

IP2014 Beta Deployment Guide for Siebel Open UI clients on 8.1.1/8.2.2: Document 1499842.1

September 5, 2014

Holger Herbert, Siebel CRM

Oracle

Table of Contents

Revision History	5
Overview	7
User Experience	7
Modern Design.....	7
New Look and Feel: Setting End User Preferences	8
Providing a Highly Optimized User Experience through Design.	8
Deployment and Integration.....	8
Deployment & Integration Features	9
True standards based Web Based client.....	9
High-Level Migration Tasks	9
Development: Configuration and extension of the UI.....	10
Summary	10
Feature overview and changed in Siebel Open UI.....	10
Features that change in Open UI	10
High Interactivity mode features available in Siebel Open UI	13
Open UI capabilities for converting Standard Interactivity (SI) mode content	16
Open UI Client Requirements	17
W3C Accessibility ARIA & WCAG2.0 AA.....	19
Accessibility Implementation: Landmark Roles in WAI-ARIA	19
Application Wide Navigation Model and Landmark Roles	20
Keyboard Navigation within Widgets/Applets.....	20
Keyboard Shortcuts.....	20
Accessibility features	20
Open UI Server Deployment Characteristics	21
Siebel Releases Supporting Siebel Open UI	21
Security	21
Server Side Memory Footprint for Clients	22
Firewall and Proxy/Reverse Proxy Server requirements	22

Web Server Performance tuning for Siebel Open UI	22
Client-Server Communication & Network Characteristics.....	23
Open UI Performance	23
Best Practices for Customized Applications: Optimizing Virtual Screen layout.....	24
Implementation Differences Related to Browsers.	24
Use of Popup Blockers:	24
Multi select features in Siebel	25
Multi Select List control Features	25
Multi Add Item Features in Customer Order Management	25
Multiple Sessions and Tabbed Browsing:	25
Impact of dynamic patching of browsers, browser interoperability:	25
Validating W3C compliance of Siebel client side code and customization.....	25
Features of the Open UI clients	26
Choosing a UI for Siebel CRM	26
Customizations of Open UI	27
Mitigating Deployment Risks	27
The Public JavaScript API	27
CSS Changes	27
Migration Tasks related to Open UI.....	28
Open UI and Legacy Browsers	28
Running HI Clients in Open UI environments	28
Migrating SWT Files	29
Migrating JavaScript and CSS to RWD (Responsive Web Design).....	29
Migrating SRF files.....	29
Migrating browser scripting.....	29
Document integration with Open UI	30
File System use in a Web based application	30
Behavior of SI Views inside RIA Mode	31
Migration Considerations for Calendar features	31
Migrating Existing COM Integration	32
Use of SWE API in the application.....	32
SWE API support in Portlet Integrations. (8.1.1.11+/8.2.2.4+).....	33

Disconnected Use with Open UI	34
Use Case #1:	34
Use Case #2:	34
Use Case #3:	35
Integrations related to Open UI	35
Test Automation	35
Out of Box public test automation interfaces for Open UI (applies to 8.1.1.11+/8.2.2.4+)	35
Oracle Automation Test Suite (OATS) Integration	36
OATS Functional Testing and Siebel Test Automation for Open UI (STAO) (applies to 8.1.1.11+/8.2.2.4+ only)	36
Oracle Automation Test Suite (OATS) Integration with Siebel	37
OATS Load Testing	37
Oracle Business Intelligence Enterprise Edition (OBIEE) Integration	37
Oracle UPK Integration	37
Siebel Product Configurator (8.1.1.14/8.2.2.14 and above only)	38
Side by Side Support Product Configurator Support for Open UI and HI	38
Migrating AIA integrations	38
Enabling Signature Capture	38
Reference Information	38
Browser Performance	38
Other Resources	39
Siebel Bookshelf	39
On My Oracle Support	39
Data Sheets	39
Standards supported by Siebel CRM / Open UI	39
Open UI Portal Standards Support	40
Client Authentication Options	40
Lightweight Directory Access Protocol and Active Directory Services Interfaces	40
Web SSO Integration Requirements	41
Client Mandates	41
Common Criteria Certification, Evaluation Assessment Level 2	41
FIPS-140-2 standard	41

Open Web Application Security Project (OWASP).....	41
Oracle Software Security Assurance (OSSA) guidelines and policies.....	41
Payment Card Industry Data Security Standard (PCI-DSS).....	42
Accessibility VPAT	42
VPAT information.....	42
EU privacy directives:.....	42
Open UI Web Authentication Options:.....	42
SAML and Single Sign-On	42
IPV6	42
Open UI Web Servers and IPV6.....	42
Open UI Clients and IPV6	43
Third-Party products in Open UI (8.1.1.14/8.2.2.14).....	43

Revision History

Version	Change
14.0	Initial Beta Guide

Disclaimer

This document in any form, software or printed matter, contains proprietary information that is the exclusive property of Oracle. Your access to and use of this confidential material is subject to the terms and conditions of your Oracle Software License and Service Agreement, which has been executed and with which you agree to comply. This document and information contained herein may not be disclosed, copied, reproduced or distributed to anyone outside Oracle without prior written consent of Oracle. This document is not part of your license agreement nor can it be incorporated into any contractual agreement with Oracle or its subsidiaries or affiliates.

This document is for informational purposes only and is intended solely to assist you in planning for the implementation and upgrade of the product features described. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described in this document remains at the sole discretion of Oracle.

Due to the nature of the product architecture, it may not be possible to safely include all features described in this document without risking significant destabilization of the code.

Overview

To respond to the need for broad support of clients and devices, to ease deployment, and to improve the user experience for end users, Oracle provides the standards based, open client. Oracle's Open UI uses a single framework to meet the demand for a variety of use cases. Employee or customer facing, mobile or desktop, touch or keyboard /mouse use cases as well as accessibility can all be supported with this framework.

Oracle's Open UI client is will work on any modern HTML5 browser and will be easy to deploy as an employee and partner-facing rich internet application (RIA), providing a modern alternative to Siebel high-interactivity (HI) applications or (SI) applications or other CRM applications in the market place.

User Experience

Users continue to leverage the familiarity gained through prior Siebel versions but supplement usability from modern JavaScript based consumer applications, used by roughly 60% of all leading web based applications. Many aspects of the UI have been redesigned to bring new highly intuitive features to the client.

Modern Design

- Full support for mobile devices: Fully device driven layout using responsive web design principles for the application to match the browser and device form factor
- Multiple navigation personalization options, such as tab, left hand and "Side" Menus
- Fully accessible without configuration
- Simplified user experience through graphic icons and new themes.
- Personalization options themes.
- Easy to customize HTML5 based web templates.

The screenshot displays the Oracle Open UI interface for Service Requests. The top navigation bar includes 'ORACLE' and various menu items like 'File', 'Edit', 'View', 'Navigate', 'Query', 'Tools', and 'Help'. Below this is a breadcrumb trail: 'Home > Accounts > Contacts > Administration - Product > Opportunities > Quotes > Fleet Management > Administration - Fleet Management > Sales Orders > Service > Calendar'. The main content area is titled 'All Service Requests' and shows a table with columns: SR #, Account, Last Name, Owner, Priority, Recommendation, Sub-Area, Summary, SR Type, and Area. The table lists several requests, with the first one (SR # 83602-370817) selected. Below the table, a detailed view for SR 83602-370817 is shown, including fields for SR #, Work Phone #, Summary, Last Name, First Name, Description, Account, Site, Date Opened, Area, Status, Endment, Date Committed, Subarea, Substatus, Product, Date Closed, and Source. The 'Status' is set to 'Open' and 'Substatus' is 'Unassigned'.

Figure 1: Modern user experience on any device

- Modern Controls

- Grid with ability to support forms
- Map Integration ability (requires configuration)
- URL Integration ability
- Date and time fields
- Can be used keyboard only or tablet only.
- WAI-ARIA Navigation model
- Calculator
- Lists a& Forms with image support
- Multiple file upload/download Control
- Analogous Pick lists and MVG's
- New Menus and toolbars
- Calendar
- New Look and Feel
 - SmartScript
 - Task Based UI
 - Customer Order Management
- Native Features:
 - Browser history management
 - Print and print preview
 - Zoom
 - Search
 - Save Page
- Development features:
 - Ability to Extend Siebel UI

New Look and Feel: Setting End User Preferences

Open UI provides new user preferences under Tools, Preferences, and then Behavior. Users can set left-hand navigation or tabbed UI options as well as setting themes for the UI.

Providing a Highly Optimized User Experience through Design.

Open UI enables implementing highly tuned design patterns to help our customers envision the design possibilities and simplification they can bring to their Siebel implementations. UX Direct <http://www.oracle.com/webfolder/ux/applications/UXDirect/index.html> introduces a User Centered Design Implementation process providing tools and resources to get started. Open UI allows for a highly optimized user experience through its rich extensibility features when combined with those design best practices. Oracle also provides content on the usable apps site at: <http://www.oracle.com/webfolder/ux/applications/index.html>

Deployment and Integration

The Siebel application architecture consists of multiple layers;

- Data layer (database tables and columns)
- Objects Layer (BOs and BCs)
- UI Layer (Views, Applets controls)
- Presentation Model and Physical Renderer (View and Applet Web Template).

Siebel Tools allows configuration of Data Layer, Objects Layer and UI Layer. Bindings between UI layer and Physical UI Layer were also created in Siebel tools.

Open UI provides the ability to fully modify the physical UI layer. Customers are open to create unlimited customization. The new client provides true Web-based client deployment. To log in, users must have a URL and credentials. There is no requirement for browser configuration.

Deployment & Integration Features

- Supports full extension of the presentation model and physical renderer to meet any CRM use case
- Application Focus State management capabilities
- Full branding capabilities
- Hierarchical, embeddable Presentation Model.
- Partial refresh
- iframe based portal integrations

True standards based Web Based client

- URL & Credentials only
- HTML5 – minimum requirement
- CSS3 – supports advanced styling options.
- JavaScript (Standard ECMA-262 conformant)

The latest versions of Internet Explorer 10/11, Google Chrome, and Mozilla Firefox are fully supported via HTML5 standards. This platform supports Android and IOS platforms such as iPad, iPhone, Android phones and tablets, RIM, Windows Phone as well as other standards compliant client and mobile platforms. It should be noted that Open UI is not optimized for Gesturing but does take advantage of the native gesturing features in mobile browsers. Siebel Mobile provides the optimized mobile solution that tightly integrates with mobile device features using standards.

High-Level Migration Tasks

The migration to Open UI can be achieved in parallel to the existing environment without a significant development effort. The user experience will be similar enough not to require retraining of end users. (It is recommended that this is verified during the deployment process.) Open UI can be added in parallel to the regular upgrade so that it does not affect the existing environment, if planned properly. The migration consists of adding object managers with the open UI parameter, which becomes available as part of Open UI releases of Siebel. Once enabled, the customer can start offering the Open UI option to a subset of the user population to test the new client to ensure the readiness for the customer's specific functional requirements. As the features enabled by Open UI meet acceptance requirements of the test population, other users can start using the new client with the enablement of a sufficient number of Open UI enabled Object Managers. This activity has no bearing on the functioning of the existing ActiveX client or SI user population. The URL for Open UI will be different from the standard URL and available in parallel to the existing ActiveX client.

