|  |
| --- |
| [Supervisor Change Tool] |

High Level Design

Revision 1.0

ARICENT™ CONFIDENTIAL

|  |
| --- |
| **Copyright** © **2012 Aricent Inc. All Rights Reserved.** No part of this document may be reproduced, stored in a retrieval system or transmitted, in any form, or by any means, electronic or otherwise, including photocopying, reprinting, or recording, for any purpose, without the express written permission of Aricent.  **Printed in \_\_\_\_\_\_\_\_**  **TRADEMARKS** ARICENT and THE ARICENT LOGO are trademarks of Aricent Inc. in the U.S. and other countries. The use of any of these trademarks without Aricent prior written consent is strictly prohibited. Other trademarks and trade names may be used in this document to refer to either the entities claiming the marks and names or their products. Aricent Inc. disclaims any proprietary interest in the trademarks and trade names other than its own.  **DISCLAIMER The** information in this book is provided “as is”, with no warranties whatsoever, including any warranty of merchantability, fitness for any particular purpose or any warranty otherwise arising out of any proposal, specification or sample. This document is provided for informational purposes only and should not be construed as a commitment on the part of Aricent. Information in this document is subject to change without notice.  **REQUESTS** For information or obtaining permission for use of material of this work, please submit a written request to: Corporate Marketing and Legal, 3460 Hill view Avenue, Palo Alto, CA. 94304.  **DOCUMENT No.: ARICENT** |
| **ARICENT™ Ltd.** Sigma Soft Tech Park Whitefield Bangalore – 560056 Karnataka (INDIA) Tel: +91-124-2346666/2455555 Fax: +91-124-2455100/2455101 E-mail: info@aricent.com Visit us at: <http://www.aricent.com> |

1. Table of Contents
2. [1. Introduction 4](#_Toc329788127)

[1.1 Purpose 4](#_Toc329788128)

[1.2 Scope 4](#_Toc329788129)

[1.3 Intended Audience 4](#_Toc329788130)

[1.4 Glossary 4](#_Toc329788131)

[1.5 Document Organization 5](#_Toc329788132)

1. [2. Background 6](#_Toc329788133)

[2.1 Overview 6](#_Toc329788134)

[2.2 Project Requirements 6](#_Toc329788135)

1. [3. Solution Architecture 7](#_Toc329788136)

[3.1 Solution Overview 7](#_Toc329788137)

[3.1.1 Mobile Business Objects 8](#_Toc329788138)

[3.2 Solution Architecture and Design 9](#_Toc329788139)

[3.2.1 Server Tier 10](#_Toc329788140)

[3.2.2 Enterprise Information System (EIS) 10](#_Toc329788141)

[3.2.2.1 Web Service Layer 11](#_Toc329788142)

[3.2.3 DMZ Layer 11](#_Toc329788143)

[3.2.4 Relay Server 11](#_Toc329788144)

1. [4. Web Service API 12](#_Toc329788145)

[4.1 Introduction 12](#_Toc329788146)

[4.2 Web Service Flow Diagram 13](#_Toc329788147)

1. [5. Feature Highlight 14](#_Toc329788148)

[5.1 MBS based Integration 14](#_Toc329788149)

[5.2 Workflow application / Hybrid Application Development 14](#_Toc329788150)

[5.3 Client Initiated Workflow 14](#_Toc329788151)

[5.4 Security 14](#_Toc329788152)

[5.5 Application Flow Diagram 15](#_Toc329788153)

[5.5.1 Get details of the Employee/Manager for Supervisor Change 15](#_Toc329788154)

[5.5.2 Updating Supervisor of an Employee 16](#_Toc329788155)

1. [6. High Level Interface Design 17](#_Toc329788156)
2. [7. References 20](#_Toc329788157)
3. [8. Revision History 21](#_Toc329788158)

Chapter

1

# Introduction

## Purpose

This document explains the high level design of the [Supervisor Change Tool], an enterprise mobile application for various mobile devices (Android/Blackberry) which will enable the Manager to change the supervisor of a particular employee. Document also describes various interfaces that are being used in application and the communication between various server layer and client layer.

## Scope

The scope of this document is to explain the high-level design details of the [Supervisor Change Tool]. This document does not include low-level design details.

## Intended Audience

The intended audience of this document comprises of the development teams who would use, enhance or modify this mobile application.

