



# Saracen IT

**Saracen Framework for EAI v.1.2**

*Introduction*

<b>Document Owner:</b>	<b>David Audoire</b>
<b>Sign Off Responsibility:</b>	<b>Waleed Al-Hussaini</b>
<b>Date:</b>	<b>26<sup>th</sup> May 2004</b>
<b>Version:</b>	<b>1</b>
<b>Status:</b>	<b>Released</b>



Filename: Saracen EAI Framework v.1.2 - Introduction.doc

## Version History

Version	Amendment(s)	Date	Amended By	Reviewed By	Signed Off By
1	First Release	26/5/2004	David Audoire	Waleed Al-Hussaini	

## Distribution List

The information contained in this document is subject to change and is updated periodically to reflect changes to the applicable software. Although every effort has been made to ensure the accuracy of this document, Saracen IT Limited (Saracen) assumes no responsibility for any errors that may appear herein. The software described in this document is furnished under a License Agreement and may be used or copied only in accordance with the terms of such License Agreement. Printing, copying, or reproducing this document in any fashion is prohibited except in accordance with the License Agreement. The contents of this document are designated as being confidential and proprietary; are considered to be trade secrets of Saracen; and may be used only in accordance with the License Agreement, as protected and enforceable by law. Saracen assumes no responsibility for the use or reliability of its software on platforms that are not supported by Saracen.



© 2004 by Saracen IT Limited. All Rights Reserved.



## Table of Contents

<b>1</b>	<b>INTRODUCTION .....</b>	<b>6</b>
1.1	Background .....	6
1.2	Intended Audience .....	6
1.3	Document Organization .....	6
1.4	Additional Information .....	6
<b>2</b>	<b>FEATURES.....</b>	<b>7</b>
2.1	Error reporting .....	7
2.2	Message Auditing .....	8
2.3	Performance monitoring .....	8
<b>3</b>	<b>ARCHITECTURE.....</b>	<b>10</b>
3.1	Overview.....	10
3.2	Seebeyond Plug-In.....	11
3.3	Framework Repository .....	11
3.4	Framework Manager .....	11
<b>4</b>	<b>FURTHER READING .....</b>	<b>13</b>



## **1 INTRODUCTION**

This chapter provides the introduction, general purpose, scope, and organization of the Saracen Framework for EAI v.1.2 Introduction document.

### **1.1 Background**

The Saracen Framework for EAI provides an “off the shelf” solution for delivering enhanced management and maintenance functionality to Seebeyond e\*Gate and SRE EAI solutions.

At Saracen, we recognized that our consultants were regularly developing customized solutions to address a series of common requirements across a broad range of customers.

- Error Reporting
- Message Auditing
- Performance Monitoring

The Framework was developed to address these challenges by providing a standardised and fully tested solution that can be used “out of the box” with any SeeBeyond eGate solution, thereby reducing ad hoc developments and mitigating the associated risks.

The Framework has a proven track record of shortening development cycles, increasing product quality and reducing costs.

### **1.2 Intended Audience**

The reader of this guide is presumed to be involved in evaluating or learning about the Saracen Framework for EAI. This person must have basic knowledge of SeeBeyond e\*Gate and SRE development principals.

### **1.3 Document Organization**

This document provides a detailed description of the functionality and architecture of the Saracen Framework for EAI.

### **1.4 Additional Information**

Additions, Errata and latest news about this product may be found at the Saracen IT Website at <http://www.saracen-it.com>.



## 2 FEATURES

The Saracen Framework for EAI provides an “off the shelf” solution for delivering comprehensive management and maintenance functionality to Seebeyond e\*Gate and SRE EAI solutions.

The Framework provides simple and effective error reporting, message auditing and performance monitoring functionality with minimal development effort, functionality that would otherwise have to be developed in-house.

The Framework has been designed and developed under the following principals:

- Non-intrusive - No new coding necessary
- Easily implemented on existing schemas
- Provides monitoring, error and performance statistics in a Web browser
- An ideal mechanism for system, and performance testing
- Highly scalable, reliable and extensible
- Low footprint

### 2.1 Error reporting

An “out of the box” Seebeyond solution provides limited functionality for reporting errors. The limitations of these approaches become especially apparent when complex business processes are modelled within your Seebeyond solution.

To address these limitations, Seebeyond developers either have to implement complex rules in the Seebeyond alerting sub-system or must design and develop error reporting systems from scratch. This Saracen Framework for EAI removes the effort and risk associated with these activities from your EAI project and allows effort to be focused on the task of integration.

The Framework addresses these shortcomings by offering a powerful mechanism to manage business logic errors in your integration projects. The Error Manager is quick and easy to implement whilst also being incredibly powerful and fully customizable.

