA allows web developers to build Single Page Applications (SPA), which is the equivalent to building a client/server app, but instead of the UI being a rich-client interface talking to a server, the client is the browser. SPAs run entirely inside the browser and the only time the server is accessed is for reading, inserting, updating and deleting data or executing server-side logic. Since all of the server calls are done asynchronously, the network traffic between the client and server is greatly reduced. Users will find a noticeable performance improvement with the new LANSA web applications, because there is no need to refresh the entire web page with SPAs – it's a like super-fast desktop style application running in the browser.

Visual LANSA Framework

The Visual LANSA Framework (VLF) is an operational web and Windows application that allows developers to assemble business software rapidly with minimal coding. Developers can assemble prototype applications and use the same design for both desktop and web applications. It is appropriate to use when these requirements apply:

- Deliver an application quickly
- Develop medium to large business applications for transactional and back-office contexts
- Maximize design productivity and include significant user involvement in the project
- Use a single application design for both Windows and web

The VLF minimizes the amount of coding required and includes a program coding assistant that can automatically generate most of the code. As each component of the application is built and approved, it snaps into the VLF. The VLF uses an XML-based design schema and its modular design supports many developers working on different parts of the application at the same time. The VLF's component structure means enhancing and maintaining applications is efficient and manageable. Developers can snap in new and maintained components without disturbing the whole application.

Web Applications for Mobile Devices

Using Visual LANSA, developers can build mobile web applications or hybrid mobile applications by including Visual LANSA's native mobile app, available for iOS, Android, Windows 10/8.1 tablets and PCs.

Visual LANSA provides tools for the user interface design, business logic and database access. In the case of mobile web applications, the browser manages the user interface and for hybrid applications, the app can also access mobile device features including the camera, location services, and offline storage.

The mobile web and hybrid applications can run on smartphone, tablet, laptop and desktop devices from the same application code.

Develop and Deploy in the Cloud

You can develop applications in the cloud and/or deploy applications in the cloud using Version 14 and LANSA's Amazon Web Services (AWS) and Azure offerings.

Using your cloud account, you can create your own instance of the Visual LANSA development environment and build applications. LANSA

Figure 1: Packaging Desktop and Server Applications

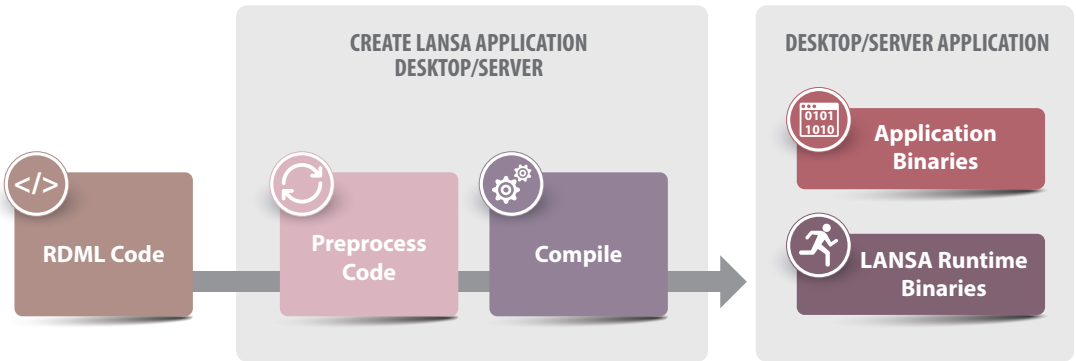
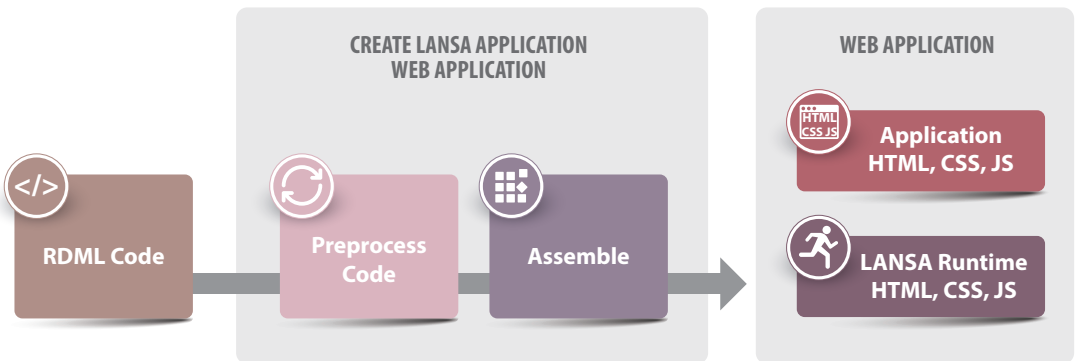


