

Digital Engagement Center Solution Blueprint

Reference Architecture

Authors: Pawel Bublewicz, Angelo Cicchitto

Version: 1.6

Status: DRAFT

Published: 9/1/2016



Table of Contents

- 1 INTRODUCTION 1**
 - 1.1 Document Overview..... 1
 - 1.2 Intended Audience..... 1
- 2 DEFINITIONS, ACRONYMS, AND DOCUMENT STANDARDS 2**
 - 2.1 Definitions 2
 - 2.2 Glossary 2
 - 2.3 Document Conventions..... 3
- 3 OVERALL ARCHITECTURE..... 4**
 - 3.1 Logical Architecture Model 4
 - 3.2 Functional View 5
 - 3.2.1 *Chat* 5
 - 3.2.2 *Email*..... 6
 - 3.2.3 *Short Message Service (SMS)*..... 8
 - 3.2.4 *Co-browse*..... 8
 - 3.2.5 *Web Engagement*..... 9
 - 3.2.6 *Mobile Engagement & Callback*..... 10
 - 3.2.7 *Social Engagement*..... 12
 - 3.3 Standard Use Cases 13
 - 3.4 Component View 14
 - 3.4.1 *Common components* 15
 - 3.4.2 *Workflow components* 16
 - 3.4.3 *Chat* 17
 - 3.4.4 *Email*..... 18
 - 3.4.5 *SMS*..... 18
 - 3.4.6 *Co-browse*..... 19
 - 3.4.7 *Web Engagement*..... 19
 - 3.4.8 *Mobile Engagement and Callback* 21
 - 3.4.9 *Social Engagement*..... 21
 - 3.4.10 *Genesys Digital Solution Components*..... 22
 - 3.4.11 *Required Genesys Common Component Services* 23
 - 3.4.12 *3rd Party Components – load balancers and reverse proxies* 24
 - 3.4.13 *Additional 3rd Party Components* 30
 - 3.5 Limits and Constraints 31

4	DEPLOYMENT VIEW	32
4.1	Centralized Deployment.....	32
4.2	High Availability Deployment.....	34
4.2.1	<i>Web Chat HA</i>	35
4.2.2	<i>Email HA</i>	40
4.2.3	<i>SMS HA</i>	40
4.2.4	<i>Co-browse HA</i>	41
4.2.5	<i>Web Engagement HA</i>	41
4.2.6	<i>Mobile Engagement HA / Callback HA</i>	42
4.2.7	<i>Interaction Server HA</i>	43
4.2.8	<i>Interaction Server proxy HA</i>	44
4.2.9	<i>Classification Server HA</i>	44
4.3	Dual Data Center Distribution.....	44
4.4	Database Configuration	45
5	INTERACTION VIEW	47
5.1	Interaction Flows.....	47
5.1.1	<i>Generic flow for interaction handled by Interaction Server</i>	47
5.1.2	<i>Reactive chat flow</i>	48
5.1.3	<i>Email flow</i>	48
5.1.4	<i>SMS page flow</i>	49
5.1.5	<i>SMS session flow</i>	50
5.1.6	<i>Co-browse flow</i>	50
5.1.7	<i>Web Engagement flow</i>	51
5.1.8	<i>Social Engagement flow</i>	51
5.2	Network Considerations.....	51
5.3	External Interfaces	52
5.4	Operational Management.....	54
5.4.1	<i>Network Management Systems</i>	54
5.4.2	<i>Serviceability</i>	55
5.4.3	<i>Monitoring Details</i>	56
6	IMPLEMENTATION VIEW.....	57
6.1	Solution Sizing Guidelines	57
6.1.1	<i>Solution Sizing</i>	57
6.1.2	<i>Database Sizing</i>	62



6.1.3	<i>Network Sizing and Readiness</i>	62
6.2	Configuration Guidelines	63
6.3	Security	63
6.3.1	<i>Secure Connections</i>	64
6.3.2	<i>Data Security Considerations</i>	64
6.3.3	<i>VM and OS hardening</i>	64
6.3.4	<i>Secure deployment for Internet facing components</i>	65
6.4	Localization and Internationalization	68
APPENDIX A	COMMON COMPONENTS SUMMARY	70
APPENDIX B	LOAD BALANCER CONFIGURATION	73
		73
		73
	Load Balancer: Co-browse	73
	Load Balancer: Web Engagement	75

Table of Figures

FIGURE 1 – DIGITAL BLUEPRINT LOGICAL OVERVIEW 4

FIGURE 2 – CHAT LOGICAL VIEW 6

FIGURE 3 – EMAIL LOGICAL VIEW 7

FIGURE 4 – SMS LOGICAL VIEW 8

FIGURE 5 – CO-BROWSE LOGICAL VIEW 9

FIGURE 6 – WEB ENGAGEMENT LOGICAL VIEW 10

FIGURE 7 – MOBILE ENGAGEMENT AND CALLBACK LOGICAL VIEW 11

FIGURE 8 – SOCIAL ENGAGEMENT LOGICAL VIEW 12

FIGURE 9 – DIGITAL COMPONENT OVERVIEW 14

FIGURE 10 – CENTRAL DEPLOYMENT MODEL - DIGITAL 32

FIGURE 11 – DIGITAL EXTERNAL INTERFACES 52

FIGURE 12 CO-BROWSE IN DMZ 67

FIGURE 13 – CO-BROWSE WITH REVERSE PROXY 68

FIGURE 14 COMMON COMPONENTS BLUEPRINT - HIGH LEVEL 71

Table of Tables

TABLE 1 - GENESYS COMPONENT LIST 23

TABLE 2 – REQUIRED COMMON COMPONENTS 24

TABLE 3 – LOAD BALANCING PARAMETERS 24

TABLE 4 – 3RD PARTY COMPONENTS 31

TABLE 5 - DATA CENTER – DIGITAL COMPONENTS 34

TABLE 6 – HIGH AVAILABILITY MATRIX 35

TABLE 7 - EXTERNAL INTERFACES 54

TABLE 8 – SIZING INPUTS 59

TABLE 9 – SERVER AND COMPONENT DISTRIBUTION 62

TABLE 10 – DATABASE SIZING 62

TABLE 11- NETWORK SIZING GUIDANCE 63

Revision History

Rev	Date Published	Author	Reason for Revision
0.1		Pawel Bublewicz	Initial release based on previous blueprint

0.2		Angelo	New content. Chapters: 3.1 Logical Architecture diagram 3.2.1 Chat diagram 3.2.2 Email diagram 3.2.3 SMS diagram 3.2.4 Co-browse diagram 3.2.5 Web Engagement diagram 3.2.6 Knowledge Center diagram 3.2.7 Social Engagement diagram 3.2.8 Callback diagram 3.4.13 Load-Balancers (Co-browse, Web Engagement) 4.2.1 High Availability (Chat, Email, SMS, Co-browse, Web Engagement, Mobile) 4.2.2 Survivability Matrix
0.3	07/26/16	Don Huovinen	Reviewed and updated based upon PS and Engineering feedback General review for readability Added appendix on Common Components
0.4	9/1/16	Don Huovinen	Updated disclosure language.

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS DOCUMENT ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS DOCUMENT ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THIS DOCUMENTATION IS BEING PROVIDED GRATUITOUSLY AND, THEREFORE GENESYS SHALL NOT BE LIABLE UNDER ANY THEORY FOR ANY DAMAGES SUFFERED BY LICENSEE OR ANY USER OF THE GENESYS DOCUMENTATION. UNDER NO CIRCUMSTANCES, INCLUDING NEGLIGENCE, SHALL GENESYS BE LIABLE FOR ANY INCIDENTAL, SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES ARISING OUT OF OR RELATING TO THIS DOCUMENTATION. SOME JURISDICTIONS DO NOT ALLOW THE LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES SO THIS LIMITATION MAY NOT APPLY TO YOU.

1 Introduction

The purpose of this document is to provide a standard Blueprint Architecture for the Digital Engagement Center Solution. It provides a prescriptive list of components (both Genesys and 3rd party) that should be included in the solution. It also provides deployment guidance, including sizing considerations, and addressing several system concerns such as security, high availability, disaster recovery and serviceability.

The Digital Blueprint is provides the architecture foundation to support both standard digital uses cases and various custom use cases which may be required.

1.1 Document Overview

The document contains the following sections:

- Chapter 2: Definitions and Acronyms
- Chapter 3: Overall Architecture
- Chapter 4: Deployment View
- Chapter 5: Interaction View
- Chapter 6: Implementation View
- Appendix A: Common Components Summary

1.2 Intended Audience

The Blueprint Architectures provide Genesys Solution Consultants, Professional Services and partners with information on the general architecture design and considerations for the solution. The information provided in this document should meet the needs of pre-sales and provide appropriate general guidance and practices for professional services. Configuration level information for professional services are not included in this document.

Describing system and solution architectures can be difficult as there are multiple audiences each with different expectations. This document is intended for multiple audiences with various chapters being more interesting to some readers than others. Readers should already have knowledge and training on Genesys products. This document provides high-level information for completeness.

The Overall Architecture and Deployment View are likely meaningful to most audiences. However, the Interaction View and the Implementation View may be of more interest to those configuring the network and components.

2 Definitions, Acronyms, and Document Standards

2.1 Definitions

This document uses various abbreviations and acronyms commonly used in Genesys product documentation and the telecommunications and contact center industries. The following table defines terms referenced subsequently in this document.

2.2 Glossary

CME	Configuration Management Environment, another name for the Configuration Layer
CS	Configuration Server
CSP	Configuration Server Proxy
CTI	Computer-telephony integration, the adding of computer intelligence to monitoring and control of telephone calls
DB	Database
DBMS	Database Management System
DHCP	Dynamic Host Configuration Protocol
DN	Directory number
DNS	Domain Name System
FTP	File Transfer Protocol
GA	Genesys Administrator
GAX	Genesys Administrator Extension
GCB	Genesys Co-browse
GIM	Genesys InfoMart
GI2	Genesys Interactive Insights
GMS	Genesys Mobile Server
GUI	Graphical User Interface
GVP	Genesys Voice Platform
GWE	Genesys Web Engagement
GWS	Genesys Web Services
HA	High Availability
HTTP	Hypertext Transfer Protocol
ICON	Interaction Concentrator
IP	Internet Protocol
IVR	Interactive Voice Response

IWS	Interaction Workspace
IXN	Interaction Server
LAN	Local Area Network
LCA	Local Control Agent
LM	License Manager
OEM	Original Equipment Manufacturer
ORS	Orchestration Server
OS	Operating System
RDBMS	Relational Database Management System
SCS	Solution Control Server
SCXML	State Chart XML: State Machine Notation for Control Abstraction
SQL	Structured Query Language
TLib	TServer Library
URS	Universal Routing Server
VM	Virtual Machine
WAN	Wide Area Network
WS(s)	Web Socket protocol. (s) stands for secure version.

2.3 Document Conventions

The document adopts following documentation and naming conventions:

- Code and configuration property names & values will appear in console font.
- References to other documents are encapsulated inside brackets ([]).

3 Overall Architecture

The Digital Engagement Center Solution covers all non-voice media used by customers to interact with the contact center. The Digital Blueprint is modular as many customer deployments start with a few channels and then add in more channels later as their business requirements evolve.

The Digital Blueprint leverages the Common Component Blueprint for foundational elements used in orchestration, reporting and configuration/management. A summary of the Common Components is included in Appendix A. The Digital Blueprint also relies heavily upon external components to deliver interactions and manage connectivity. We have included detail on 3rd party components utilized within the blueprint and their role.

Note: The Digital Blueprint is a living document, being updated through the Genesys Commercialization process. Early Adopter products, which presently include Video, Knowledge Center, etc. are not included in the solution blueprint and will be added as these products obtain Conditional Status.

3.1 Logical Architecture Model

The following is a logical model of the Digital Blueprint architecture. The diagram reflects the high level components covered in the Digital Blueprint.

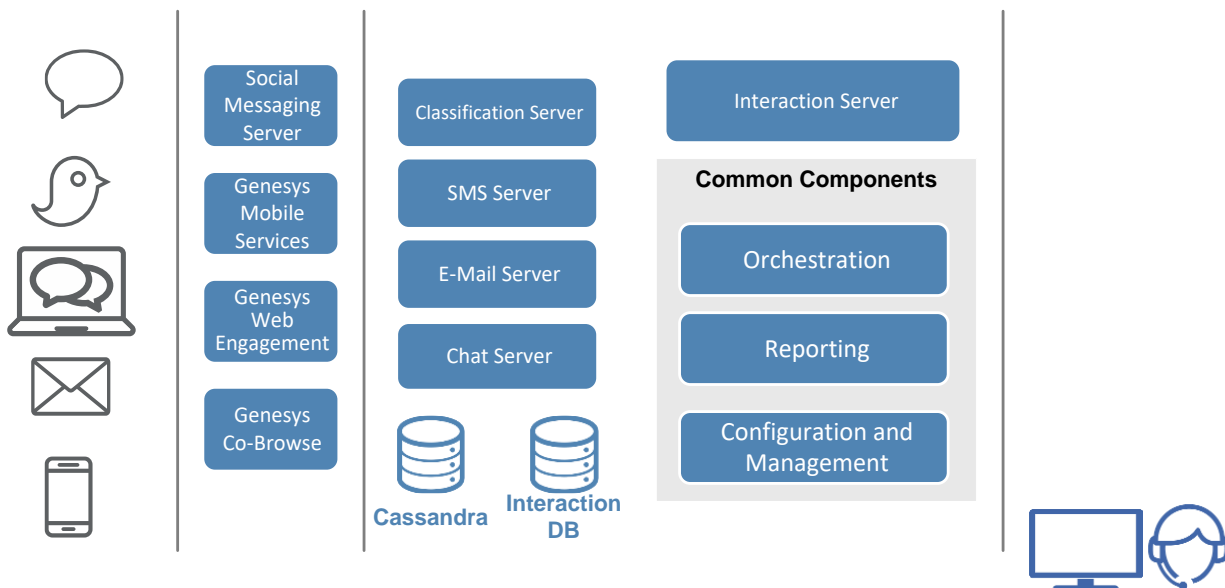


Figure 1 – Digital Blueprint Logical Overview

3.2 Functional View

The Digital Blueprint covers the non-voice customer interaction channels managed by Genesys. At a high level the Genesys architecture for digital consists of:

- Media interfaces – which provide connectivity for each media type and perform the translation between media specific protocols and the Genesys environment. These interfaces include general APIs which provide a flexible option to interact with Genesys.
- Workflow components – which manage the digital interactions received by the Genesys environment. With digital these components perform text processing to extract additional context from the interactions which can be used to more intelligently route and respond to interactions. Workflow components are common components that can be shared by multiple channels.

The Digital Blueprint also leverages the Common Components for standard routing and reporting.

The Digital Blueprint covers following channels:

- Chat
- Email
- SMS
- Co-browse
- Web Engagement (Proactive Engagement)
- Callback
- Social

Other Digital capabilities such as Knowledge Center and Video will be added to the Digital Blueprint through the normal Commercialization process as those products move into Conditional availability.

3.2.1 Chat

Genesys Chat lets your agents provide live assistance to customers via chat. Chat interactions are managed by the same business logic and routing rules that apply to other interaction type and share the same integrated monitoring and reporting tools. Chat interactions can originate via multiple devices such as standard web browser or through a mobile applications allowing customers multiple ways in which to request a chat session. Chats may be invoked reactively by a customer inquiry or proactively through Web Engagement. Live customer service chat is an essential element in a modern Digital solution.

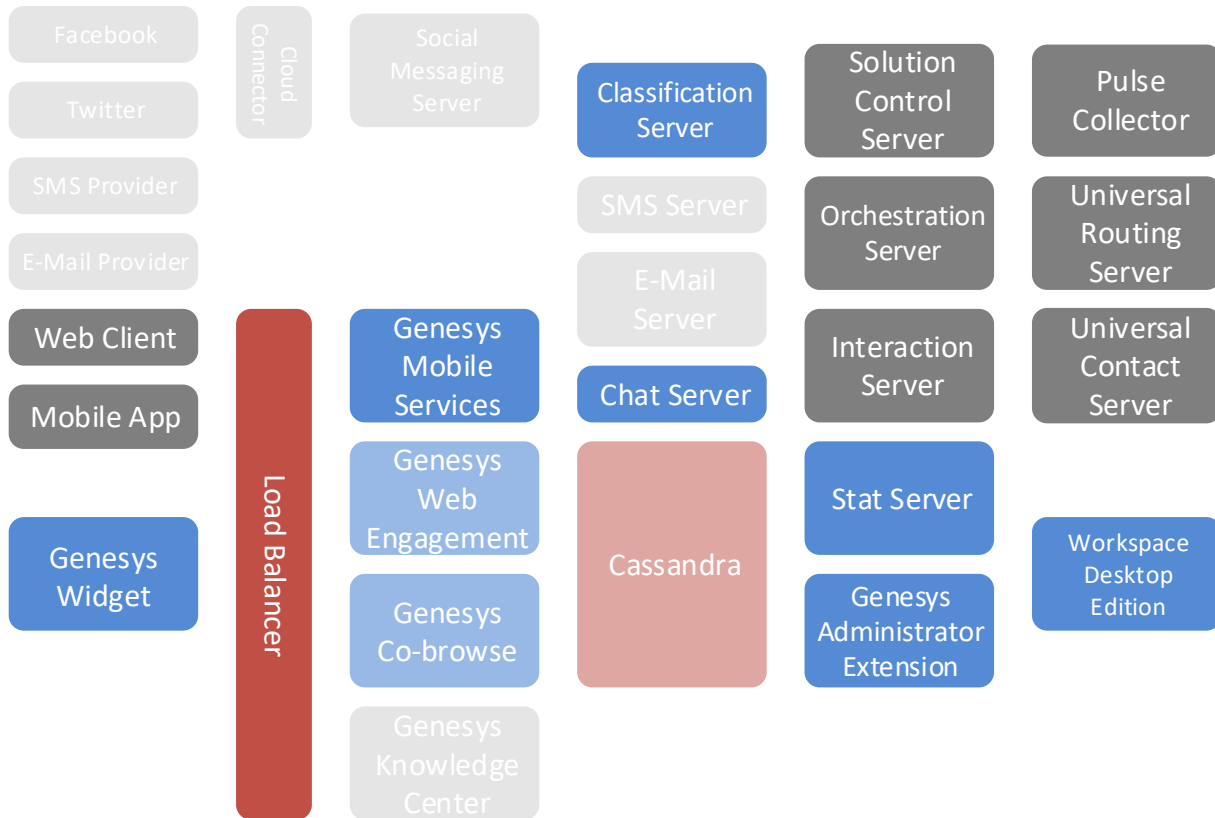


Figure 2 – Chat Logical View

Genesys Chat enables integration of live chat capability from any device that customers use to contact an enterprise organization. Genesys Chat features include:

- A ready to use customizable chat widget which makes it easy to quickly integrate live chat to an HTML web site
- Support for live chat conferencing allows customer service agents to conference other agents or supervisors into a live chat interaction to increase first contact resolution and improve customer satisfaction.
- Chat supervision allows supervisors to monitor live chat sessions for training and coaching.
- Native chat integration with Genesys Workspace provides access to Suggested Responses and delivers agents a single interface for monitoring voice, email, chat and other interactions.
- Chat routing and distribution provided by the Genesys Customer Experience (CX) platform enables consistent interaction and journey management to be applied, leveraging the same business logic and routing rules as other interactions.

3.2.2 Email

Genesys Email provides text analysis, routing and distribution of customer service emails to the desired agents. Genesys integrates to multiple enterprise email systems using POP, IMAP and EWS (Exchange Web Services) for email retrieval and SMTP or EWS for email submission.

For customers leveraging their web presence or mobile applications as a customer service portal Genesys E-mail allows emails to be directly posted to Genesys via a web form. This approach provides an advantage over traditional SMTP email services as it supports standard HTTPS web server security for submission and eliminates the need for Genesys to pool an enterprise email system to obtain the email.

Genesys can also analyze and classifying email content using either pre-configured rules or through trained statistical language models. The contextual information gained from this analysis can help improve the routing. Advanced contextual analysis can also be used to determine potential automated responses or suggested responses which can increase the agent’s efficiency.

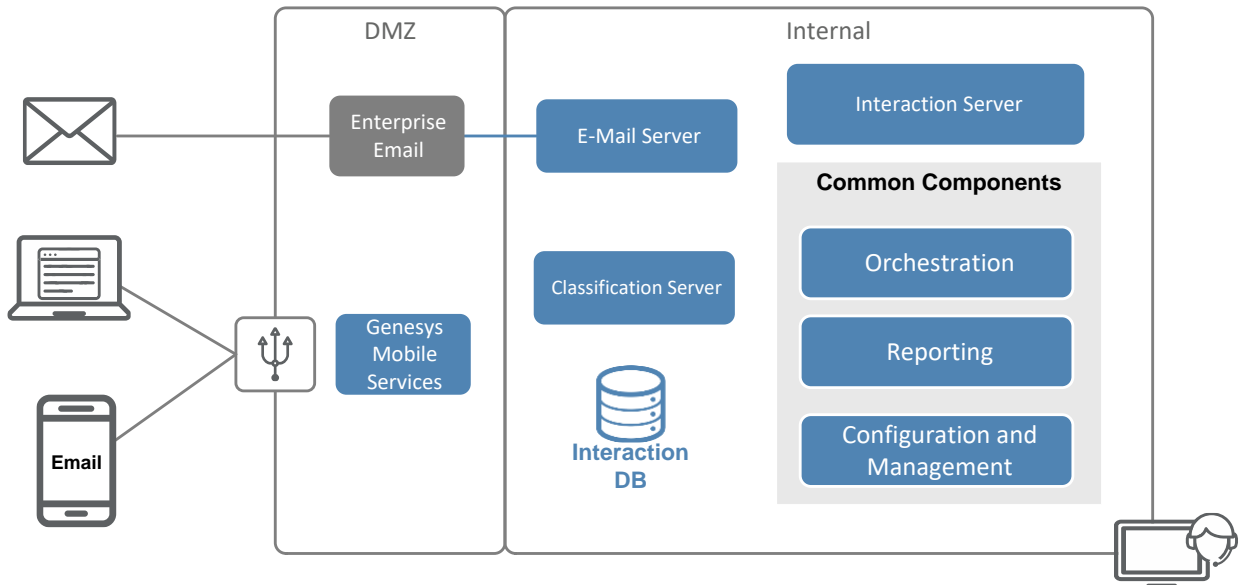


Figure 3 – Email Logical View

The real power of Genesys Email comes in leveraging Genesys Digital components of the CX Platform. The digital architecture allows customers to create a complete business process flow for the handling of email messages. The email flow can include pre-processing steps before routing the interaction to an agent, such as screening messages for account numbers, content analysis of the text to improve routing or provide suggested responses and post processing steps such as content analysis of the agent’s response to scan for misspelled or profane words.

3.2.3 Short Message Service (SMS)

Genesys SMS enables a customer service organization to add text messaging (short messaging service, or SMS) as a communications and support channel for attracting and retaining customers. Adding SMS capabilities as a customer service channel makes it possible to target customers how, when, and where they want to be contacted; provides an ideal platform for promoting new products and services; and creates an effective notification channel that can also be actionable. SMS provides a communications channel that can achieve a rapid return-on-investment, and quickly become an effective method for interacting with customers.

