



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

Deployment Guide

What is Genesys Web Engagement?

What is Genesys Web Engagement?

Contents

- **1 What is Genesys Web Engagement?**
 - 1.1 Overview
 - 1.2 Basic Architecture
 - 1.3 Events, Categories, and Rules
 - 1.4 Routing and Pacing
 - 1.5 Web Engagement Applications

If you're just getting started with Web Engagement, check out [A First Look at Genesys Web Engagement](#).

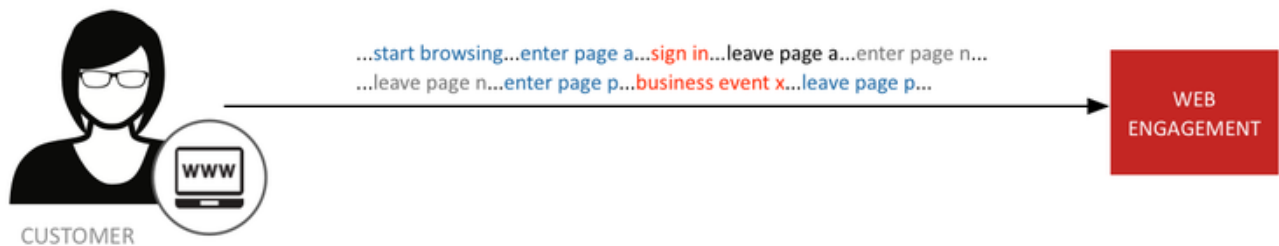
Overview

Genesys Web Engagement enhances customer experience by bringing together three related technologies:

- **Monitoring** records the customer's web browsing activity.
- **Notification** lets a customer know about an offer to engage via chat so they can accept it or decline it.
- **Engagement** provides the actual chat window to the customer's browser. (Starting with GWE 8.5.1, engagement technology is deprecated. Instead, you can integrate GWE 8.5.1 with [CX Widgets](#).)

Here's how they work together:

Monitoring

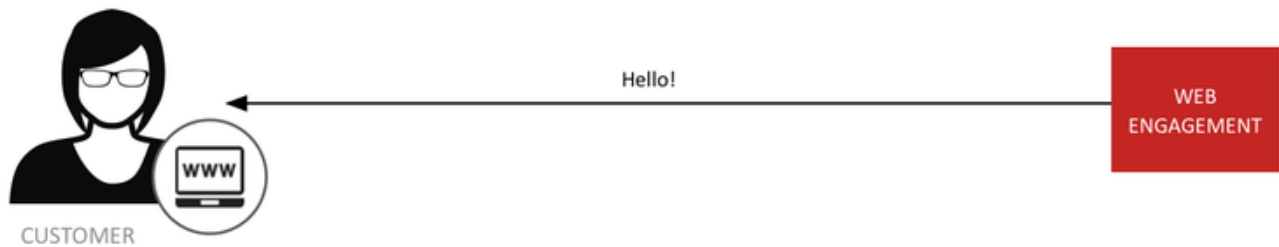


Monitoring allows you to gather data on customer browsing behavior so you can analyze it.

For example, if you've noticed that a lot of customers reach a certain point on your site and then abandon their transactions, you may want to offer a chat as soon as other people reach that point.

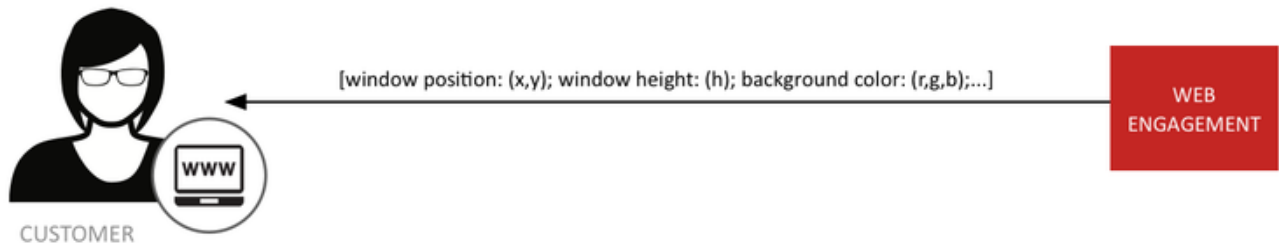
Another case might be where someone has been looking at a certain type of product, and you want your agents to know what it was, so they can respond to the customer's queries more intelligently.

Notification



While the concepts behind **notification** are simple, the power it gives you can lead to a lower abandon rate and higher levels of customer satisfaction—all because Web Engagement provides easy-to-use JavaScripts that allow you to reach out to those customers who need help.

Engagement



Engagement uses simple JavaScript technologies to display a chat invitation window to the appropriate customers.