Figure 2: Packaging Web Applications



provides a ready to go virtual machine image for you to deploy applications in the cloud. You run a script with a simple user interface to instantiate a working virtual machine. After creating your instance of the virtual machine, you provide your LANSA application MSI (Microsoft/Windows Installer), which is automatically installed for you and your application is ready to go.

You can develop and deploy in the cloud, or develop on-premise and deploy in the cloud.

Business Rules Repository

LANSA's unique business rule and data definition repository makes future maintenance and enhancement easier, faster, and less risky. You are more productive because you don't have to code business rules and data definitions (including type, size, presentation and validation). The repository centralizes data definitions so that all applications use the same definitions. When a data element attribute changes, you only have to change the definition once, and don't have to examine every application that uses that data element and make the same change. This reduces the cost of application maintenance significantly. For example, 5 lines of RDML code can create a typical, working, multi-column, multilingual list bound to its data source.

Getting Web Development Right is Essential

Businesses are in two industries. One industry is the domain of their products and services and the other is information technology. Making wrong choices or failing to deliver products and services can put you out of business. The Internet is a dominant and influencing factor driving the success of any business and that is why getting web development right is essential.

Four Types of Web Applications

We at the LANSA Product Center think that there are four types of websites.

1. Information

Information websites include publishing and encyclopaedia style websites (think "brochure-ware"). Content is static and edited by the owner of the website or registered editors. Typically, these websites provide search and a table of contents, and users retrieve information by searching and following links.

2. Social and community

Social and community websites include WordPress, Facebook, Twitter, Instagram, blogs, and bulletin boards. They have more business logic than information websites. The content is dynamic and edited by all participants, including the website owners and users.

3. Transactional

Transactional websites support business-to-consumer (B2C) and business-to-business (B2B) commercial transactions between customers, suppliers and business partners. Examples are shopping, banking, shipment tracking, travel booking, procurement, and many other business related activities. Transactional websites require integration with back-office systems and content is dynamic. Users of transactional websites operate desktop and tablet computers to interact with the site. The architecture of these applications varies, examples are bespoke, on-premises packages and software-as-a-service.

4. Back office

Back office websites are corporate applications, including ERP and other core line-of-business systems, finance and accounting, customer relationship management, business document exchange and employee self-service portals. Content is dynamically served from relational and document databases. Typically, users are employees operating large tablet or desktop computers with one or more screens. Back office systems integrate with transactional systems and provide data, business logic and workflow services.

How Visual LANSA Helps you get Web Development Right

Recognizing the characteristics of website types helps architects and developers decide how to design applications and identify tools best suited to building them. Developers can use Visual LANSA to build any of the website types but where is it the most suitable tool?

1. Information

Information websites can be built with Visual LANSA, but it may not be economically viable. The competitive space is full of great tools that specialize in these type of websites, including templates and non-programming tools.

2. Social and community

Developers can build social and community websites using Visual LANSA. However, many open source frameworks are available on which to assemble social and community websites.