## Glossary

Table 1‑1: Glossary

| Acronym | Description |
| --- | --- |
| SUP | Sybase Unwired Platform |
| EIS | Enterprise Information System |
| MBO | Mobile Business Object |
| SCC | Sybase Control Center |
| MBS | Message based synchronization |
| WSDL | Web Services Definition Language |
| CDB | Consolidated Database |
| SOAP | Simple Object Access Protocol |
| HTTP | Hyper Text Transfer Protocol |

## Document Organization

The document is organized as follows:

Table 1‑2: Chapter Description

| Chapter | Name | Description |
| --- | --- | --- |
| 1 | Introduction | Provides a brief project introduction |
| 2 | Background | Gives an overview of [Supervisor Change Tool]application |
| 3 | Solution Architecture | Describes the flow of information and also interface integration details |
| 4 | Web Service API | Describes web service methods and flow diagram |
| 5 | Feature Highlight | Application Flow Diagrams describing the control Flows. |
| 6 | High Level Interface Design | Describes various interface/screen design of the application. |
| 7 | References | References used to create this document |
| 8 | Document History | Document History |

Chapter

2

# Background

## Overview

Aricent IT has many internal applications. Supervisor change is one of the functionality in RM tool as a web application, which is used by the Manager to change the supervisor of a particular employee. For the same an enterprise mobile application is developed which will provide the same functionality via mobile and other smart devices and we have a web service as an interface, which will interact with the database (SQL) and provide functionalities required by the application.

This document describes the high-level design of the Supervisor Change Tool, and the control flows from accessing the application by the mobile user to updating the information in the Back-End EIS.

## Project Requirements

The following are the basic requirements of the Supervisor Change application:

* Change the Supervisor of an Employee of Aricent Group, who is currently being supervised by the User.
* Search for an employee, using a minimum of three characters and the search result will contain all the employees that match the search criteria and are currently being supervised by the user( who is applying for the supervisor change).
* Search for other Aricent Employees holding a grade of E5 or higher, using a minimum of three characters.
* In the case of the search result holding an abundance of records, the functionality of paging will be provided on the device(s) of the end user(s).
* The application should return valid error messages to the user, in case of failure scenarios.

For further details refer the doc : SRS\_Supervisor Change Tool X.X v .

Chapter

3

# Solution Architecture

## Solution Overview

The following figure below depicts the solution overview of the Supervisor Change Tool using Sybase Unwired Platform. With the Sybase Unwired Platform we can create and manage multiple mobile applications that securely connect a variety of back-end data sources to all major device types. Sybase Unwired Platform enables us to embrace mobility across our entire organization with a development platform that is consistent, but highly adaptable.



**Sybase Unwired Server**

WE

B

SE

R

VI

C

E

**MBO**

**Workflows**



**DB**

**SOAP Web Service**

**Interfaces**

**Connections**

**Sybase Control Center**



**Enterprise Info System**

**Multiple Device Type** Support

Figure 3‑1: Sybase Unwired Platform- High Level Overview

The architectural design divides the platform into three separate areas that provide the ideal operating environment for delivery of mobile applications: Device Management, Mobile Middleware and Data Services. Including platform-wide services, security, administration and development tooling surrounding and support these concerns.

As the runtime server, Unwired Server handles enterprise data source and application access, communication between the backend data source and the mobile device, security, transaction processing, and scheduling. The Unwired Platform architecture includes several major components associated with tiers: server tier, data tier, and client tier.

Unwired Server subcomponents include:

* **Core and administrative subsystem (MMS)** – responsible for core services which include device client application interaction management, security enforcement, notifications processing, and runtime component management. MMS administrative services provide the server management interface used by Sybase Control Center.
* **Data services (DS)** – responsible for back-end interaction, data integrity, and transactions.
* **Cache database (CDB)** – is a runtime cache database used by Unwired Server. By default, an embedded SQL Anywhere® database server is used as the CDB, as shown in the diagram. However, you can configure Unwired Server to use an existing SQL Anywhere instance as its CDB server.
* **Messaging subsystem** – responsible for transport of messages required for maintaining data integrity on device client applications. It uses a JMS service to upload and download data changes between devices and the cache database.

### Mobile Business Objects

The cornerstone of the solution architecture is the concept of the mobile business object (MBO). Mobile business objects help form the business logic for mobile applications and mobile workflows by defining the data you want to use from your backend system and to expose through your mobile application or workflow.

MBO development involves defining object data models with back-end EIS connections, attributes, operations, and relationships that allow filtered data sets to be synchronized to mobile devices. MBOs are built by developers familiar with the data and transactional requirements of the mobile application, and how that connects to the existing EIS data sources.