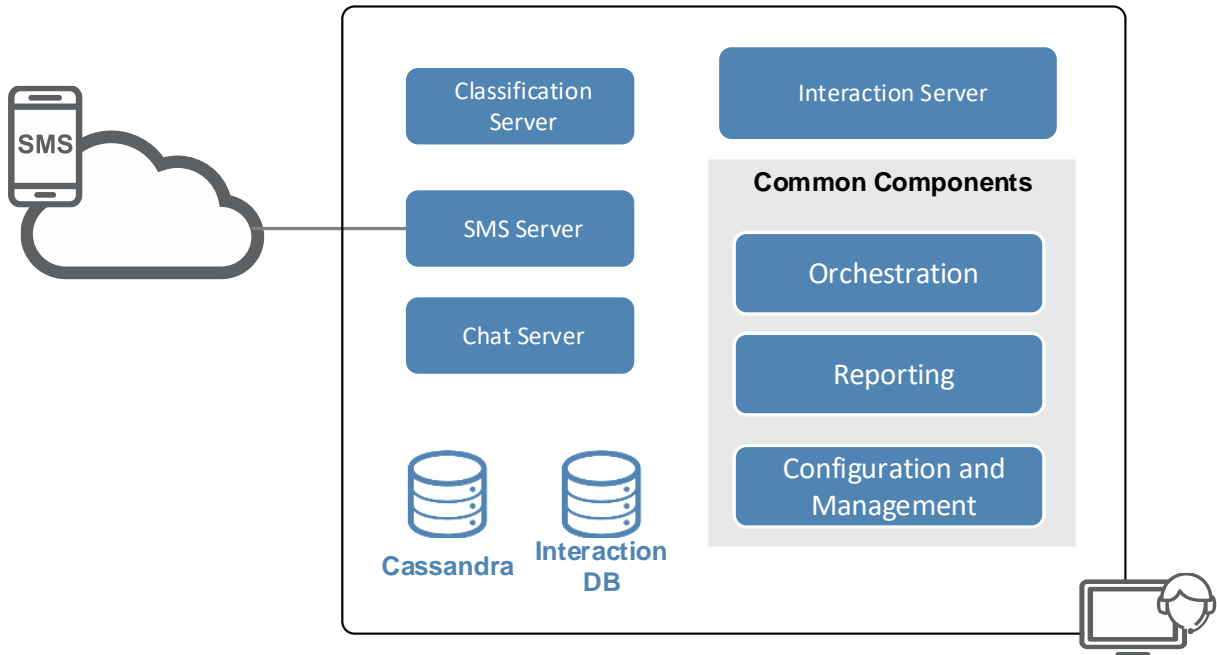


Figure 4 – SMS Logical View

Genesys SMS enables customer service representatives to incorporate this channel easily into their established workflows using the same tools. With Genesys agents can easily manage concurrent SMS sessions with multiple customers and blend SMS interactions with other media.

Genesys Workspace desktop enables customer service representatives to easily track multiple SMS sessions managed in either paging mode (for a single one-way or two-way interaction) or chat mode (for an ongoing string of messages with a given customer). It also provides support for multimedia messaging service (MMS) messages, making it possible for agents to receive and review photo, audio, and video files when necessary.

3.2.4 Co-browse

Genesys Co-browse lets customers browse and navigate web pages with contact center agents or knowledge experts in conjunction with real time chat or phone support. This real-time visibility of the customer’s experience helps agents to provide more effective, personalized on-line assistance.

In a Genesys Co-browse session, both the agent and the customer share the same instance of the browser session, as opposed to a conventional screen sharing application, where one of the parties sees an image of the other party's browser instance.

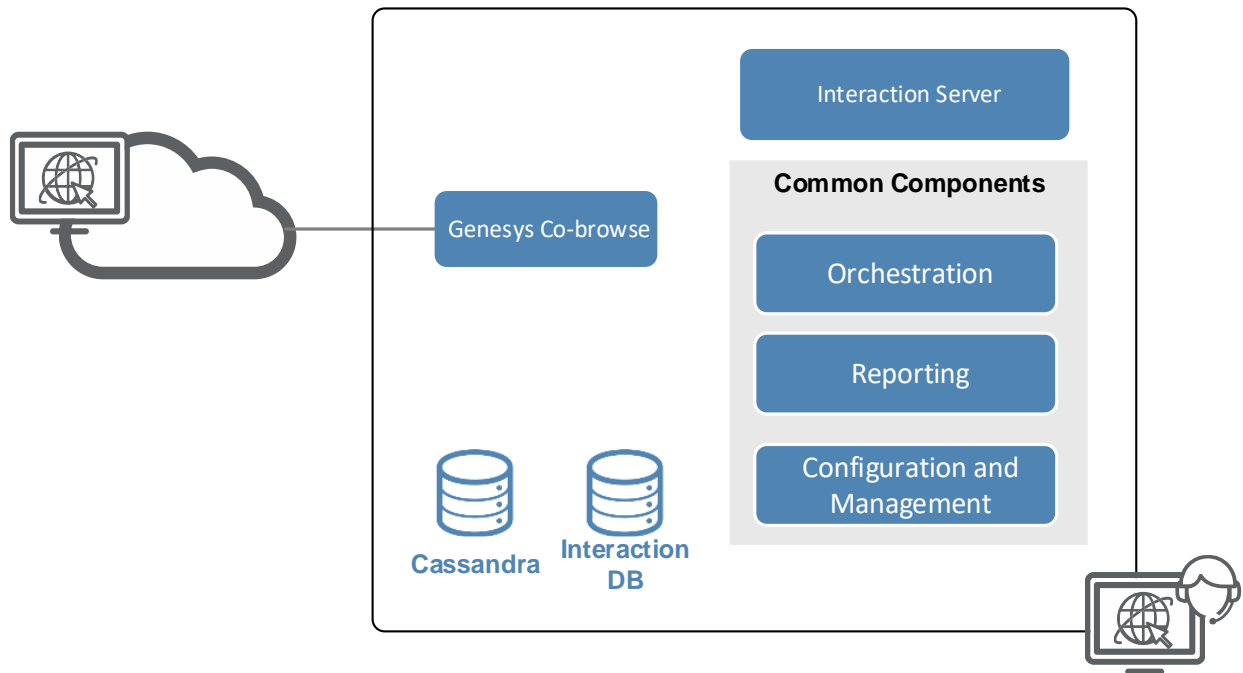


Figure 5 – Co-browse Logical View

Initiating a Co-browse session requires another active channel like chat or voice to which allows the customer and agent to share the Co-browse session-id and start sharing the browsing session. This active channel can continue to be used to allow the customer and agent to converse during the browsing session.

There are multiple ways to protect customer's sensitive information and privacy during a co-browse session, so that pre-defined sections of customer's web page cannot be viewed by the agent. Additionally the customer is always in full control of the co-browse session.

Note: Other channels such as SMS or email could be utilized to share the session-id however they do not allow real-time communication therefore they are not typically effective to complement a co-browse session.

3.2.5 Web Engagement

Genesys Web Engagement provides the ability to monitor, identify, and proactively engage web visitors in conversations that match business objectives. Business rules allow you to identify the customers you want to engage, providing a simple and comprehensive means for recognize key customers based on their behavior on the website and their business value.

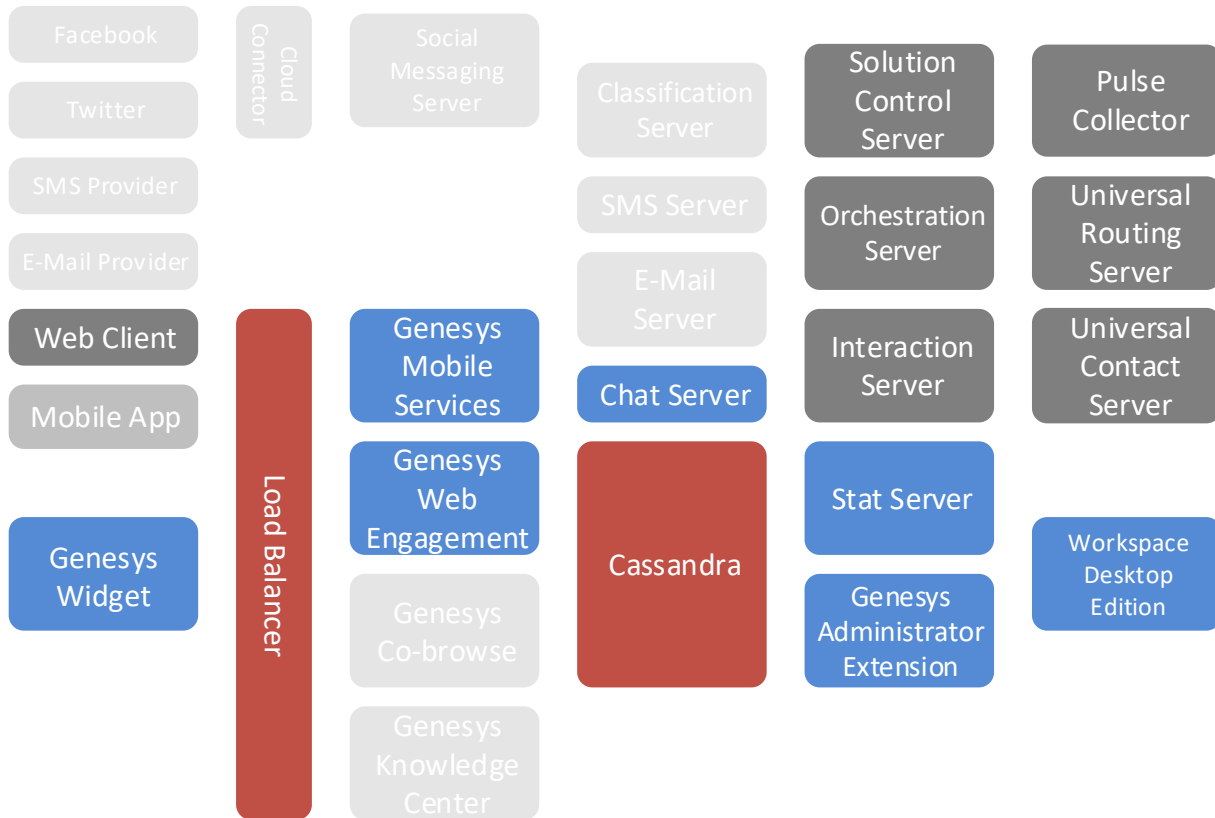


Figure 6 – Web Engagement Logical View

Genesys enables evaluation and engagement of key customers, matching them with the most appropriate agents, in order to meet business objectives such as new customer acquisition, product sales, or customer support.

Genesys Web Engagement integrates the browsing activity of your web visitors into the overall Genesys customer service process. It records the customer web-browsing history, gathers accurate information, evaluates the suitability of each web session based upon the on-going web activity and when it matches your configured business rules converts this activity into Genesys interactions.

In addition to its web monitoring capabilities, Genesys Web Engagement enables you to engage the customer by either chat or web callback. You can mix and match engagement invitations for available media channels any way you want, customize the look and feel of the invites, and even provide promotional advertisements.

3.2.6 Mobile Engagement & Callback

Genesys Mobile Engagement and enables customers to request customer service on their terms from mobile phones or other devices. Genesys Mobile Engagement provides a robust set of APIs which act as a service gateway to enable a seamless integration between channels such as native mobile applications and the Genesys Customer Experience Platform. These APIs which include immediate callback, scheduled callback, mobile push notifications can also be used for other channels such as enabling callback via a web-site or through an IVR.

Genesys Callback which is enabled through Genesys Mobile Services allows mobile and other applications such as IVR and Web to make it easy for customers to request a callback rather than waiting in queue or schedule live voice support at a time convenient to them. Callback places an outbound call from the contact center to the customer which provides the benefit that the call can be free of charge to the customer and because the customer is not waiting on hold reduce inbound telephony or infrastructure costs required by the business to queue the call.

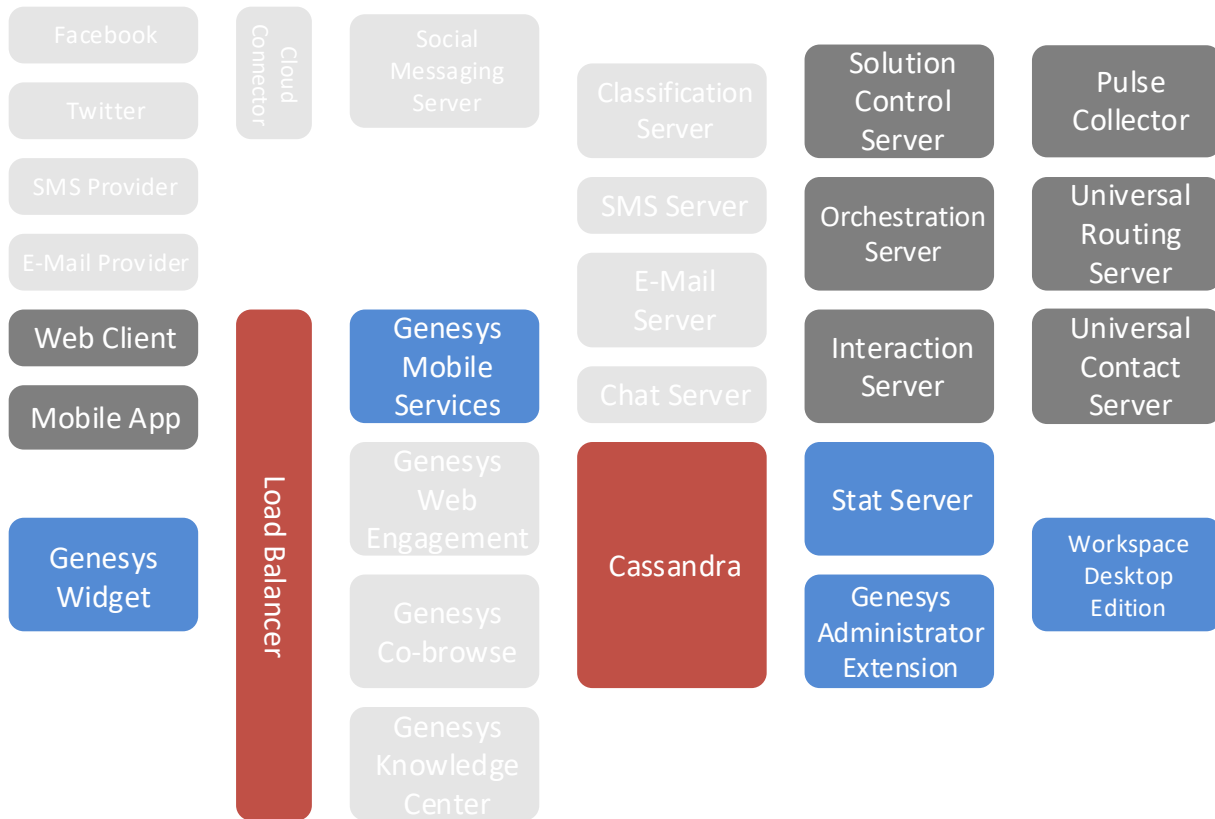


Figure 7 – Mobile Engagement and Callback Logical View

Genesys Mobile Services provides a set of REST APIs which can be invoked via any channel (Mobile, IVR, or Web) and provides the following features:

- A multi-channel, integrated solution that provides customer service access to context, such as customer profile, history, and location
- Optimized REST and Ajax Push interfaces for mobile, web, and IVR
- Estimated Wait Time calculations consistent with rest of Genesys platform including routing and reporting
- Genesys Callback allows mobile and other applications such as IVR and Web to make it easy for customers to request and schedule live voice support at a time convenient to them, in the form of a telephone dialed by the contact center, and therefore free of charge to the customer.
- Flexible connection for adding callback anywhere in the interaction: click-to-connect voice, push notification, chat, delayed or immediate
- Tight integration with Genesys routing that does not disrupt queuing or other service metrics

For Genesys Callback typical usage scenarios include:

- Scheduled, immediate or delayed callback requests.
- Preview callback requests.
- Proactive notification.
- Schedule callback with enhanced multimedia confirmation.
- Schedule an immediate return call or a callback at a convenient time, based on operating business hours.
- Check and display agent availability by providing estimated wait times.
- Support for mobile push notification to provide an alert when agent is available.

3.2.7 Social Engagement

Genesys Social Engagement enables enterprises to monitor social media sites, evaluate social media interactions based upon attributes such as sentiment, actionability, and social influence and then convert these social interactions into Genesys managed interactions. Genesys Social Engagement provides direct integrations with Facebook and Twitter. Genesys also provides an open API enabling customers to integrate the power of Genesys Social with other social media sites.

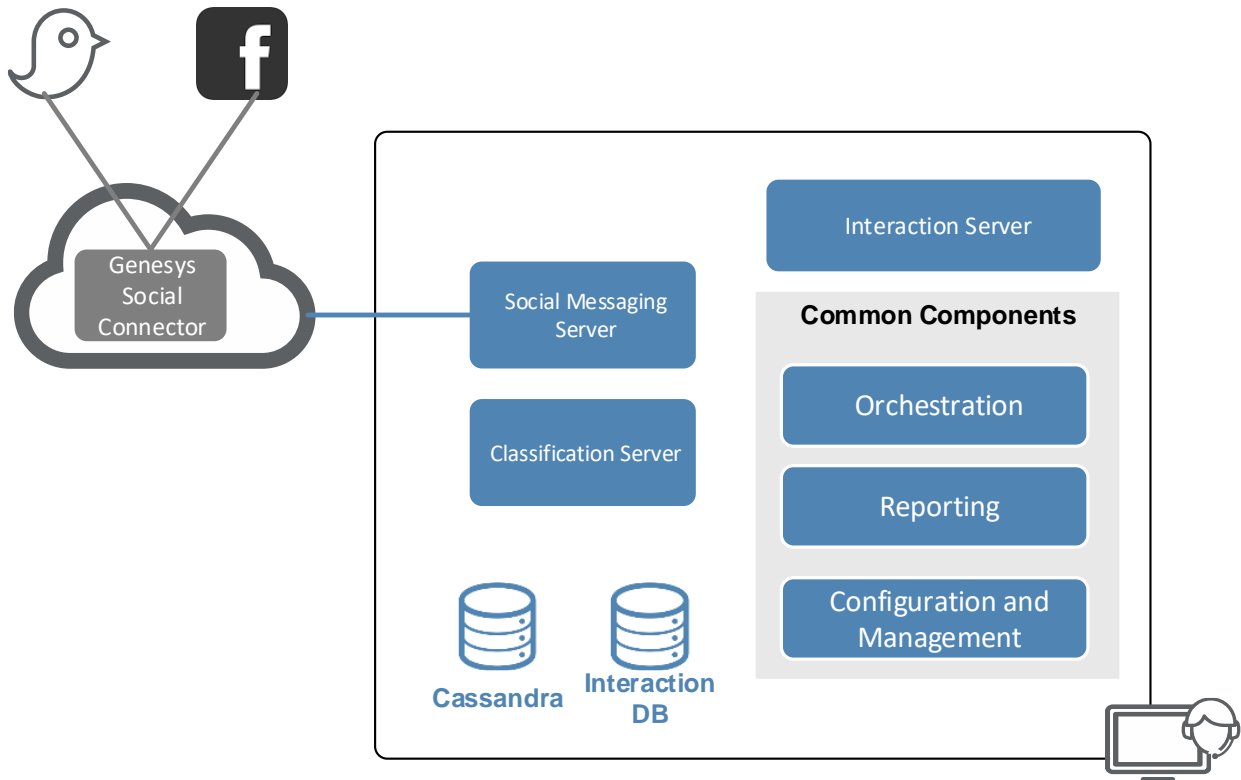


Figure 8 – Social Engagement Logical View

Genesys Social Engagement can evaluate 3 main attributes of any social interaction within the established business strategy. Genesys Social will look at:

- Actionability – Some social interactions are simple comments that don't warrant any specific action. The Genesys solution can review the social interaction and determine if an action is warranted. Messages that are considered non-actionable may be routed as low priority, or simply ignored.
- Sentiment – Social interactions can convey a range of emotions such as delight to disappointment and indifference. The system evaluates the emotional content or sentiment of the social interaction and can prioritize those with a negative tone above those with a neutral or positive tone.
- Influence - The influence of the author of a post is an important factor in determining the post's priority. Influence is typically defined as reach of one's social network. There are many different calculations for determining influence. Most calculations include both the number of followers, 2nd degree connections, the number of posts that they send, the number of posts that are forwarded or resent, etc.

Based upon the business rules that you have established Genesys will then route the social interactions to agents and allow the agents to respond in channel, through direct message or via alternative channels.

3.3 Standard Use Cases

The Digital Blueprint architecture supports the following standard use cases:

- Chat Basic - The customer can request a chat session with an agent from the company's web site on a specific topic. Configurable routing logic distributes the interaction to the best available agent depending on the subject and the agent skill. The agent receives the entire customer context (including requested subject).
- Proactive Chat for Sales - Offer an invite to a chat session between the website visitor and the contact center agent based on specific customer behavior on the website.
- Proactive Engagement with Co-browse - Offer an invite to a chat or voice session between the website visitor and the contact center agent based on specific customer behavior on the web site. The customer is then able to initiate a Co-browse session and connect with the current agent for assistance.
- Inbound Email Management Basic - A customer sends an email to a company email address. The email is captured by the Genesys system. The email is then queued and automatically distributed to the appropriately skilled agent.
- Inbound Email Management Advanced - A customer sends an email to a company email address. The email is captured by the Genesys system and a content analysis is performed to assign a category to the email. The email is then queued to the best available agent with the skill set corresponding to the category. After the agent has compiled the email answer, the email is reviewed by a supervisor depending on the agent.
- Proactive Notification via SMS - Ability to proactively send customers notifications via SMS to keep them informed on the status of their request or account.
- Callback Immediate Use Case - A customer initiates a service to place an outbound call from the contact center that connects him to an agent via the Web or Mobile Application. The contact center agent is provided with context of the request.
- Callback Scheduled - A customer requests an outbound call from the contact center at a specific

time that connects him to an agent via the Web or Mobile App. The Contact Centre Agent is provided with context of the request.

For more detailed information about each use case, including interaction flows, please refer to [Use Cases](#) which is contained in Genie under Sales Resources, SC Method.

3.4 Component View

The Component View provides an overview of the components which are included in the Digital Blueprint. The diagram below shows the main components involved in the Digital Blueprint.

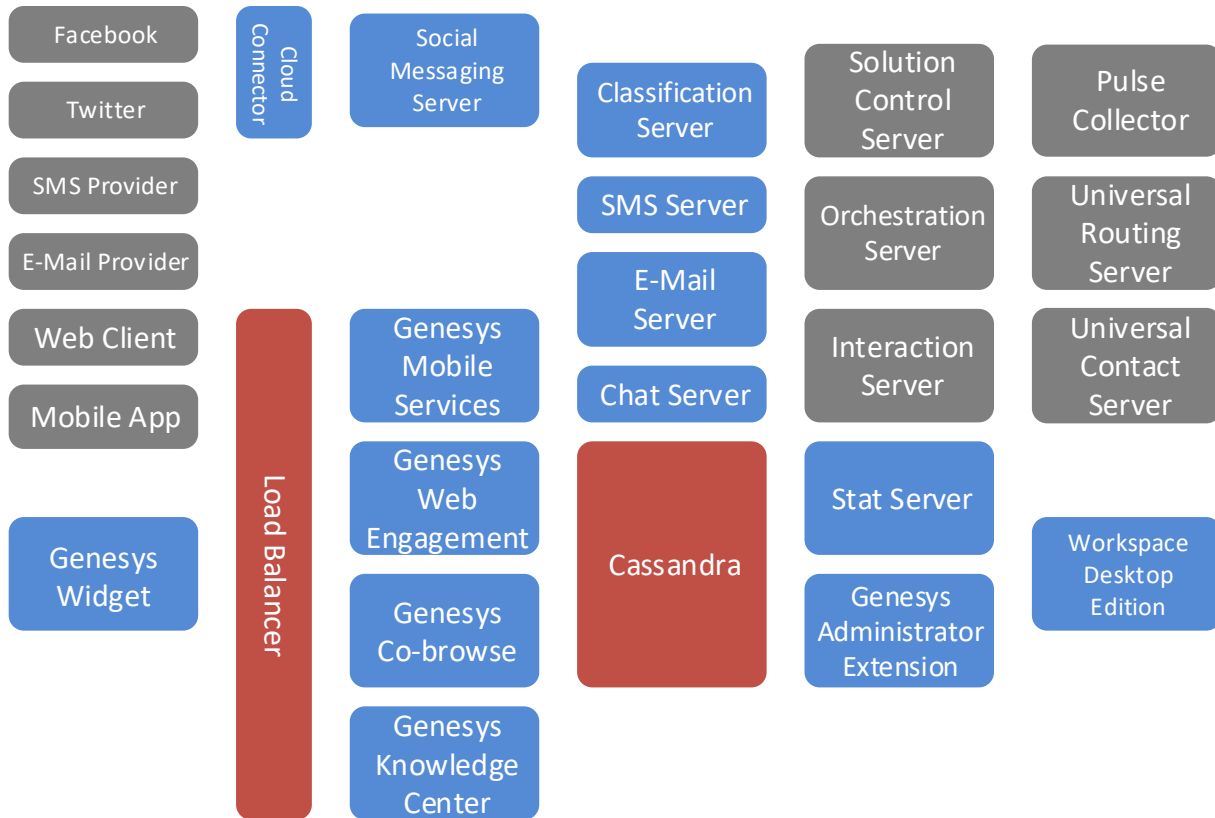


Figure 9 – Digital Component Overview

This chapter groups components into three categories:

- Common components – Components used by most channels which provide supplemental functions
- Workflow components – Used by all channel and required for interaction routing
- Channel specific components – Components dedicated to one channel

Refer to *Genesys Common Components Architecture Blueprint* for a description of general Genesys component used for service orchestration and routing, reporting and configuration and management. An overview is included in [Appendix A – Common Components Summary](#).