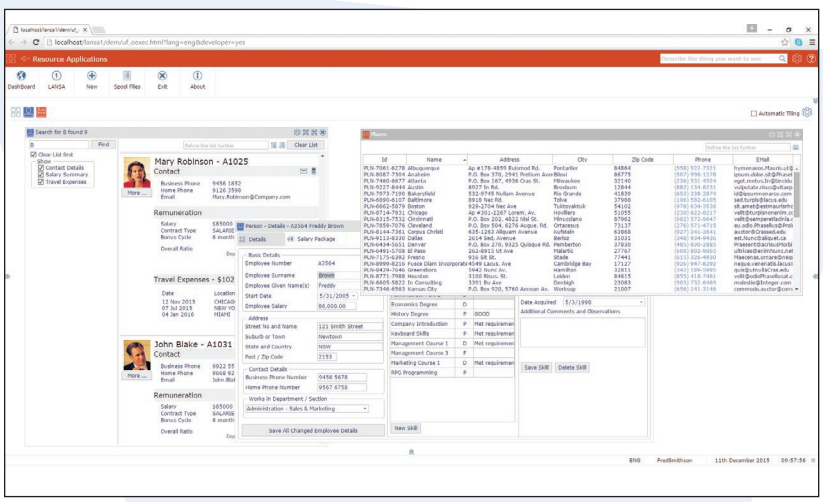
3. Transactional

Visual LANSA is very well suited to build the business logic and application functionality of a transactional website, as well as to create a sophisticated web user interface, and to integrate the website with back office systems.

4. Back-office

Visual LANSA is very well suited to build applications to manage back office activities. The Visual LANSA Framework also fits perfectly into this type of web application. Enhancements and new applications plug into the framework without disruption to existing framework applications.

Visual LANSA is best suited to building transactional and back office web applications. Developers use one language to build business logic, validation and user interface and can use LANSA Integrator to support integration interfaces with other systems. It's all right up LANSA's alley. ■



Visual LANSA is very well suited for developing transactional websites, back office applications and self-service portals.

A tour of Visual LANSA Version 14 features

Development features	
Develop web applications without coding HTML, CSS or JavaScript	Build web applications for smartphone, tablet, laptop and desktop devices without coding HTML, CSS, JavaScript, PHP, or RPG.
A single high-level language for all development	Use one programming language to build client-side (browser) and server-side components of web applications.
Web applications that look like desktop applications	The user interface is generated in the same way as it is for desktop applications. Visual LANSA manages screen rendering and updates only those areas of the screen that change. The whole page is never reloaded.
Powerful client-side processing	The Visual LANSA runtime engine runs in the browser which simplifies client-side development allowing developers to build stateful, desktop style web applications.
Browser agnostic	Applications work with all browsers and no browser specific program code is required.
Out-of-the-box application templates	Application templates are provided for developers to use as the basis for generating their web applications.
Write once reuse many times	Build web applications and user interfaces with components that can be used over and over to ensure consistency across applications.
Third party JavaScript libraries and frameworks	Developers can also use popular JavaScript frameworks and libraries and add their own JavaScript to applications.
Mobile and offline features	
Applications can use mobile device features	Integrate with mobile device features such as the camera and geolocation.
Applications can operate offline	Build web applications that can operate with or without a connection and then synchronize data when a connection is available.
Design features	
Easy responsive design	The Layout Manager includes features to make responsive interface design easier; e.g. use drag-and-drop to design multiple user interface formats.
Business rules repository	All business rules reside in a single repository so that data validation, event triggers and other rules can be isolated from the web applications that use them.
Data element definitions repository	Data element definitions include type, size, presentation and validation, e.g. the way that a telephone number is displayed and validated is defined in one place and used across all applications.
Cloud features	
Cloud based development option	The Visual LANSA Integrated Development Environment is available to code and test applications using AWS or Azure.
Completed web applications can be deployed in the cloud	Version 14 includes a deployment tool to upload and configure web applications to AWS and Azure.