Rudimentary error reporting is delivered without having to write a single line of new code. This has been achieved by delivering a pluggable software component that is only visible to the eGate Architect. This is then used to manage the collaboration rule from the outside, rather than the developer having to maintain the same error handling logic in all his/her collaborations.

The Framework provides the following error reporting functionality:

- Error message dictionary:  
A centralised registry of all the error messages that may be reported by the business logic



- **Error reporting:**  
Error messages are collected in a single location that may be accessed by support personnel.
- **Message editing and replay:**  
Once an error has been reported, a suitably authorised user may recover the error, by first editing the original message that caused the error, and re-inserting it at the relevant point in the business logic.
- **Error tracking and management:**  
All errors may be viewed, even after they have been resolved, and where a fix has been applied, the support analyst may then refer back to previous fixes of the same type, thus reducing the time taken to resolve issues. Management tools are available to identify **who, when, why, what** and **how** errors have been resolved.

## 2.2 Message Auditing

The Framework provides the capability to produce a complete audit trail of messages through mission critical e\*Gate systems.

It can be configured to take snapshots of messages at every stage of a business process. Messages may be retrieved and viewed through a simple and user-friendly GUI in real-time. It is also possible to “thread” together auditing snapshots to create a complete historical record of a message as it passes through the Seebeyond business processes.

Message auditing provides the following benefits:

- Full tracking of audited messages through an entire chain of transactions
- Real view of the individual messages at each step in the business process
- Customizable business data reporting
- Access the performance statistics relating to individual transactions

## 2.3 Performance monitoring

The Framework’s performance monitoring functionality provides detailed performance statistics of your solution architecture in real-time, providing drill down reports with three levels of granularity:

- **Interface level:**  
The overall performance of a series of components that make up an interface.
- **Component level:**  
The performance and throughput of each business logic component.
- **Code segment level:**  
The performance of individual segments of code within collaboration rules.

The performance information supplied by the Framework is often invaluable at both development-time where it can be used to make decisions in order to find optimal designs, and also at run-time in situations where system performance is critical.

At run-time the performance monitoring data shows the number and duration of transactions passing through the Seebeyond solution and has been used to :



- Identify short-term and long-term trends and patterns. Preempt integration issues
- Spot sub-optimal business processes
- Monitor performance against service level agreements and contracts

Drill-down reports allow developers to view performance statistics on each Seebeyond module on a component-by-component basis to:

- Identify performance bottlenecks
- Optimize Seebeyond architecture
- Guide hardware sizing decisions

Further to this, performance monitoring statistics can drill down even further to display data on individual functions within software components to identify poor performing code.

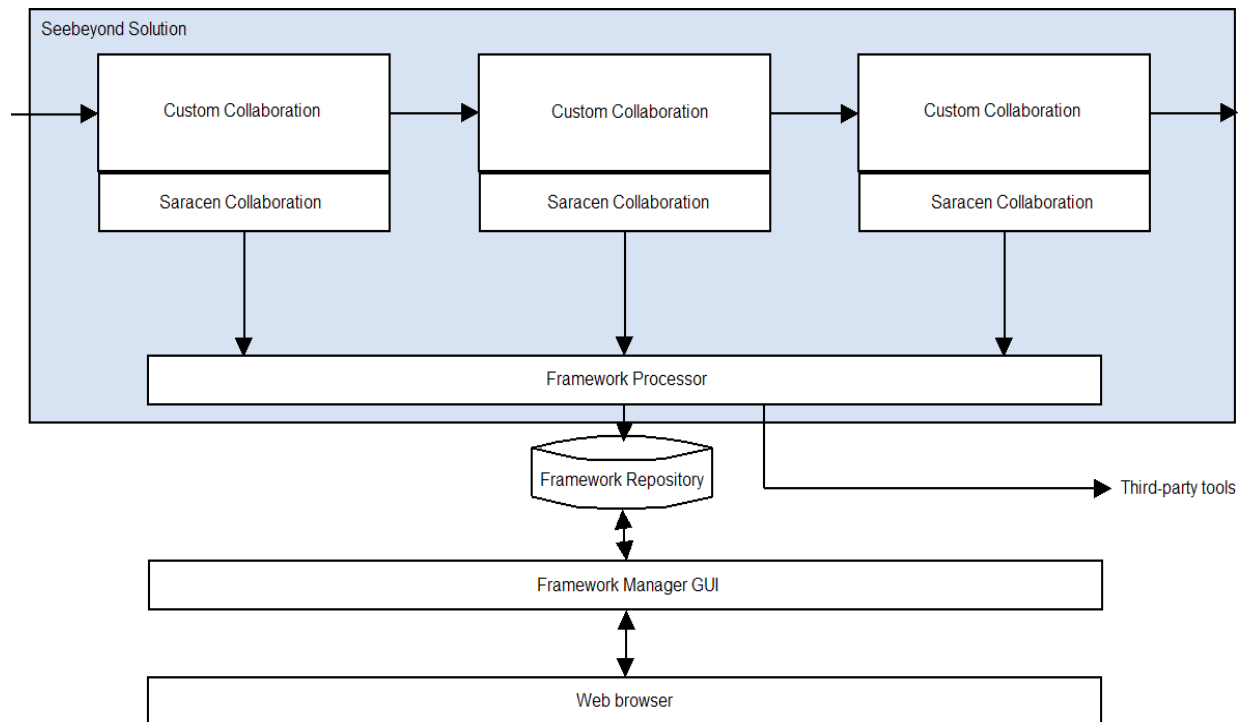
In this respect, the Framework becomes an invaluable decision-support tool and an invaluable support tool where system performance is critical.