The advised high-level roadmap for customers follows:

- Deploy the latest available Siebel release compatible with Open UI
- Apply latest [patch set](#) applicable to the release

- Read release documentation provided for the migration and assess impact:
 - Migrate from SEA to SIA if on a SEA release.
 - Assess and Run the IRM upgrade steps
 - Assess and convert Client side browser scripting
 - Assess and convert COM integrations
 - Assess mail Integration changes.
 - Review lifecycle support for Oracle and Third party integrations such as OBIEE, BIP and Java
- Add an object manager for the UI, and set it to Open UI using the Open UI enablement settings.
- Test the Open UI client rich internet application (RIA) deployment. This can be tested in parallel when running the environment if using HI or SI
- Perform user-acceptance testing for all the deployed client UI options.

Development: Configuration and extension of the UI

To achieve the best possible user experience, Open UI provides a fully published JavaScript API to build powerful CRM solutions. That API allows customers to easily extend the capabilities of Oracle provided features. The new API supports anything from simple validations to replacement of whole UI components that require rendering in special ways. In addition Siebel's new implementation of CSS allows extreme ease of branding the user experience. Literally any aspect of colors fonts, layout and styling can easily be manipulated in Siebel.

Customers can continue to use tools they are accustomed to or use the new web development extensibility to build features previously unachievable using traditional means.

Summary

These changes enable the ability to provide improved usability, deployment, development and integration options. Customers implementing Siebel today can expect many benefits of deploying Open UI. More information on Open UI can be found at <http://my.oracle.com/go/Siebel>

Feature overview and changed in Siebel Open UI

Customers are advised to carefully review the feature mappings provided in the tables below. Siebel Open UI can reside side-by-side on the same server with existing Siebel CRM implementations and can run against the same SRF as existing SI or HI deployments

Note: It is strongly recommended to apply the latest available Fix Pack and [patch set](#) for Open UI for best functionality and support.

Features that change in Open UI

Open UI deviates for specific features from the way Siebel has worked in the past. This should be considered when implementing Open UI,

- Attachments: Attachments can only be edited "on the fly" in Open UI if Java is present on the system. If not, end users must download the document into a secure location specified by the browser and upload when finished editing. This is a standard behavior of true web based

applications. The section on File System use in a Web based application below, provides alternatives of this feature in a true web based application.

- Applet Location Editor is not required as Open UI Dynamically locates UI elements.
- Client Side logging is not required as all logging happens on the server side.
- Drag and Drop: Drag and drop from Outlook to the file attachment applet is not supported only from a file directory or desktop.
- Notifications for Universal Inbox are not available in Open UI.
- Outlook/F9 integration: F9 integration can only be supported via the Siebel email client due to integration limitations between client side desktop applications and web browser security models. Using the integrated Email client within Siebel provides many benefits. The internal email client is able to capture customer responses, not just outbound mails in the customer profile. So all communications are captured, not only the outbound half.
- Targeted Account Selling (TAS) Coach is not available in Open UI
- Asset Dispatch board is not available in Open UI
- Artifacts in the UI: Some of the features not available will still have artifacts present in the user interface to maintain consistency with the existing user interface rendering. Notable Items are:
 - SI applet artifacts rendered in Open UI
 - Buttons for features not available in the application toolbar such as
 - legacy printing
 - spellcheck options

Customers have the option to remove those UI elements by manipulating Siebel configuration via Siebel tools or using supported web development methods such as post-loading CSS style sheets to remove the artifacts. In some cases planned target releases of the related features are shown below in the “Roadmap Target” column.

- Advisor/eAdvisor is replaced by Need Analysis for Commerce solution which offers similar or greater capabilities than Siebel Advisor, including the use of OPA’s natural language rules modeling and interview automation
- Auditing for the Print Feature: Since Siebel Open UI makes use of native print features of the browser, rather than proprietary printing features, customers are advised to use external audit capabilities to audit printing use of end users. Open UI does not support print audit features.
- Double click: Double click causes novice users great difficulties and since it conflicts with the single-click interaction style of the Web and many mobile devices in addition to offering great draw-backs to accessibility, has been eliminated to improve usability. Double click is not common to all operating systems.
- Right click: Right Click is not supported as it lacks good usability characteristics. It is not discoverable, does not meet accessibility requirements and is very difficult to implement on mobile devices. As a result it has been removed from the application user experience.
- Escape key to undo: Escape key is only supported for a single undo operation. Pressing the undo key (ESC) multiple times may result in differing behavior depending on the user interface element/ control used.

- Calendar: Open UI requires an explicit click to make a selection instead of predicting a selection based on current date.
- Close/ Exit browser: Exiting the browser using the “X” / Close rather than using logoff will not terminate the session in a W3C application as this does not inform the server the client has disconnected. This scenario is handled via session timeout. IT is recommended to set session timeout as short as supportable for usability, while training users to use the logoff button.
- Forms: Open UI form applets support drilldowns, while HI forms do not.
- Implicit commit on browser back button, Logoff on Exit: These functions cannot be supported in a standards based implementation due to web browser/W3C limitations and thus behave differently than in HI.
- Images and screens:
 - Login/logout pages have a new look (can be reverted to old page look and feel)
 - Task bar icons are more modern with higher resolution and color depth (can be reverted)
 - Buttons are updated with higher resolution and color depth (can be reverted)
 - Oracle Logo font and color change (can be reverted)
 - Tab bar has no images for consistency
 - Updated checkbox look (can be reverted)
 - New “columns displayed” icons (can be reverted)
- MVG/Shuttle applet: Shuttle applets have been redesigned for improved usability and to make them more consistent with pick lists. This may require review with users that have historically been accustomed to shuttle applets.
- Pick lists and LOV’s do not select a new value until a unique match is made. If using auto fill without an exact match values will revert to original on tab.
- Product Promotion Schedule administrator’s view is now managed through Administration -Product -> Promotion view.
- Scrollbars are a common feature of Windows applications and ActiveX based applications. They do not work well with mobile devices, where swiping is the preferred use case. Because scrollbars tend to deteriorate usability there are no horizontal scrollbars used out of box. This deterioration is exaggerated for large datasets which make scrollbars hard to use as well as the associated performance degradation. To emulate specific behaviors scrollbars can be added using configuration changes. For large datasets customers are encouraged to use search specification features to narrow down datasets represented in fields.
 - Vertical Scrollbars. Since most lists in Siebel tend to be long only the forward and back button for moving through lists are supported out of box.
- Quotes are not included when you select from an LOV in Open UI. This is a change from HI/SI. For users that transition from HI/SI to Open UI. Double quotes are required for strings.
- Notes fields are dynamically sizeable and in line with the user interface, but upon resizing these may alter the screen layout dynamically.

High Interactivity mode features available in Siebel Open UI

The following table provides a high level list of features that are available in Open UI. The Notes/Siebel Versions column indicates that the initial Siebel Open UI release providing this feature. Additional data is made available in the Siebel Bookshelf with the standard bookshelf updates.

Table 1: HI/RIA Features Enabled

Product Feature	Product Feature Delivered	Notes/ Version
Application Menu	Menu selection: Eliminate duplicate features, support keyboard use and accessibility	n/a
Application Toolbar / Buttons	Buttons in the Application Menu. For information on Buttons, see the associated features	n/a
Browser History & Bookmarking	Full native support for Browser History implemented in URL's and back / forward button of browser.	n/a
Calculator	Calculator: supports additional features such as percentages,	n/a
Calendar	Calendar related functions.	n/a
Charts	Charts	n/a
Chat	Chat: Call center chat API	(8.1.1.11/8.2.2.4+)
Checkbox	Checkbox	n/a
(COM) integration	Implemented via Cross platform JavaScript API : works with all browsers	n/a
Customer Dashboard	Customer Dashboard and associated button in Call Center	(8.1.1.11/8.2.2.4+)
Customer facing self-registration for RIA	Self-Registration : customer and partner portals, self-registration in Customer facing applications	(8.1.1.11/8.2.2.4+)
Customer Order Management (COM)	Browse Catalog (Quote), Browse Catalog (Order) Views, Edit Promotion UI, Unified Messaging: product recommendation Unified Messaging framework validation for passive (list applet), active (pop-up)	(8.1.1.11/8.2.2.4+)
CTI Toolbar	CTI communications toolbar in Call center.	n/a
Date picker, Date features	Date picker, Date features	n/a
Dialogs and Custom Popups	Dialogs and Custom Rendered Popups	n/a
Dispatch Board	Dispatch Board in Field Service	(8.1.1.11/8.2.2.4+)
Drag and Drop column and calendar Features	Drag and Drop Column reorder and drag and drop within calendar	(8.1.1.11/8.2.2.4+)
Email response /	Native Outbound Mail Client and native Email response advanced tracking in Service	n/a
File upload and download	File upload and download	n/a

Product Feature	Product Feature Delivered	Notes/ Version
Email Invoice	Ability to email an invoice from the Invoice Screen in the Service Application.	
Find Features	Application Level find, Binocular Find: Database Search using Find across application; Binocular find	(8.1.1.11/8.2.2.4+)
List View / List Applet	List View options: Drag and drop for columns, save state for column options. Settable Column width	(8.1.1.11/8.2.2.4+)
Images (BMP/PNG/GIF)	Images (BMP/PNG/GIF) : improved accessibility through accessibility metadata, higher image quality	n/a
Loyalty Program Designer	Loyalty Program Designer Open UI Control has an improved user interface with the new control in IP2014.	(8.1.1.11/8.2.2.4+)
Marketing Allocation	Manages marketing programs and Campaigns and segmentation	(8.1.1.11/8.2.2.4+)
Marketing Calendar	Marketing Calendar : Used to render the Event schedule	(8.1.1.11/8.2.2.4+)
Marketing HTML & Rich Text Editor	Marketing HTML & Rich Text Editor: Design HTML email offers, Web offers, event web banners.	(8.1.1.11/8.2.2.4+)
Message Notification	Notification UI comprising Notification counter, Summary Pane and Detail Pane. Distinguish Read/Unread notifications and dismiss notifications	n/a
MVG/Pick list	MVG/Pick list: MVG's and pick lists now provide a similar look and feel for ease of use.	n/a
Product Configurator	Documented user Interfaces	n/a
Customer Order Management (COM): Catalog Navigator	Browse Catalog View: Catalog Navigator Pop up list in Brose catalog quotes/ order has been replaced by new Browse Catalog View functionality in Open UI" in Customer Order Management	n/a
Print/Print Preview	Print Preview: Native browser print dialogs and previews.	n/a
Search, Applet Level	Applet level structured Search, PDQ	(8.1.1.11/8.2.2.4+)
Spellcheck	Native Spellchecker support.	n/a
Proposals	Proposals & Attachments in Sales and Marketing	n/a
PRM	PRM(HI Mode)	(8.1.1.11/8.2.2.4+)
Query Application Toolbar	New Query, Execute Query	(8.1.1.11/8.2.2.4+)
Reports/ BI Publisher	Selected records reports using multilingual templates, administration, scheduled or parameterized reports. frame integration or Reports notifications using notification UI in Reports	(8.1.1.11/8.2.2.4+)
Resource Scheduler	Resource Scheduler for Field Service, Marketing Calendar, Hospitality function-space diary. (Gantt chart)	(8.1.1.11/8.2.2.4+)
HTML Editor	HTML Editor for general use	n/a