A mobile business object (MBO) is derived from a data source (such as a database server, Web service, or SAP® server). MBOs are deployed to Unwired Server, and accessed from mobile device application clients.

## Solution Architecture and Design

The figure below depicts the logical architecture of the [Supervisor Change Tool] with all the sub modules of SUP and their connectivity.

The application is going to be implemented using SUP and back-end as Web Service over the Aricent network.

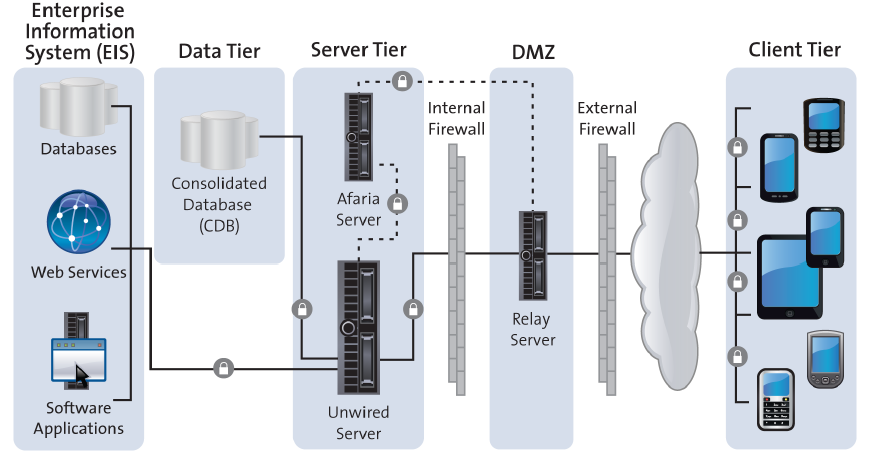


Figure 3‑2: Logical Architecture of Supervisor Change Tool with SUP

The Unwired Platform architecture includes several major components associated with tiers: server tier, data tier, and client tier.

* Server tier – Integrates the server components with back-end enterprise systems, data access and transaction services, device and application deployment, and system management functionality.
* Data tier – Stores data retrieved from the backend data sources and other runtime related metadata.
* Client tier – Consists of device applications built on top of the Unwired Platform client runtime. You can employ different secure application communication styles—replication and messaging—between the client and the server tiers.

The runtime landscape includes multiple components that are part of or interact with Unwired Platform components to provide the platform mobility solution. These components include: Unwired Server, the Data tier, Relay Server, and Afaria Server.

### Server Tier

Server Tier consists of the following modules:

* **Unwired Server**

Allow developers to package and deploy one or more mobile application from the development environment. From the unwired server user can deploy the application to mobile devices or optionally to Afaria Frontline Management for provisioning.

* **Afaria Server (Optional)**

Afaria's software platform delivers centralized control of all mobile devices and tablets including iPhone, iPad, Android, Windows Mobile and BlackBerry, as well as the apps that run on them. With keeping technology in mind, Afaria offers Enterprises the flexibility to deploy on premise or partner hosted.

* **MBO Services**

Mobile business objects help form the business logic for mobile applications and mobile workflows by defining the data you want to use from your backend system and to expose through your mobile application or workflow, and the methods and operations to perform.

* **Cache Database**

Unwired Server keeps a local copy of enterprise data in a consolidated database (CDB). Ensure that Unwired Server is using the correct CDB instance to act as its data cache. Unwired Server manages updates between the CDB and the EIS servers. When data is updated, the remote client database eventually retrieves this updated data from the server's local copy in the CDB.

* **Cluster Database**

The cluster database is, by default, a SQL Anywhere database used by the Unwired Server and associated command line utilities. It contains configuration information about the cluster for which the database is installed, as well as data used to coordinate cluster components.

* **Monitoring Database**

It monitors the activities in unwired clusters.

### Enterprise Information System (EIS)

Unwired Platform supports connections to several types of EIS and databases, which serve as the back end for mobile applications. It supports databases mainly (SQL Anywhere 11.x, Adaptive Server® Enterprise 12.5.x, 15.x,Microsoft SQL Server 2005, 2008,Oracle 10g, 11g,IBM DB2 UDB 9.1 ) and web services(SOAP/REST). We will be using a web service layer for the Supervisor Change Application.