3.4.1 Common components

Genesys Mobile Services

Genesys Mobile Services (GMS) is a generic component, which exposes predefined Genesys services to the external world. Within the Digital Blueprint GMS provides the following high-level services:

- Enable chat clients through a general API
- Expose contact center statistics
- Submit web forms as email interactions
- Allow both immediate and callbacks to be initiated
- Enable seamless context transfers from external environments such as a mobile application to the call center

Workspace

The Genesys Workspace desktop delivers an agent facing applications that enables agents to manage customer interactions across a variety customer touch points such as voice, email, chat, social media, sms, etc. Workspace integrates both the various channels and delivers agents tools to enhance managing the customer relationship such as access to standard responses, customer history, KPI and other key information.

Workspace enable agents to handle customer interactions, manage their status, and interact with other resources in the contact center.

Workspace provides a set of atomic and composite views on the agent's workstation desktop that enables the agent to perform privileges assigned to the agent's role. Atomic views typically have a single function, such as viewing case data or specifying a disposition code. Composite views enable an agent to perform multiple functions such as previewing and accepting interactions, or managing their status, meetings, and contacts. Workspace can be easily customized providing detailed control over enabled features. Agents can also personalize the look and feel of the application through settings specific to font size, font color, column order, etc. Workspace Desktop Edition contains several plug-ins which are product specific are bundled together to enable the agent with additional product capabilities.

Workspace connects directly to the enabled Genesys components and is an HA aware application allowing it to automatically work with the high availability provided by Genesys.

Genesys Widget

The Genesys Widget is set of web components (HTML, JavaScript, stylesheets and graphics files) that provides pre-packaged, HTML5 responsive elements that can be used by enterprises to enable Genesys Digital channels. The Widget integrates easily within a customer's existing webpage and allow quick deployment. The Widget can also be branded to match your web presence.

Currently the Widget supports following use cases:

- Reactive chat
- Proactive chat with web engagement

Note: The Widget is a general enabler to make adoption and integration of Genesys Digital channels much easier for end customer. At present the only channel which is considered approved is Chat but other channels will continue to be added and matured as the Widget is also part of the Genesys commercialization process.

Cassandra

Cassandra is an open source, fault-tolerant, and highly scalable NoSQL database for mission-critical data. It provides built-in support for replication of data across multiple data centers. Cassandra is typically used in many high profile, high volume systems including Facebook. Cassandra is a 3rd party component and is part of the Apache Software Foundation.

The following digital components utilize Cassandra:

- Chat server – for temporary storage of chat sessions for resiliency
- Co-browse – for temporary storage of browsing sessions between the customer and agent
- Web engagement server – for storing data about all sessions which are monitored by the Genesys Web Engagement server and for storing historical data aggregated by elastic search
- Genesys Mobile Services – for storage of callback information and temporary storage of data for other scenarios

Customers are required to provide and support Cassandra in the same manner as other 3rd party databases. Customers may deploy a single Cassandra cluster which is shared by the various Digital components. It is possible to use the same Cassandra cluster that is required as part of the Common Component Blueprint or utilize different Cassandra cluster. For redundancy and data consistency each Cassandra cluster should have at least three nodes.

3.4.2 Workflow components

Interaction Server

Interaction Server manages the interactions once they are received by Genesys and interacts with the Orchestration layer and routing strategies to handle the interaction flow.

- It receives the interaction and operational data from the media interface.
- It stores the operational data in a cache (a database) while receiving and transmitting information about the interaction. This cache also contains queues through which the interaction passes as part of its processing.
- It works in concert with the Orchestration components to route interactions according to interaction workflows and routing strategies.
- It enables agents to log in and manage their availability for each Digital channel.
- It is easily extended through open capture points to support the addition of custom media channels
- It can provide interaction/event logging for more detailed reporting

Interaction server uses a database to store the current state of all interactions currently in-progress.

Interaction Server can also be configured with an Event Logger to store detailed information on interaction processes and event messages in a database. The data generated by the Event Logger is leveraged by other components such as product specific data marts.

Interaction Server proxy

Interaction Server proxy helps the offload processing of agent desktop and Stat Server connections from Interaction Server. The proxy keeps a single connection open to Interaction Server and separate connections for each Interaction Server client (desktop or Stat Server). By consolidating the connections and multiplying traffic out to all the client the proxy helps and reduce the load on Interaction Server.

Interaction Server proxy will also play a main role in Interaction Server clusters to mediate the connection between applications and a cluster of Interaction Servers. **The current blueprint does not cover this design.**

Classification Server

The Classification Server can be used during interaction process to analyze the text of the interaction and extract additional context. Screening rules can provide basic pattern-matching queries performed on interaction contents to extract additional information. Genesys Context Analyzer can also be used which applies trained natural language based models to categorize incoming interactions. This classification process can provide additional context on the interaction which can be used by a routing strategy to better optimize routing decisions. Both the screening rules and models are stored in the Universal Contact Server database.

Classification server is also used to apply privacy rules to find and mask sensitive data in interaction content. With chat interactions the privacy rules are actually applied by Chat Server in real-time, rather than Classification Server, as messages are exchanged between the agent and customer.

Classification server is only used on text based interactions. It is possible to select on which data to apply classification rules – interaction content, interaction user data or arbitrary data provided in request.

eServices Manager Plug-in for Administrator

The eServices Manager Plug-in for Genesys Administrator (GAX) allows users to create and store Knowledge Manager objects (categories, screening rules, standard responses, field codes) in a multi-tiered hierarchical structure. Those objects are later used in strategies through Classification server or by Genesys Workspace (WDE/WWE).

The current release of the eServices Manager Plug-in for Administrator offers essential Knowledge Manager Functionality. Additional functions will be added in subsequent releases. Genesys Content Analyzer is not currently included in the eServices Manager Plug-in. Content Analyzer and other functions which are not yet enabled in the plug-in can be managed through the legacy Knowledge Manager application.

Privacy Manager Plug-in for Administrator

The Privacy Manager Plug-in for Genesys Administrator allows users to create and test rules for masking sensitive data in interaction content. The rules use regular expressions to search for sensitive data, such as a credit card number, and contain an action to perform such as replace the data which was found. The rules are executed by Classification Server. The content of the rules is stored in UCS.

3.4.3 Chat

Chat Server

Chat Server is used to open, conduct, and close chat interactions between agents and customers. Chat sessions can be initiated through:

- Calls from GMS or through the prior Web API server
- Calls from SMS server for session mode
- Calls from customer adapters written for third party messenger apps such as Facebook private messages

The Chat Server component should never be directly exposed to Internet traffic.

During a chat session this component not only relays messages between the parties but also masks sensitive data defined by the Privacy Manager Plug-in, stores the final chat transcript in the UCS database and stores active chat sessions in the Cassandra database (if configured) to allow session continuation in case of component failure.

3.4.4 Email

Email server

Email server is responsible for retrieving emails from a corporate email system and submitting them to Genesys. It also receives emails from Genesys and sends them to corporate email systems. The email server supports following protocols:

- POP3 – for receiving emails. This protocol does not allow emails to be left on the email server which may be required by some customers
- IMAP – for receiving emails
- SMTP – for sending emails
- EWS (Exchange Web Services) – for both receiving and sending emails with a Microsoft Exchange email system
- Genesys Web API – for receiving emails submitted through web forms. Both GMS and the Genesys Web API server can provide this functionality.

All these protocols can be configured for either encrypted (TLS) or unencrypted communication.

In addition it is possible to add custom handlers written in java to pre-process the message body before it is submitted to Genesys (for inbound emails) or before it is sent to corporate email server (for outbound emails). This customization can be used to handle emails which contain an encrypted body.

For detailed instruction about this feature called “MIME Customization” please follow this link

<https://docs.genesys.com/Documentation/ES/8.5.1/Depl/CustMime>

3.4.5 SMS

SMS Server

SMS Server receives and handles SMS and MMS messages sent from a mobile client. SMS Server uses SMPP v3.4 protocol for SMS support, and MM1, MM7 protocols for MMS support.

SMS Server supports two operational modes:

- Paging mode – where each SMS message is managed separately. An individual SMS from a mobile client will be routed and the agents response will be sent back (paging inbound) or an individual SMS message may be sent to a mobile client from the contact center (paging outbound).
- Session (chat) mode – creates an interactive conversation between a mobile client and an agent in the form of a conventional chat session. All messages received and sent during this session are associated with one interaction, which corresponds to this SMS session.

For MMS only receipt and routing of inbound messages is supported. Agents cannot sent outbound MMS MMS content.

Genesys SMS Server does not support nonstandard protocols like web service calls or email-2-sms gateways for receiving and sending SMS. Custom integrations to such services can be provided as part of the Genesys Digital Solution utilizing Genesys APIs.

3.4.6 Co-browse

Co-browse Server

The Genesys Co-browse server manages and synchronizes the browsing session between the customer and agent. Co-browse keeps a full copy of the web page the customer is visiting and renders this page to an agent. When this page is rendered the server will also applying any rules established for security and masking data ensuring that agents can only view the approved sections of the web page. To optimize traffic the co-browse server does not download or modify in any binary files referenced in the page like images. Those files are directly downloaded and rendered by the agent's browse. Co-browse uses Cassandra for short-term storage of the co-browse session.

Co-browse Plug-in for agent application

Genesys Workspace includes a plug-in which allows the agent to join and interact with a co-browse session. The plug-in uses a REST API to control the session and manage the page display. It is possible for customers to create similar plug-in for a custom agent desktop application. Details on creating such a plug-in are available at: <https://docs.genesys.com/Documentation/GCB/8.5.0/Developer/AgentApps>

Co-browse JavaScript

Genesys provides co-browse JavaScript code must be included on every page where a customer wants to enable a Co-browse session. This code is responsible for synchronizing the content displayed to customer and agent. It communicates using REST and CometD APIs with the Co-browse server. Details on the co-browse JavaScript are available at:

<https://docs.genesys.com/Documentation/GCB/latest/Deployment/WebsiteInstrumentation>

3.4.7 Web Engagement

3.4.7.1 Web Engagement Server

The Genesys Web Engagement (GWE) Server is the primary component of the web engagement solution. It is responsible for controlling the logic of engagement by processing a user defined sequence of events coming from any source. The current release only supports web events. as source of events.

The GWE server receives events through a REST API and executes the user defined rules to determine if the events are actionable. For those events that are actionable the server executes Orchestration logic to decide how to respond to the event and if an engagement is required which resource/agent should be used. If there is a need to provide a response to the customer the GWE server also can notify any client currently connected and trigger an action (like displaying a chat popup or simple ads). In addition, the GWE server provides an API to access the raw data gathered from monitoring sessions and aggregate data providing a high level overview.

The GWE server uses Cassandra to store all data and elasticsearch to index the data for quick access. GWE also uses Kibana to visually display the aggregated data.

3.4.7.2 Reporting Server

The GWE Reporting Server is design to process and report on the large amount of data gather by the GWE Server. It provides reporting data about the customer engagement on the web site. Data is gathered and aggregated by reporting server. Data is then pushed to Pulse providing summary views and details are displayed using Kibana which is embedded in GWE server.

A description of available reports can be found here

<https://docs.genesys.com/Documentation/GWE/latest/Report/Welcome>

Due to the heavy load required during aggregation processing the reporting server requires additional dedicated Cassandra nodes in a cluster.

3.4.7.3 Web Engagement WDE Plug-in

The Web Engagement Plug-in for WDE allows the agent to view all the pages (url's) the customer visited prior to accepting the engagement chat session. It also shows the pages that customer is currently viewing. The Plug-in only allows the agent to see the content of pages if there is no requirement for user authorization or the pages are not using session information to display dynamic data. If there is a need to see the current customer page and it required authentication or session information then it is recommended to add Genesys Co-browse as part of the solution. Enterprise customers can also create a similar Plug-in for a custom desktop by using the History API to retrieve details on the users browsing history.

3.4.7.4 Web Engagement JavaScripts

Genesys Web Engagement includes three java scripts which should be integrated by customers to provide monitoring, notification and engagement agent.

The **Monitoring Agent** records the web browsing activity. It generates basic system events such as VisitStarted, PageEntered, and additional custom business events such as 'add-to-shopping cart'. These events are sent to the Web Engagement Server for further processing.

The **Notification Agent** allows a web server to push data to a browser without the browser explicitly requesting it, providing an asynchronous messaging channel between the server and browser.

The **Engagement Agent** provides the engagement mechanism and initialization for chat communication and web callback. If the Genesys Widget is used this script is not required.

3.4.7.5 Web Engagement Plug-in for Administrator

The Web Engagement Plug-in provides two administrative functions:

- For developers/admins it provides the Script Generator tool which allows you to create JavaScript code snippets that you will added to your web pages to enable Web Engagement monitoring.
- For business users it allows to define categories which are used to implement the simple engagement model. Web Engagement categories contain business information related to URL or web page titles. These categories are used in the CEP rule templates, which provide rules that

define when to submit actionable events to Web Engagement—this is what starts the engagement process.

3.4.8 Mobile Engagement and Callback

3.4.8.1 Genesys Mobile Services

Genesys Mobile Services provides a set of RESTful APIs which are used to deliver multiple services to mobile, web and other channels. In addition to the APIs the Genesys Mobile Services (GMS) platform provides several management capabilities for callback services. It provides:

- A user interface to manage the provisioning and deployment of callback services
- Callback service monitoring allowing the user to view the number of callback in queue, number of tries and cancel callback requests.
- The ability to push notifications and updates to native mobile applications using iOS or Android push notification services

When a callback is generated a voice application validates the recipient prior to routing the called party to an agents. Genesys provides a Composer project which contains a voice callback application. This application can be directly modified by customers to meet their needs.

Genesys Workspace allows the agent to receive callback notifications or reschedule callbacks directly from Workspace. If the customer has a custom desktop it is possible to integrate it with the Genesys callback solution using the standard APIs to deliver similar functionality.

3.4.9 Social Engagement

3.4.9.1 Social Messaging Server

The Genesys Social Messaging Server interfaces with social media sites to bring social interactions into the Genesys system. The server uses drivers to connect to different social media sites. Genesys delivers out of the box drivers for:

- Facebook, including Facebook Messenger
- Twitter

Currently the Social Message Server may connect to the social media sites directly or connect to Genesys cloud which manages the connectivity to the social media sites. For all new deployments Genesys recommends that connectivity is performed through Genesys cloud which in turn manages the direct connectivity to Facebook and Twitter. This approach insulates customers from the frequent API changes that the social media sites make and when API changes are made Genesys is able to directly manage the updates to ensure API compatibility.

With the cloud based architecture each customer has dedicated account in Genesys cloud where they configure the criteria used to capture social media data from Twitter and Facebook. The data received is buffered in Genesys cloud. The Social Media server at the customer premises connects to the cloud using this dedicated account and retrieves the previously captured data which is then submitted for processing

in Interaction Server. When responses or posts are sent to social sites a similar flow occurs as the Social Media server sends a response to Genesys cloud where it is buffered and then sent by Genesys cloud to Facebook or Twitter.

3.4.9.2 Custom driver

Genesys Social Engagement can integrate with additional social networks through the creation of a custom driver. This integration is performed through the API described at:

<http://www.genesyslab.info/repository/eServices/OpenAPI/index.html>

The custom driver uses the the API to submit captured interactions to the Social Media server and provides the ability to respond to those interactions.

Genesys Professional Services has already developed drivers for following Social networks:

- Google+
- Lithium
- Radian 6

3.4.9.3 Social Media Plug-in for Workspace Desktop

Genesys Social Engagement includes a plug-in for Workspace Desktop Edition which enables agents to handle social media interactions. The plug-in renders content of Twitter and Facebook interactions providing the agent with additional information allowing to better handle it.

3.4.10 Genesys Digital Solution Components

The following table lists the Genesys components that make up the Digital Blueprint. Optional components are noted in the table.

Category	Component	Version	Notes
Workflow components	Interaction Server	8.5.1+	
	Interaction Server proxy	8.5.1+	
	Classification Server	8.5.0+	
	eService Manager	8.5.1+	
	Privacy Manager	8.5.1+	
Common components	Widget	8.5+	Customer facing UI
	Workspace Desktop Edition	8.5.111.21+	Minimum version required for native callback support
Channel Specific Components			

Chat	Chat Server	8.5.104.10+	Required for Cassandra persistence and proper usage of privacy filtering
Email	Email Server	8.5.0+	
SMS	SMS Server	8.5.1+	
Co-browse	Co-browse Server	8.5+	
	Co-browse WDE plug-in	8.5+	
Web Engagement	GWE Server	8.5+	
	GWE Reporting Server	8.5+	
	GWE WDE plug-in	8.5+	
	GWE GAX plug-in	8.5+	
Social	Social Messaging Server	8.5.1+	
	Cloud driver for Twitter	8.5.3+	
	Cloud driver for Facebook	8.5.3+	
	Social WDE plug-in	8.5.3+	

Table 1 - Genesys Component List

3.4.11 Required Genesys Common Component Services

The Genesys solutions use a common set of component for configuration/management, routing and reporting. These components are detailed within the Common Component blueprint.

The following is a list of the mandatory Genesys Common Components required by Digital Blueprint. This table lists any dependencies and any minimum versions required.

Category	Component	Minimum Version	Notes
Orchestration	ORS, URS, Stat Server, UCS, Genesys Mobile Services, Genesys Rules	N/A	Base versions used in Common Component blueprint are fine
Reporting	ALL		Base versions used in Common Component blueprint are acceptable with the additions noted below
	Stat Server	8.5+	Java extension for Digital
	Interaction Concentrator, Info Mart		Support for focus time and callback reporting

Configuration & Management	ALL (SNMP Master Agent is optional)	N/A
	Genesys Administrator Extension (GAX)	8.5.220.20+

Table 2 – Required Common Components

3.4.12 3rd Party Components – load balancers and reverse proxies

Most Digital products normally expose an HTTP interface for client-server communications. 3rd party components such as HTTP Load Balancers and Reverse Proxies mediate this communication and typically collapse this communication into a single device from a Genesys perspective (that is, most Reverse Proxies would normally Load Balance HTTP traffic to their backend nodes). Distribution of HTTP traffic will normally rely on Cookie header to establish create a “sticky session” to a given node, as described further below.

Origin	Destination	Protocols	LB Stickiness	Note
Customer browser	Co-browse Cluster	HTTP(s), WS(s)	Cookie: <i>gcbSessionServer</i>	Cookie <i>gcbSessionServer</i> is added by Javascript so that both the customer and agent connect to same Co-browse node
Workspace Desktop Edition	Co-browse Cluster	HTTP(s), WS(s)	Cookie: <i>gcbSessionServer</i>	Cookie <i>gcbSessionServer</i> is added by Javascript so that both the customer and agent connect to same Co-browse node
Customer browser	Web Engagement Cluster	HTTP(s), WS(s)	Cookie: <i>GWROUTEID</i>	Web Engagement server adds Set-Cookie: <i>GWROUTEID</i> to the first HTTP response, forcing the Browser to add <i>GWROUTEID</i> cookie to subsequent HTTP requests
Workspace Desktop Edition	Web Engagement Cluster	HTTP(s), WS(s)	Cookie: <i>GWROUTEID</i>	Not used

Table 3 – Load Balancing parameters

3.4.12.1 Co-browse

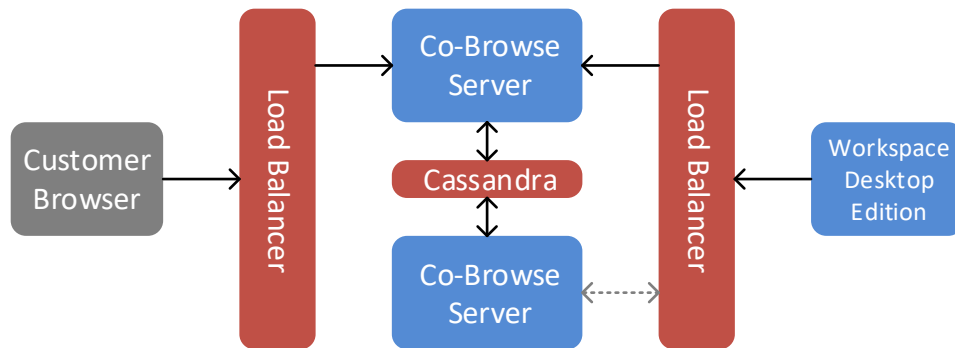
3.4.12.1.1 Product behavior

Genesys Co-browse relies upon HTTP load balancing to provide scalability and high availability of Co-browse nodes. The following client-server HTTP communications would traverse a load balancer:

Origin	Destination	Protocols	Cookie
Customer web browser	Co-browse server node	HTTP(S), WS(s)	<i>gcbSessionServer</i>
Workspace Desktop Edition	Co-browse server node	HTTP(S), WS(s)	<i>gcbSessionServer</i>

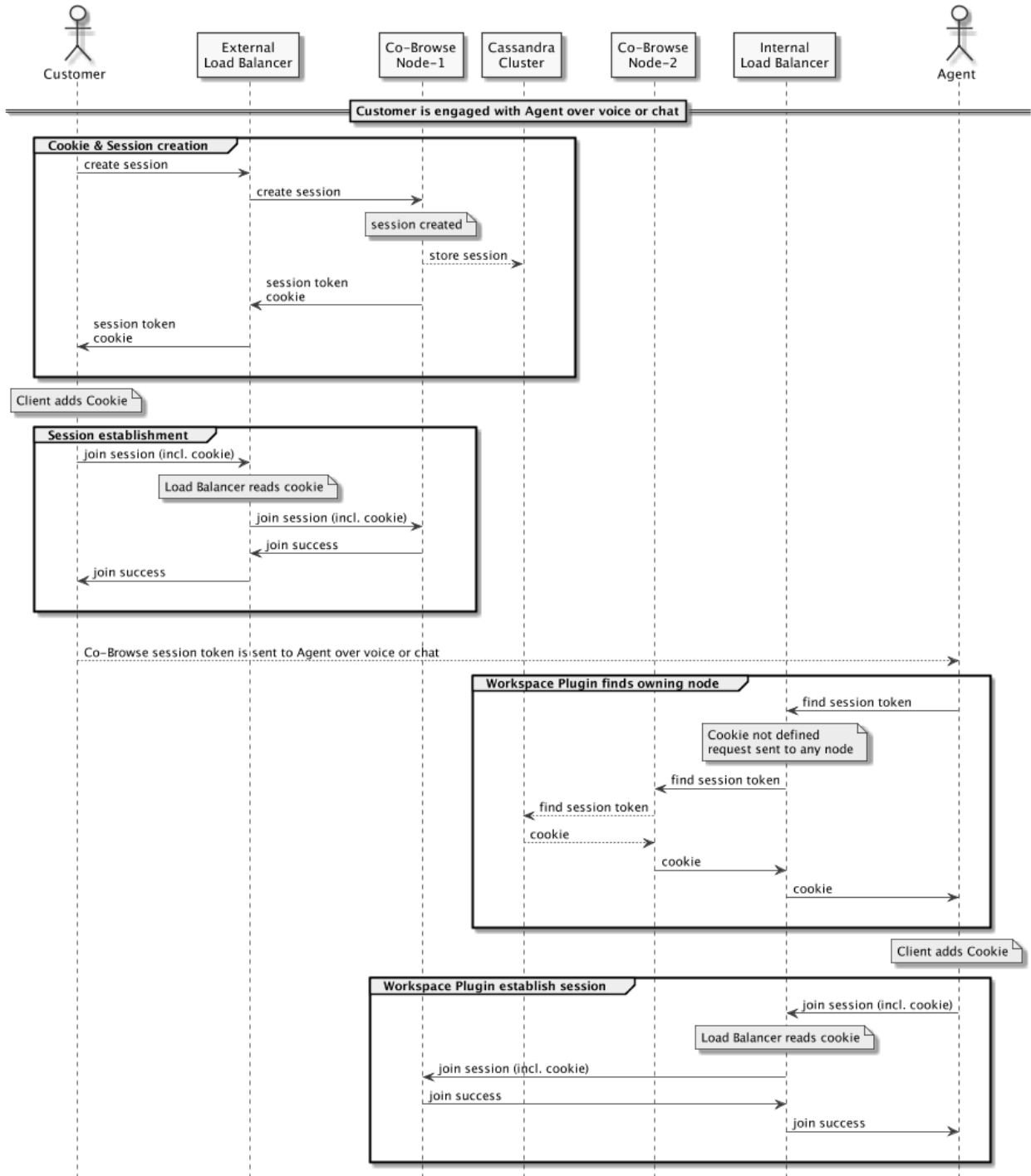
3.4.12.1.2 High Level Architecture

Both the customer and agent must join the Co-browse session on the same Co-browse Server node. Genesys Workspace may need to query multiple Co-browse Server nodes to find the node owning the Co-browse session.



