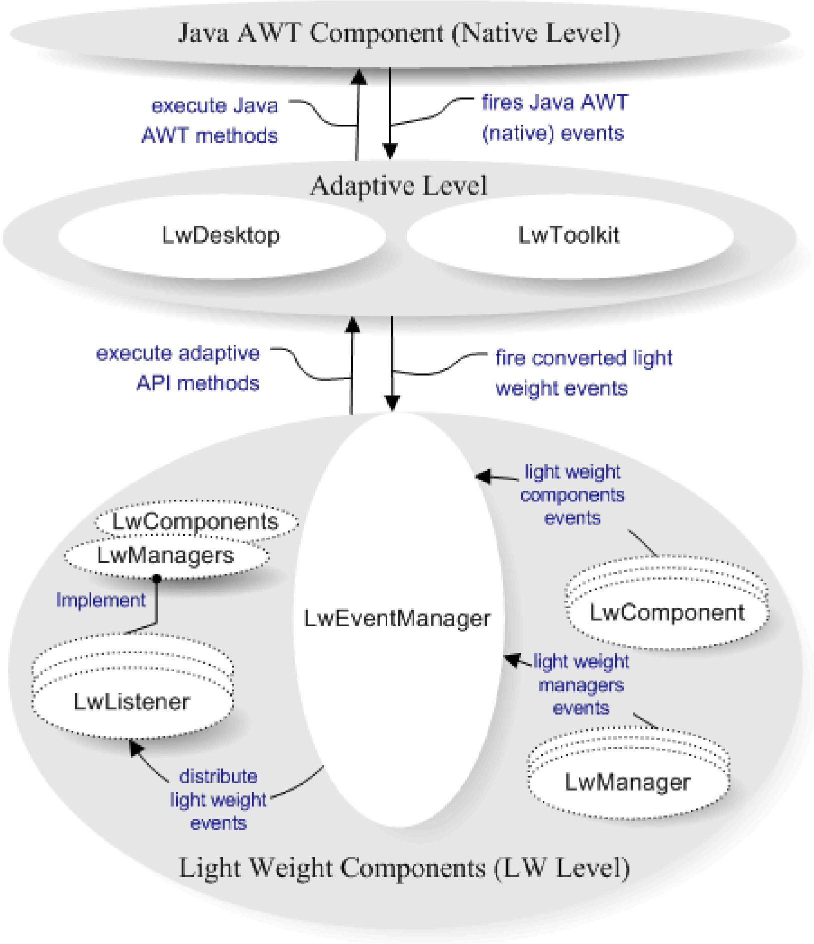
Home
Products
Library Overview
Library Architecture
Technical summary
Competitive Analysis
Documentation
- IT Specification
- View API Online
- Tutorial
Software Acquisition
- Online Demo
- Download
- License it!
- Online Demo (Night build)
- 3rd-party Extensions
Quality Assurance
- Quality policy
- Test Protocol
- Test Plan
Maintenance and Support
- Support
- Further Product Plans
- Contact Info
Legal Information
- Distribution License

Architecture

The core idea is to reject **java.awt.Component** inheritance. The library provides **LwComponent** interface that is basic for all components of the library, instead of **java.awt.Component**. The library architecture is shown below.


Java AWT Component (Native Level)
execute Java AWT methods
fires Java AWT (native) events
Adaptive Level
LwDesktop
LwToolkit
execute adaptive API methods
fire converted light weight events
light weight components events
LwComponents
LwManagers
Implement
LwEventManager
LwComponent
light weight managers events
LwListener
LwManager
distribute light weight events
Light Weight Components (LW Level)

The lightweight library has three levels:

1. Level 0 (**Native**). This is Java AWT area. In common case the level represents a Java UI implementation. As rule this level provides *java.awt.Graphics*, *java.awt.Image*, *java.awt.Canvas* or their analogues. Actually it is enough for Adaptive level to be easy implemented.
2. Level 1 (**Adapting**). This level adapts native events to appropriate lightweight events, provides ability to work with graphical context and has a set of native-specific methods that are necessary for the library. *LwDesktop* adaptive class is responsible for native events conversion process and it provides set of methods to work with a graphical context. *LwToolkit* adaptive class provides set of native-dependent methods. This level is very important to use the library on other systems that have their own Java UI implementations (for example portable devices). To adapt the library it is necessary to re-implement the level classes and you shouldn't touch lightweight components set.
3. Level 2 (**Light Weight**). This is a core level. This level provides all classes and components that are necessary to develop your own systems and components based on the lightweight library.
   Level can be divided as follows:
   a. Lightweight components. This is set of widespread GUI components that are ready to be used.
   b. Lightweight managers. The **LwEventManager** is the basic manager for the library. The manager defines message delivering strategy and should be defined for the library. Another important manager is a paint manager. The manager is used with the lightweight component implementation to paint itself. The managers can be listeners for different lightweight events. For example, focus manager should listen mouse event to pass focus between components. The managers are initialized with

lightweight manager (**LwManager**). Any managers can be overwritten by anybody. The manager deployment concept is powerful and easy to use.

So, LwVCL has 3-Tier architecture that separates functionality on three independent levels. This feature makes easier porting the library to other systems (portable devices, for example) and makes the library more flexible. The library event distributing mechanism can be easily modified according to the user requirements.

For more information about the product, please contact Zaval CE Group directly in a free form.

Thank you,
The Zaval CE Group.