### 3 ARCHITECTURE

#### 3.1 Overview

The Framework is designed around a tiered software model, outlined in Figure 1.



**FIGURE 1**

The Seebeyond plug-in component resides within the Seebeyond solution and provides all the functionality for collecting error messages, performance statistics and message auditing snapshots from the custom-designed collaborations. The plug-in writes all the management data to the next tier, the Framework repository, or may instead pass this management data onto external reporting tools.

The Framework Repository is a relational database which stores all of the data generated from the Framework.

The Framework Manager GUI is a Web-Based application which provides a view of the data in the Framework Repository. It provides views of audited messages, drill-down performance reports, management of error messages including error resolution functionality, and also tools for configuration of the Framework.



## 3.2 Seebeyond Plug-In

The Framework Seebeyond plug-in components are the elements of the Framework which are implemented within the Seebeyond solution. There are two elements to the plug-in's, the Framework processor and the Saracen collaboration.

The Saracen collaboration provides error reporting, message auditing and performance monitoring functionality to custom-developed business rules coded in Java collaboration rules. It is a thin layer of code which is configured to support each custom-developed collaboration rule. The collaboration is inserted into each rule through a simple configuration exercise which requires no coding. Once this exercise has been completed, the collaboration rule is then enabled to report data to the Framework.

Each collaboration rule which implements the Saracen collaboration reports its management data to the second element, the Framework processor.

The Framework processor is a delivered Seebeyond schema that collects the Framework data from each of the Framework enabled collaboration rules. The processor's role is to interpret and aggregate the Framework data and populate the Framework repository. The Framework processor may also be extended in order to forward the management data on to other third-party management tools. E.g. SNMP events.

## 3.3 Framework Repository

The Framework Repository is a relational database which is used to store all management data generated from the Framework enabled Seebeyond schema.

The database model is documented in the Saracen Framework for EAI v.1.2 Data Dictionary and is open for customers to generate custom-reports from the Framework management data.

## 3.4 Framework Manager

The Framework Manager is a Web-based application that provides a view on the Framework management data stored in the Framework repository. The Manager provides powerful yet easy to use browser-based access for configuring the Framework, viewing and resolving error reports, inspecting audit and performance data.

All business errors that are reported by eGate are displayed in the Manager. When an error is reported by the Seebeyond solution, the error is reported to the Framework and a copy of the message that caused the error is also stored in the repository. The Framework Manager allows users to view all the errors reported by the custom Seebeyond solutions and in addition, view the message that caused the error to be reported. Further to this, the Framework Manager allows the user to make amendments to these messages and re-insert them back into the Seebeyond solution thereby resolving the error.

The Manager also supplies a view on message audit snapshots. The user is supplied with a list of all audited message and may also drill down to view the contents of the message at the point where the snapshot took place. The auditing tool is also able to group together auditing



snapshot into threads. If a message is audited at various Seebeyond processes the Manager will associate the snapshots together to provide a chronological history of the message as it is transformed through the system.

Further to this, the Manager also features powerful system performance reporting tools. Graphical and tabular representations are available that provide an overview of your Seebeyond solution performance and also drill down reports show how each of the collaborations within your custom solution are performing.

Component performance reports are invaluable at design and development time to identify performance bottlenecks and support system scaling decisions while the higher level system performance reports facilitate management of the solution at runtime.



#### **4 FURTHER READING**

This document provides a basic high-level description of the functionality of the Saracen Framework for EAI.

For further detailed implementation guidelines, please see the User Guide.