3.4.12.1.3 Sequence diagrams

3.4.12.1.3.1 Normal Flow



3.4.12.1.4 Load balancer requirements

To deploy a Co-browse cluster (N+1 nodes) in a production environment Genesys requires the following:

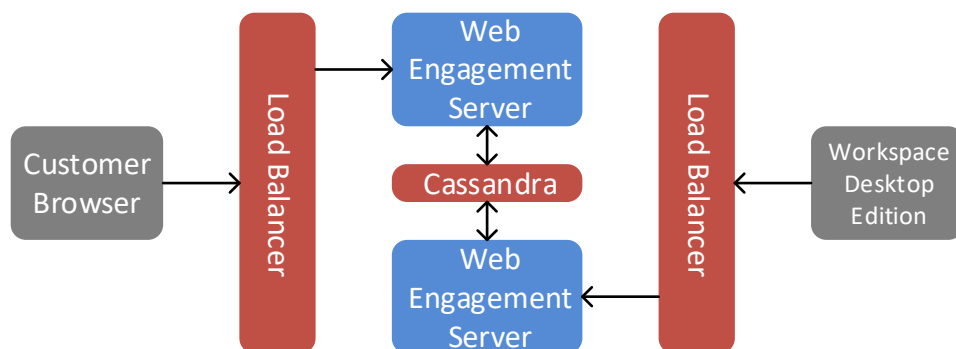
1. A Third Party HTTP load balancer is mandatory. Load balancers are not provided nor validated by Genesys.
2. The load balancer must support [health check monitoring](#) of each node (200 OK response to GET /cobrowse/health). Monitoring allows the load balancer to detect failure of a Co-browse node. Failure of Co-browse node cannot be recovered gracefully however health detection is required to notify node failure to the client (and thereby allow the session to be manually restarted).
3. The load balancer must support [cookie based session stickiness](#). Cookie 'gcbSessionServer' is added by Genesys components into HTTP requests, and should be used by the load balancer to distribute requests to appropriate Co-browse node.
4. [WebSocket protocol support is highly recommended](#). Use of WebSocket protocol (WS) improves performance and considerably reduces the request throughput rate generated by each customer's browsing session.
5. If WebSockets are enabled the load balancer must support both HTTP and WS balancing in order to handle scenarios where a WebSocket session cannot be established due to client/infrastructure capabilities and HTTP is required.
6. Secure Socket Layer (SSL) support is typically required. Since Genesys Co-browse relies on application generated Cookie headers, if the incoming HTTP traffic is SSL encrypted (HTTPS) the Load Balancer must be configured to perform SSL offload so that it can decrypt the incoming HTTPS traffic and access the Cookie header used for session stickiness. The resulting outgoing traffic from the load balancer to the Genesys Co-browse server could either be re-encrypted (SSL Bridging/Re-encryption) or remain un-encrypted (HTTP) to reduce load on the Co-browse server (SSL Offloading). The load balancer logic relies on application Cookies added at the HTTP protocol layer so the load balancer must have access to HTTP headers as mentioned at previously.

3.4.12.1.5 Further details on the Load balancer configuration is provided in Appendix A.

3.4.12.2 Web Engagement

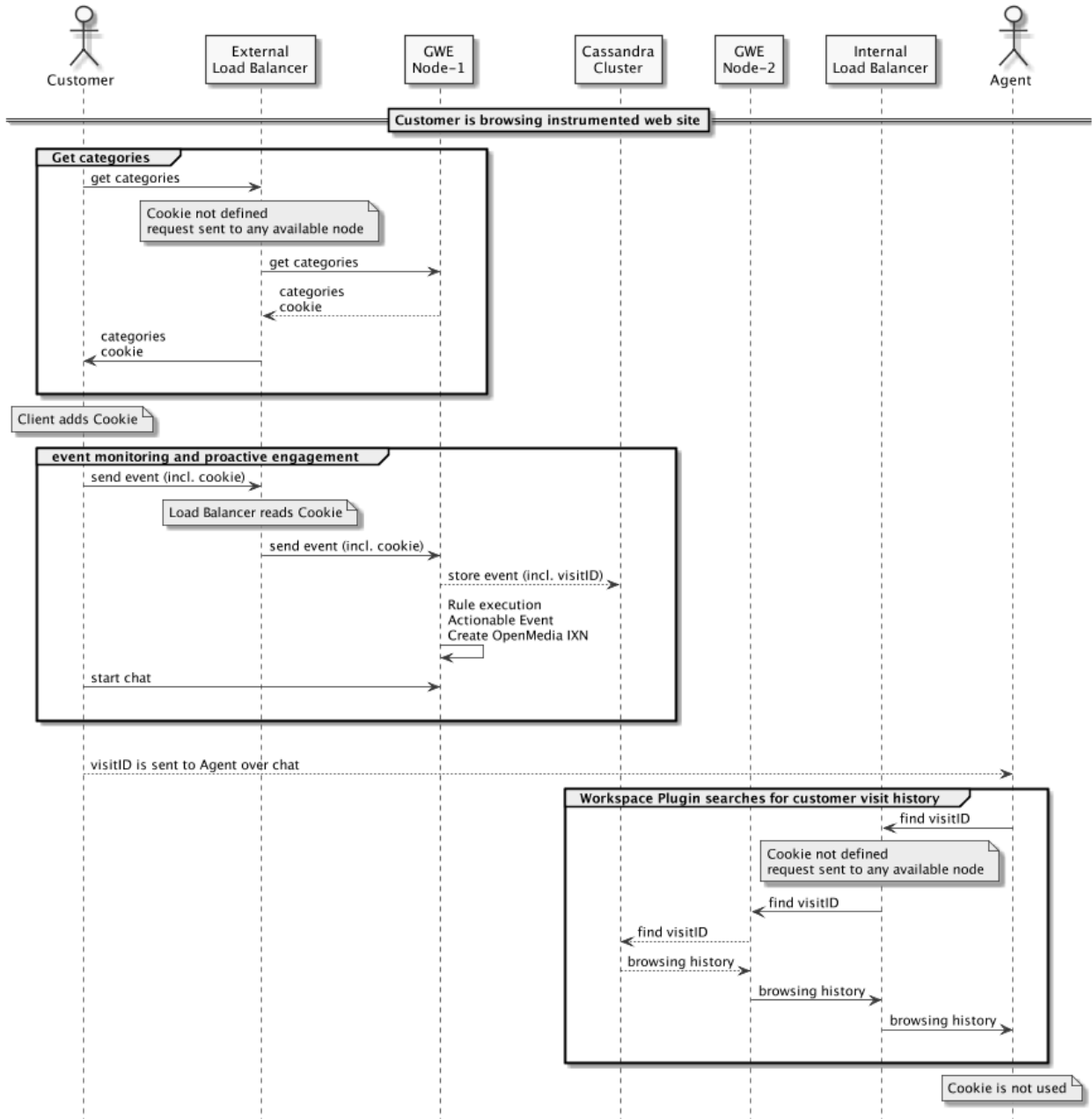
3.4.12.2.1 High Level Architecture

With Genesys Web Engagement as the customer navigates the website the customer browser will be sending monitoring and business events to Web Engagement Server node. Workspace Desktop Edition can access the customer’s browsing history from any Web Engagement Server node since the browsing history is accessible through the underlying Cassandra Cluster.



3.4.12.2.2 Sequence diagrams

3.4.12.2.2.1 Normal Flow



3.4.12.2.3

3.4.12.2.4 Load balancer requirements

To deploy a Web Engagement cluster (N+1 nodes) in a production environment Genesys requires:

1. A HTTP load balancer is mandatory to achieve N+1 scalability and high availability. Genesys does not provide nor validate any specific Load Balancer, therefore proper Load Balancer configuration must be defined during project execution based on requirements described in this section.
2. Genesys Web Engagement (GWE) is designed to add a server side Cookie the first HTTP response ("Set-Cookie: **GWROUTEID**=.<app-name>; Path=/"). Once the cookie has been added this GWE node expects to receive subsequent HTTP requests containing the specified Cookie header. If request goes to a different GWE node, then request will be processed, but **alarms will be raised against Management Layer** by the other GWE node since GWROUTEID Cookie received will not match the local application name. The new GWE node will also force a re-write of Cookie.
3. The requirement to use the GWROUTEID Cookie in a load balancer configuration is particularly important for internet facing Load Balancer
4. If TLS is required we recommend to enable SSL Termination/Offloading (decryption) or Bridging (decryption + encryption) on the Load Balancer, so that Load Balancer can access the GWROUTEID cookie from HTTP messages. With SSL Pass-through the load balancer would not be able to correctly send the traffic to the correct GWE node which would result in alarms being raised.
5. In case a GWE node fails the load balancer can detect the node failure (based on periodic health check URL), and send subsequent HTTP(S) requests to a surviving GWE node. Monitoring and Chat sessions will continue on the surviving GWE node, which will access previous session data from Cassandra. There is a possibility that as a consequence of specific transitions happening during failover, some events may not properly trigger an engagement strategy.
6. The load balancer must be able to monitor the cluster status for each node using following URLs. The Status is considered healthy is the load balancer receives a successful response to either:
 - I. `http(s)://Web Engagement Server Host:Web Engagement Server Port (secured port for https)/server/about`
 - II. `http(s)://Web Engagement Server Host:Web Engagement Server Port (secured port for https)/server/isAlive`
7. The first HTTP connection from Workspace through the load balancer could end up on any GWE node in the cluster. This is acceptable as Workspace only needs to retrieve the visit history from Cassandra.
 - If a GWE node fails the Workspace plug-in will automatically re-establish the session to a different node (through the load balancer).

3.4.13 Additional 3rd Party Components

Component	Recommended	Version	Note
Database	MSFT SQL Server 2012, Oracle 12c		
Storage	Cassandra	2.2+	Required for Co-browse, Web Engagement and Chat server
Virtualization	VMWare ESXi Hyper-V	5.1+ 2012 R2	Note: Hyper-V is supported for the Digital components but not yet supported for all the Common Components

Table 4 – 3rd Party Components**Note:**

- Operating systems are not listed. Genesys recommends either RedHat Enterprise Linux 6.0 64-bit or Windows Server 2012 64-bit. Some components may only be supported on a specific OS. When a component is supported only on a single OS this is specifically called out.
- Databases are required as part of the Digital Blueprint, such as the Interaction Server database, but not listed. RDBMS is often a customer preference. Genesys recommends either Microsoft SQL Server 2012 or Oracle 12c. In general for business continuity/DR there may also be specific requirements on the database features utilized.

3.5 Limits and Constraints

The following are limits and constraints in both the scope of the solution and the underlying products:

- Genesys Email Server only supports a single outgoing protocol per instance. If for security reasons the customer doesn't want to allow emails to be sent with different **From** headers through single SMTP account the customer has to deploy a separate Email Server for each SMTP account they want to use for sending emails.
- Genesys Email Server does not support MAPI

4 Deployment View

4.1 Centralized Deployment

The Digital solution is typically deployed in a centralized model. This model assumes that the customer has a data center that is reachable by all agents and that the network has the capacity to support the traffic between the solution components in the data center and between the agents' desktop/endpoints and the data center.

A typical deployment will be similar to that depicted in the following diagram. Note the diagram shows a logical diagram highlighting the distribution of the components. The specific components distribution across servers would vary based upon the capacity requirements and sizing. Further details showing the signaling is included in the Interaction Views.

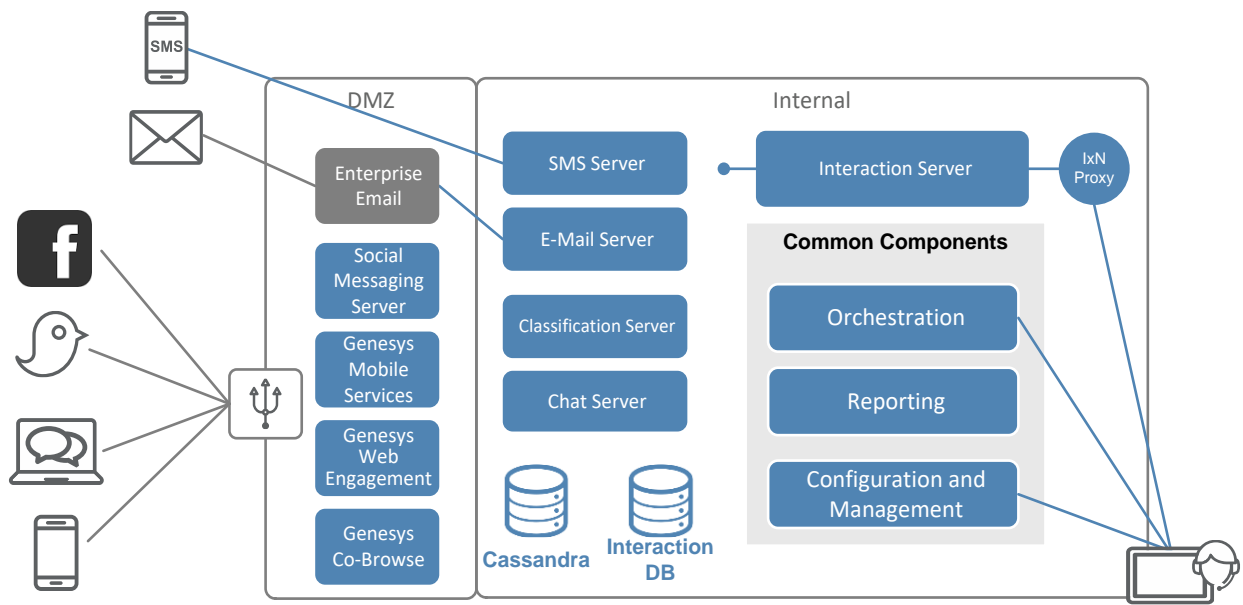


Figure 10 – Central Deployment Model - Digital

The Digital Engagement Center Solution consists of numerous channels as depicted above. The distribution of the Genesys components in an architecture will vary based upon which components are involved, the communication requirements and the established security zones with some components deployed in a DMZ and other components deployed in an internal zone.

Interactions from the Digital channels enter the enterprise and Genesys environment via two mechanisms:

1. HTTP traffic – HTTP requests/APIs which are submitted by external elements such as the web browser or mobile applications. These requests will be managed by firewalls and HTTP load balancers to provide security and high availability. The Genesys components which will receive this communication are typically deployed in a DMZ as depicted in the diagram.
2. Channel specific protocols – Email and SMS both use non HTTP based protocols such as POP3/SMTP or SMPP to handle receipt and sending of interactions. Genesys Email may also

use EWS for integration with Exchange in which case it may be potentially located in the DMZ.

The Digital interactions typically follow a common flow. As they enter the Genesys environment there may be some initial pre-processing of the request to determine whether it is actionable. For example if the case of Genesys Web Engagement there is an on-going set of events which will be sent to the Web Engagement server as the customer navigates the web site and only if the appropriate rules fire and there is an actual engagement will an interaction be routed to agents.

The communication are ultimately translated from the channel specific components into Genesys interactions and submitted to Interaction Server. Interaction Server may perform additional processing on the interaction, such as applying rules to determine if an interaction is actionable in the case of social media. The interactions will then be held in a queue and then submitted to the Orchestration layer.

The Orchestration layer will invoke the strategies and business logic to determine how the interaction should be processed. The orchestration strategy may invoke the Classification Server to extract additional context from the text of the interaction to better understand the objective and perform better routing. Based upon the information extracted from the interaction Orchestration which will perform additional logic such as determining which agents are the best candidates or targets for the interaction and when an agent is finally selected informing Interaction Server to transfer the interaction to the selected agent.

For real-time interactions such as chat or co-browse once the interaction has been routed and communication is established all on-going communication occurs directly between the customer and agent.

Once an agent has completed the interaction additional actions may be performed. With chat could include emailing a copy or the chat transcript to the customer or in the case of email performing additional post-processing prior to sending the email to perform a quality review or check for acceptable language. The specific post-processing steps may be performed automatically through configuration or based upon the strategy.

The Digital Blueprint utilizes elements from the Common Component Blueprint to provide Orchestration, Management and Reporting. Web based user interfaces are provided for administering the solution and providing real-time and historical reporting. The Configuration and Management node provides the OAM&P functions including configuration, solution monitoring, managing high availability/failover and alarming. The Reporting node provides both real-time and historical data collection. The historical data which is captured is transformed into the data warehouse for reporting and directly accessible through a web based reporting solution which includes standard default reports.

The Digital architecture may include proxies such as the Interaction Server Proxy and UCS Proxy to reduce the load on the underlying server components.

The following table lists the components that make up each of the different layers. The actual component distribution may vary based upon the final architecture.

The following table lists the components that make up each of the nodes in the deployment model.

Node	Component	Comments
	Interaction Server	

Digital Workflow components	Interaction Server proxy	
	Classification Server	
	eService Manager	Plug-in for Genesys Administrator
	Privacy Manager	Plug-in for Genesys Administrator
Digital Common Components	Widget	Currently supports Chat
	Workspace Desktop Edition	

Digital Channels

Channel	Component	Comments
Chat	Chat Server	
Email	Email Server	
SMS	SMS Server	
Co-browse	Co-browse Server	
	Co-browse WDE plugin	
Web Engagement	GWE Server	
	GWE Reporting Server	
	GWE WDE plug-in	
	GWE Admin plug-in	Plug-in for Genesys Administrator
Social	Social Messaging Server	
	Cloud driver for Twitter	
	Cloud driver for Facebook	
	Social WDE plugin	

Table 5 - Data Center – Digital Components

Please see section 6.1 Solution Sizing Guidelines for details on the sizing of each virtual machine.

4.2 High Availability Deployment

Genesys recommends deploying all components with High Availability (HA). There are multiple ways to implement HA, depending on the Genesys component.

Some Genesys components utilize Genesys standard high availability which consists of a primary and backup process. The backup process takes over from the primary if it fails. Other Genesys components, primarily those which use HTTP communication, can take advantage of HTTP based load-balancing which allows for an N+1 model.

The following table summarizes the high availability levels supported by each component.

Genesys Component	HA Mode	Notes
Interaction Server	Warm standby	Cluster is restricted for now

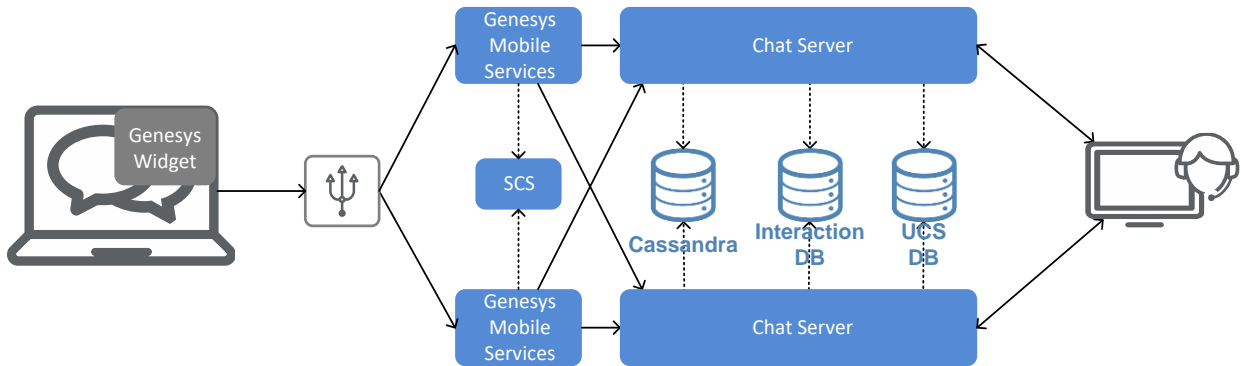
Interaction Server Proxy	Warm standby	
Chat Server	Load balancing (N+1) and Warm standby	Load Balancing (N+1) is recommended and does not require HTTP Load Balancer
Email Server	Warm standby and load balancing	
Classification Server	Warm standby and load balancing	
SMS Server	Warm standby	
Social Messaging Server	Warm standby	With the Genesys cloud architecture all interactions will be kept in the cloud and available once connectivity is re-established
Co-browse Server	Load balancing (N+1)	HTTP Load Balancer required
Web Engagement	Load balancing (N+1)	HTTP Load Balancer required
Mobile Server and Callback	Load balancing (N+1)	HTTP Load Balancer required

Table 6 – High Availability Matrix

The following sections provide information on high availability for the various components of the solution.

4.2.1 Web Chat HA

This section describes the end-to-end high availability (HA) for Web Chat functionality. The scope includes communication from the customer browser to the Agent Workspace. For simplicity not all components are covered. Both GMS and Chat Server and implemented following a N+1 HA model. The following diagram includes all components involved in Web Chat HA and recovery logic in case any node fails.

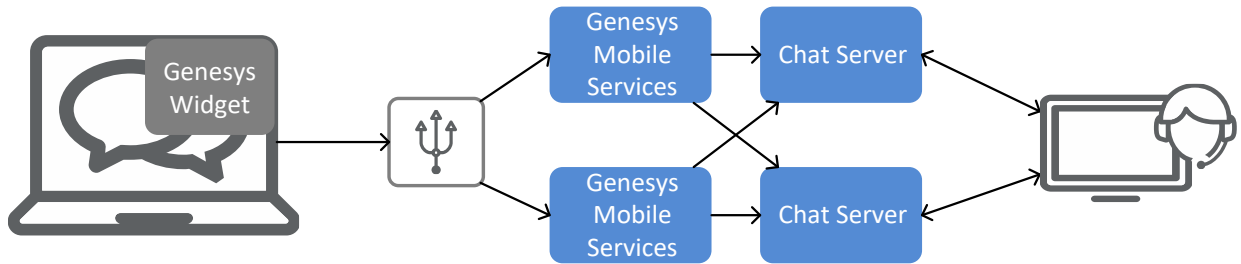


GMS has an active connection to SCS which is used by GMS to obtain the current availability of the Chat Servers and select which Chat Server will be used.

4.2.1.1.1 Genesys Mobile Services

Genesys Mobile Services can be deployed in an N+1 model as part of a cluster. For Chat functionality GMS does not rely on Cassandra. GMS uses a stateless protocol for sessions between the Widget and Chat Server node. HTTP requests from the Genesys Widget include the Chat Server DBID, so that effectively any GMS node can process this request and forward to proper Chat Server.