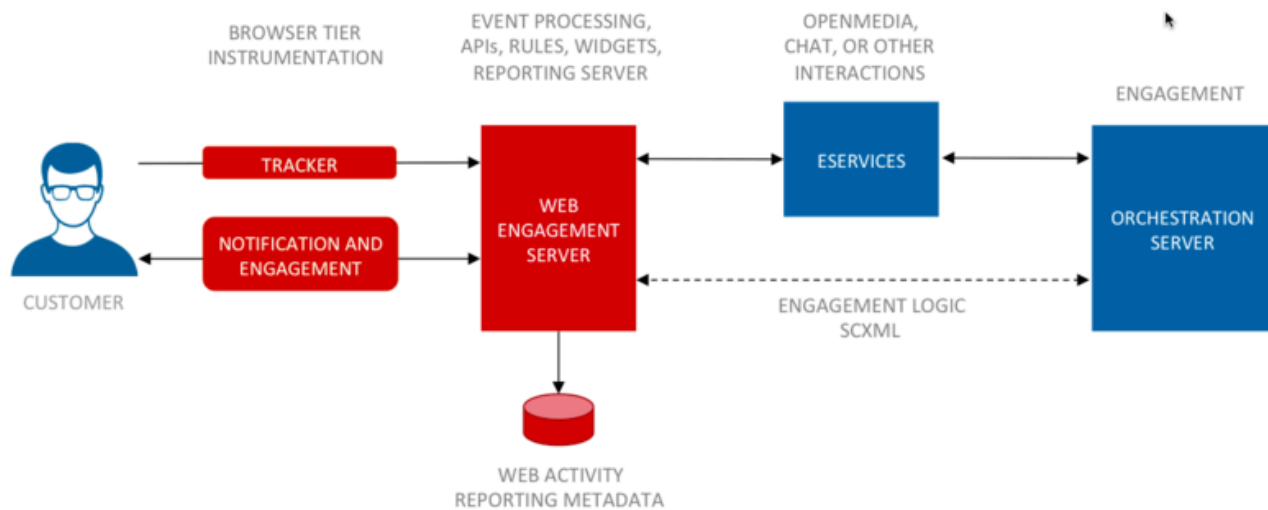
Basic Architecture

Browser Tier

Each of these three functions has an agent in the browser tier, as shown in the diagram below.

The browser tier connects to a load balancer that in turn communicates with a cluster of Web Engagement Servers. This cluster is connected to the rest of the Genesys solution, as well as to the Web Engagement database, which is frequently located in the Enterprise Tier.

What is Genesys Web Engagement?

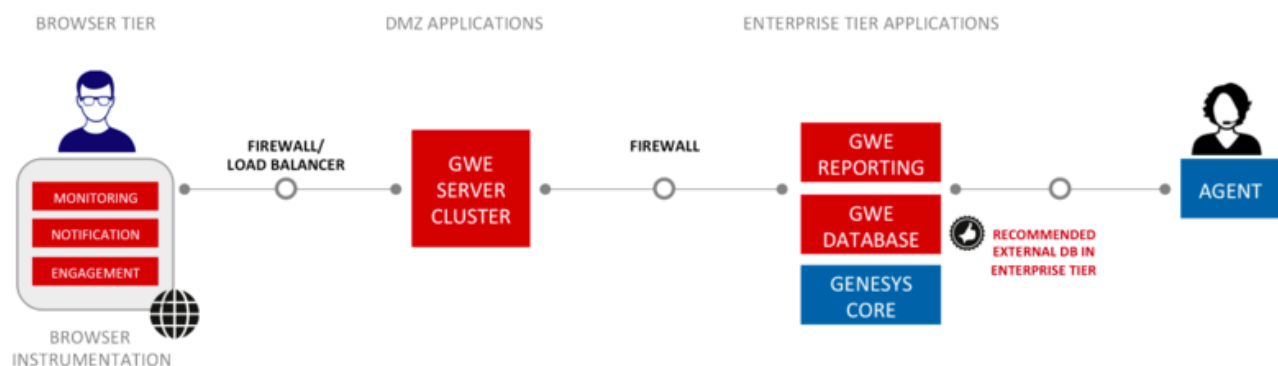


Enterprise Tier

Let's take a closer look at this architecture, with a somewhat sharper focus on the enterprise tier.

Web Engagement uses Cassandra 2.2 for database storage and Elasticsearch as an indexing engine. These two components work closely together to provide high-performance access to large amounts of data, both for operational needs and as input to your reporting. As mentioned, the database is often housed in the enterprise tier, but these components can live anywhere within reach of the Web Engagement Server cluster.

The rest of the enterprise tier consists of the various Genesys components that work together with Web Engagement, backing it up with the full power of the Genesys CX solution.



The Web Engagement Tier

The Web Engagement Cluster is built on top of an N + 1 architecture to support an easily extendable

set of Web Engagement Servers. These servers facilitate a complicated set of services that bring together the power of Web Engagement. Let's take a look at some of the most important ones.