Product Feature	Product Feature Delivered	Notes/ Version
Right to Left Language support	RTL Support: RTL Language support for ARA and HEB languages	(8.1.1.11/8.2.2.4+)
Search, Advanced	Search Center Advanced Search and Search button	n/a
Search, OSE	using Oracle Secure Enterprise Search (OSSES)	n/a
Siebel Test Automation	STAO interface using new integration	(8.1.1.11/8.2.2.4+)
SmartScript Player	SmartScript UI Renders the questions, answers and input controls.	n/a
SWE API	SWE API: GotoView Login, Logoff, ExecuteLogin, GotoView SWE API in portlets	(8.1.1.11/8.2.2.4+)
Sum total for list view	Sum Total List View	(8.1.1.11/8.2.2.4+)
Tabs	Tabs: supports dynamic resizing, keyboard use and accessibility	n/a
Task Based UI	Provides the Task Based UI	n/a
Time Picker, Time Features	Time Picker, Time Features	n/a
Tooltips	Tooltips	n/a
Tree Control	Tree Control	n/a
Zoom	Zoom: zoom features are now fully supported via browser	n/a
Alerts & Alarm, Siebel	Used for immediate notification	(New in 8.1.1.14/8.2.2.14!)
Attachments	Ability to save directly to File System	(New in 8.1.1.14/8.2.2.14!)
Barcode Toolbar	Barcode Toolbar: In Open UI, the barcode query capability is integrated with the Global Search and the New/ Update capability is provided in the relevant View	(New in 8.1.1.14/8.2.2.14!)
Drag and Drop Feature (Calendar)	Provides drag and drop items from an applet onto Siebel Calendar. Life Sciences calendar allows dragging a contact record from a list applet onto the Siebel Calendar in order to create a contact visit record.	(New in 8.1.1.14/8.2.2.14!)
Expression/Rules Designer	Design Time: Personalization business rules designer.	(New in 8.1.1.14/8.2.2.14!)
Funnel Chart	Design & run Time: sales and sales-phases pipeline analysis charts in Sales	(New in 8.1.1.14/8.2.2.14!)
Gant Chart/Resource Scheduler	Consumer Goods Promotions, Asset DB Planned Gantt Chart Applet, Life Sciences Call Planning, , eAuto Dashboard	(New in 8.1.1.14/8.2.2.14!)
HTML Healthcheck	Checks browser compliance with the Open UI feature set	(New in 8.1.1.14/8.2.2.14!)
Hierarchy View,	Provides hierarchical view of accounts, Contacts	(New in 8.1.1.14/8.2.2.14!)

Product Feature	Product Feature Delivered	Notes/ Version
IHelp	Unstructured help	(New in 8.1.1.14/8.2.2.14!)
List View / List Applet	Lock/freeze on list columns	(New in 8.1.1.14/8.2.2.14!)
Marketing Program Designer	Design Marketing Programs and Multi-stage campaigns in Marketing. . The new control delivers better user experience and a fresh new look at designing a Campaign.	(New in 8.1.1.14/8.2.2.14!)
Maps Integration	Mapping ability using Oracle Spatial out of box.	(New in 8.1.1.14/8.2.2.14!)
Order to Cash	Siebel AIA Order to Cash features with Oracle Configurator in Customer Order Management	(New in 8.1.1.14/8.2.2.14!)
Product Configurator	Administration, Product/Definition, Constraint (Rule) Builder	(New in 8.1.1.14/8.2.2.14!)
Query on checkbox	Checking for checkbox values inclusion of a “view all” option	(New in 8.1.1.14/8.2.2.14!)
Resource Scheduler for CG	Life Sciences Call Planning, Consumer Goods Promotions	(New in 8.1.1.14/8.2.2.14!)
Signature Capture	Signature Capture	(New in 8.1.1.14/8.2.2.14!)
SmartScript Designer	Ability to design SmartScript, Smart Script Path Flow Chart Admin View	(New in 8.1.1.14/8.2.2.14!)
Siebel Anywhere	Siebel Anywhere support in Remote	(New in 8.1.1.14/8.2.2.14!)
SWE API	SWE API: GotoView Login, Logoff, ExecuteLogin, GotoView SWE API fully accessed from external integrations, session detail available through business service from browser and server script.	(New in 8.1.1.14/8.2.2.14!)
TAS Competitive Analysis	Organizational Analysis in Sales	(New in 8.1.1.14/8.2.2.14!)
Web Services	Web Services Administration Inbound/Outbound	(New in 8.1.1.14/8.2.2.14!)

Open UI capabilities for converting Standard Interactivity (SI) mode content

SI client deployment features are not supported for automatic conversion to Open UI, and not all SI features allow manually migration to Open UI. Migrating applets or views to open UI requires HI equivalent applets or views in Siebel Tools. If RIA/HI view equivalents are available, Open UI can be used to render the views and user interface. Customers are strongly advised to convert SI features to Open UI. Examples of features that can be supported via Open UI that traditionally only exist in SSSI are:

- Application Menu (SI)
- Info Center (SI)
- Helpdesk Online (SI)

- Forgot Password (SI)
- Login (SI)
- Partner Locator (SI)
- Salutation (SI)
- Self-Registration (SI)

These applet types can be partner or customer facing but can be supported by Open UI, either by conversion or customizing for new deployments.

Open UI Client Requirements

Open UI client requirements are listed on [My Oracle Support Certifications](#) for all certifications of Siebel. Many questions about Certify are answered in: [My Oracle Support Certification Search - Frequently Asked Questions \(Doc ID 1491004.1\)](#).

Siebel Open UI will perform best on modern browsers that comply with W3C standards. Open UI is built on the HTML5 standards. Oracle is committed to supporting Web standards to ensure a problem-free user experience while using Siebel CRM in Open UI.

Siebel Open UI supports browsers if they conform to the following browser standards:

- **HTML 5:** HTML5 is the minimum requirement for a browser to take full advantage of the features in Open UI. <http://html5test.com/results/desktop.html> provides an excellent overview of HTML5 compliance.
- **JavaScript.** Siebel Open UI requires support of JavaScript (ECMA-262) for the browser
- **CSS.** CSS3 is the minimum requirement for a improved end user experience.
- **WAI-ARIA** Accessible users accessing the browser must use a WAI-ARIA compliant Accessibility Technologies such as screen readers etc. for full accessibility support. It is important to choose the most accessible components available and Oracle recommends requesting input from the accessible user population for commonly used technologies.
- **Java (Optional)** If Java is present on the client system File attachments can be edited in place and saved back to the file system.
- **JavaScript Performance:** The performance of the chosen browser and device/hardware has a large impact on client performance. Browsers can be compared using <http://octane-benchmark.googlecode.com/svn/latest/index.html> The Appendix contains performance numbers for common browsers.

Oracle tests Open UI internally against commonly found enterprise browsers:

- Internet Explorer 11 including latest released versions)
- Chrome 30 (and above including latest)
- Firefox 29 (and above including latest)
- Safari 5.1.7 (and above including latest – Mac Only)

Support of browser specific defects is handled by the browser vendor

- Apple Safari: <https://bugreport.apple.com/> or Apple Support
- Microsoft Internet Explorer: or Microsoft Support
- Mozilla Firefox: <https://bugzilla.mozilla.org/page.cgi?id=productdashboard.html> or Mozilla Support
- Google Chrome: Settings > About Chrome > Report an issue or Google Support

Some, but not all, known limitations related to standards and KPI's required by Siebel Open UI for specific browsers (Safari Internet Explorer, Firefox, and Chrome) are listed below. For other less common browsers please use the links to review support of your browser.

- **Chrome (All Versions), Safari (All Versions):** Accessibility can only fully be supported on WAI-ARIA browsers. Browsers listed with partial support at the following link are not recommended for accessibility <http://caniuse.com/wai-aria>.
- **Internet Explorer (All Versions), Safari 3.2 (and below) :** Browser implements Textarea resize in CSS such as notes fields using vertical scrollbars or not at all, while other browsers implement as resizable text boxes in multiple dimensions leading to differing user experience <http://caniuse.com/css-resize> , http://www.w3schools.com/cssref/tryit.asp?filename=trycss3_resize
- **All Browsers (Optional):** If Java is not present on the client system:
 - File attachments cannot be edited in place and saved back to the file system without intermediate storage on the local system.
 - CTI hoteling support will not be available.
 Oracle recommends use of latest versions of Java for security. Use of Java is optional. <http://java.com/en/download/index.jsp>
- **Internet Explorer 9 (and below):** Enhanced long column field validations such as in Description fields are only supported in HTML5 Browsers. <http://caniuse.com/#search=validation>. Text Field length cannot be controlled, causing potential data loss on saving field values.
- **Internet Explorer 9 (and below):** Potential login logout/session timeout issues with SSO related to XMLHttpRequest support. <http://caniuse.com/xhr2>
- **Internet Explorer 9 (and below):** Performance on most recent desktop system does not reach an Octane 2.0 score of 12000+ on a I7 processor 2.0GHz system. Browsers can be compared using <http://octane-benchmark.googlecode.com/svn/latest/index.html>
- **Internet Explorer 9 (and below):** Animated Chat notification for multiple Chat windows will not be available as these use the following CSS3 feature: <https://developer.mozilla.org/en-US/docs/Web/CSS/animation>
- **Internet Explorer 9 (and below):** Bookmark features / browser history and fully defined URL's are features only available in HTML5. <http://caniuse.com/#search=history>
- **Internet Explorer 9 (and below):** Drag and Drop support in Open UI can be supported only in HTML5 capable browsers. <http://caniuse.com/#search=drag>

Lack of use of fully HTML5 compliant browsers or other requirements above may limit user functionality across the application.

Open UI has no specific hardware requirements beyond the minimum requirements posted by the browser, device or operating system vendor it is running on. Modern hardware and browsers are

recommended as these will provide the best user experience and performance. It is recommended to perform acceptance testing against the actual environment that is used.

Note: Customers are strongly advised to carefully choose browsers and perform acceptance against the provided and customer based performance success criteria. Modern browsers or improved hardware may assist in providing resolution to performance issues.