The following diagram includes components involved in GMS Node failure and recovery

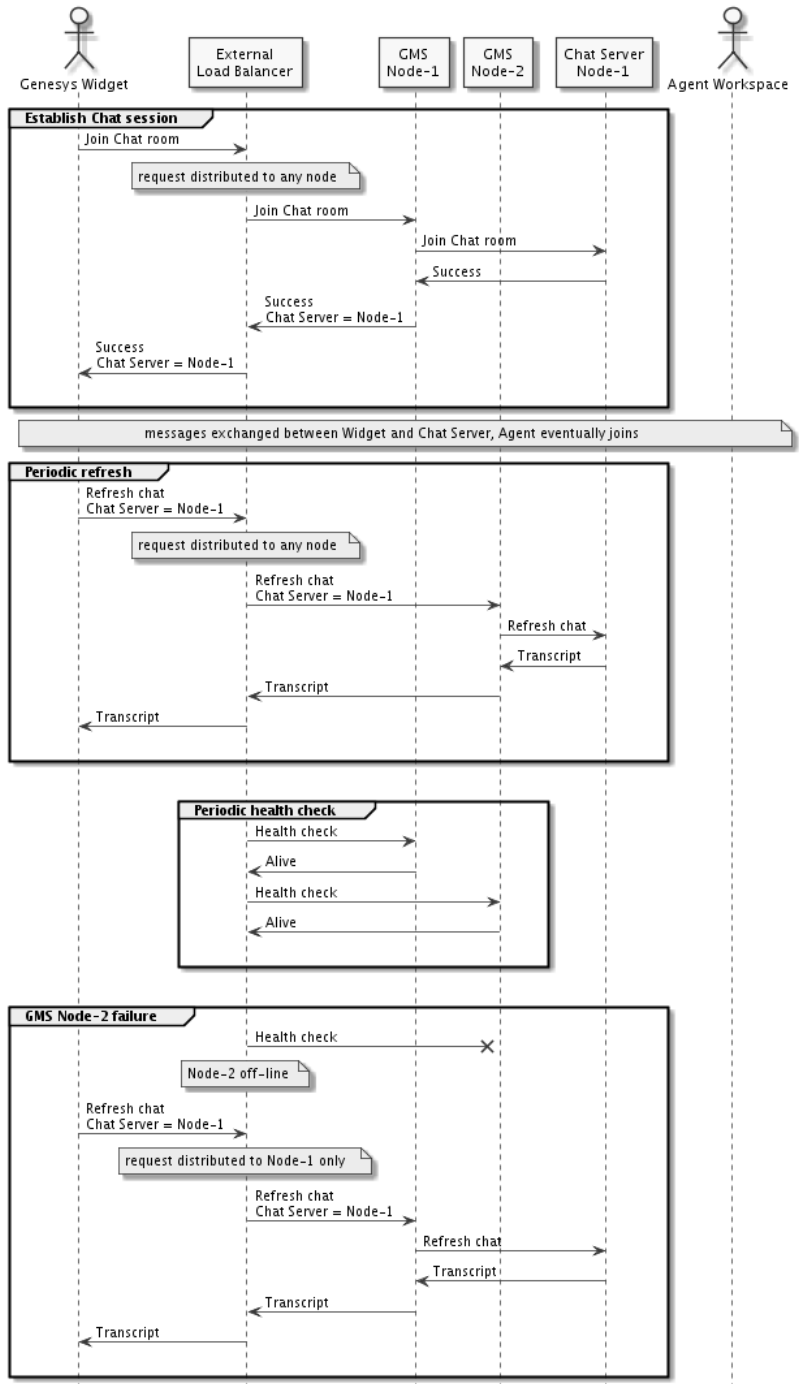


4.2.1.2 GMS Node failure

1.1.1.1 Existing live sessions

The load balancer is responsible to detect GMS node failures based upon the [Node API](#). Once detected, new HTTP requests from the Genesys Widget will be forwarded to the surviving GMS node. Currently the Genesys Widget refreshes the session every 3 seconds, therefore the chat session would remain idle for similar order of magnitude (depending upon the load balancer health check timeout). The session is fully recovered on the surviving node and any message sent by the agent during failover is retrieved from the chat transcript

1.1.1.1.2 Sequence diagram



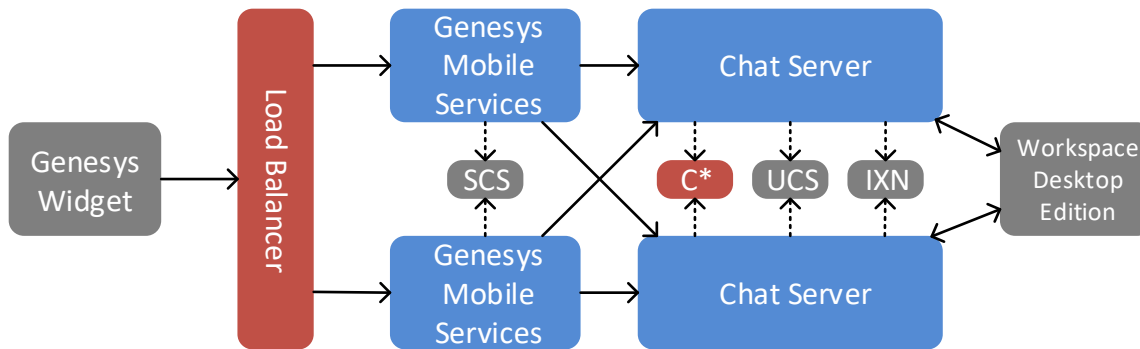
1.1.1.1.3 New sessions after failure

New Chat sessions started after a GMS node failure are properly handled by surviving GMS nodes, provided that the load balancer implements proper health check detection to ensure new sessions are only sent to available GMS nodes.

4.2.1.2.1 Chat Server

Chat Server can be deployed in an N+1 model as part of a “cluster”. To handle node failure Chat Server relies on either UCS or Cassandra to store the current state (transcript). Cassandra is an optional component that stores the intermediate transcript while UCS stores the final transcript. Every time the chat transcript changes this is persisted either the Cassandra cluster or UCS, along with sessionId. Each time a client requests a chat refresh the request includes the sessionId. If there is a failure of the current Chat Server any Chat Server can resume processing and use the sessionId to provide an up to date transcript back to client.

Following diagram includes all components involved in Chat Server node failure and recovery.



- Solution Control Server (SCS) is used by GMS to monitor the status of each Chat Server
- Cassandra and UCS are used to store temporary and final transcripts to persistent storage
- Interaction Server (IXN) will notify all chat parties about the session relocation by providing Attached UserData to Workspace with the new host:port information of the Chat Server owning the chat session

4.2.1.3 Chat Server Node failure

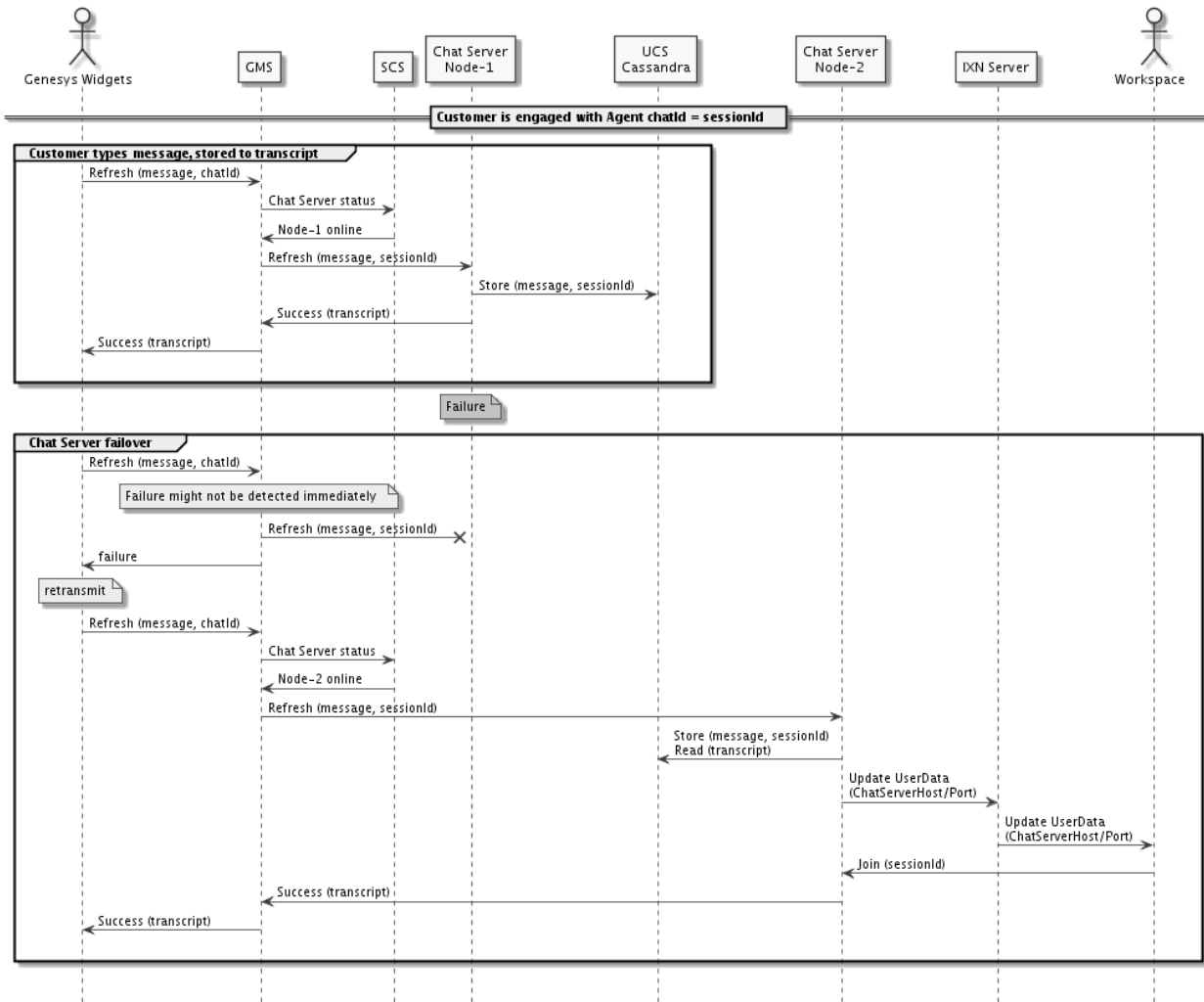
1.1.1.1.4 Existing live sessions

In case the Chat Server node fails:

1. When any client request is received the GMS node checks availability of the Chat Server. If a failure is detected the GMS node will attempt to contact other the Chat Server reported as online and after timeout will notify the client (Widget) with an error code, so that the client (Widget) can resend request
2. When the Chat Server receives the request (from GMS node) for an unknown (i.e. currently not serving by this instance) chat session the Chat Server starts the restoration procedure:
 - a. Reads the transcript from UCS or Cassandra,
 - b. Verifies that the requesting party has the right to participate in this chat session (including comparing secure key),
 - c. Updates Attached UserData of the interaction with the new Chat Server host/port and other information
 - d. Confirms (or rejects) the restoration of chat session with GMS node.

3. Workspace detects Chat Server failure and it will wait for an update of the corresponding Attached UserData from Interaction Server containing the new Chat Server host/port and then send a new Join Request to enter the chat room
4. The workflow (strategy) could encounter errors when sending ESP messages to the chat session. In this case the workflow may need to repeat the attempt to send the ESP message several times. Each time it must read "ChatServerAppName" from userdata and use it for sending the request.

1.1.1.1.5 Sequence diagram

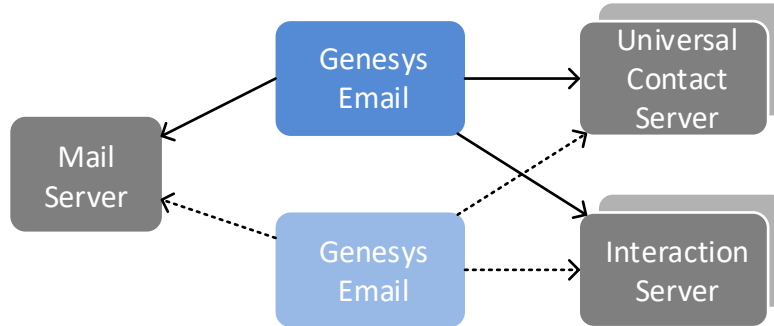


1.1.1.1.6 New sessions after failure

As described in the prior paragraphs, new Chat sessions started after a Chat Server node failure are handled properly by any surviving Chat Server node.

4.2.2 Email HA

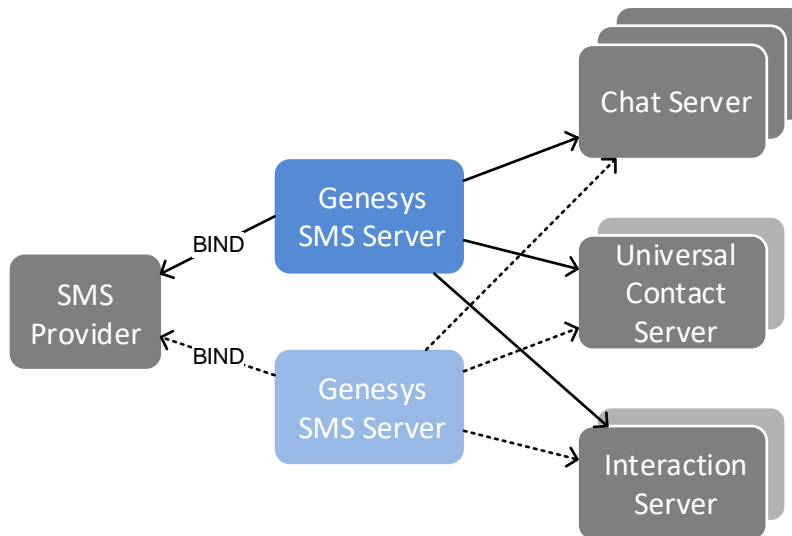
Genesys Email supports the Warm Standby HA model. Genesys Email server is considered a stateless application. At any given time only one Email server is running in Primary mode and retrieving emails from email server for a given mailbox. Genesys Email will then store the retrieved emails into the UCS database.



In case of a failure the backup instance of the Genesys Email server will become Primary and take over email handling for new and existing interactions.

4.2.3 SMS HA

Genesys SMS supports the Warm Standby HA model. At any given time only one SMS server is running in Primary mode and receiving SMS from an SMSC (Short Message Service Center) for a given range of numbers. The primary SMS will register as an External Short Message Entity (ESME) against the Short Message Service Center (SMSC) for specific number range. Genesys SMS will store the SMS content into UCS database, therefore Genesys SMS server is stateless.



In case of a failure the backup instance of the Genesys SMS Server will become primary. It will take over SMS handling for the given number range by registering as new ESME against the SMSC.

4.2.4 Co-browse HA

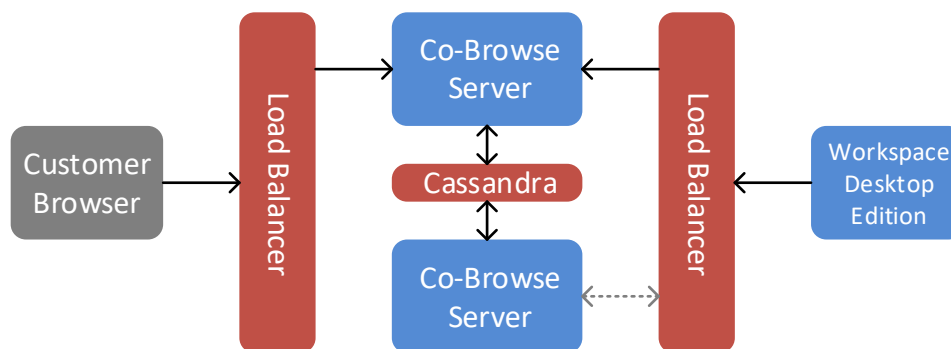
Genesys Co-browse (GCB) high availability and scalability is based on the N+1 model. Sticky sessions must be supported by all HTTP Load-Balancers which front the N+1 GCB nodes within the cluster so that load balancer can direct communication to the established co-browse session.

Currently, there is no fail-over support for co-browse sessions. If a Co-browse Server node becomes inaccessible the co-browse sessions hosted by this server will terminate.

Co-browse failover uses the following logic, assuming a Co-browse cluster of N+1 nodes:

1. Co-browse Session is established on Node-1
2. Node-1 becomes unavailable
3. The Load Balancer detects the Node-1 failure and reroutes all HTTP requests to Node-2
4. As Node-2 doesn't "own" the session, the session will end for both agent and consumer
5. On the customer side the Co-browse button will re-appear on the browser, therefore the customer can manually re-start new session

Proper HA behavior relies on the Load Balancer to detect the Co-browse Node failure (step 3) and for new Co-browse nodes to disconnect old sessions (step 4) allowing clients to manually re-start a new Co-browse session.



Appendix B provides detailed description on Load Balancing configuration for Co-browse.

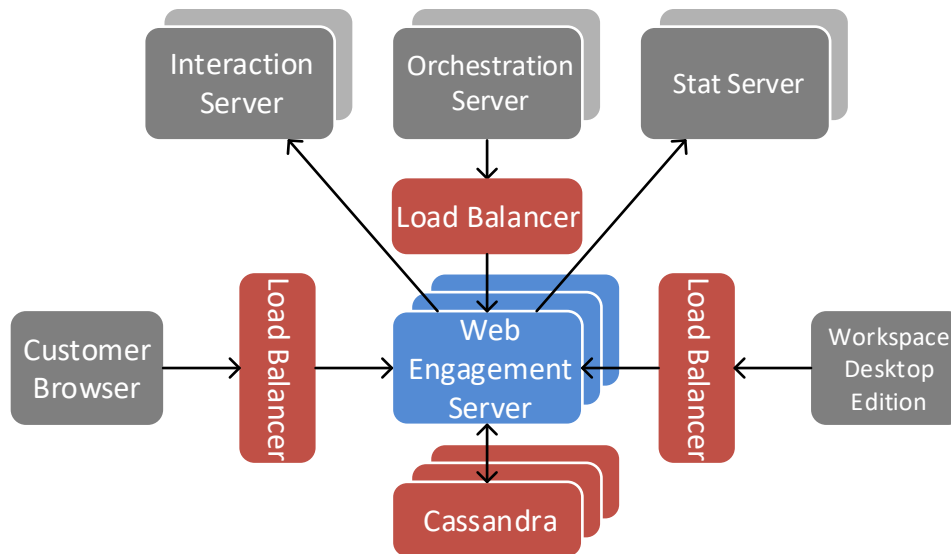
4.2.5 Web Engagement HA

Genesys Web Engagement (GWE) using a basic N+1 model to provide high availability and scalability. It requires sticky session (session persistence) to be supported by the HTTP Load-Balancer which manages the requests to the GWE nodes belonging to the cluster.

After the initial GWE monitoring session starts on a given node, each subsequent request from the customer browser is directed to this node. If a GWE node fails the Load Balancer (LB) can detect the node failure based on a periodic health check URL, and send subsequent HTTP requests to a surviving GWE node. The Monitoring and Chat sessions will continue on the surviving GWE node which will retrieve session data from Cassandra. Due to the specific transitions happening during failover there is a possibility that some events may not trigger the engagement strategy. Rules execution (based on CEP templates) are handled in memory by the GWE node, so in the event of failover, the CEP rules session on the failed node are lost and will be restarted on the surviving GWE node after the first new event for the visit is received from customer's browser.

Requests from Workspace to retrieve the customer's browsing history can use any GWE node. In the case of a GWE node failure the Load Balancer will simply direct new requests from Workspace to a surviving GWE node which can retrieve the latest customer browsing history from the common Cassandra cluster.

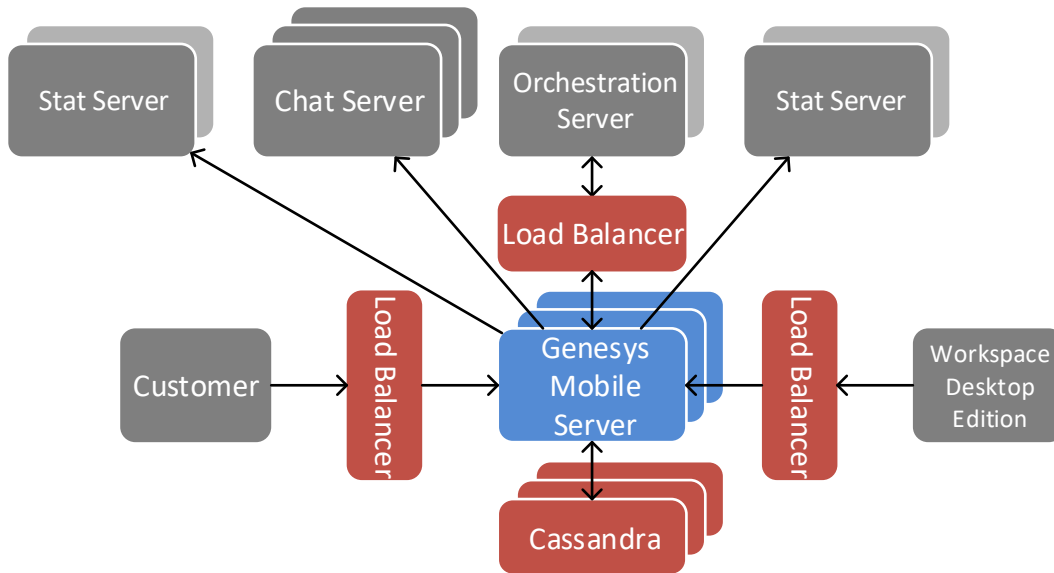
HTTP POST requests from ORS to start an Engagement will normally traverse the HTTP Load Balancer and may target any GWE node. This behavior is being improved with a future GWE release (8.5.000.22 not yet released at present time).



Appendix B provides detailed description on Load Balancing configuration for Web Engagement.

4.2.6 Mobile Engagement HA / Callback HA

Genesys Mobile Engagement (GME) and Genesys Callback use a basic N+1 model to provide high availability and scalability. An HTTP Load-Balancer is required to distribute load across the Genesys Mobile Server (GMS) nodes belonging to the cluster.



HTTP Requests from clients, including ORS, towards the GMS cluster are stateless. Any HTTP request can be directed to any GMS node in the cluster, as long as it belongs to same data-center. In the event of a GMS node failure any other node in the cluster can continue processing the ongoing session.

The one exception to this stateless behavior is Chat API V1 which uses CometD, and requires session stickiness. Chat API V1 is being obsoleted by Chat API V2, which is completely stateless protocol and would make failover nearly transparent for all GMS interfaces. For more details around Web Chat support with GMS review section 4.2.1

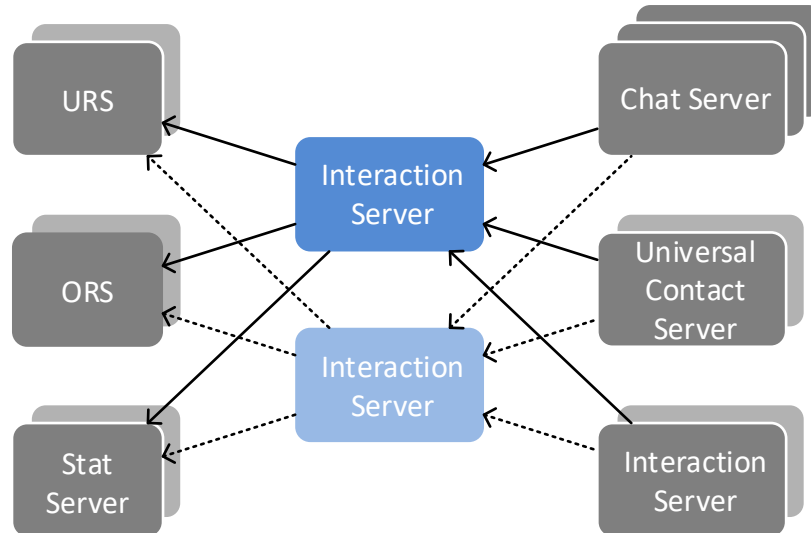
4.2.7 Interaction Server HA

Interaction Server supports the Warm Standby HA model. At any given time only one Interaction Server is running in Primary mode and processing interactions. During processing Interaction Server stores the interaction state and properties in the database. If the primary Interaction Server fails clients of Interaction Server will need to reconnect to the backup, which will have been promoted to primary. The state of interactions is recovered from the database. There are five types of clients which connect to Interaction Server:

- Agent applications – After an Interaction Server switchover the agent application will need to login the agent again and set agent state. If there were interactions present the application will need to pull them from Interaction Server.
- Media Servers – If an interaction was submitted by one of the media servers (email, chat, SMS, etc.) and the server received a confirmation from Interaction Server there is nothing required after switchover. The interaction will be persisted in the database and the new primary Interaction Server will process it accordingly.
- Routing engine – After switchover both URS and ORS should stop processing any interactions coming from the failed Interaction Server. After the new Interaction Server becomes primary it will automatically resubmit interactions for routing.
- ESP clients – Any ESP client that sent request before the switchover and didn't receive a response will need to resubmit the request. When a switchover occurs the ESP clients will see a

disconnection and will have to open a new connection to the Interaction Server which is primary.