Events, Categories, and Rules

System Events

The basic goal of Web Engagement is to help you decide when to invite a customer to engage with your agents. You can use Web Engagement's six **System Events** to help with this. These events are:

- **VisitStarted**—Includes the URL the customer first visited on your site and the time they started the visit
- **PageEntered**—Includes the URL and time when a customer entered a page
- **PageExited**—Includes the URL and time when a customer exited a page
- **UserInfo**—Reflects that a visitor is recognized, but not signed in
- **SignIn**—Reflects that the user has signed in
- **SignOut**—Reflects that the user has signed out

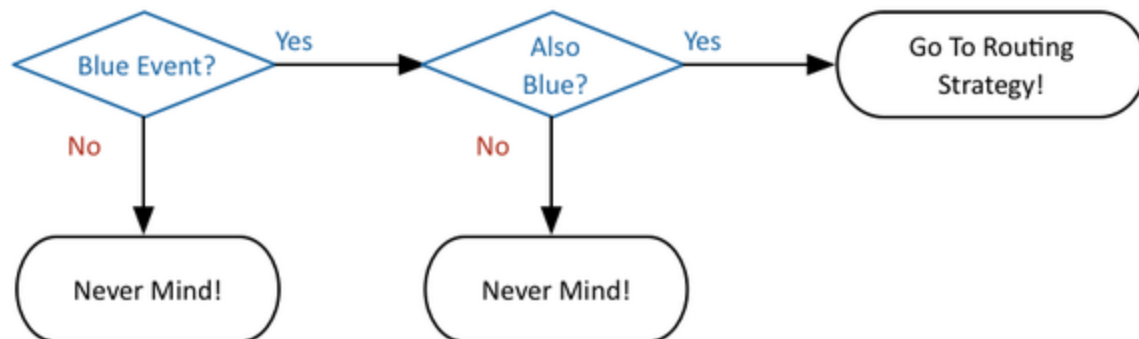
Sometimes you just need to know how long someone has been on your site. But in many situations, you are looking for more complicated patterns. These six events give you a lot to work with.

Categories

<u>Electronics</u>	<u>ExpensivePurchase</u>	<u>ContactUs</u>	<u>None</u>
PageEntered	AddToCart	PageEntered	
Timeout-30	RemoveFromCart	Timeout-30	
PageExited		LocationMapClick	
		PageExited	

And then there are the times when you need to pay attention to certain *types* of activity on your site. For example, you may be having a special on 40-inch HDTVs, so you want to keep track of customers who are visiting the pages for 40-inch models. You can set up a **Category** using the Web Engagement Management interface that allows you to tag the **PageEntered** events for these pages so you can track them.

Rules



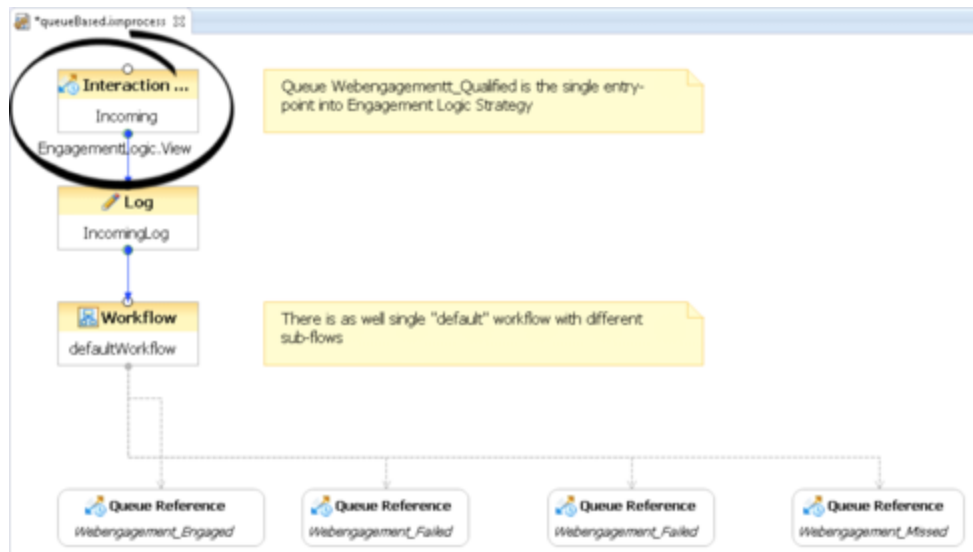
Once you know which system events and categories you're interested in, you need to write some logic that decides what to do with the information you have gathered about them. This logic is known as a **Rule**.

Business Events

In some cases, the system events and categories don't tell you everything you need to know. In those situations, you may want to define your own **Business Events**. As with system events, you can often get a lot more out of your business events by using them in conjunction with categories.

Routing and Pacing

Routing Strategies



Having identified your events and categories, and after writing your rules, you can finally decide who to engage. Interactions for these customers will be processed by the appropriate **Routing Strategy**.

Pacing

When you start sending out chat invitations, you need to make sure you don't send more than your agents can handle. Web Engagement includes intelligent **pacing** technology to help you optimize the number of invitations you send at any given time.

Web Engagement Applications

To make all of these features work together, you must develop a Web Engagement **application**, so that your website can start using the power of Genesys Web Engagement. The Web Engagement application is basically a set of site-specific resources designed to be deployed into the Genesys Web Engagement cluster.

Important: Before using Genesys Web Engagement, you must either **create** and **deploy** your own Web Engagement application or deploy the provided sample application. Otherwise, the product will not work.