W3C Accessibility ARIA & WCAG2.0 AA

Siebel CRM has added the benefit of keyboard-only support, which can be leveraged by high-performance keyboard users and accessible users alike. It can be used in conjunction with screen readers and other assistive technologies supporting WAI-ARIA.

WAI-ARIA <http://www.w3.org/WAI/> is an initiative by W3C to provide rich internet applications with accessibility. WAI-ARIA aims to systematize the design of Web-page accessibility for such applications. Open UI aggressively meets web standards from W3C, including WAI-ARIA markup and is a leading enterprise application in adopting assistive technologies standards from the W3C.

Note: To use Siebel Accessibility end users must use WAI-ARIA compliant Browsers and AT's such as Screen readers and other assistive technologies to fully leverage Siebel Accessibility.

WAI-ARIA compliant screen readers and other accessibility technologies provide facility to show the navigable regions and to interpret WAI-ARIA markup present in the Siebel application. WAI-ARIA provides a role attribute to mark Landmark regions. Only a subset of these features is available without support of WAI-ARIA.

Open UI has been designed from the start to meet WCAG2.0 AA accessibility standards where possible and practical. Section 508 and draft mandate 376 are covered in large part via WCAG2.0 AA conformance. Oracle has taken leadership in not only attempting to meet mandates, but to provide very practical and usable business solutions for accessibility enabled business users and consumers.

Accessibility Implementation: Landmark Roles in WAI-ARIA

WAI-ARIA provides a collection the landmark roles that can be applied to each of the navigable regions identified for the Siebel application. Users of all abilities can navigate directly to the various application widgets like Application Menu, Toolbar, Applet, and so on by using Assistive Technology shortcut keys.

Open UI has marked up the following landmarks:

- Application Page identifies the Application Page Region.
- Application Menu identifies the Application Menu Region.
- Application Toolbar identifies the Application Toolbar Region.
- Application Screen Bar identifies the Application Screen Bar Region for navigation to views.
- Predefined Queries identifies the region that contains the Predefined Queries.
- Visibility Dropdown identifies the region that contains the Visibility Dropdown.

- Main View identifies the Main View Region that contains the various applets.
- Applet identifies the Applet Region.

To understand landmarks in more detail, see: http://www.w3.org/TR/wai-aria-practices/#kbd_layout

Application Wide Navigation Model and Landmark Roles

WAI-ARIA improves Siebel CRM navigation features. To understand WAI-ARIA guidelines for tabbed navigation between widgets, see: http://www.w3.org/TR/wai-aria-practices/#kbd_general_between

The Tab and Shift Tab keys move focus among widgets and standard HTML controls. As an example, a typical List Form view, the tabbing sequence would be:

Application Menu Bar -> PDQ dropdown -> Toolbar -> Visibility Dropdown -> List Applet Menu -> List Applet Buttons (one by one) -> List Applet Column Headers (one by one) -> List Applet Grid First Row First List Column Control -> Form Applet Menu -> Form Applet Buttons -> Form Applet Controls (one by one).

Keyboard Navigation within Widgets/Applets

Once within the widget use the following keys to handle tasks:

- The arrow keys to navigate.
- The enter key to drill in.
- The escape key to back out.

Top Level composite widgets, such as the Application Menu Bar, the Applet Menu, Tabs at any of the four possible levels, the List Applet Grid, and so on have complex navigation requirements between their constituent elements and widgets. There are a set of general recommended principles to follow for such navigation. Details are available at: http://www.w3.org/TR/wai-aria-practices/#kbd_general_within

Keyboard Shortcuts

Siebel CRM continues to support keyboard shortcuts, similar to the HI and SI client shortcuts. In some cases, the shortcut combinations have changed, in particular in cases where they interfere with common browser shortcuts by browser vendors for browsers now supported via standards support, or where they interfere with accessibility standards. In general the WAI-ARIA based keyboard navigation model described above is simplest to use, but keyboard shortcuts can be very useful for performing of repetitive and frequently used tasks. More general information can be found at http://www.w3.org/TR/wai-aria-practices/#kbd_shortcuts . Siebel specific shortcuts are referenced in the *Siebel Open UI Fundamentals* Guide.

Accessibility features

Open UI has many features designed specifically for accessibility. A brief list follows:

- Always on accessibility. Accessibility is always enabled for all users, not just accessible users. No action is required to enable accessibility.
- Dynamically generated accessibility data for the entire rich internet application

- Full screen reader compatibility through use of WAI-ARIA metadata for labels images and other application information.
- Enablement of accessibility for dynamic content within the application; WAI-ARIA markup for dynamic events like notifications, popups and dynamic visual clues such as flashing buttons or hourglass-type indicators

Open UI Server Deployment Characteristics

Siebel Releases Supporting Siebel Open UI

Siebel Open UI requires the adoption of the latest available Siebel fix pack and [patch set](#) for Siebel CRM version 8.1.1 or Siebel 8.2.2. To deploy Open UI, customers must have migrated to the SIA version of Siebel. The latest software continues to support the high interactivity and standard interactivity clients in addition to the new Open UI.

Security

Siebel Open UI is the most secure Client available in Siebel. Oracle strongly recommends using Open UI for demanding implementation supporting strong security requirements.

- Limited Attack Surface: Siebel Open UI uses only three technologies to render the client code: CSS, HTML, and JavaScript. The small set of underlying technologies required and the absence of third-party plug-ins, such as ActiveX and Java reduces the Open UI client's vulnerability to attack while maximizing its capability.
- Transparent Technology: The Siebel Open UI client is built in accordance with World Wide Web Consortium standards, a variety of modern inspection tools can be used to validate the security compliance of the Siebel implementation.
- Compatibility with hardware security features and virtualization: Because the Open UI client is a scripted client, it is fully compatible with DEP/NX operating system features and with virtualization functionality, and is flexible in supporting a variety of implementation techniques for secure environments.
- Session security and concurrency: To protect data integrity and system security, Siebel Business Applications do not allow concurrent Siebel sessions to run in a browser. The application provides a list of options when this condition is detected. The Siebel Server also implements a sophisticated session security model to provide a very secure end user experience.
- FIPS-140: Siebel Open UI can support the FIPS-140-2 standard with applicable configuration.
- Common Criteria Certification: Siebel Open UI supports Common Criteria Certification for Siebel CRM version 8.1 and later to level Evaluation Assessment Level 2. For additional information, see: http://www.commoncriteriaportal.org/files/epfiles/st_vid3026-vr.pdf for additional information.

Siebel Open UI implements a number of JavaScript controls from third-parties. Oracle tests these controls to the same rigorous standard that is applied to Oracle code. For example, Oracle validates the security of the Open UI client code:

- Using industry-leading Web application security assessment solutions
- By working with the ethical hacking community to ensure code undergoes practical, real-world, security testing

Note: Siebel Charts/ Visual Mining Netcharts use is not recommended with Open UI due to concerns with the Heartbleed vulnerability potentially affecting embedded Tomcat Web Server versions in Visual Mining Netcharts. Customers are advised to remove Netcharts until the concern with these vulnerabilities has been remediated. Siebel Open UI does support a JQuery based alternative to Netcharts which is secure.

Oracle takes part in the Open Web Application Security Project (OWASP) and Oracle's Software Security Assurance (OSSA) policy conforms to the guidelines provided by OWASP. The Oracle Software Security Assurance policy requires developer training in secure coding, and comprises guidelines, policies, and secure coding standards that all Oracle product development teams are required to follow.

Server Side Memory Footprint for Clients

Resource use for Open UI is very similar to the ActiveX UI. The work done on the server is similar for Open UI as the HI object managers. Because end users tend to use only one client at a time, the two object managers if used in parallel for the same users, do not necessarily require additional hardware resources.

Firewall and Proxy/Reverse Proxy Server requirements

When deploying Siebel applications across a firewall, ensure that the firewall supports the HTTP 1.1 protocol. While supported, using HTTP 1.0 will result in reduced performance. If the firewall/proxy does not fully support HTTP 1.1, ensure the following:

- Web server compression is disabled. Set the DoCompression parameter to FALSE in the eapps.cfg file
- Ensure that the firewall proxy can handle cookie wrapping or other proxy-specific features which enable cookie handling and forwarding.
- Ensure that headers passed to the SWSE plug-in do not contain HTTP 1.1 protocol content. This can be automated by setting the proxy to strip HTTP 1.1 header content.

Web Server Performance tuning for Siebel Open UI

Though Siebel Web Server installations attempts to perform some amount of optimization, it is recommended that customers run performance tuning plugins such as YSLOW (which runs on Firefox) to verify good configuration tuning of the web server. More information can be found at the following links: <http://developer.yahoo.com/performance/rules.html> and <https://addons.mozilla.org/en-US/firefox/addon/yslow/>. YSLOW for well configured system will generally provide a "grade" of 85 or above. More information about performance rules can be found at: <http://developer.yahoo.com/performance/rules.html>. Settings that should be made for the web server platforms follow: used are:

Note: For IIS7 and above these settings are automatically set correctly in the IIS configuration at install time. For Oracle Iplanet and IBM HTTP Server these should be optimized.

- [Minify](#) customization using a tool such as [YUI compression](#) for customer JavaScript and CSS.
- [Enabling gzip](#) compression on the web server.
- [Disabling ETag](#) on the web server. This will generally improve performance on multi-server deployments.
- [Header Expiration](#) should be set to 5 days for production deployments. Do not set this for development environments.

Client-Server Communication & Network Characteristics

Because the client server communication is very similar for both UIs the networking characteristics are similar. In support of reduced network traffic, Open UI client supports the following new features:

- “Partial refresh” that allows only distinct regions of the UI to be refreshed, rather than the entire screen. This feature saves network bandwidth.
- Push notifications which eliminate polling for communications server messages, such as CTI and other messages.

These features improve bandwidth requirements and in some cases reduce power consumption on mobile devices, by reducing “chattiness”. Siebel CRM can be run on modern wired or wireless networks, and it supports use over low-bandwidth connections.

Open UI Performance

To enable a broad variety of platforms, Siebel Open UI runs on code interpreted on the browser or JIT compiled in the browser. There are performance differences that are a function of customization of the interface when running cross platform code inherent to Open UI. With these considerations taken into account, Siebel Open UI can perform extremely well on modern browsers.

Siebel Open UI provides new features to optimize performance such as

- Partial refresh that support granularity in updating UI regions efficiently.
- Improved asynchronous operation improves performance characteristics in specific use cases of the client.
- Open UI caches client files and static Web templates and updates automatically with updated client files

Two general areas of review must be taken into consideration:

- Browser requirements related to performance are listed in the “Browser requirements” section. Browser performance impacts the speed at which code is executed
- Tuning of customization toward the use case of standards based web browser features reduces the amount of code being executed.

Performance of applications considering these application aspects can perform similar to the HI client. Customers are advised to review customization guidance prior to testing for performance in user-acceptance testing efforts.