- Reporting clients – Reporting clients are only reading events feeds from Interaction Server they only have to reconnect to new the primary Interaction Server and re-register for all relevant events.



4.2.8 Interaction Server proxy HA

Interaction Server Proxy supports the Warm Standby HA model. At any given time only one Interaction Server Proxy is running in Primary mode and processing interactions. In case of the primary Interaction Server Proxy failure the backup Interaction Server Proxy will be promoted to primary. Clients of Interaction Server need to implement the Genesys proprietary Interaction Management Protocol and reconnect to new primary Interaction Server Proxy.

4.2.9 Classification Server HA

Classification Server provides a stateless N+1 high availability model. Each client request can be distributed to a different Classification Server. In addition, if there is no response from a failed Classification Server the client can resubmit the same request and as long as there is another instance available it will process the request.

4.3 Dual Data Center Distribution

Many customers require a highly survivable architecture, which support geographic redundancy and will continue to provide service even in the event that a primary site become unavailable. The Common Components Blueprint provides information on establishing a foundation for dual data center distribution and resiliency.

At present the Genesys Digital solution does not have an overarching product approach to provide survivable service in the event of a site outage. There are specific component level recommendations. We recommend that Interaction Server proxies and UCS proxies should be deployed at each data center

and communication to the server processes should be funneled through the local proxies. The use of proxies helps provide consistency and simplify the management.

For other components within the Digital Solution possible approaches to provide a Dual Data Center distribution are:

1. **Split High Availability** – In this approach, primary/backup components such as Interaction Server are split between data center sites. If a split HA approach is used it requires alignment of HA with those components detailed in the Common Component Blueprint, such as Solution Control Server (SCS) which would also need to be split across between the data center sites.
2. **Cold Standby** – Instead of deploying active components at a secondary data center, you may elect to deploy a replica of the appropriate Digital components in the secondary data center in a cold-standby mode. This may consist of components which are deployed on separate servers but idle. Alternatively, customers may utilize VMWare or other technologies for survivability. Appropriate hardware would be available at the secondary data center and in the event of a data center failure the logical servers and snapshots would be replicated and restarted at the secondary site.

As neither architecture has been formally validated if there are requirement for active survivability of the Digital deployment you should work with the Product Management, Engineering and Solution Architecture teams.

In the past customers had tolerated cold-standby for many Digital components in the event of a data center loss. Some channels such as email are not real-time and therefore if there is latency of minutes or a few hours in processing an email it may be acceptable for the business. Other channel such as chat are real-time and as Digital has become critical for many businesses they are no longer willing to tolerate architectures which do not provide for active geographic redundancy. In the event that a service is unavailable the clients such as web pages which will offer chat, co-browse or mobile applications should be designed to handle any gracefully handle any errors including an impact to service availability.

One of the primary constraints in establishing a dual data center architecture which can provide active disaster recovery is data replication. The databases used by the Digital Solution such as the Interaction database and UCS database much be configured with transaction replications between sites to provide disaster recovery and eliminate loss of interactions of interaction history.

4.4 Database Configuration

The RDBMS is a customer provided component of the solution and must be provided as part of the solution. Genesys-specific databases need to be setup within the database system and made accessible to the Genesys components. Follow the installation guides specific for each product and database vendor. Note that appropriate language/character sets need to be configured for some product databases.

Each database vendor has various strategies for providing high availability for their database system and customers may have their own setup which needs to be adhered to. Genesys should always be communicating with a single logical database.

To ensure site survivability the databases must be replicated to an alternative location such as the secondary data center. Business continuity is typically accomplished via some form of replication or clustering of databases. Genesys recommends either Microsoft SQL 2012 or Oracle. There may be

specific requirements or features utilized such as Oracle Golden Gate or Microsoft SQL AlwaysOn Clustering to provide business continuity.

For NoSQL databases such as Cassandra customers should configure the solution to provide appropriate availability and redundancy. At a minimum 3 Cassandra nodes should be configured for redundancy. Customers may configure additional data replication and/or additional nodes based upon their desired level of redundancy and survivability.

The specific configuration such as transaction replication, batch replication, etc. will also define what is possible for the Recovery Point Objective (RPO) and Recovery Time Objective (RTO).

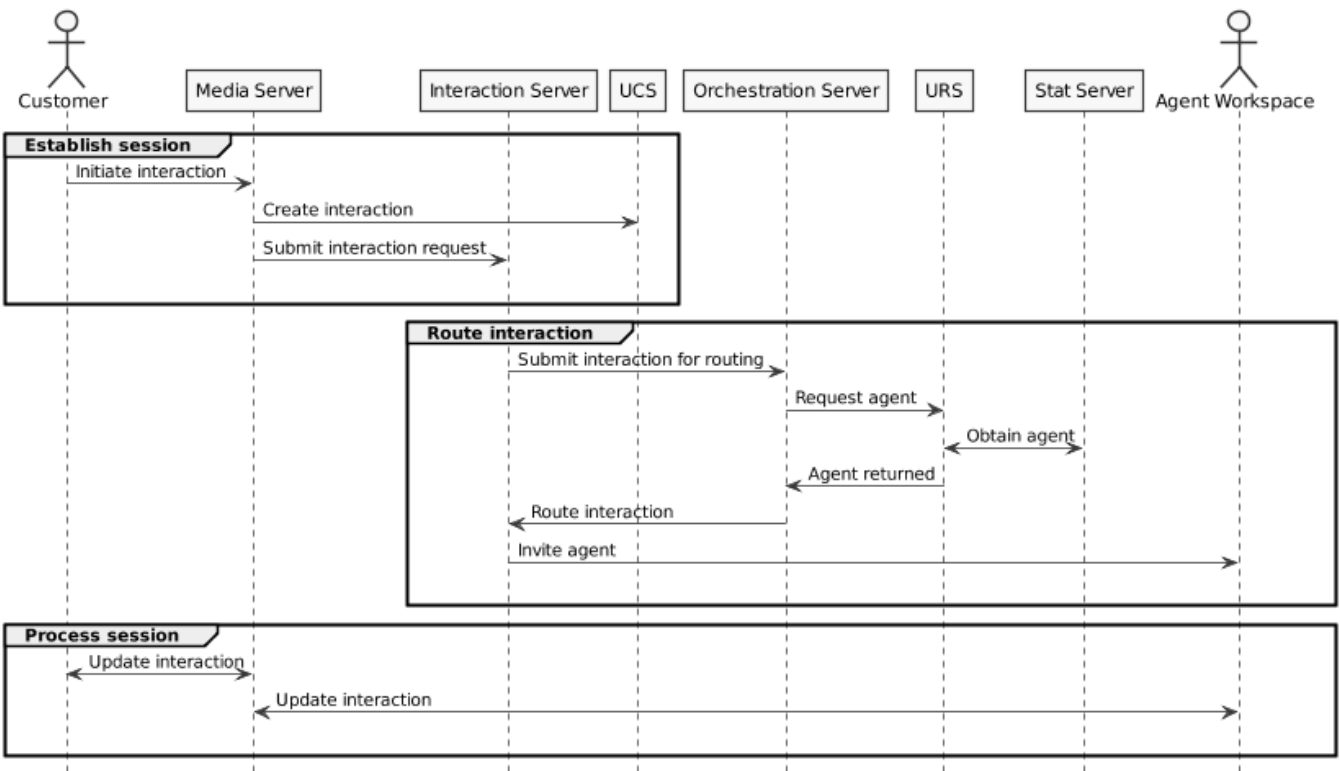
5 Interaction View

5.1 Interaction Flows

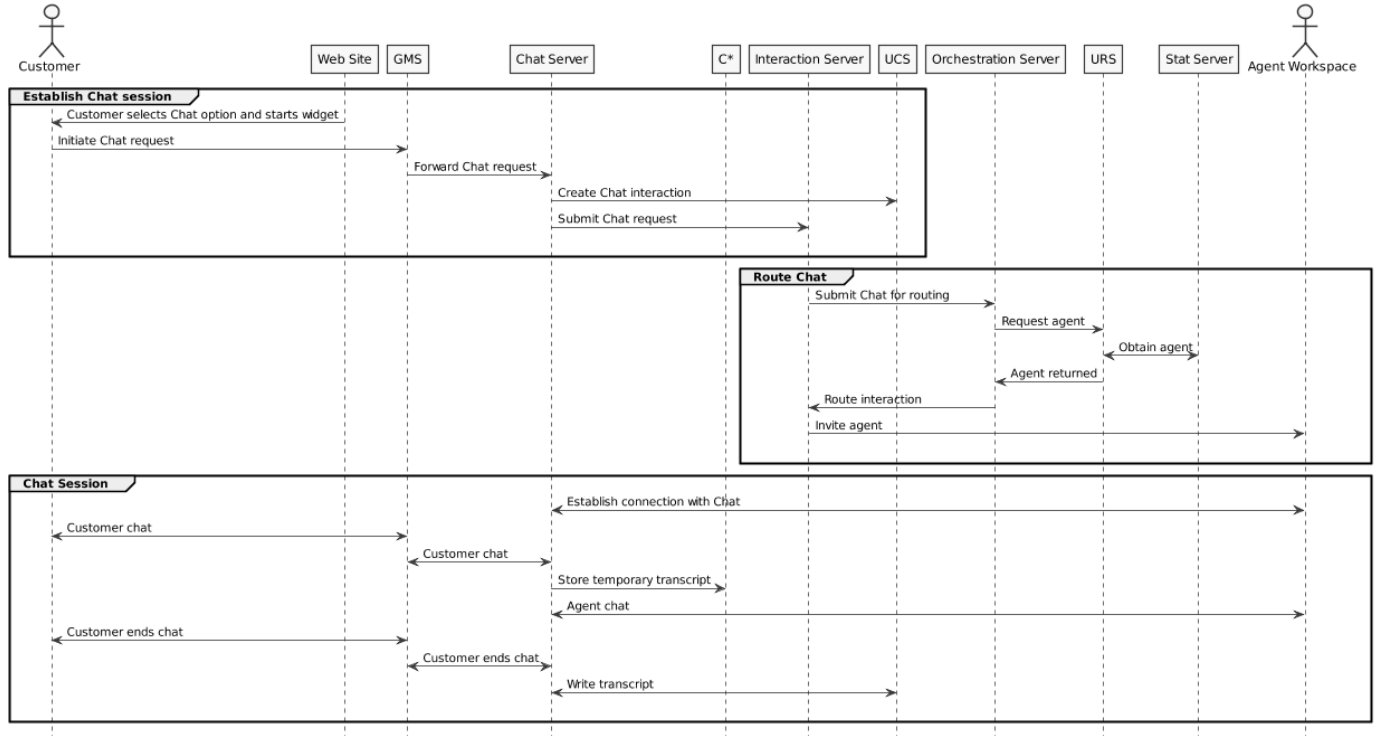
The following section documents the typical interactions between components in the Digital Blueprint. The objective of these flows is to clarify the communication between components to better illustrate how the various components work together in the overall solution.

The interaction flows have been simplified as certain messages and notifications have been removed for clarity.

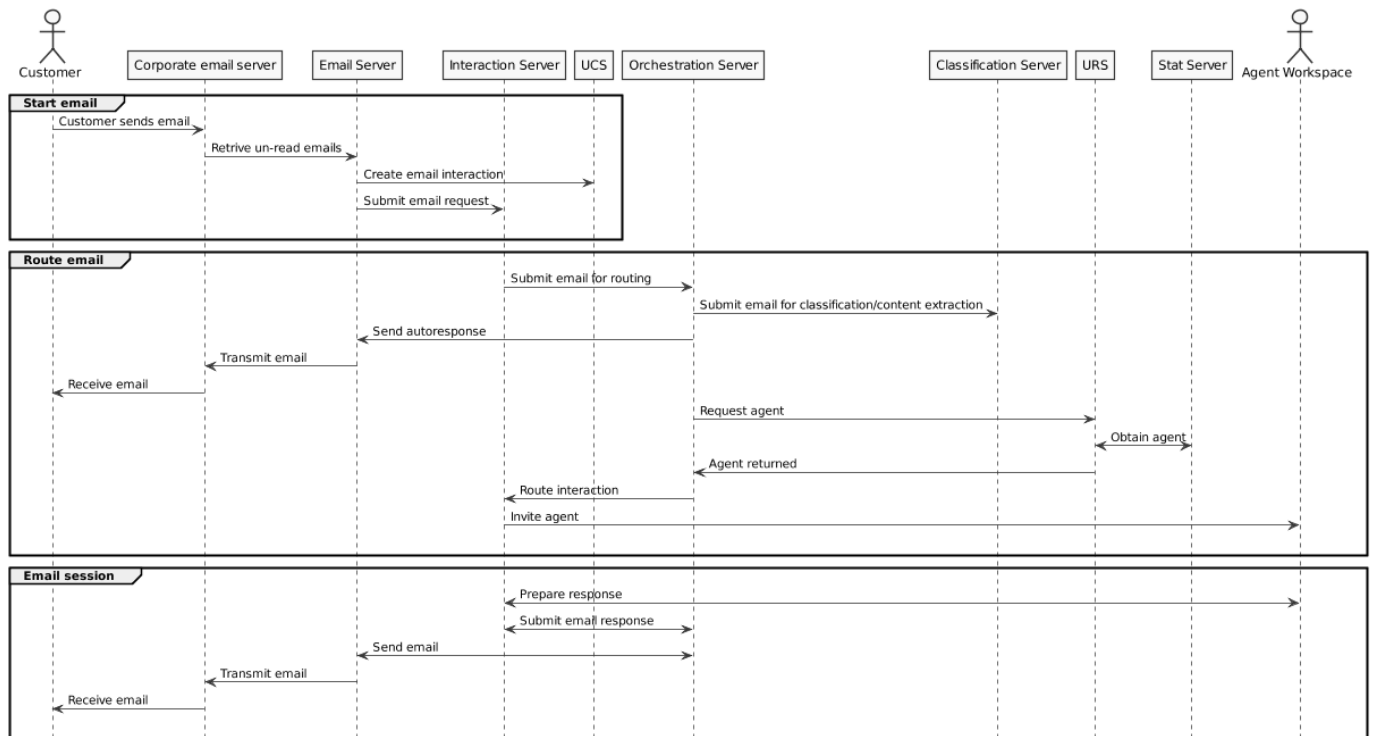
5.1.1 Generic flow for interaction handled by Interaction Server