#### Web Service Layer

Web service will be designed to expose web methods based on the requirement. These web methods will be consumed by clients. Basically this web service will be developed on .NET framework which will receive a SOAP request and in return it will send back the corresponding SOAP response. Web service will perform required validations & business logics on each method call before placing the DB call. Web service will perform necessary DB operation based on the business logic. Supervisor Change Tool Application (SUP) will use this web service as back-end EIS.

### DMZ Layer

A DMZ (sometimes referred to as a perimeter networking) is a physical or logical sub network that contains and exposes an organization's external services to a larger unreliable network, usually the Internet. The purpose of a DMZ is to add an additional layer of security to an organization's local area network (LAN); an external attacker only has access to equipment in the DMZ, rather than any other part of the network. The name is derived from the term "demilitarized zone", an area between nation states in which military action is not permitted. Component that resides on this layer for security purpose is:

### Relay Server

A relay server is similar to proxy server in addition; it also supports load balancing and across-the-firewall deployment without opening any internal firewall ports for enterprise mobilization.

A relay server accepts and forwards requests from remote clients to Unwired Platform components. A relay server is implemented as a pair of Web extensions that run in a Web server.

Relay server supports two Web servers: IIS on Windows, and Apache on Linux. A relay server can be configured to use either HTTP or HTTPS.

Chapter

4

# Web Service API

## Introduction

A Web service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-process able format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards.

Thus we are developing a Supervisor Change web service API which will expose web methods that will be consumed by the clients. Methods will be created based on the application requirement.

**Web Methods:**

* **searchEmployee** **:** The web method fetches the employee ID and name of the employee from the database based on the search filters (Name)
* **getSCEmployeeDetails** **:** The web method fetches the details of the employee from the database based on the search filters employee ID and manager ID(new supervisor)
* **updateSCEmployeeDetails :** The web method updates the details of the employee in the database and returns status flag/message for the operation performed

For further details please refer doc : API\_Supervisor Change Tool v X.X

## Web Service Flow Diagram

1. Initiates SOAP Request

2. Fire wall Validation

F

I

R

E

W

A

L

L

****

****

**IIS 6.0**

6. SOAP

Message returned

7. Successful response to client

3. Request received at IIS which identifies the web service & forwards the call to respective method

5. SOAP response

**Web Service Methods**

4. Database call for data transaction.

****

Figure 4‑1: Flow Diagram

Chapter

5

# Feature Highlight

## MBS based Integration

This approach uses Sybase Unwired Messaging Service (Messaging Service) that provides a dedicated channel that is always open to asynchronously push data changes, requests, and notifications between client and server. We use this option for use with Mobile Workflow applications. Ideal when several users interact with each other using both replicated and non-replicated (message) data to complete a workflow process. For example, Supervisor Change Tool application can be used by various users of Aricent in same time.

## Workflow application / Hybrid Application Development

Because this is simple request response application, where user will give some parameter to search employee and get list as paging. So there we message based communicate with Online request type. This approach is ideal for low data volume, simple display and limited offline support. Benefit of this approach is platform independent application. Standard that use in this application is HTML5/CSS/JS and XML.

## Client Initiated Workflow

Application can be started from the client, by giving request query to SUP server. Then server will fetch the data from the Aricent Database by calling web service method. This is done as a MBO operation in SUP server. Because there is no need where server initiates the transaction from server to client this is normal Pull mechanism.

## Security

As transport layer, all message based communication happen as encrypted, so there is high data security in this application. Here Device security is being take care by SUP server or if we will use Afaria then all devices need to be registered first with servers. Only valid devices can be used to communicate with servers. We will use Relay server as proxy and load balancer to provide External access security.

## Application Flow Diagram

### Get details of the Employee/Manager for Supervisor Change

1. Initiates Request by selecting employee / manager name

2. Validate and forward to server

**DMZ Layer**

**Relay Server (Proxy)**

**MBO**

**SUP Server**



6. SUP Workflow engine generates HTML5 response

7. Data of the employee is retrieved & displayed in the detail screen of the application

3. Executed MBO query (Call web service method)

5. WebService Response with details of the employee/ manager

**Web Service**

4. Database call for data transaction.

****

**Fig 5-1: Flow Diagram to get details of the employee/manager.**

### Updating Supervisor of an Employee

1. Manager updates the fields in the details screen of the application and clicks on update option

2. Validate and forward to server

**SUP Server**

**DMZ Layer**

**Relay Server (Proxy)**



**MBO**

6. Workflow engine returns the result from server.

7. success/ error message

3. Executed MBO query (Call web service method)

5. Web Service Response with success/error

**Web Service**

4. Database call for data transaction.

****

Fig 5-2: Flow Diagram to change the supervisor of an employee.

