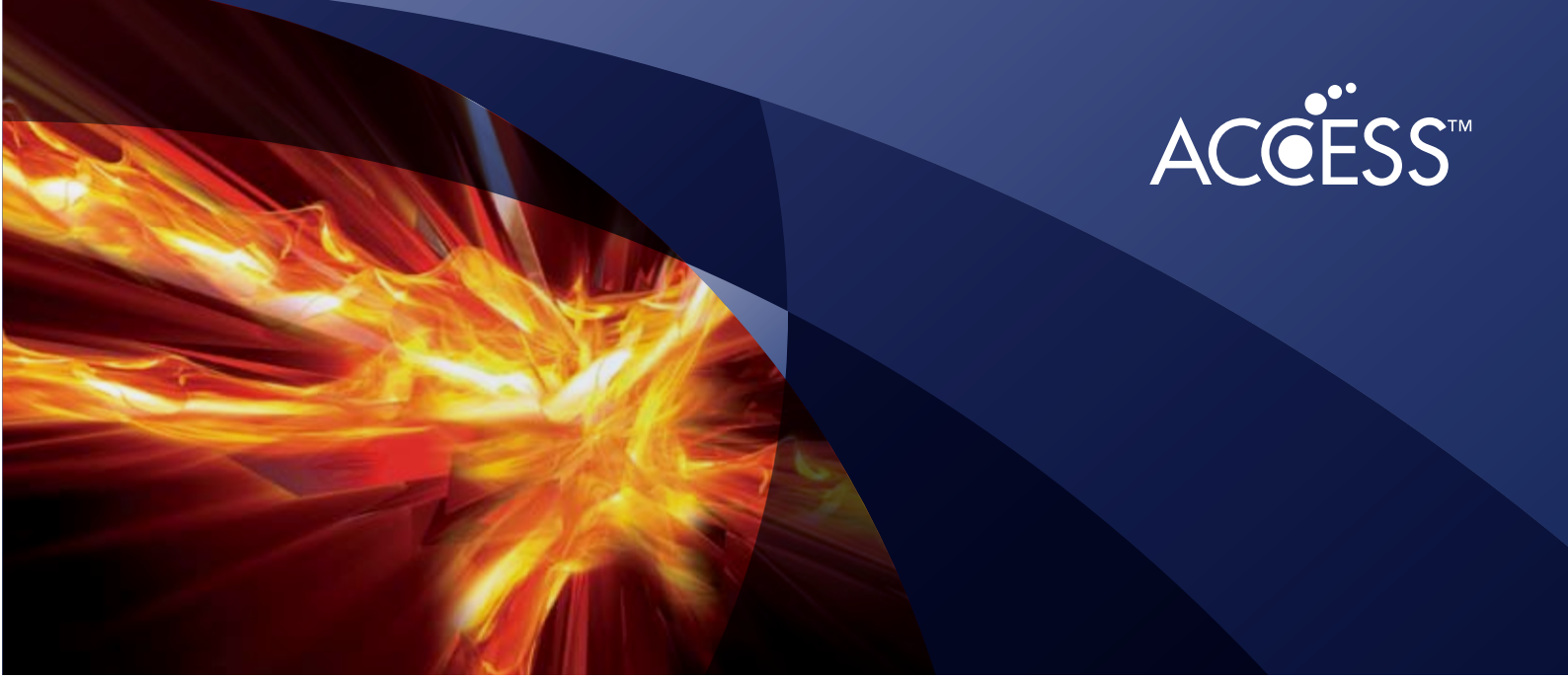

ACCESS White Paper

Open Creativity with the ACCESS Linux Platform™ Advanced UI Engine

Open Creativity with the ACCESS Linux Platform™ Advanced UI Engine

Overview

Today's mobile device, more than ever, must offer a great user experience. The device has to ease navigation and also encourage the use of new services. As a result of these converging needs, what has emerged as a key factor in the users' buying criteria is the user interface (UI). To provide the user with a fun and engaging experience, the UI therefore, must make use of high impact visual effects such as smooth transitions when switching applications. The UI must intelligently expose the features from the increasing list of services and applications that run concurrently on mobile operating systems. While these needs are well understood by mobile operators and handset manufacturers, there are still many barriers in their quest for a competitive look and feel. Indeed, even if visual and interaction designers are given carte blanche, their creativity has to take into account what factors limit their designs in the area of visual effects, the customization and operating system features. Modern UI engines, the software components in charge of displaying elements on the screen such as buttons, text or images, must remove the boundaries on the designers' imagination. The right UI engine should open creativity, not set limits.