Best Practices for Customized Applications: Optimizing Virtual Screen layout.

A web application can render content beyond the limits of the data displayed on the screen. This effectively increases the amount of processing done to render content without end user benefit. It is recommended to build UI's that render only content in the visible area of the user interface for this reason. Large rendered surfaces, the interpreted nature of code in modern web applications and the rich interactivity of the application can compound to reduce performance and response times of the UI. It is recommended to review the deployment with a focus on optimized content. In light of this there are several considerations that can be followed as a best practice.

- Simplify the use case. End users desire simplicity. This assists with reduced amount of UI and also helps on mobile devices where screen space is limited. Is the process really relevant to the requirement? Can process be made smaller without losing business process information?
- Applets: For performance and usability, do not to add more applets to a screen than the typical user can consume without scrolling.
- Modularize business flows to match screen space. Scrolling interrupts work, does not lead to good user experience on desktop and mobile devices and the content not displayed still requires rendering on screen refresh. Scrolling typically involves mouse use which interrupts more efficient keyboard activity
- Displayed columns in list views: Consider limiting columns available and set good defaults for columns displayed. For most desktops no more than 7 columns should be displayed by default as a rule of thumb. Even the components hidden have to be processed by the Rendering engine.
- Remove UI components that can be gathered by analyzing/summarizing data to leverage the ability to add value to end user tasks and reduce visible data. This could be as simple as automatically reducing data based on filters but as complex as deriving answers via analytics.

The above components have a significant impact on what all must be rendered and the various factors must be balanced and optimized for usability.

Implementation Differences Related to Browsers.

Open UI tries to hide specific differences in rendering between browsers within the UI, but there are browser specific differences. Customers are encouraged to identify differences in rendering and to choose browsers based on the best usability offered.

Use of Popup Blockers:

Open UI is compatible with popup blocker use. Ensure Siebel application servers and third party iframes are whitelisted or excluded if popup blockers are used.

Multi select features in Siebel

Multi Select List control Features

- Multi select Add is available. (8.1.1.11/8.2.2.4+)
- Checkbox based multi select Add is available in (8.1.1.14/8.2.2.14+)
- Multi select delete is not be available but the delete button will only allow focus on one record)8.1.1.14/8.2.2.14+)

Multi Add Item Features in Customer Order Management

- Multi item Add in ordering for single quantities are available. Implementing this will require minor configuration change (8.1.1.11/8.2.2.4 [latest patch set](#))

Multiple Sessions and Tabbed Browsing:

Tabbed browsing is supported with Siebel applications for only one active session. Multiple Web pages may be opened using tabbed browsing, but only one tab can have a live connection to a Siebel.

Multiple sessions can be supported only using the memory isolation options on individual browsers. These options make sure multiple tabs do not share memory space. These options are enabled differently depending on the browser chosen:

- Internet Explorer: In Internet Explorer 10 and above, use the “-noframemerging” command line option: [http://msdn.microsoft.com/en-us/library/ie/hh826025\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/ie/hh826025(v=vs.85).aspx)
- Chrome: In Chrome use the Multiple User profiles option to create a profile for each parallel session: <https://support.google.com/chrome/answer/2364824?hl=en>

Impact of dynamic patching of browsers, browser interoperability:

Specific changes in browsers may prove to be regressive to Siebel functionality. Though this is an uncommon occurrence two options exist to mitigate risks.

1. Lock browsers to specific versions: Though this does mitigate the risk of introducing regressive interaction into the Siebel environment, it also introduces risks of not patching imperfect browser code and does not help in addressing functional and security fixes in browsers.
2. Dynamically updating browsers: It is strongly recommended to mitigate overall functional and security browser risks by dynamically patching browsers. If regressive interactions occur, Oracle is committed to addressing these in a timely manner.

Validating W3C compliance of Siebel client side code and customization.

To validate a Siebel pages against the W3C standard, a number of W3C and other hosted validation services are available. These validations services include:

- HTML: W3C Html validator: <http://validator.w3.org/>
- HTML: Html5 validator : <http://html5.validator.nu/>
- CSS: CSS validator: <http://jigsaw.w3.org/css-validator/>
- WAI-ARIA : Accessibility Validator [HTML Code Sniffer Bookmarklet](#)

- LINT, ESLINT and derivatives: <http://www.eslint.com/>

Features of the Open UI clients

JavaScript Framework

The performance of underlying JQuery components and the Siebel provided application are evolving quickly. The JavaScript based client is easy to deploy and secure while not requiring any plugins for the browser.

Flexibility

In Open UI client all aspects of rendering can be controlled. Colors, fonts, borders, backgrounds, the controls themselves used for rendering can be exchanged to offer just about any user experience desired. This creates somewhat more overhead in regard to performance, when combining this attribute with improper application tuning. It is recommended to tune customized applications to take this into consideration.

Interpreted Code

Open UI: Open UI has a strong dependency on the performance of the browser's rendering engine. The section on Browser performance provides guidance on choosing the best performing clients.

Choosing a UI for Siebel CRM

Siebel CRM supports side-by-side use of all user interfaces in a single deployment and on a single server: SI, HI, and Open UI. However, each user interface must each have a configured object manager. This allows running Open UI in parallel to existing Siebel client instances. It should be noted that each UI type SI, HI, and Open UI will receive its own URL as well.

Siebel Open UI supports the existing SRF without any change to it and Siebel Web Templates that have been customized. When using Open UI, it is recommended that any customized Web templates are moved to the newly created customer files locations, as described in the bookshelf documentation *Configuring Siebel Open UI*.

Open UI provides benefits for specific deployments. It allows the migration to Open UI without any change in the developed content. This is a good option to use when migrating with minimal effort. Customers can continue to use existing browser scripting as used in HI mode without requiring any changes.

The new JavaScript API provides a more modern means of achieving what can also be performed using client-side browser scripting. Customers can write JavaScript using their own IDE and extend the provided JavaScript API. Guidance on converting browser scripting to JavaScript API content is provided in the "Configuring Siebel Open UI" document in Siebel Bookshelf.

When using the new file deployment structure, Open UI allows customers to get dynamically updated Web templates from Oracle, without the possibility of overwriting customer file content. This allows for maintenance and improvement of templates for issues related to performance, usability and standards compliance of out of box templates as part of maintenance releases. Customers are able to lock

templates from being upgraded by moving them to the customer directory structure. Customers migrating existing HI mode Web templates will be required to physically move the files into the new directory structure. Guidance for this activity is provided in the “Configuring Siebel Open UI” document in Siebel Bookshelf.

Customizations of Open UI

Developing in Open UI enables a lot of new possibilities for customers. Configuration can be performed using Siebel Tools or with standards based Web development tools to create a unique and pleasant user experience adapted to business use cases for end users using Siebel applications.

Though Open UI does not require Web development activities, developing with the new JavaScript API provides a powerful tool to extend the platform.

Open UI continues to work much like the current client. If you want to customize the SRF and Web templates or write browser scripting or just leave it in a standard deployment, then you continue to have that option. Additional information will be provided in “Configuring Open UI” guide in Bookshelf

Mitigating Deployment Risks

The migration to Open UI can be achieved in parallel with the existing environment with minimal development effort. Because the migration can be performed and the addition of Open UI does not affect the existing deployment, it does not impact the existing environment, if planned properly.

Customers must take into account the constraints of what browsers are supported by each client, for example, standards compliant browsers for Open UI, while the Siebel HI client requires IE version 8 or below. Meeting both requirements allows switching back to the legacy deployment, as needed using the same browser.

If browser constraints are considered properly, then there is no risk in maintaining both environments until user acceptance tests are completed with successful results.

Note: *Before getting started, please review the list of features unavailable in the Siebel Open UI referred to earlier in the document.*

The Public JavaScript API

The JavaScript API extension is a very powerful method for experienced web developers familiar with JavaScript to extend Siebel features in Open UI. HI or SI clients do not benefit from these changes.

The preferred Integrated Development Environment (IDE) may be used to write native JavaScript code on top of the public API that Siebel CRM uses or with the JavaScript API that it uses. For detailed information, see “Configuring Open UI” in Siebel Bookshelf

CSS Changes

Oracle implements CSS for styling and layout. Open UI uses a fully updated implementation of CSS. Open UI CSS changes only benefit the Open UI client and no legacy clients.

To change CSS behavior, customers must post load the changes to CSS as opposed to changing out of box files. Post-loading CSS is a common capability and method to change CSS. Complying with this approach means customers can continue to leverage CSS improvements made as part of improving Siebel. CSS changes may be required to accommodate legacy resolutions such as 4:3 and small form factor screens.

Migration Tasks related to Open UI

Open UI and Legacy Browsers

Customers may review the following options for deployments currently using Internet Explorer 6 and 7 or 8, due to their performance and conformance limitations:

- Deploy a second modern and standards based browser for use with Open UI because of performance and standards conformance considerations. Third parties such as [Browsium's Catalyst](#) product can route traffic to browsers based on enterprise requirements
- Continue to deploy HI or SI object managers for users that only have Internet Explorer 6 and 7 or 8 support available.
- Consider use of a web browser layout engine plug-in that meets standards and performance requirements to update browsers to fast versions. [Google Chrome frame](#) may provide such an option if it meets user acceptance criteria for Siebel and other enterprise applications. Please note this option is also limited in performance, but can work for environments limited to older IE browsers.

For additional information on deploying Chrome Frame for specific applications please visit <http://www.chromium.org/developers/how-tos/chrome-frame-getting-started/chrome-frame-faq>

Note: Plug-in solutions are solely dependent on the third parties building them and customers are advised that Oracle has no part in support of those solutions.

For more information on running Open UI and HI clients in parallel please see the section "Running HI Clients in Open UI environment" below.

Running HI Clients in Open UI environments

Customers that require Internet Explorer 8 to run in conjunction with Open UI environments have several alternatives to consider. Should a requirement exist to support older IE browsers, deployment of HI clients is recommended for users on such browsers. The users can be sent to appropriate browsers using a landing page that identifies the end users browser and directs it as appropriate.

Internet Explorer 8 continues to be supported by Microsoft per the schedule outlined on the Microsoft site at [Windows 7 Lifecycle information](#). Mainstream support is available for Windows 7 until early 2015.

The recommended options are:

1. Use a secondary browser, Chrome or Firefox, to render Open UI
2. Use Citrix or Windows Terminal Services clients running on compatible client platforms such as Windows 8, Windows 7, Windows Vista and Windows XP to host IE8 on a Windows 2008 server running HI Client
3. Use Windows XP Mode to run virtualized Windows XP Clients on IE8, this option is available on Windows Vista and Windows 7 only and is supplied as a free download by Microsoft.

NOTE: If deploying Open UI and HI in parallel on releases prior to 8.1.1.11/8.2.2.4 please consult the Siebel Open UI Supplement article that is part of this MOS article for additional steps.