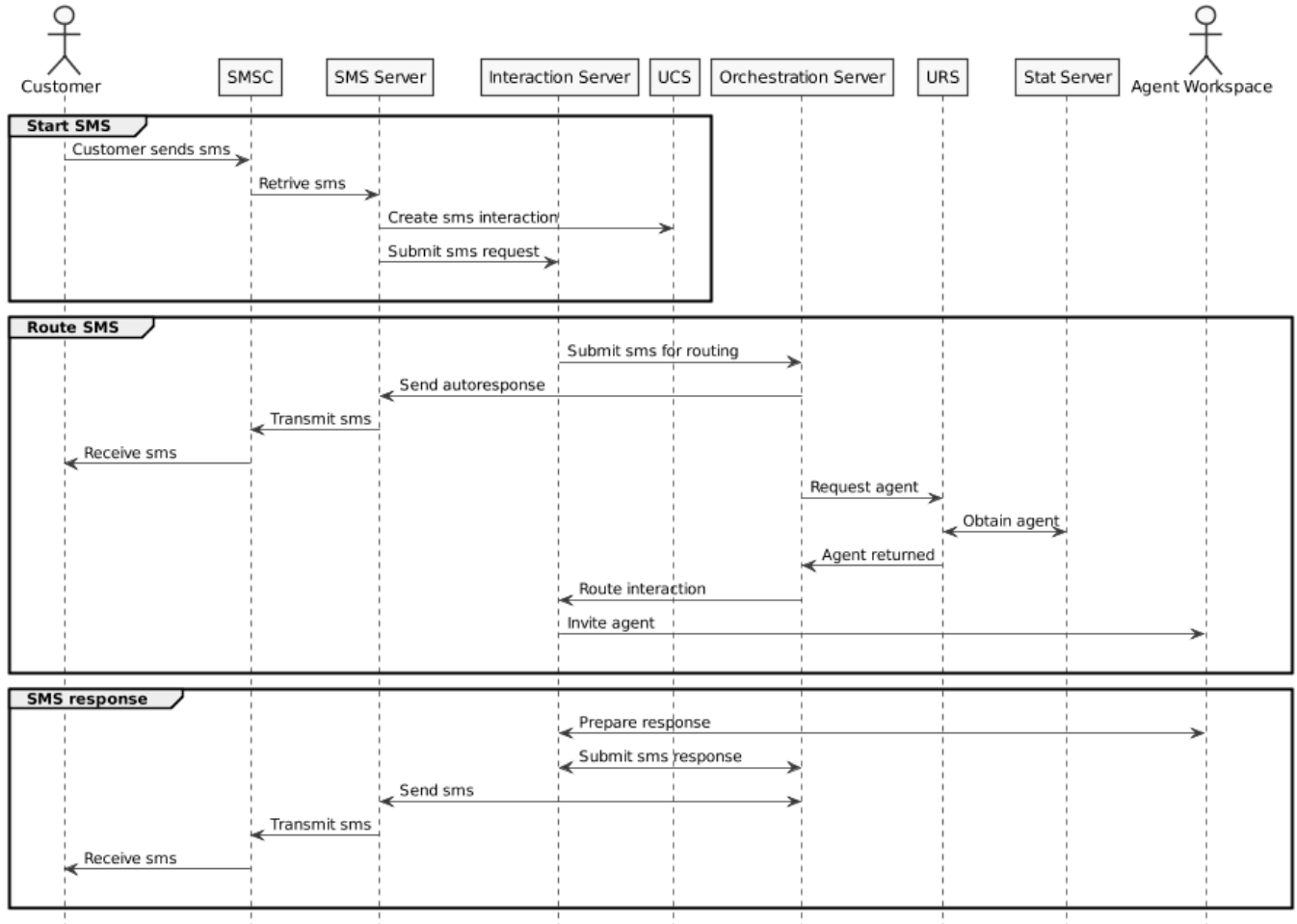
5.1.2 Reactive chat flow



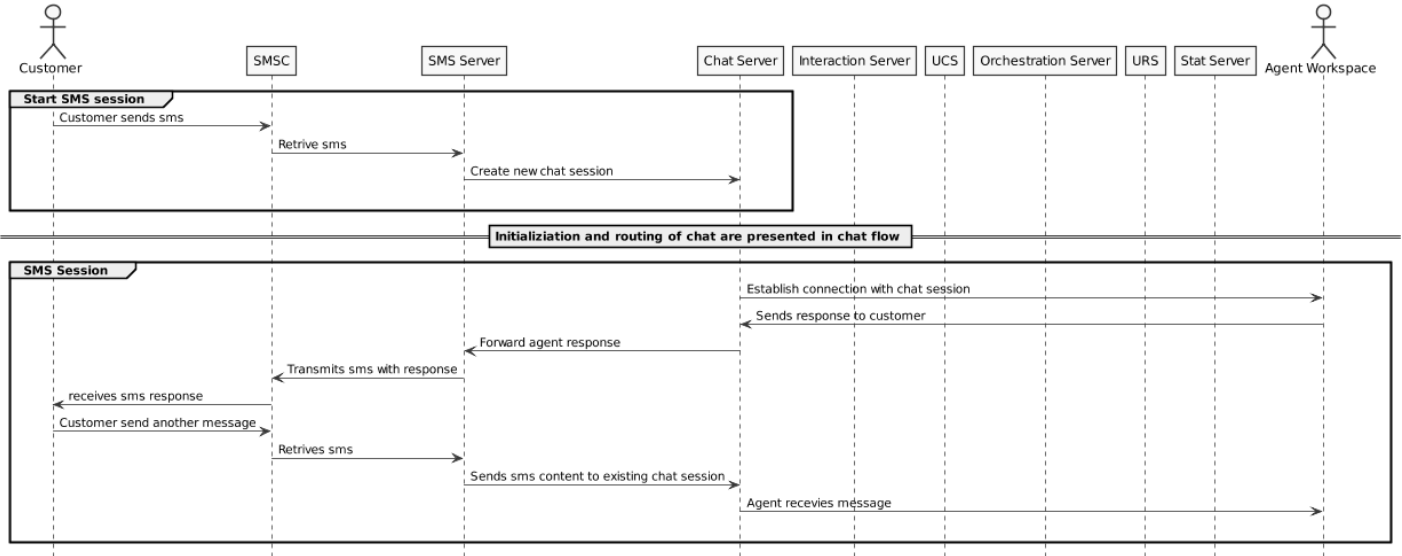
5.1.3 Email flow



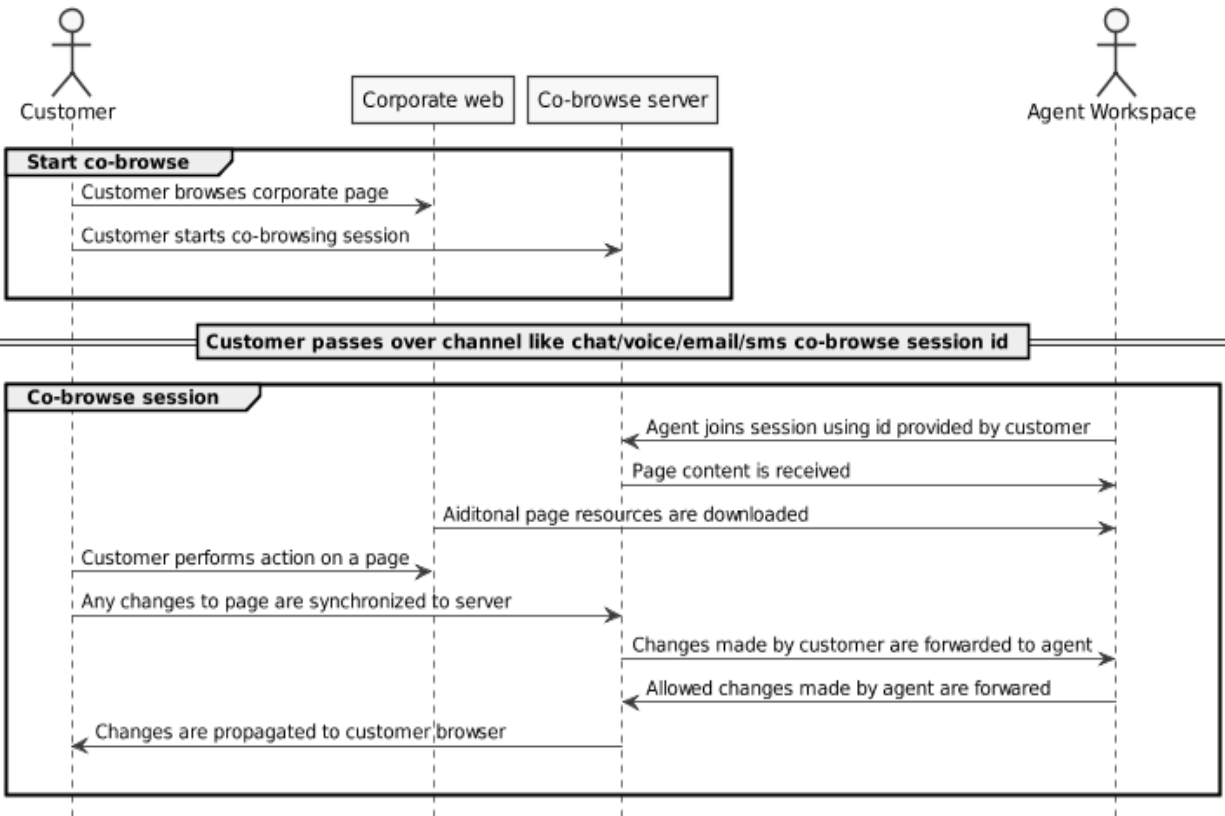
5.1.4 SMS page flow



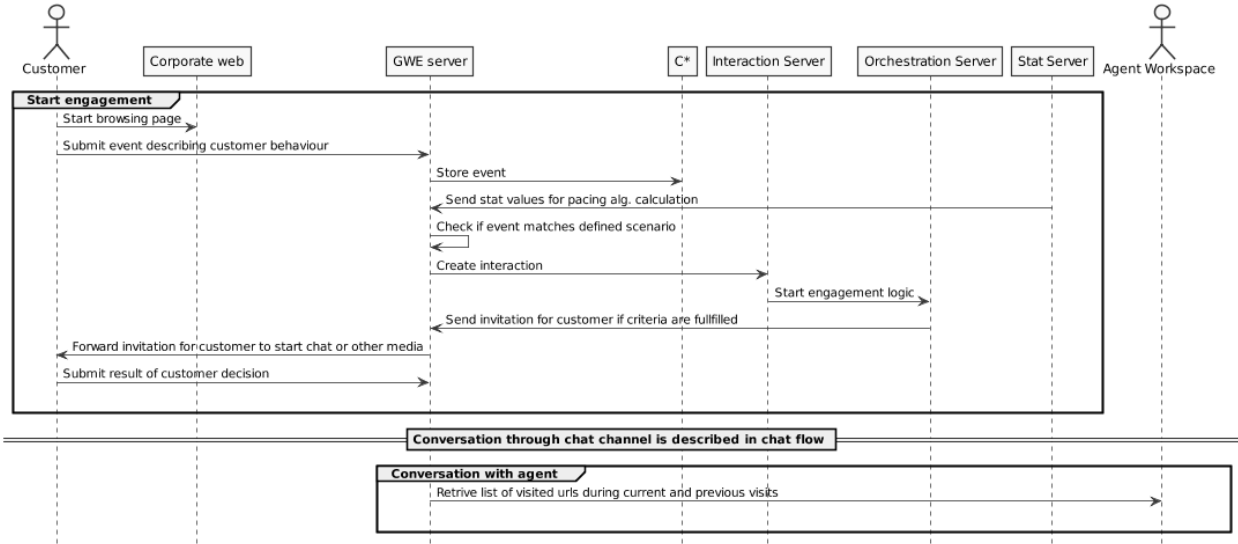
5.1.5 SMS session flow



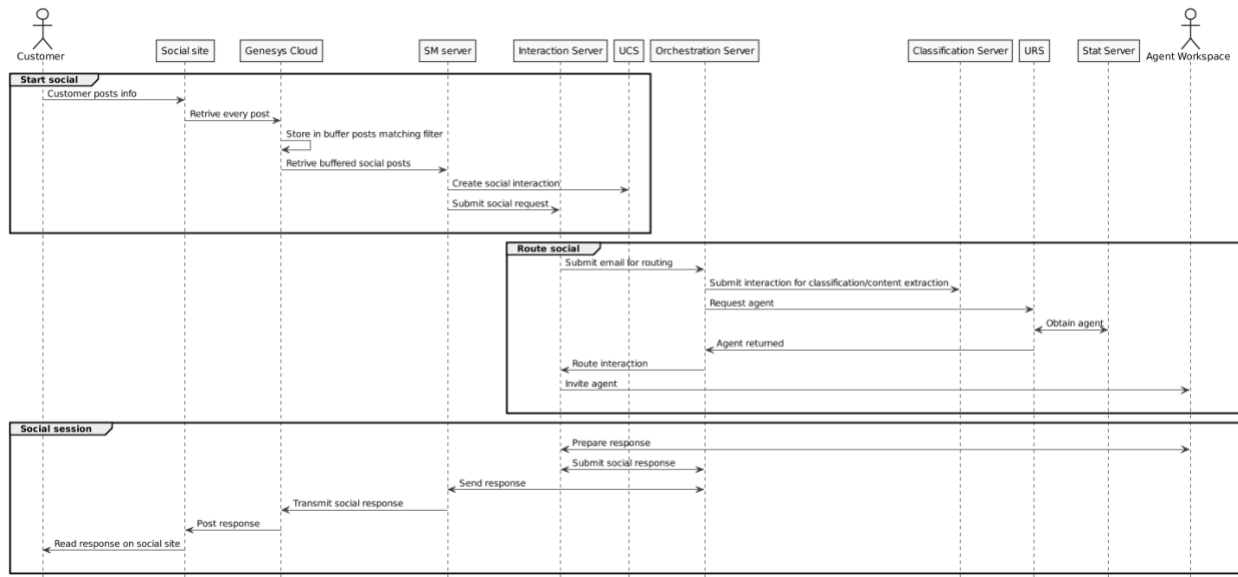
5.1.6 Co-browse flow



5.1.7 Web Engagement flow



5.1.8 Social Engagement flow



5.2 Network Considerations

The Digital Solution and channels it supports does not require real-time communication or use real-time communication protocols therefore there are no special network requirements such as network latency or jitter between the customer and Genesys components or between Genesys components and agents. Bandwidth requirements are also very minimal as the communication is primarily text based.

There are some network dependencies based upon the solution and technologies utilized. For example special consideration must be taken when if you will deploy Genesys Web Engagement and intend to

span Cassandra across multiple data centers, specifically reviewing network bandwidth, latency and firewalls.

For the Digital Solution the most critical network consideration is the proper network security design, which is described in detail in chapter 6.3.4, and proper load balancer design which is detailed in Appendix B.

5.3 External Interfaces

This section describes the external interfaces used within the solution. These become the integration points between solution components and the elements in the customers’ premise.

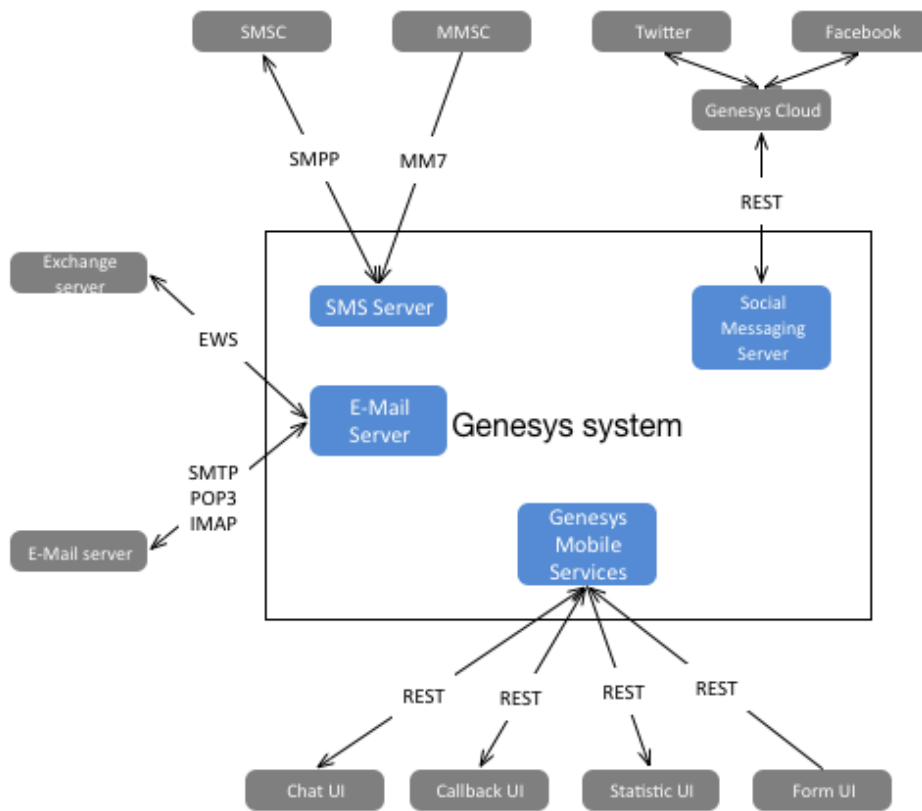


Figure 11 – Digital External Interfaces

The following table details each of the external interfaces, its protocols, the components within the solution that are impacted or connected to these external interfaces and lists the integration tasks required to setup the external interfaces.

Interface	Protocol	Solution Components	Integration Tasks	Description
Email system	POP3	Email server	Retrieve email from external system	Interface used to retrieve emails from corporate email servers. This interface doesn't allow to copies of emails to be left on the email server.
Email system	IMAP	Email server	Retrieve email from external system	Interface used to retrieve emails from corporate email servers. This is used instead of POP3 when the customer requires Genesys to leave the email on the server.
Email system	SMTP	Email server	Send email through external system	Interface used to send emails through the corporate email server. It is possible to configure only one SMTP account per email server With Exchange SMTP does not leave a copy of the message on server and if this is required EWS should be used.
Exchange server	EWS (Exchange Web Services)	Email server	Retrieve email from external system Send email through external system	Web services interface used with Microsoft Exchange servers to retrieve and send emails. When using the EWS protocol to retrieve after retrieving the email the email server will delete it without leaving copy on Exchange server. When emails are sent through EWS the Exchange server will allow copies of each sent email to be stored on the server.
Customer facing chat UI	REST(over http(s))	GMS	Send chat message to agent Receive chat message from agent/contact center Send notification Receive notification	Interface used to build chat UI element. It allows the customer to send and receive chat messages with the agent. It can be used to push a URL and in future versions files. It provides notification about agent/customer typing. This interface can also be used to send commands between agent application and chat UI element – for example to join a specific co-browse session
Customer facing form UI	REST(over https(s))	GMS	Submit form with message	Interface is used to submit a message and fields as a form. This message and form is translated to a Genesys email interaction and the message and fields containing additional information are used for routing.
Customer facing statistic UI	REST(over http(s))	GMS	Retrieve statistic value	Interface is used to query selected object and statistics and display it to the customer

Interface	Protocol	Solution Components	Integration Tasks	Description
Customer facing callback UI	REST(over http(s))	GMS	Submit callback request	Interface is used to submit callback requests. It can also be used to query for exiting callbacks or query for available slots before submitting new callback request.
SMSC	SMPP	SMS Server	Send SMS Receive SMS	Interface used to connect to a telecom operator to retrieve and send SMS messages. More details can be found at: https://docs.genesys.com/Documentation/ES/8.5.1/Admin/AdmSMS
MMSC	MM7	SMS Server	Receive MMS	Interface used to retrieve MMS messages
Genesys cloud	REST (over http(s))	Social Media Server	Receive social interactions captured by Genesys cloud Send response to social interactions capture by Genesys cloud	Interface used to send and receive interactions captured by Genesys cloud for supported social media providers. Currently Genesys cloud supports Facebook and Twitter. The configuration defining what is captured by Genesys Cloud is managed through web UI.

Table 7 - External Interfaces

5.4 Operational Management

Once a Genesys solution is in place, managing the solution becomes a primary concern of the customer. There are two approaches to operational management that need to be considered for the solution.

1. If Genesys components are the main focus of the operation, then using Genesys Administrator and Genesys Administrator Extension become the primary mechanism for administering the solution.
2. If Genesys is part of a larger operation, then integration into the customer's operational management tool is recommended.

In both cases, Genesys Administration and Genesys Administrator Extension needs to be installed and configured to manage the solution.

5.4.1 Network Management Systems

If the customer has a Network Management System (NMS), then Genesys components need to be integrated into the customer's NMS. This is typically done by setting up Net-SNMP to send SNMP events and info to their NMS.

Examples of supported NMS include Zabbix, HP OpenView and OpenNMS (an open source NMS - <http://www.opennms.org/>).

In addition to the monitoring and alarming performed directly by Genesys the following recommendations should be considered:

- Monitor JVM status, especially memory usage. Note that a regular saw-tooth pattern should be observed due to Java garbage collection.
- Set alarms for specific disk and cpu thresholds
- Incorporate additional SNMP traps

It may also be beneficial to use ELK (ElasticSearch, LogStash & Kibana) or Splunk to harvest logs and build alarming for specific conditions within the logs.

Genesys recommends that customers also establish monitoring strategies for 3rd party components such both SQL and NoSQL. Detailed guidance should be provided by your DBA or others who are responsible for the management of these technologies in your environment.

5.4.2 Serviceability

Serviceability relates to the ability of technical support to identify issues and defects within the system. Many customers or partners will perform initial triage and analysis to determine whether Genesys Care should be engaged. If Genesys Care needs to be engaged it is critical to retrieve the required logs and configuration information and pass this information back to Genesys Care. The following recommendations provide guidance on improving serviceability which can accelerate issues analysis and resolution.

Logging

Setting up logical logging locations is a best practice that makes it easier to collect logs and reduce the time to send logs to support. Configuring 3rd party components to log into the same location is ideal as well. Genesys recommends to setup a “log” directory on a separate partition from the Operating System and applications:

```
D:\GCTI\log  
/log
```

Many problems can occur when trying to retrieve the log files necessary for troubleshooting. Common problems include:

- The log files for the time when the problem occurred have been overwritten or otherwise lost.
- Log files delivered are not within the event time frame.
- Log files provided were created with log levels not detailed enough for the investigation.
- The set of log files provided is inaccurate or incomplete.

The Genesys Log File Management Tool (LFMT) is an intelligent, configurable log collection utility developed by Genesys Customer Care intended to minimize these issues, and thereby reduce the time required to resolve customer problems. It is recommended to include LFMT as a standard part of every deployment.

Log Analysis

To assist customers with performing log analysis of messaging Genesys recommends a network diagram should be maintained by customers and kept up to date help with analysis. It is recommended to have this information readily available and, if possible, provide it to technical support together with the initial problem description and logs, to help reduce overall resolution time.

Future Tools

In 2H 2016 Genesys will be releasing the Genesys Care Workbench which is a suite of troubleshooting tools that can help you quickly and easily identify and resolve issues in your Genesys environment. Workbench collects data from multiple sources, analyzes it, and displays aggregate data and important data correlations in its Current and Historical dashboards as well as some specialized consoles.

Types of information displayed on the Workbench Dashboard include:

- *Configuration Server changes – Workbench monitors Configuration Server events for all Application objects, and displays recent configuration changes in the environment*
- *Alarms – Workbench configures a default set of alarms in Solution Control Server and displays alarms when thresholds are triggered. If you subscribe to Remote Alarm Monitoring, additional alarms may be displayed.*
- *Log events – If Log File Management Tool is deployed, Workbench can monitors log files from supported Genesys applications and display important events for troubleshooting.*

Once Genesys Care Workbench is released it is recommended that it is included as a standard part of any deployment.

Proactive Monitoring

Genesys can provide proactive monitoring services which delivers the most complete servicing of a customer's environment. Genesys has the ability to perform proactive monitoring through our Premium Care offering. For details on Premium Care options consult the Genesys Account Team and Genesys Customer Care.

5.4.3 Monitoring Details

The following provides details on recommended monitoring:

- Numerous SNMP traps can be provided by the Digital components. Ensure that these traps are properly configured.
- The Common Components Blueprint includes details on recommended UCS monitoring which should be implemented.

6 Implementation View

The Implementation View describes details such as sizing, security and configuration of the solution based on the previous deployment and interaction views.

6.1 Solution Sizing Guidelines

This section provides guidelines on sizing the solution components to determine the server requirements. Providing a simple and accurate sizing guideline is difficult as there are many variables beyond the number of agents. The variables which impact the sizing include the mix of media channels, peak interaction volume, interaction duration, routing strategy complexity, etc.

The approach taken is to assume certain variables such as busy hour interaction volumes, defined AHT and interaction flows for each media, and a specific traffic mix. Based on these assumptions, the CPU, memory and data requirements are calculated based on the deployment target of 1,000 agents. The sizing accounts for a centralized deployment.

The sizing assumptions and architecture provided below are from the **Integrated Sizing Calculator**. The sizing assumes use of virtualization and includes the specific performance requirements for the underlying hardware (CPU profile) are also specified. *Please treat this sizing estimate as a rule of thumb. Changes to any variable can impact the overall sizing.*

Media Considerations

Often customers will look for a solution where the agents can be blended and support any media type. Different media exert very different loads on the components in the solution, e.g. Genesys Email vs. Genesys Co-browse. When sizing the system it is important to understand and document the traffic mix which needs to be supported because two identically licensed deployment may have significantly different architecture requirements based upon the planned traffic which needs to be supported.

6.1.1 Solution Sizing

This section provides the sizing for a centralized deployment supporting 1,000 digital only agents.

The following assumptions are made regarding the sizing of this solution.

Input Assumptions	
Agents	1000
Chat	
• Peaks chats per hour	3,500
• Concurrent chats per agent	2
• Agent Utilization	80%
• Chat Duration	400 sec
• Messages per chat session	10

• Chat message size	0.1 KB
Email	
• Emails per day	20,000
• Agent Utilization	80%
• Email Duration	400 sec
• Email Size	25 KB
SMS	
• Peaks SMS per hour	350
• SMS Handling Time	60
Web Engagement	
• Peaks chats per hour	1,800
• Concurrent chats per agent	1
• Agent Utilization	80%
• Chat Duration	400 sec
• Messages per chat session	10
• Chat message size	0.1 KB
• Web visits per second	2
• Page views per visit	10
• Events per page	2
• Categories per page	1
Mobile Chat	
• Peaks chats per hour	700
• Concurrent chats per agent	1
• Agent Utilization	80%
• Chat Duration	100 sec
• Messages per chat session	4
• Chat message size	0.1 KB

Co-browse	
• Peak Co-browse per hour	2,800
• Sessions per agent	1
• Agent Utilization	80%
• Handling time	400 sec
Additional Settings	
Log retention	Debug 2 weeks
Reporting History	2 years
Non-aggregated Reporting History	3 months

Table 8 – Sizing Inputs

6.1.1.1 Hardware/Virtualization Assumption

The underlying hardware used will impact the overall performance of any virtualization solution. For sizing the following hardware requirements are assumed:

- CPU Score per Core = 30
- Hyper-threading = Off
- Number of Chips (NUMA Nodes) = 2

In addition, it is assumed that each hardware server will be running ESXi v5.4 or greater.

6.1.1.2 Virtual Machine Sizing

The following table details the virtual machine sizing for CPU, RAM and disk. For further details please see the accompanying **Integrated Sizing Calculator**.

Note: The Integrated Sizing Calculator only provides sizing for:

- *Chat*
- *Email*
- *Callback (Genesys Mobile Services)*

The following products in the Digital Blueprint are included in the Sizing Calculator however the calculator does not have sizing data therefore the outputs are placeholder and should be refined by consultation with Product Management and Engineering

- *SMS*
- *Co-browse*
- *Web Engagement (Proactive Engagement)*

Lastly the following products are not yet included in the Sizing Calculator:

- Social

Enhancements to the Integrated Sizing Calculator are currently planned to bring this tool into alignment with the Digital Blueprint. Until the Integrated Sizing Calculator is complete please work with Product Management and Engineering for any sizing on these components.

The sizing calculator indicates that 22 VMs are required for the components which it covers. The VMs which are specifically covered by the Digital Blueprint are highlighted below in yellow.

VM Name	vCores	Memory, GB	Components
vm_common_fw	4	8	common_cfg
			common_msg
			common_scs
			VM Total
vm_common_prx	4	8	common_ss-cl
			common_cfg-prx
			digital_ixn-prx
			common_ucs-prx
			VM Total
vm_digital_ixn_icon	4	4	digital_ixn
			digital_icon
			VM Total
vm_digital_stat_router	4	4	digital_ss
			digital_urs
			VM Total
vm_digital_web	4	4	digital_web
			VM Total
vm_digital_chat_email_sms_class	4	4	digital_chat
			digital_email
			digital_sms
			digital_class
			VM Total
vm_common_fw-b	4	8	common_cfg-b
			common_msg-b
			common_scs-b
			VM Total
vm_common_prx	4	8	common_ss-cl
			common_cfg-prx
			digital_ixn-prx
			common_ucs-prx
			VM Total

vm_digital_ixn_icon-b	4	4	digital_ixn-b
			digital_icon-b
			VM Total
vm_digital_stat_router-b	4	4	digital_ss-b
			digital_urs-b
			VM Total
vm_digital_web	4	4	digital_web
			VM Total
vm_common_db	8	8	common_db
			common_db-cl
			common_sql
			VM Total
vm_digital_ors_cas	4	4	digital_ors
			digital_cas
			VM Total
vm_digital_ucs	4	4	digital_ucs
			VM Total
vm_digital_db_ixn_ucs	4	4	digital_sql-ucs
			digital_sql-ixn
			VM Total
vm_common_gim	4	8	common_gim
			common_java
			VM Total
vm_common_prx	4	8	common_ss-cl
			common_cfg-prx
			digital_ixn-prx
			VM Total
vm_digital_ors_cas-b	4	4	digital_ors-b
			digital_cas
			VM Total
vm_digital_ucs-b	4	4	digital_ucs-b
			VM Total
vm_digital_chat_email_sms_class	4	4	digital_chat
			digital_email
			digital_sms
			digital_class
			VM Total
			digital_gwe
			digital_gms

vm_digital_gwe_gms_gcb	4	8	digital_gcb
			VM Total
			digital_gwe
			digital_gms
			digital_gcb
vm_digital_gwe_gms_gcb	4	8	VM Total

Table 9 – Server and Component Distribution

6.1.2 Database Sizing

The following table summarizes the database sizing requirements for Genesys components stored within the RDBMS. These are estimates based on the sizing assumptions and should be treated as a starting point. Other customer factors can impact the overall data requirements.

Awaiting updated Sizing Calculator

System	1000 agents
<i>Configuration Server</i>	_ MB
<i>Message Server</i>	_ MB
<i>Genesys Administrator Extension</i>	_ MB
<i>Universal Contact Server</i>	_ MB
<i>Interaction Server</i>	_ MB
<i>Outbound Contact Server</i>	_ MB
<i>Interaction Concentrator</i>	_ MB
<i>Genesys Info Mart</i>	_ MB
<i>Genesys Interactive Insights (GI2)</i>	Insignificant
<i>Pulse</i>	TBD
Total SQL database storage	__ GB

Table 10 – Database Sizing

6.1.3 Network Sizing and Readiness

The success of the Digital Blueprint deployment hinges on ensuring that the network is ready and has the appropriate bandwidth.

As guidance, the following network load has been calculated for the solution.

Awaiting updated Network Bandwidth Sizing Calculator

Network Traffic	1000 Agents
------------------------	--------------------

Within Data Center	
Between Data Centers (Business Continuity)	
Between Branches and Data Centers	N/A

Table 11- Network Sizing Guidance

6.2 Configuration Guidelines

The following provides a high-level recommendations for configurations within the Digital Blueprint deployment based upon Genesys Engineering and Professional Services experience

- Interaction Server Proxy and UCS Proxy are recommended for large architectures to handle desktop and Stat Server load and reduce the impact on Interaction Server and UCS
- It is recommended that E-Mail Server Java should be configured to use IMAP instead of POP3 due to the improved handling of large messages. Note that using Exchange Web Services is another valid option if Exchange is the target corporate e-mail server.
- Review the Apache configuration parameters to ensure that the number of available thread handles is configured correctly to support the planned load. If Apache is not configured properly it can run out of thread handles at high load.
- Databases should be kept co-resident with the primary processes and it is recommended that HA pairs for servers reside within the same data center. If DR is required it should be accomplished with a cold standby environment at the DR site.
- When Genesys Mobile Services is used for Chat the GMS server will communicate with SCS to check the available Chat servers. As GMS is commonly deployed in a DMZ the firewall must be configured to allow connections from GMS to SCS.

