Servoy Design Principles

Servoy is both a Rapid Application Development (RAD) tool and a deployment platform, designed and build to create rich, data centric applications, that can be securely deployed over the internet or locally to any platform with as little code and effort as possible.

Using industry standard components and protocols, Servoy offers a scalable, secure and stable enterprise-level computing platform, while at the same time offering unparalleled developer productivity.

Servoy's unique architecture provides developers with a single solution to create applications that, from a single codebase, can be deployed on any platform, on many different devices, in whatever deployment model one chooses, supporting different databases and multiple languages at the same time.

The Integrated Development Environment (IDE) of Servoy, Servoy Developer, allows developers to focus on creating the required functionality to solve business requirements in the most agile way possible.

In This Chapter

Servoy Design Principles

The Servoy platform is built around the following principles:

Scalable, Secure & Stable

The Servoy platform is designed from the ground up to provide a scalable, secure and stable platform. Servoy builds on proven industry standards that are combined together and made easy to use. It provides out of the box clustering support, database connection pooling and active data management.

Using a multi-tiered setup, it provides separation of concern and unparalleled scalability.

Agile development

Out of the box productivity:
Servoy's aim is to bring 4GL-like productivity, without the usual limitations that come with 4GL. From a single codebase, applications can be deployed on any platform, any database, on any device and in the deployment model of your choice, without having to write specific code or making special builds.

Features like Security, Audit Trails, Internationalization are built in, though fully configurable.

Servoy defines a lot of behavior out of the box, but also provides the developers with the means to override the default behavior to fit their needs.

Client, Database, platform and device agnostic development

When developing a solution with Servoy, it will out of the box run in all Servoy Clients (Smart Client, Web Client, Headless Client and Batch processor), can be deployed on any platform, connected to any database and in any deployment model one chooses.

Servoy's built-in datalayer through which all the interaction with database is done will make sure that a Solution will run on any database. For example, during development the developers can use the bundled PostgreSQL, while during deployment at Customer A the Solution is ran on Oracle 8, at Customer B on Oracle 11 and at Customer C on MySQL.

Solutions developed with servoy will be deployed to end-users through either the Servoy Smart Client or the Servoy Web Client. The Servoy Web Client runs in all modern browsers on any platform and the Servoy Smart Client uses Java WebStart technology to deploy the Smart Client to Windows, OSX, Linux or any other Unix variation.

WYSIWYG UI Design
The Servoy Developer IDE comes with a full-featured, What You See Is What You Get User Interface designer, to visually create UI's.

Data-centric development/Automatic databinding

The Servoy platform included a ORM that exposes records and tables from the database as Objects within the Solution, with a extensive API for manipulating the data without having to write any SQL statement. The exposed Objects (called FoundSets and Records) expose the columns as so-called dataproviders which can be bound to the UI by simply setting the dataprovider property on the UI element. Servoy will automatically keep the UI, the datalayer inside Servoy and the backend database in sync.

Business Logic scripting

Business Logic in Servoy is written in the world most used programming language, JavaScript. Servoy provides all the tools to make writing JavaScript a breeze: a full featured debugger inside the Servoy Developer IDE, designtime code validation, full refactoring support, code formatting, code completion (also known as IntelliSense or autocompletion), extensive built-in scripting library for all common actions, with samplecode on each function etc.

Instant Developer feedback

While developing Solutions with Servoy, debug clients can be started from the Servoy Developer IDE to instantly run and test the code or start a debug session. More importantly, any changes made to the design of the Solution are instantly broadcasted to the connected debug clients. so, without having to restart the debug client or having to go through a compilation cycle, the changes can be seen/tested straight away

Modular & Object oriented development

Solutions can be built, maintained and versioned in modular fashion. A Solution can be used as a module in another solution, building a entire tree of modules, that together make up the end solution.

Besides modular development, Servoy also supports inheritance within the Solution.

And finally a UI, called a Form in Servoy, is an object that can be reused in different locations and nested into other Forms using so-called container elements like TabPanels and SplitPanes

Backwards compatibility

New Servoy version are always backwards compatible: take a Solution developed with Servoy 2.2 and import it on Servoy 6.0 and it will just work!

A new version might provide new and improved ways to do things, but old functionality is deprecated, never removed.

Single codebase

Cross platform
solutions built on the Servoy platform can be deployed to all modern operating systems, including windows, Mac, Linux, Solaris etc., without the need to write or compile platform specific code.

Multi client
A Solution developed in Servoy can be run using one or more of the different types of clients that Servoy offers:

- Servoy Smart Client: a native client
- Servoy Web Client: a browser-based, 100% HTML, CSS & JavaScript client
- Servoy Headless Client: a server-side client, without UI that exposes all business logic of the application programatically
- Servoy Batch-processor Client: a server-side client to automate tasks in the background

Cross device
Through the different clients that Servoy offers, applications built using the Servoy technology can run on a multitude of devices, including tablets, desktops, laptops and mobile phones.

Database agnostic
Servoy is not a database, as there are a plethora of excellent database engines out there, both commercial and free. Servoy can connect to any database that supports the JDBC protocol, which nowadays it nearly any database. Servoy comes bundled with PostgreSQL, an open source and free database, but, as Servoy can connect to any database, it's not mandatory to use PostgreSQL.

Due to Servoy's incorporated ORM (Object Relational Mapping) layer that provides an easy to use API for all interaction between the actual implementation of the data-layer (database x, y or z) and the application, any application created with Servoy is out of the box database agnostic, meaning that it can be deployed on any database.

The ORM layer exposes the components of the data-layer of the application as objects with an easy to use API for interaction (standard CRUD operations and more) and allows for easy linking of UI elements like fields and grids to the data-layer, eliminating the need for manually writing SQL statements.

Deployment model agnostic
Whether in the Cloud (public or private) or on Premises, whether in Client-Server or stand-alone setup, whether single or multi-tenant (SaaS), Servoy supports all these deployment models out of the box. The development of applications in Servoy is to a great extend agnostic to the deployment model and where differentiation is needed (for example single vs. multi-tenant) this differentiation can be incorporated within the same code-base.

Easy deployment

Zero install Clients
Servoy applications do not have to be installed on the client machine.

In case of the Servoy Smart Client, the application will automatically download and install itself, by accessing just a URL on the Servoy Application Server from the client machine.

In case of a Servoy Web Client, just accessing a URL on the Servoy Application Server will launch the application.

Remote management

The Servoy Application Server hosts the Servoy Admin page: a web-based portal to administer the Servoy Application Server remotely

Auto update

Rolling out a new version of a Solution with Servoy is a breeze: From Servoy Developer an export is made of the Solution, which is then imported through the Admin page of the Application Server, after which the new version is available.

By default, on import on the Servoy Application Server Servoy will update the datamodel to the version required for the new version of the Solution, but like with all things, the default behavior can be overridden by the developer to create their own mechanism.

The Servoy Smart Client will automatically update itself on launch when it detects that there is a new version available on the Servoy Application Server. The Web Client is launched on the Servoy so will also always automatically runt he latest version

Built-in versioning

Each version of a Solution that is imported on an application Server is recorded and can be rolled back with the click of a button through the Admin page

Standards based

Servoy tries hard to not invent the wheel, but to bolt all the useful wheels out there onto one car and make them all work together.

Servoy utilized many open standards and makes them easy to use.