A competitive and differentiated UI needs more from the graphical engine than just displaying objects and images on the screen. It requires platform-wide high performance graphics rendering capabilities allowing cutting-edge effects involving advanced visual treatments such as translucency, dynamic shadowing, and object manipulation. Nevertheless, since open mobile operating systems UI engines provide a similar level of features with support for a limited set of animations and effects, UI designers have little room for liberty and thus competitiveness driven by differentiation.

Another key aspect of the mobile device look and feel is that it allows handset manufacturers to differentiate and mobile operators to convey their own branding. Whether the mobile software platform is open or closed, you have to customize it based on the targeted market segments and their specific needs. The operating system and applications UI have to be completely remodeled for each segment. This activity incurs significant delays and costs in software refactoring and thus in time-to-market delivery. In addition, customization is often limited to icons, fonts, color scheme, button shape and so on. Therefore, the customization is not as deep as expected and does not completely fit the market segment.

Interaction and visual designers also have to deal with the important number of events that can occur on a mobile device and operations that the user can concurrently perform. The generalization of mobile data services, including pushed data, such as e-mails and social networking requires a deep interaction between applications to exchange data such as multimedia files, contact information and so on. To absorb the high degree of complexity introduced by these services and to create a consistent and seamless user experience, the UI engine has to be deeply integrated to take full advantage of the platform capabilities.

The design of a competitive user interface is a very complex task that should not have to take into account architecture and software limitations. A quick look at today's open mobile platforms, able to power high-end mobile devices, and their UI engines, shows that only a few have visual effects enabled applications other than the Home Screen--the first application you see when you power on your mobile device. Instead of contributing to the operating systems' UI benefits, this mix of standard and boosted

applications with their own look and feel introduces a lack of consistency in terms of user experience. The user may wonder why one application has a menu but not another or why there is a transition effect in one application but not in the other applications. The presence of only a few visually boosted applications reveals information about the platforms they are running on for the following reasons: 1) the software components used to handle visual treatments in those applications are not part of the platforms' core UI engines; all applications would otherwise benefit these components 2) UI engines used on those platforms do not provide support for fancy effects; if so, there would be animations and transitions everywhere 3) the UI customization capabilities of those platforms are very limited; you can easily name each platform by simply launching a couple of applications. This is the current state of the market. It does not have to remain this way. Why is it that a mobile operating system could not embed a UI engine with Hollywood-style visual effects? Why not provide this UI engine to the whole platform? And why not provide a UI that licensees and operators can endlessly customize to look like what they want. It is possible to have advanced, open, platform-wide, easily accessible and complex UI design capability all at once. ACCESS provides these capabilities. ACCESS has designed the ACCESS Linux Platform advanced UI engine to allow the creation of compelling, engaging, and highly customizable user interfaces available across the platform.

The Advanced UI Engine

The ACCESS Linux Platform advanced UI engine leverages the most recent hardware chipsets with support for graphical hardware acceleration. It relies on open-source projects and industry standards, such as OpenGL® ES, to make maximum use of hardware capabilities for graphical processing. Its architecture allows rendering applications in a single process to handle concurrent applications with different graphical environments, such as 2.5D and 3D user interfaces, Java, and GTK and Garnet applications.

The advanced UI engine comes with a complete set of standard and advanced UI widgets, common buttons, checkboxes, lists, and so on. Each UI widget has two parts: the data held by a component called a control and the visual aspect held by another component called the actor (see Figure 1). The UI widgets expose properties and behaviors letting developers chose between simple controls and complete customization of the UI.

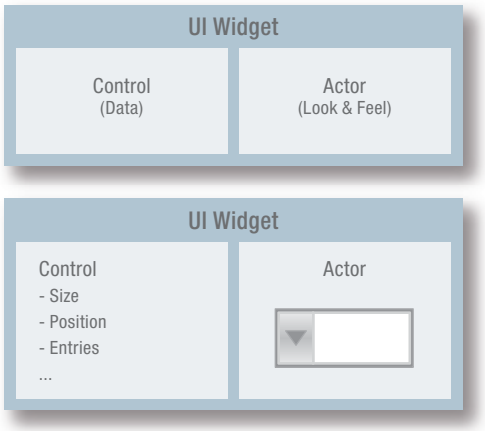

Figure 1: *UI Widget Division (Control and Actor) and a Combo Box Example*

Applications developed using ACCESS Linux Platform advanced UI engine are divided into two different parts: application logic that focuses on the data the application manages or the services it provides, and the UI description files where all the visual elements a re handled. The scriptable interface provided by the UI engine allows detaching the application from the visual aspect of the UI widgets. Thanks to this architecture, the complete look-and-feel of the ACCESS Linux Platform can be changed by simply replacing the actors without modifying the application code, thereby reducing development effort and time to market.