Migrating SWT Files

Siebel Tools development and changes in Siebel Web Templates can typically be deployed to both client types the HI client and the Open UI client. Support for these modifications follows the convention that has always been in place for Tools development on Siebel. No special changes are required for migration of SWT files with new features.

Note for Siebel 8.1.1.11/8.2.2.4: Siebel Tools does not support SWT work with non-IE browsers.

Migrating JavaScript and CSS to RWD (Responsive Web Design)

Open UI is enhanced to support RWD starting IP2014. RWD is enabled automatically in IP2014. RWD is not a mode that can be turned on or off. It's an evolution of the Open UI Framework.

RWD will not override or impact existing customization from prior releases. It only impacts the look and feel as it is a new theme that is delivered in IP2014 that adopts the RWD principles. The new theme also provides a fresh new look to the applications and enables it to run on multiple screen size devices, including touch devices.

RWD also introduces new customization methods that are not mandatory to uptake, but recommended to fulfill the RWD goals within customizations. Thus if RWD features are desired there may be impact on CSS and JS extensions in support of RWD.

Migrating SRF files

In general use cases there is no need to recompile the SRF to run Siebel Open UI outside of exceptions mentioned in this guide. Siebel Web templates can be stored to the new custom location for custom SWT's or be used from its existing location if not found at the new location. Moving the files to their new location is detailed in the "Configuring Siebel Open UI" document in Siebel Bookshelf.

Migrating browser scripting

There should be no requirement to change the client side browser scripting to run Siebel Open UI. Customer must ensure however that no browser specific JavaScript is implemented in existing browser scripts to ensure function will support cross platform browser support requirements.

Specific methods in Siebel have been enhanced or changed for a variety of reasons such as security across all user interfaces. These methods include:

- SetProfileAttr() has been disabled for HTTP calls for security reasons. If there is a browser script reference to SetProfileAttr(), it will not work for security reasons if invoked on the client side. To resolve this using browser script/tools it is recommended to create a server side business service to set the profile attribute and call it from the client. There are options to override this change, but it will reopen the security vulnerabilities. So the decision to override must be done after careful consideration of the risks if available. SetProfileAttr() access can be turned on by setting the server parameter "EditProfileAttr" value to TRUE.

Document integration with Open UI

W3C based applications typically run inside of a web browser container and are limited in their interaction with client side applications and files on end user systems for security reasons.

This dictates specific best practices when designing hosted deployments involving Siebel and the padding of data into documents and templates, both for structured or unstructured data. It is recommended to work with web centric data and forms where possible, such as XML and HTML. Where possible it is recommended to integrate Open UI with other hosted applications such as Outlook web access or other true W3C compliant client applications. These applications can be integrated via web services in the back office or via the JavaScript API at the client side, depending on requirement. By integrating only true web based client components many deployment challenges are simplified.

File System use in a Web based application

When using the file attachments and correspondence features in Open UI a changed interaction between user and Open UI Web client with legacy client side proprietary documents and applications takes place. Implementers have several options:

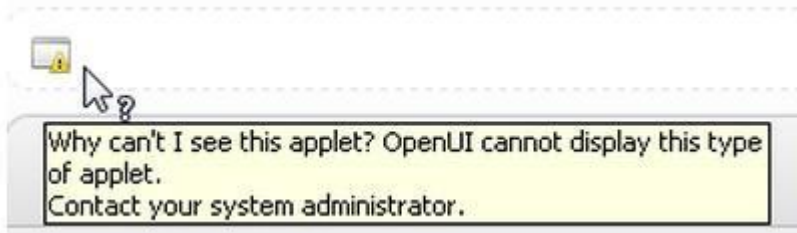
1. To save files to a directory or desktop prior to attaching to the file system. This use case also commonly occurs in Siebel when an end user tries to drag an email to Siebel file system from applications such as outlook to attachments. To best support such a use cases there are several considerations that can assist in simplifying user interactions. Users must have available a directory locally, on the network or the desktop itself to store modified files or content dragged from Outlook. This is due to the fact that such applications use different drag and drop and storage structures than most desktop operating systems.
2. Consider moving content edited in the common attachments into structured fields of data or unstructured data fields in Siebel. Only generate physical documents when all content is completed. This allows:
 - a. Formatting of content using Document Server or BI Publisher
 - b. Support of audit trails and approvals as content is changed and added by users
 - c. Obviates the need to manage content in documents.
 - d. Supports more ability to easily search the content that is modified.
3. Installing Java on a Siebel Open UI client will permit the client to download and edit attachments in place; however this does not assist in the use case between Outlook and attachments.

Behavior of SI Views inside RIA Mode

The HI client displays certain SI mode applets. When converting HI applications to Open UI there are specific rules for converting those SI applets because Open UI does not support SI mode applets. Open UI Framework will support the following scenarios only when "EnableOpenUI=TRUE" AND "HighInteractivity = TRUE". If this parameter is FALSE (Open UI is disabled), then all legacy checks will apply as shown above.

An SI applet will be handled as follows in Open UI:

- An applet with no Business Component (BC): An applet that does not have a BC, will auto upgrade to RIA since it has no custom renderers if it is SI mode.
- Applets outside of a view:
 - An applet that is SI only because of it is outside of the view (all other conditions pass for RIA qualification), will auto upgrade itself to RIA.
 - An applet outside of the view, which has custom renderers attached, will be blocked.
- Applets with renderers as base SI renderers (CSSPopupSIRenderer, CSSListPopupSIRenderer, CSSHtmlSIRenderer); will auto upgrade to HI, to the extent possible.
 - An applet that was SI mode only by the class definition in the repository is a first class candidate for a straightforward conversion. Such applets will auto upgrade to RIA. Rendering might not be complete because not all support for SI features such as links exists. Recent records applets are implemented in Open UI as client side artifacts, as an example.
- If View has multiple applets with a combination of SI and HI applets, then HI applets will be rendered and SI will be rendered as a yellow bang with a rectangular outline as shown.



In this use case the image shown with a tooltip explanation will be rendered to inform the applet cannot be rendered.

Migration Considerations for Calendar features

There are two recommended options as a workaround to retain legacy calendar behavior. The Open UI calendar out of box has been enhanced to provide additional capability.

Option 1:

This will render the basic calendar in Open UI, but retains existing function in SI/ HI to the pre-open UI function.

1. Export the following objects from the pre-migration repository and import into the migrated repository
 - a. eCalendar Daily Applet Home Page
 - b. eCalendar Daily Applet

- c. eCalendar Screen
2. Export the following objects from the migrated repository and store for use on open UI SRF if using option #2
 - a. eCalendar Daily Applet Home Page
 - b. eCalendar Daily Applet
 - c. eCalendar Screen
3. Import the objects from Step 2 into the migrated SRF

Option2:

This provides new full calendar on the home screen and one of the calendar views, while providing the old one in SI/HI.

Install separate object managers with separate SRF, one for open UI using the existing migrated SRF and one using SI/HI object managers using the SRF from Step #3.

Migrating Existing COM Integration

COM integration is a proprietary interface for the Windows platform. COM interface function in Open UI is replaced by multiple interfaces depending on the requirement.

- Oracle’s Open UI JavaScript API Interface allows extending features within the web browser to extend and integrate functionality with Siebel using JQuery controls and custom JavaScript code. The Siebel [Bookshelf Guide for configuring Siebel Open UI](#) is a helpful reference in such integrations. The JavaScript API can be best implemented with other web based interfaces.
- Web services/server side integrations. These are listed above and are recommended options for backend data integrations that may have been performed on the desktop in the past.

Note: Existing COM integrations typically require thorough review as customers should consider changing not only the integration to JavaScript but also make sure the products integrated to are also web based.

Use of SWE API in the application

SWE API's are exposed as part of XML interface to deliver data and Metadata content as described in http://docs.oracle.com/cd/B40099_02/books/PortalFrame/PortalFrameDelConExtWebApps25.html

Starting with 8.1.1.11+/8.2.2.4+FP release, Siebel Siebel portlet integrations can be performed. SWE XML features are not available in Open UI.

A summary of all SWE functions available is provided below:

Supported Values	Called within application	Called within portlet container	Called into portlet container	Description
CanInvokeMethod	yes	yes	No	Checks whether a method can be invoked on an applet, a business service, a buscomp, or the SWE application.

Supported Values	Called within application	Called within portlet container	Called into portlet container	Description
ExecuteLogin	yes	not applicable for use case	Yes	Executes login for a user.
GotoView	When invoked from the browser address bar by refresh and history navigation.	yes	yes	Goes to a Siebel view, only for view navigation
InvokeMethod	yes	No	No	Invokes a method on an applet, a business service, a business component, or the SWE application.
LoadService	yes	yes	No	Loads a business service on the server side.
Login	yes	not applicable in Open UI	Not applicable (uses SSO or similar)	Loads the login view or login page. SWE first looks at the Acknowledgment Web View property of the application object in the repository for the login view to show. If not specified, the default is the "Acknowledgment Web Page" property to show the login page.
Logoff	yes	not applicable in Open UI	No	Executes the database logoff, and then shows the logoff view or page. SWE first looks at the Logoff Acknowledgment Web Page property of the application object in the repository for the login page to show. If none is specified, SWE will show the login view or login page, depending on how you log in.
ReloadCT	yes	yes	No	Reloads personalization info. SWE loads the initial personalization on startup, and when the personalization rules are changed, SWE does not update the info automatically since there is cost in performance, so SWE provides this command to reload the info.

SWE API support in Portlet Integrations. (8.1.1.11+/8.2.2.4+)

Open UI allows the following SWE API calls in the current implementation:

Starting with 8.1.1.14/8.2.2.14 CSRF protection,

- Exposes an API to return SRN for external integrations protect from CSRF vulnerabilities
- External integrations via GET don't require SRN.

- SRN is read and sent to client only during application initialization, and SRN is only to POST
- SRN is not used in GET commands. SRN is escaped when read in GET
- SRN temporary cookies are unused.
- All applet, buscomp and service methods can be accessed only through POST (for both read and write: Explicit white listing is needed to allow GET for read only operations)
- Explicitly white listed applet/service/buscomp methods are required to allow use via GET for external integrations/bookmark operations

Disconnected Use with Open UI

The following are common disconnected use cases.

Use Case #1:

Disconnected support using remote then migrating to a next generation disconnected solution. (For existing Remote customers this is a preferred choice.)

- Migrate from HI/Remote to Open UI/ Remote in the short term
- Migrate from Open UI / Remote to next gen disconnected within the medium term. (6 -18 months)?

Benefits

- Barcode reading is supported using USB, Bluetooth etc.
- All disconnected support is there as has been in phase one and updated in phase 2
- Any Windows/Intel systems up to Windows 8.x can be used including Surface Pro 3 and convertible tablets.
- No development required in both phases.
- Meets platform architecture nicely
- Migration is low impact.