Chapter

6

# High Level Interface Design

**This is a Sample Application whose GUI’s are subject to change accordingly.**

1. **Initially a link/icon is provided to Supervisor Change Application**



Figure 6-1: Application icon/link

1. **Start screen is displayed with two input field (Employee name / Manager Name) :**
2. **Employee Name**: Search option is provided for assistance which on selecting will display all employee names that match the search criteria and are currently placed under the manager.
3. **Manager Name:** Search option is provided for assistance which on selecting will display all manager names that match the search criteria and have grades E5 & above (ENGG) and grade E4 & above (Non ENGG).

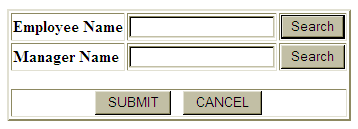


Figure 6-2: Start Screen

When user clicks on submit option control navigates to next screen contain detailed information of the employee whose supervisor has to be changed.

1. **Search result screen which displays Employee no. / Employee Name :**

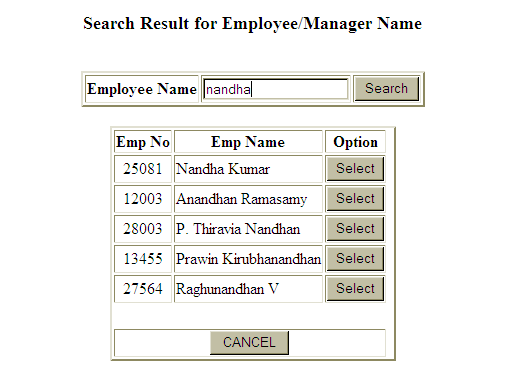
****

Figure 6-3: Search Result Screen

On selecting the required employee from search result screen controls goes back to the start screen with Employee Name and Manager Name populated with the selected name value.

1. **Employee Detail Screen which displays all the attributes of the Employee as well as the Manager details :**

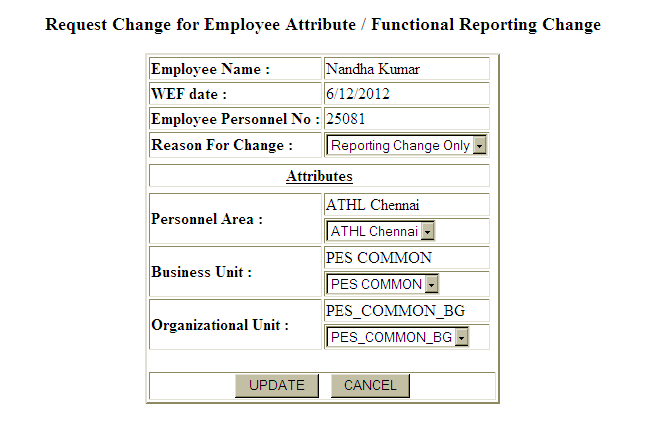
****

Figure 6-4: Detail view after selecting one particular record from the list

**Employee Name: ABC XYZ**

Once all the attributes values are set/modified user can select Update option for successfully changing the supervisor for the given employee.

Chapter

7

# References

* 1. Supervisor Change Tool: SRS v 1.0
  2. Sybase Web Help (<http://www.infocenter.sybase.com>)

Chapter

8

# Revision History

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Author: | | | 1st Reviewer | | | Kumaresan Ettiyannan | | | | |
| Hari haran Venugopal | | | 2nd Reviewer | | |  | | | | |
| 3rd Reviewer | | |  | | | | |
|  | | | | | | | | | | |
| Document Status: | | Initial | | | Internal | Draft | Approved | | Obsolete | Other |
| X | | |  |  |  | |  |  |
|  | | | | | | | | | | |
| Version | Date | | | Author | | | | Comments | | |
| 1.0 | 10/7/2012 | | | Hari haran Venugopal | | | | Initial Draft | | |
|  |  | | |  | | | |  | | |
|  |  | | |  | | | |  | | |
|  |  | | |  | | | |  | | |
|  |  | | |  | | | |  | | |
|  |  | | |  | | | |  | | |
|  |  | | |  | | | |  | | |
|  |  | | |  | | | |  | | |
|  |  | | |  | | | |  | | |
|  |  | | |  | | | |  | | |