Unlike other mobile device UI Engines, the ACCESS Linux Platform advanced UI engine does not impose a specific execution model on application developers. Its tight integration with the operating system allows leveraging the advanced features as well as the power and openness of a Linux® based platform. Based

on the application requirements, applications can be developed to initiate concurrent tasks in order to gather information from various sources, such as contact database, web pages, video, and so on. They can also manage their own lifecycle and decide when to be executed and even exited. For example an application can be automatically launched when a data connection is available, interact with the user, and run until completed without closing any other applications.

ACCESS Linux Platform and the new UI engine provide all the key ingredients for creating a successful user interface and positive user experience.

Key Features

High performance, server-based graphics rendering: The ACCESS Linux Platform advanced UI engine client-server architecture allows for the separation of applications from the visual aspect. Since everything is rendered in a single process, there are no limits in terms of visual manipulation. It is possible to mix various applications interfaces into a 3D-based environment for performing state-of-the-art transition effects. Other applications are not even aware of what is currently happening on the screen.

Hardware accelerated visual processing: The advanced UI engine leverages hardware capabilities through the use of OpenGL ES 2.0 functions. Consequently, it takes advantage of the graphical processing unit to perform highly complex rendering. This means that the visual processing is isolated and does not drain cycles from the overall ACCESS Linux Platform processes.

Simple and high-level description interface: The advanced UI engine provides a simple and efficient interface for developers to create rich and complex applications. This flexible interface allows developers to choose between simple high-level control over objects and fine control of specific parameters or specific objects creation.

Replaceable look-and-feel: In the ACCESS Linux Platform advanced UI engine, applications and the system user interface employ a specific interface to communicate with the elements that implement the actors. To change the look-and-feel, only the actors have to be replaced, which has no impact on the application code.

Rapid application development through scripts: The advanced UI engine only needs two scripts to process the visual part of an application: the scene description file, which handles the visual layout and transition effects; and the control description file, which handles the data associated with each actor rendered on the screen, such as labels and status.

Coexistence with other application environments: The advanced UI engine allows rendering in different application environments. This means that ACCESS Linux Platform applications can co-exist with applications from other environments, such as GTK+, Java, and Garnet.

Tight integration of the UI engine and middleware: The advanced UI engine is designed to fully leverage the platform's middleware features. Thanks to this architecture and tight integration, applications developed for ACCESS Linux Platform have no limitations over the execution model. They can leverage capabilities in terms of multi-tasking, interruption management, and navigation.

Advanced UI Engine Architecture

The heart of the ACCESS Linux Platform advanced UI engine architecture is the Render server. The Render server runs in its own process and performs the entire rendering into a device's display. Many games use this model, which helps prevent the rendering from stalling due to system load. This also enables the creation of complex, fluid transition effects between the visual components of applications.

An advanced UI component consists of a control and an actor.

- **Controls:** A control consists of a component that runs in the application process (a client control) and a component that runs in the URS process (the server control). A form is a special control that provides a container for one or more UI widgets. The control-actor model separates the business logic and interactions of an activity from its visuals. This enables the application user interface to be completely replaced without modifying the application itself.
- **Actors:** Each server control is associated with an actor running in the Director. The server control provides the mechanisms for the actor to request the data it needs from the application and to pass user events back from the actor to the corresponding client control running in the application's process.

How it Works

An application specifies its UI in its associated control description and scene description files (CDF and SDF). When an application is launched, the Render server reads the CDF and SDF files, then creates all the controls and factors of the application's form. The CDF specifies the set of controls, their layout, how data is associated with the controls (properties) and persisted between application invocations, and which UI events the controls respond to and their behavior upon receiving those events. Depending on the criteria, the Render server also loads a corresponding SDF. The SDF specifies which actors that the Render server uses for rendering, thus creating the look-and-feel of the application. One or more SDF files may be associated with an application's CDF. Depending on the configuration of the device running the application (for example, display dimensions and orientation), the Render server will load the corresponding SDF, as shown in Figure 2.