Considerations

- No iPad/Android support
- Features used must be factored in

Use Case #2:

Connected only support temporarily (Disconnected use is a future requirement)

- Use Open UI without disconnected
- Add next gen disconnected later 6-18 month timeframe

Benefits:

- Barcode reading is supported
- All architectures can be used including IOS Android
- No development in both phases
- Meets the platform architecture nicely.
- Migration in both phases is low impact

Considerations

- No disconnected use for 6-18 months
- Features used must be factored in

Use Case #3:

Use Siebel Mobile disconnected (for new custom applications with limited disconnected data requirements using IP2014 for limited offline data use cases. Can also be used for use cases where limited amounts of data can be used short term with a medium term goal of larger offline data use cases)

- Go to Siebel disconnected using IP2014
- Optional) Go to Siebel next gen disconnected container application later (6-18 months)
- Features used must be factored in

Benefits:

- Barcode reading is supported
- All architectures can be used including IOS Android
- Requires development for custom app in one (possibly both) phase(s)
- Disconnected open UI Architecture from the start
- Development for both phases has development impact due to multiple offline architectures

Considerations

- Requires changes to accommodate in disconnected architecture for phase 2 such as container
- Limited scale for disconnected use data for 6-18 months

Oracle partners offer additional offline solutions that work with Open UI and these may meet other use case requirements.

Integrations related to Open UI

Test Automation

There are multiple options for automation with Open UI:

- 1.) Use Oracle's Web standards based test automation interface to build integration. (See section below)
- 2.) Use Oracle Automation Test Suite with the Siebel Test Automation interface for Open UI – STAO. (See section below)
- 3.) Use Oracle partner solutions built on Oracle's public test automation interface.

Out of Box public test automation interfaces for Open UI (applies to 8.1.1.11+/8.2.2.4+)

Siebel Open UI supports test automation with any test automation suite that can support automation testing against true web standards based applications. Though these interfaces are rudimentary they provide a lot of capabilities required for basic task automation.

Open UI can be automated by traversing through DOM to perform operations. This requires setting of Automation tag attributes required for Client side HTML elements. These tag attributes help in unambiguously describe the location of any element in the hierarchy of Open UI and allow full integration capabilities with any automation instrumentation.

Automation tag attributes: When Siebel server side parameter "EnableAutomation" is set to "True" the below tag attributes will be available for a broad set of client side HTML elements.

RN - Siebel repository name in the form of <Applet Name><Control Name>

UN - Display name that logically identifies the HTML element

OT - Object type of Siebel HTML element

Below are examples of the differences between the DOM structure for New Button/Combo box **with and without** Automation tag attribute support enabled:

DOM structure for New Button/Combo box with Automation support:

```
<button id="s_1_1_7_0_Ctrl" class="appletButton" type="button" data-display="New"
title="Accounts:New" ot="Button" rn="NewRecord" un="New" style="">New</button><input
type="text" name="s_2_1_87_0" value="NY" aria-labelledby="State_Label" aria-label="State"
style="height: 24px; width: 120px;" maxlength="10" class="ui-autocomplete-input siebui-input-popup"
autocomplete="off" role="textbox" aria-autocomplete="list" aria-haspopup="true" ot="JComboBox"
rn="State" un="State" aria-disabled="true" readonly="readonly" aria-describedby="s_2_1_87_0_icon">
```

Note that the remaining HTML element ID's in the above example can be ignored by automation scripts even if changing under recompilation.

*DOM structure for New Button/Combobox **without** Automation support:*

```
<button id="s_1_1_7_0_Ctrl" class="appletButton" type="button" data-display="New"
Title="Accounts:New" style="">New</button><input type="text" name="s_2_1_87_0" value="NY" aria-
labelledby="State_Label" aria-label="State" style="height: 24px; width: 120px;" maxlength="10"
class="ui-autocomplete-input siebui-input-popup" autocomplete="off" role="textbox" aria-
autocomplete="list" aria-haspopup="true" aria-disabled="true" readonly="readonly" aria-
describedby="s_2_1_87_0_icon">
```

If the Automation is enabled in the environment to be verified; OT/RN/UN attributes must be available for Client side HTML elements.

The SweCmd=AutoOn command is required when launching the application.

Example: http://www.testtest.com/callcenter_enu/start.swe?SWECmd=AutoOn

Oracle Automation Test Suite (OATS) Integration

OATS Functional Testing and Siebel Test Automation for Open UI (STAO) (applies to 8.1.1.11+/8.2.2.4+ only)

Oracle Automated Test Suite supports functional testing starting with Siebel 8.1.1.11/8.2.2.4 release using Siebel Test Automation for Open UI (STAO). STAO provides a value added set of test automation libraries for functional testing, that greatly simplify accessing the DOM by abstracting use of metadata away from the test automation developer and thus greatly simplifying the task of test automation when using Oracle ATS over the public automation interface.

The STAO API is not compatible with the Siebel Test Automation (STA) interface for HI/SI or third party solutions for test automation and is reserved exclusively for use with Oracle Test Automation (OATS).

Test scripts for HI only run on QTP and are not compatible with (STAO). Siebel Test Automation for Open UI will be distributed as part of Oracle Test Automation Suite. Customers must record scripting using the STAO API and Oracle ATS tools when using STAO. Existing scripts for QTP will continue to work with HI and any future tools deemed compatible by vendors that choose to maintain that compatible Script-based solutions. Siebel Mobile Test Automation is not provided as part of the initial Siebel Test Automation for Open UI (STAO).

Siebel Test automation for Open UI ships as part of OATS when used with OATS licenses. Users already using Siebel Test Automation are covered under existing license for STAO. Oracle fully supports the documented STAO API. Customized automation requires support of the implementer.

Note: STAO requires new scripts when automating Siebel as it is written in a new scripting model. Prior automation is not supported with STAO.

Oracle Automation Test Suite (OATS) Integration with Siebel

Oracle ATS Siebel test support works with SI, HI and now also Open UI. It is the only test automation suite available that is tightly integrated with Siebel Open UI. Siebel works with OATS as outlined in [MOS Certify](#). OATS products are downloaded from here:

<http://www.oracle.com/technetwork/oem/downloads/index-084446.html>

Documentation part of that release includes Open UI function library. Oracle ATS documentation can also be downloaded at: <http://download.oracle.com/otn/nt/apptesting/12.3.0.1/oats-docs-12.3.0.1.376.zip>

OATS Load Testing

The TSK (Test Start Kit) contains test scripts that can be used for benchmarking as well and provides sample scripts. These are based on Siebel 8.1.1/8.2.2 and the HI object manager. Documentation can be found at: <http://www.oracle.com/technetwork/oem/app-test/oats930-test-starter-kit-siebel-1524433.pdf>

Oracle Business Intelligence Enterprise Edition (OBIEE) Integration

Please consider the platform compatibility of clients prior to integrating with OBIEE. Client support matrix for OBIEE is listed at [OBIEE 11g - Browser Certification Advisor \(Doc ID 1615805.2\)](#)

Oracle UPK Integration

UPK supports the Open UI with version 11.1.0.2 of UPK as an integrated solution. This means Siebel customers running Open UI will be able to record UPK content against Open UI and launch the content in a context sensitive manner, just as in the past using the High Interactivity application.

For additional details regarding supported browsers please consult the Target Application Technical Specifications: <http://www.oracle.com/us/products/applications/user-productivity-kit/overview/index.html>

Siebel Product Configurator (8.1.1.14/8.2.2.14 and above only)

The Siebel Product Configurator displays a native user interface from within Quotes, Orders, Agreements and Assets AND is fully supported in Open UI. Open UI will support all documented user interface features found in the HI client with some noted exceptions outlined below:

Side by Side Support Product Configurator Support for Open UI and HI

Siebel Product Configurator can be deployed in both HI UI and Open UI in parallel. The URL to access the HI UI Siebel Product Configurator Runtime is provided to users who only need the Siebel Product Configurator under the auspices of Siebel HI UI (e.g. Universal Agent). This Configurator Runtime will access the HI UI library artifacts when displaying the Configurator Runtime user interface. An Open UI Siebel Product Configurator Runtime will access the same controls from the Open UI library and display it. Please note the Open UI and HI UI controls must be the same name/filename in the Product Definition.

Migrating AIA integrations

AIA versions 2.X 3.X and 11.X are all compatible with Siebel `Open UI releases. Because of IRM there are no specific ACR's to import for these releases Open UI. Support for AIA versions is stated on [MOS Certify](#). Note that AIA integrations require Fusion Middleware SOA server and lifecycle for AIA integrations support falls under that of the supported SOA Server. The AIA integrations are listed as Application Server dependencies on My Oracle Certifications.

Enabling Signature Capture

Signature Captures are stored as images in Open UI to avoid integration issues for external components trying to display signatures. The previous encryption algorithm did not adhere to web standards and is not supported in Open UI

Existing CIC signatures as images will not function in Open UI and must be imported as image

For more information on Signature Capture see the Configuring Open UI Guide at

http://docs.oracle.com/cd/E14004_01/books/config_open_ui/customizing_applets_and_layouts21.html

Reference Information

Browser Performance

Example data for a variety of browsers is listed in the following table. It provides a means of comparing relative computing performance in a normalized manner using the [Octane 2.0](#) JS benchmark.

Table 2: Hardware/Client Benchmarks

Browser	Octane 2.0 score	Hardware
Internet Explorer 11 x64	11751	Intel I7 2.0GHz – Windows 8.1 x64
Firefox	16374	Intel I7 2.0GHz – Windows 7

Browser	Octane 2.0 score	Hardware
Chrome x86	16469	Intel I7 2.0GHz – Windows 8.1 x64

As can be seen by the performance times, some browsers are better suited to high performance and display optimized characteristics over others.

Other Resources

Additional Information can be found at the links below and through Oracle University:

Siebel Bookshelf

- Siebel Bookshelf Guide for customizing Siebel Open UI great for **web developers**. : http://docs.oracle.com/cd/E14004_01/books/config_open_ui/config_open_uiTOC.html
- Siebel Bookshelf End User Fundamental Guide for **end users**: http://docs.oracle.com/cd/E14004_01/books/Fundamentals/FundamentalsTOC.html
- Siebel Mobile Guide for **mobile app developers**: http://docs.oracle.com/cd/E14004_01/books/ConnMobApps/ConnMobAppsTOC.html

On My Oracle Support

- Siebel University Open UI Training: http://education.oracle.com/pls/web_prod-plq-dad/db_pages.getpage?page_id=615&get_params=key:open%20ui
- [Oracle Support Document 1499842.1 \(Siebel Open UI Best Practices - Deployment Guide\)](#)
- [My Oracle Support Certifications](#)
- [Support for Siebel HI Client with Internet Explorer 9, 10, and 11 \(IE 9 IE 10 IE 11\) \(Doc ID 1627168.1\)](#)
[IP 2014 SOD](#)
- [Support Readiness for Siebel Open UI \(ACR 801\)](#)
- [Questions and Answers around Open UI](#)
- [OOW Siebel Sessions - IP and Open UI](#)

Data Sheets

- [Open UI Data Sheet](#)

Standards supported by Siebel CRM / Open UI

The following list provides the standards implemented in Siebel Open UI. Many of these assist in meeting commercial off-the-shelf procurement practices.