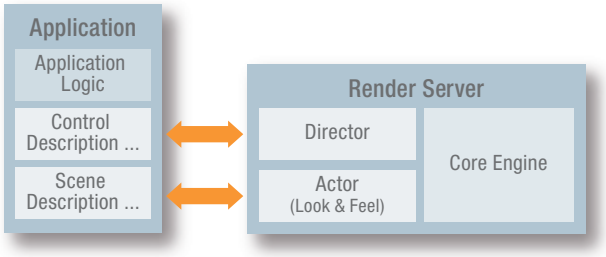

Figure 2: *Control Description File and Scene Description File*

Key Benefits for Licensees and Mobile Operators

The ACCESS Linux Platform advanced UI engine allows development of highly customized and competitive user interfaces. Its extensive toolkit gives creative freedom to visual and interaction designers so they can provide the sophisticated interaction experiences coveted by today's end users.

Handset manufacturers and mobile operators can create an engaging, interactive user experience with animations, transitions, and photo realistic 2.5D and 3D visual effects. The advanced UI engine offers flexibility for developing new user interfaces, either by developing a complete, new user interface with their own UI widgets, transitions and applications; or by reusing and customizing existing UI widgets and visual effects.

Reducing time to market is a key concern for licensees. To that end, the advanced UI engine provides a fast, easy way of customizing the standard look-and-feel of the platform and application set. Updating the look-and-feel of the actors using description files allows for the creation of a differentiated and competitive UI without involving a team of designers with advanced programming skills. The simple scripting interface also removes the need for long learning curves for development and helps reduce development complexity, which is always a source of uncertainty in planning UI design projects (see Figure 3).

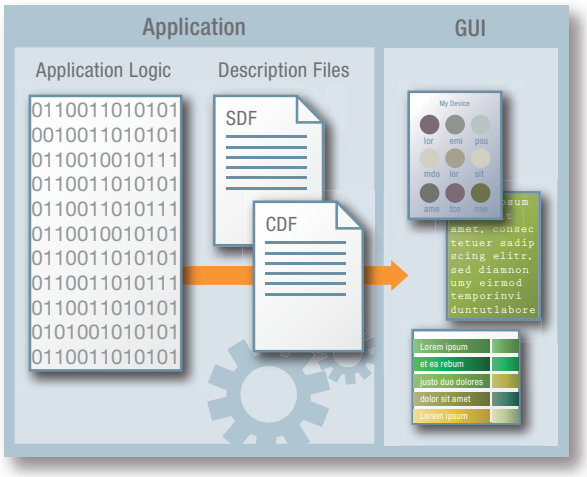

Figure 3: *Change script files to create many GUIs, without having to change your code*

The Advanced UI engine provides mobile operators with a powerful and efficient way to communicate their branding and services portfolio to customers while satisfying their demand for a sophisticated device experience. It provides capabilities beyond simple theming with merely changing basic elements, such as icons, fonts, control shapes, and color scheme. Instead, the advanced UI engine lets carriers take full control over the user interface and allows them to create and convey a branded experience. The power of ACCESS Linux Platform advanced UI engine, combined with the platform's capabilities and functionality, as well as ACCESS's expertise in developing operator extensions makes ACCESS Linux Platform the platform of choice for carriers.

ACCESS Linux Platform integrates a new advanced UI engine that removes boundaries to visual and interaction designers' creativity. It allows the development of sophisticated and engaging 2.5D/3D user interfaces through visual effects and animations. The advanced UI engine gives an unlimited control over the software platform and application's user experience. Thanks to its singular architecture, handset manufacturers and mobile operators can change the whole look-and-feel without changing the source code.

ACCESS™

About ACCESS CO., LTD.
ACCESS CO., LTD., is a global company providing leading technology, software products and platforms for Web browsing, mobile phones, wireless handhelds and other networked devices. ACCESS' product portfolio, including its NetFront™ Browser, Garnet™ OS (formerly Palm OS®) and ACCESS Linux Platform™, provides customers with solutions that enable faster time to market, flexibility and customizability. The Company, headquartered in Tokyo, Japan, operates 29 subsidiaries and affiliates within Asia, Europe and the United States. ACCESS is listed on the Tokyo Stock Exchange Mothers Index under the number 4813. For more information about ACCESS, please visit http://www.access-company.com/.