Table 3 Supported Standards

Standard	Std. Spec.	Detailed Description	Notes
W3C CSS	3.0+	http://www.w3.org/Style/CSS/	rendering of clients
W3C HTML	5.0+	http://www.w3.org/TR/2011/WD-html5-20110525/	rendering of clients
W3C HTTP	1.1 RFC-2616+	http://www.w3.org/Protocols/rfc2616/rfc2616.html	Rendering and data communications, SSO

W3C XML	1.0+	http://www.w3.org/TR/REC-xml/	Rendering and data communications
W3C XMLHttpRequest+	Working Draft 6 December 2012	http://www.w3.org/TR/XMLHttpRequest/	JavaScript Communications for Client
W3C WAI-Aria	1.0+	http://www.w3.org/TR/wai-aria/	Accessibility Screen Reader integration
OASIS WSRP	WSRP2.0	http://docs.oasis-open.org/wsrp/v2/	Portlet Integration
ISO/IEC	16262:2011	http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?cnumber=55755h\	Client processing JavaScript
IETF LDAP	RFC-2256+	http://www.ietf.org/rfc/rfc2256.txt	LDAP support
IETF LDAP	Draft Behera LDAP Password Policy V9	http://tools.ietf.org/html/draft-behera-ldap-password-policy-09	LDAP Password Handling : X.500(96) User Schema for use with LDAPv3+
IETF TLS	RFC-2818+	https://tools.ietf.org/html/rfc2818	TLS Secure web traffic
IETF SSL	3.0 RFC-6101+	http://tools.ietf.org/html/rfc6101	SSL secured Communications in the application and to clients

Open UI Portal Standards Support

Open UI supports portal integration

- JSR286: Siebel Open UI is integrated as an iframe. This allows it to be hosted inside the JSP portion of a JSR286 Portlet. In this integration style it can be addressed via the URL using Open UI's component architecture for rendering components of the UI rendering hierarchy, data and focus context. Data is typically returned using web services.
- WSRP2.0: WSRP2.0 compliant portlets can be surfaced using native iframes generated by Siebel and as such are standards based portlets.

Client Authentication Options

Lightweight Directory Access Protocol and Active Directory Services Interfaces

Siebel supports LDAP Servers that meet the following standards:

- They comply with the LDAP 3.0 standard
- Handle password management in one of the following ways:
 1. The IETF Draft password handling standard is implemented on the LDAP
 2. Password handling is not required by handling password management externally (password expiry and other password-messaging features)

Oracle provides Active Directory support that requires the native connector shipped with the operating systems that are supported for Siebel applications on Microsoft Windows servers. Siebel Active Directory support is limited to either

- Specific active directory connectors based on the operating systems supported by the release. The AD connector used to connect with any Active Directory schemas must be deemed compatible by Microsoft.
- Use of the LDAP compliant superset of features supported by Active Directory.

Web SSO Integration Requirements

Web SSO solutions integration is supported with SSO solutions meeting the following prerequisites:

- The 3rd Party SSO system should be able to send the user identity in an HTTP Header Variable using HTTP1.1 standard W3C HTTP 1.1 RFC-2616+
- LDAP V.3 via IETF LDAP RFC-2256+ compliance
- IETF LDAP Draft Behera LDAP Password Policy V9 for password handling
- Siebel Web Single Sign On configuration is configured for the Siebel Web Engine.
- Siebel Security Adapter (LDAP/ ADSI) Profile is used for authentication.
- SSO solutions using static trust token in the HTTP header.

It should be noted that HTTP headers must turn off

- Cache-Control: no-cache
- Pragma: no-cache

If a SSO solution is used for performance with Open UI. The default setting of OAM is to enable these settings. Simply uncheck the two check boxes for these two directives in OAM for the virtual directory – then confirm they are no longer being added to our Siebel content. These settings are NOT supported for Siebel.

Client Mandates

Siebel when used with Open UI can meet the following general and industry specific Mandates.

Common Criteria Certification, Evaluation Assessment Level 2

More information can be found at http://www.commoncriteriaportal.org/files/epfiles/st_vid3026-vr.pdf for additional information

FIPS-140-2 standard

Siebel Open UI can be implemented to meet [FIPS-140-2](#) standard.

Open Web Application Security Project (OWASP)

Oracle participates in [OWASP](#) to meet security challenges in the enterprise.

Oracle Software Security Assurance (OSSA) guidelines and policies

Siebel participates in the Oracle [OSSA](#) effort.

Payment Card Industry Data Security Standard (PCI-DSS)

Siebel can comply with PCI-DSS. This requires configuration and compliance audit as per https://www.pcisecuritystandards.org/security_standards/ Siebel does not support PCI-DSS out of box.

Accessibility VPAT

- WCAG 2.0aa compliant (where practical and advisable) to the W3C WCAG2.0 aa Guidelines <http://www.w3.org/TR/WCAG/>.

VPAT information

Oracle provides a VPATS for Siebel Open UI 8.1.1/8.2.2 Releases. VPAT documenting details of WAI-ARIA and WCAG2.0AA compliance will be published at http://www.oracle.com/us/corporate/accessibility/vpats/vpats-siebel_162956.html

Oracle strongly recommends use by accessibility users to meet user acceptance of accessibility features as this is the most practical method of testing the application for accessibility and encourages customers to make sure that the user population understands AT's prior to testing Siebel. Oracle welcomes any feedback on accessibility related features and user experience.

Compliance for accessibility standards for all Oracle applications is detailed at <http://www.oracle.com/us/corporate/accessibility/vpats/index.html>.

EU privacy directives:

EU mandates strong rules around use of cookies:

http://www.ico.gov.uk/for_organisations/privacy_and_electronic_communications/the_guide/cookies.aspx. Siebel uses session cookies to track the security of the established connection. Siebel Open UI requires use of cookies to function properly.

Open UI Web Authentication Options:

SAML and Single Sign-On

SAML is supported via SSO integration. SSO typically requires third party products such as Oracle Access Manager or a third party equivalent. Information on such integrations can be found at:

- Integrating Siebel with SSO ([Doc ID 1509084.1](#))
- SIEBEL– OAM WEBGATE SSO INTEGRATION ([Doc ID 1509338.1](#))

IPV6

Open UI Web Servers and IPV6

Siebel Web Servers interoperate with IPV6 for use with Siebel client infrastructure using dual stack combinations. IPV6 is transparent to Siebel applications if it is installed on the web server, and is only typically needed on the web server. This support provides IPV6 connectivity for customer-facing web sites while also using IPV4 for communication across the DMZ to the intranet.

Note: It is recommended that you use the host name instead of the IP when configuring Siebel applications in environments that use IPV6 and IPv4 in a dual-IP-stack configuration.

Open UI Clients and IPV6

Open UI clients can be implemented on a native IPV6 stack using IPv6 if a dual stack web server is implemented.

Third-Party products in Open UI (8.1.1.14/8.2.2.14)

The following third-party components are used in the user interface.

- Acid Media mobiscroll 2.9.2
- Adam Shaw FullCalendar 1.6.4
- Adobe Scrollview Plugin 1.3.1
- Amiado Group AG Candy 1.0.9
- Apache Commons Codec 1.6
- Apache POI 3.8
- Apache XMLBeans 2.3 & 2.5
- Apache HttpComponents Client 4.3.3 (update to 4.2.5 TPNO 13329)
- Beanbag Inc. ReviewBoard 1.7.19
- Benjamin Arthur Lupton history.js 1.8.0
- Chris Leonello JQPlot 1.0.8
- Christian Robertson Robotoi Font n/a
- Codehaus Jackson 1.9.11
- Coolite, Inc. Date.js Alpha-1
- David Furfero Touch Punch 0.2.2
- Douglas Crockford and Ryan Grove jshint 3
- Facebook, Inc "f" Logo (<https://www.facebookbrand.com>) N/a
- Frederico Caldiera Knabben (FredCK.com) CKEditor 4.3
- Gideon Sireling jQuery Caret Plugin 1.3.1
- Google+ Logo (<https://developers.google.com/+web/badge/>) Most Recent
- Grant McLean jquery-udraggable 0.2.0
- Icojam Blue Bits Basic and Bonsu Icon Set N/a
- Ivan Bozhanov jsTree 3.0
- JTSage JQueryMobile - SimpleDialog 1.0.1
- JUnit JUnit 4.8, 4.8.2
- Jan Sorgalla jcarousel 0.3.0
- Jean-loup Gailly and Mark Adler zlib 1.2.5
- Joda.org Joda Time 1.6.2
- Johan Säll Larsson jQuery UI Google map 3.0-RC1
- John Resig JQuery 2.1.1.0
- John Resig JQuery Calculator 1.4.1

- Josh Bush Watermark Beta 1
- Juan Peri JQuery Session Timeout Handler .73
- Klaus Hartl jQuery Cookie 1.4.0
- LinkedIn LinkedIn Logo Most Recent
- Martin Wendt Dynatree 1.2.5
- Matt Bryson TouchSwipe 1.6.5
- Matteo Spinelli Add to Home Screen 2.05
- Matteo Spinelli iScroll 5.1.1
- Modernizr modernizr 2.7.1
- Open Lab Teamwork Gantt n/a
- Ryan Scherf jSwipe .2.2
- Sebastian Tschan jQuery File Upload Plugin 5.40.0
- Shlomy Gantz JSGantt 1.2
- SourceForge ini4j 0.5.1
- SpringSource Spring Framework 3.2.2
- SpringSource Spring Framework 4.0.3 (update to 3.2.2 TPNO 13295)
- The Apache Software Foundation Apache Maven 3.0.3
- The Dojo Foundation RequireJS 2.1.11
- Thomas J Bradley Signature Pad 2.5.0
- Tom Moor Tinycon 0.6
- Tony Tomov JQuery Grid 4.6.0
- Trent Richardson JQueryUI time picker add on 1.4.1
- Twitter, Inc Twitter Logo (<https://twitter.com/logo>) N/a
- Vladimir Agafonkin Leaflet 0.6.4
- Washington Botelho jQuery Raty 2.5.2
- YouTube YouTube Logo (<http://www.youtube.com/yt/brand/using-logo.html>) N/a
- Zachary Carter Jison 0413
- commadelimited jquery.swipeButton.js 1.2.1
- jQuery jQueryUI 1.11.0
- jsPlumb jsPlumb 1.6.2
- lesscss.org Less 1.6
- p.yusukekamiyamane Fugue Icons 3.5.7
- siliconforks JSCover 1.0.6.

Copyright © 2012, Oracle and/or its affiliates. All rights reserved.

This document is provided for information purposes only and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners. Ltd. 0911.

Hardware and Software, Engineered to Work Together