Interaction Server

- Interaction Server uses a direct ODBC connection to ensure the most efficient connection when accessing the database (<https://docs.genesys.com/Documentation/ES/8.5.1/Depl/ODBC>). The database can also be partitioned if required to improve performance (<https://docs.genesys.com/Documentation/ES/8.5.1/Admin/IxnDBGen>).
- Interaction Server can also be configured with an Event Logger to store detailed information on interaction processes and event messages in a database. While the Event Logger is considered “optional” it may be required by specific products. The use of Event Logger can have a significant impact on the Interaction Server load and capacity. It is important to properly configure the Event Logger based upon the requirements. Interaction Server can be configured to store all events or a selected subset of events. Additionally it may be configured to store reporting events in multiple destinations.

6.3 Security

Protecting the customer’s infrastructure should be imperative for any solution deployment. Genesys components can typically be deployed in a secure manner. Many customers have their own security procedures that our solution needs to conform to. The Genesys Security Deployment Guide provides details on security features offered by Genesys software and how these features are configured. This document can be accessed at: <https://docs.genesys.com/Documentation/System/8.5.x/SDG/Welcome>

The following are guidelines for some of the requirements that may be encountered or should be recommended.

6.3.1 Secure Connections

Connections between components, especially those external to the solution (see 5.3 External Interfaces) should be secured. Secure connections are typically performed using SSL or HTTPS.

While secure connections can have a performance impact and addition operational considerations at a minimum Genesys recommends to secure any internal any communication which may contain sensitive data and any traffic external to the environment.

6.3.2 Data Security Considerations

The Digital Solution includes components that allow customers to define policies for masking sensitive data within interactions.

Rules for sensitive data are defined within the Privacy Manager Plug-in for GAX. These rules contain regular expressions which define the data to search for the policy defining how to replace each occurrence. These patterns are applied to different channels to remove sensitive information.

Privacy data is removed in two different ways depending on the channel:

- For chat the Chat server executes the rules which screen for sensitive information. It is possible to configure if sensitive information is removed only from transcript or also in real-time from conversation
- For channels other than Chat there are 2 new ESP services available which allow the strategy to remove sensitive information from data stored in UCS.

For more information please refer to following links in Genesys documentation

<https://docs.genesys.com/Documentation/ES/8.5.1/Admin/mask> and

<https://docs.genesys.com/Documentation/ES/8.5.1/Admin/SensDat>

6.3.3 VM and OS hardening

Operating Systems are often pre-configured for ease of use and development and not necessarily security. If the O/S is being installed or is part of a set of VMs being delivered, that O/S should be hardened to ensure that typical security holes are addressed.

The following document provides recommendations that can be used to harden the solution VMs and the OS.



Microsoft Word 97
- 2003 Document

6.3.4 Secure deployment for Internet facing components

The Digital Blueprint contains a number of components which are Internet facing. There are two possible approaches for deploying the Genesys Internet facing components:

1. Separate the Genesys components by putting Genesys external services (Genesys components processing traffic from internet clients) in the DMZ and Genesys core components in internal network.
2. Putting all Genesys components in an internal network and configure the firewall between the DMZ and internal network to prevent binary protocol passing between the zones. For this approach it is mandatory to have a reverse proxy deployed in DMZ and pointed at the Genesys external services which are installed in the internal network.

For the first approach there is no need to use a reverse proxy but the security team needs to configure the firewall correctly.

For the second approach the customer needs to configure the reverse proxy to allow traffic arriving in DMZ to be passed to the internal network. This also means the reverse proxy needs to be scaled correctly to handle additional load generated by Genesys traffic. In this approach log analysis for any type of attack needs to be performed using logs from the reverse proxy as the Genesys logs will not contain any public IP addresses.

All Genesys component need access to the Genesys configuration layer to start. Since the configuration layer contains critical system information the following measures must be taken to secure access:

1. External components should have dedicated account in Genesys used to access configuration. Using default system account is serious security breach as it would allow an attacker who gains access to the server with Genesys components to retrieve the whole Genesys configuration (including all passwords used to access databases).
2. The dedicated account should be configured to see only objects required for proper operation of service
3. External components should access the configuration through a dedicated configuration proxy. This approach prevents DOS type of attacks from affecting the rest of the Genesys infrastructure.

To provide additional security it is recommended to use client side ports. The client-side port definition feature enables a client application (of server type) to define its connection parameters before connecting to the server application. This enables the server application to control the number of client connections. In addition, if the client application is located behind a firewall, the server application will be able to accept the client connection by verifying its predefined connection parameters.

Web Security Testing

Genesys performs security testing with OWASP Zed Attack Proxy (ZAP) to make sure the Genesys solutions are protected against known attacks. ZAP is an Intercepting Proxy. It allows you to see all of the requests made to a website/web app and all of the responses received from it. For example, you can see AJAX calls that might not otherwise be obvious. Once set up, ZAP automatically passively scans all of the requests to and responses from the web application being tested. While mandatory use cases for the application that is being tested are followed (either manually or automatically), ZAP analyzes the requests to verify the usual operations are safe.

Active scanning attempts to find potential vulnerabilities by using known web attacks against the selected targets. Active scanning is an attack on those targets. ZAP emulates known attacks when active

mode is used.

Through active scanning, Genesys Co-browse is verified against the following types of attacks:

- Spider attack — Automatically discovers all URL links found on a web resource, sends requests, and analyzes results (including src attributes, comments, low-level information disclosure, and so on).
- Brute browsing (based on the Brute Force technique) — Systematically makes requests to find secure resources based on known (commonly used) rules. For example, backup, configuration files, temporary directories, and so on.
- Active scan — Attempts to perform a predefined set of attacks on all resources available for the web resource.
- Ajax spider — Automatically discovers web resources based on presumed rules of AJAX control (JS scripts investigation, page events, common rules, dynamic DOM, and so on)

In following paragraphs there more details about security for each Genesys component facing Internet.

6.3.4.1 Co-browse server security

Co-browse server is responsible for sharing pages between the customer and agent.

Traffic generated and received by Co-browse server includes:

- Bidirectional http(s)/WS(s) traffic initiated from javascript running in the customer browser which contains a copy of the page currently browsed by the customer
- Bidirectional binary traffic initiated by the Co-browse server towards the Genesys configuration layer to receive operational parameters
- Bidirectional binary traffic initiated by the management layer to monitor components running on server where Co-browse is installed
- Outbound binary traffic initiated by the Co-browse server to the management layer to send critical log events
- Bidirectional http(s)/WS(s) traffic initiated by the agent application to share the session with the customer
- Outbound traffic from the Co-browse server to the website which is being browsed by the customer to download static content. This traffic can pass through the proxy if direct connections from the server to Internet are not allowed.

The Co-browse server needs access to the following configuration objects and permissions:

- Read only access to host objects where Co-browse server and chat servers are installed
- Read only access to application objects to which Co-browse server has connections
 - o Configuration proxy/configuration server
 - o Message server

The following diagram depicts the Co-browse server installed in the DMZ with all types of traffic originating or terminating on the Co-browse server.

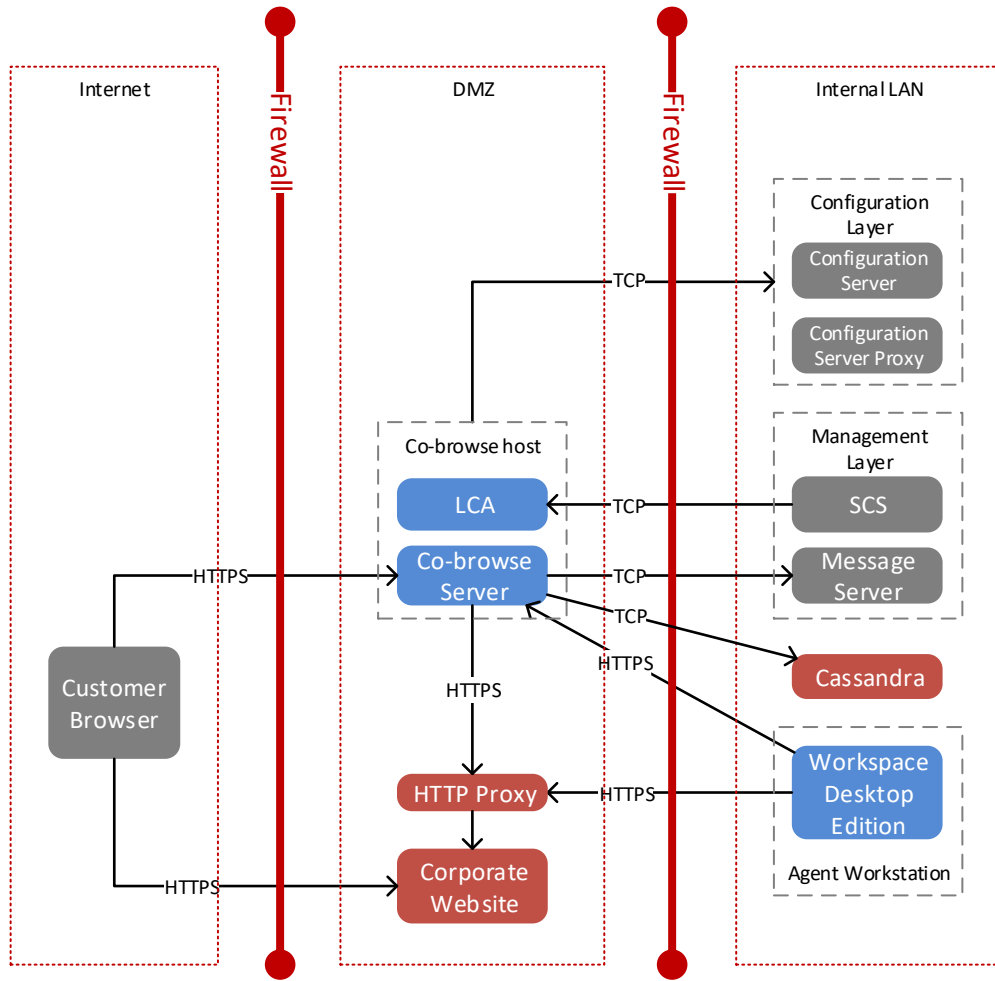


Figure 12 Co-browse in DMZ

The following diagram depicts the Co-browse server installed in an internal network and a reverse proxy used to manage all type of traffic originating or terminating on the Co-browse server.

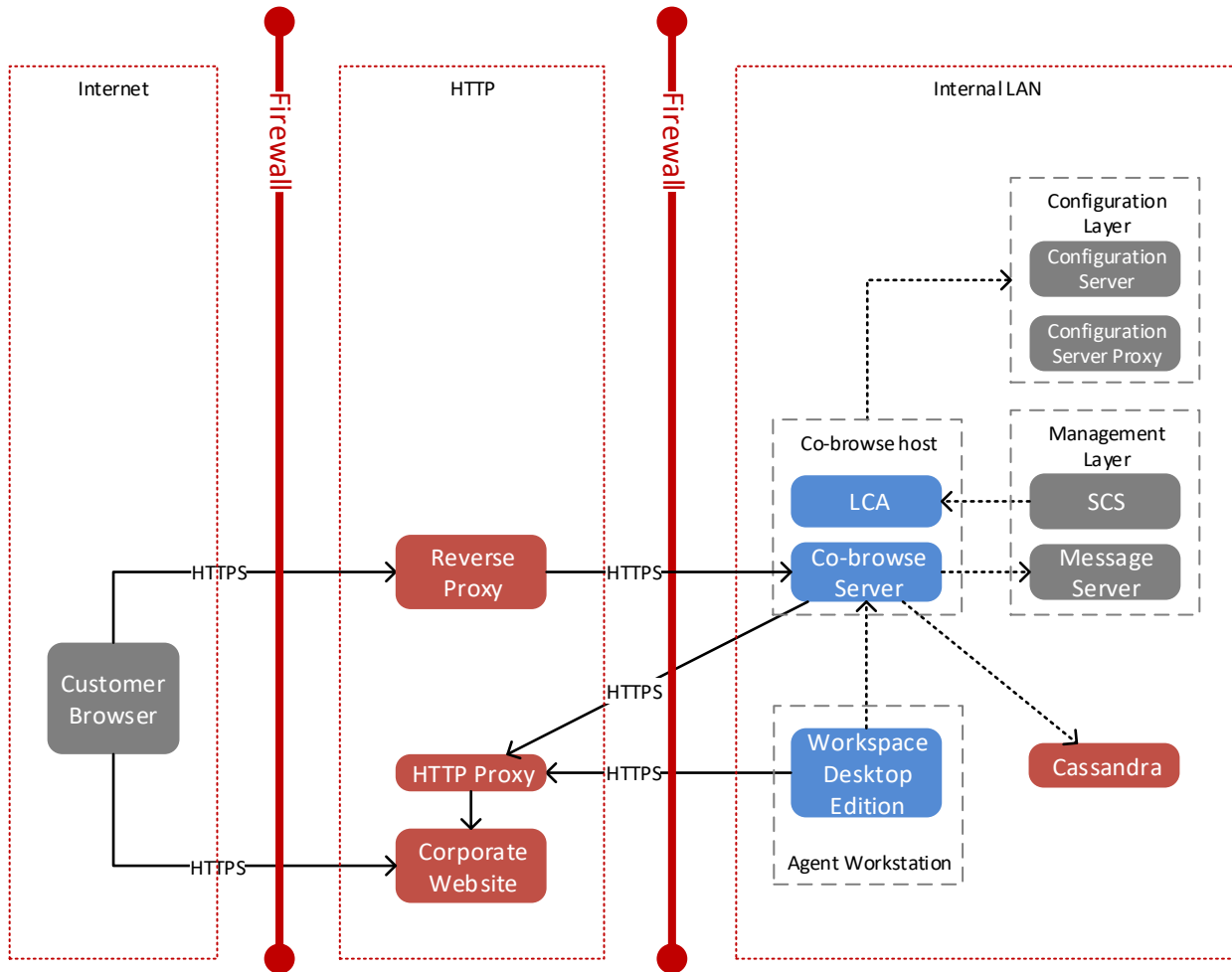


Figure 13 – Co-browse with Reverse Proxy

6.3.4.2 Callback security considerations

When Genesys Callback is integrated with a native mobile application (iOS or Android) it can notify the phone when an agent is available. To enable this functionality connectivity must be established between Genesys Mobile Server and the Cloud-based Apple or Google Push Notification Services (PNS) which will be used.

6.4 Localization and Internationalization

Localization and Internationalization are topics for numerous Genesys components, especially user interfaces and reporting. Within the Digital Solution, the main components to pay particular attention are:

- UCS
- Agent desktop software

- Reports

Under following link <https://docs.genesys.com/Documentation/MixedLangWP> Genesys published document describing how each component can be configured to properly support international characters.

Appendix A Common Components Summary

The Genesys Common Component Blueprint provides a foundational architecture which consists of elements utilized across all Blueprint architectures. The Common Components architecture is not intended as a standalone solution but should be used with other Blueprint architectures as it provides common capabilities which they utilize.

Readers are recommended to review the Common Component Blueprint for detailed information on the Common Components and the architectural considerations and recommendations. The following summary is provided as a quick reference so readers are aware of what is covered by the Common Components Blueprints.

Common Components Scope

Genesys Common Component Blueprint covers standard elements which are used by multiple solutions. The Common Component Blueprints is focused on 3 areas – Orchestration, Reporting and Configuration/Management.

Genesys Orchestration– World class routing engine which intelligently distributes interactions to the right contact center resources based contextual information and business rules and the overall state of contact center resources.

The following components are included within the Orchestration layer:

- Universal Routing Server
- Orchestration Server
- Stat Server
- Universal Contact Server
- Genesys Mobile Services (Context Services)
- Genesys Rules
- Genesys Web Services

Genesys Reporting – Real-time and Historical reporting provided by Pulse (Real-time) and ICON/Info Mart/Interactive Insights (Historical)

The following components are included within Reporting layer:

- Stat Server
- Pulse (Consists of Collector, Storage and Rabbit MQ)
- Interaction Concentrator
- Info Mart
- Interactive Insights

Genesys Configuration and Management – OAM&P layer enabling centralized configuration, management and alarming of the entire Genesys environment.

The following components are included within the Configuration and Management layer:

- Genesys Administrator / Genesys Administrator Extensions
- Configuration Server
- Solution Control Server
- SNMP Master Agent
- Message Server
- Local Control Agent
- DB Server (Used in prior Genesys version or for other Genesys components)

Common Components Scope

The following diagram shows the components included in the Common Components Blueprint.

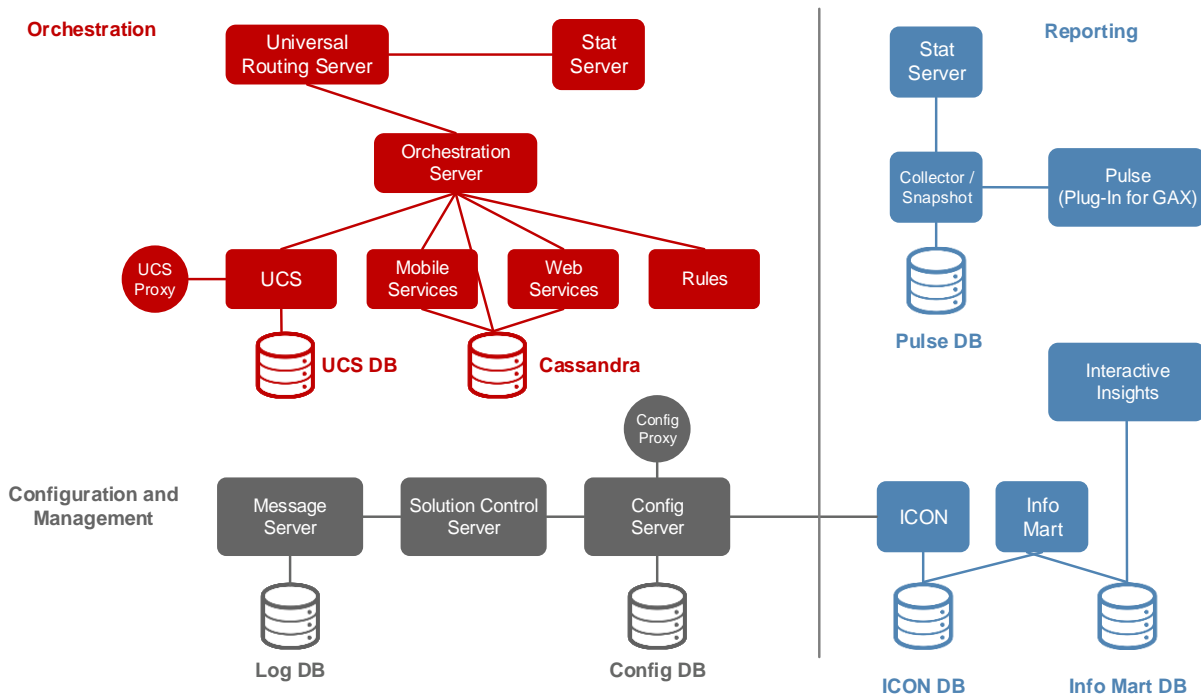


Figure 14 Common Components Blueprint - High Level

The Common Components are the foundation of any solution. The diagram is intended to provide a clear understanding of the different areas which are addressed by the Common Components Blueprint and not all components or connections are shown. The general expectation is that all elements reflected in the Common Components Blueprints will be deployed. Utilizing a standardized architecture and standard deployment strategy helps create consistency, repeatability and enables extensibility of the Genesys capabilities through the additional business scenarios without needing to install other Genesys components. For example while an Email deployment may not require Genesys Mobile Services, Web Services or Rules these components are expected to be installed. There are some exceptions to this as

those Solution Blueprints which do not directly Orchestrate interactions such as Workforce Management may not have these components installed as standard.

Appendix B Load Balancer Configuration

Genesys provides sample load balancer configuration details on the Genesys Documentation web site covering both Co-browse and Web Engagement.

Load Balancer: Co-browse

Samples for NginX and Apache are publicly available in [Genesys Documentation page - Configuring a Load Balancer for Co-browse Cluster](#).

D.1.1. List of variables

(DataPost, QueryString, Cookie...)

Name	Type (DataPost, QueryString, Cookie)	Format	Description	Frequency	Size
gcbSessionServer	Cookie (session)	Co-browse Node Application Name in CME			
genesys.cb.session	Cookie (short-lived, browser only)	Arbitrary string with JSON	Cookie is saved on web page unload and is erased on web page load, so not sent to server		
gcb-cobrowse-reload	Cookie (short-lived, browser only)	Arbitrary string with JSON	Cookie is saved on web page unload and is erased on web page load, so not sent to server		

D.1.2. List of static resources

(example: .css, .js, .gif, .html...)

Green items must be accessible publicly. Blue may be made available only for agents.

Resource	Description
/cobrowse/js/gcb.min.js /cobrowse/js/genesys.min.js /cobrowse/js/slave.min.js	Co-browse JavaScript applications
/cobrowse/slave.html /cobrowse/slaveMirror.html	Co-browse slave pages
/cobrowse/css/slave/normalize.css /cobrowse/css/slave/slave.css /cobrowse/css/toastr.css /cobrowse/css/smoke.css /cobrowse/css/mirror.css /cobrowse/css/slave/themes/wde.css /cobrowse/css/slave/themes/wde-hc.css /cobrowse/css/slave/themes/iws.css	Co-browse slave CSS
/cobrowse/chatWidget.html	Chat widget
/cobrowse/chatTemplates.html	Default chat templates
/cobrowse/nls/de-de.json /cobrowse/nls/es.json /cobrowse/nls/fr-ca.json	Built-in slave localizations. All localization files are loaded as JSONP with "callback" query variable, e.g.:

/cobrowse/nls/fr-fr.json /cobrowse/nls/js.json /cobrowse/nls/pt-br.json	
/cobrowse/js/chatAPI.min.js /cobrowse/js/chatAPI-noDeps.min.js /cobrowse/js/chatWidget.min.js	Chat API Documentation: https://docs.genesys.com/Documentation/GCB/8.5.0/API/JSChatAPI#Accessing_the_Chat_Service_API_using_the_JavaScript_Bundle

D.1.3. List of URIs and methods

Green items must be accessible publicly. Blue can be made available only for agents. Red should not be made public.

Description	Url	Methods	Comments
Session creation	/cobrowse/rest/live/sessions/create	GET	
Session creation	/cobrowse/rest/live/sessions	POST	
Session retrieval by slave	/cobrowse/rest/live/sessions/<session_token>	GET	
Session termination	/cobrowse/rest/users/<userToken>/session/stop	GET	userToken example is "12p2c7192umn611kjnip"
Health check	/cobrowse/health	GET	Might be used by Load Balancer (depends on LB config)
CometD	/cobrowse/cometd /cobrowse/cometd/connect /cobrowse/cometd/disconnect /cobrowse/cometd/handshake	WebSocket, GET, POST	Can be WebSocket, POST or JSONP GET. In case of JSONP full URL include "callback" and "_" variables, e.g. ?callback=jQuery18205012391582131386_1457533491494&_=1457533500497
DOM Restrictions	/cobrowse/rest/dom-restrictions.json	GET	
CSS proxy	/cobrowse/css-proxy?url=<URL>&for=<(1 2)>&referrer=<URL>&cobrowseUrl=<URL>	GET	
URL proxy	/cobrowse/url-proxy?url=<URL>&referrer=<URL>		
Reporting signal from slave to CB server	/cobrowse/rest/live/sessions/<session_token>/slave-rendered	GET	
Static resources	/static/*	GET	JSONP requests. Isn't used out-of-the-box, using it is up to customer. See https://docs.genesys.com/Documentation/GCB/8.5.0/Deployment/ServingJSONP
Get session history	/cobrowse/rest/history/sessions/<sessionHistoryId>	GET	

Load Balancer: Web Engagement

Samples for NginX and Apache are publicly available in [Genesys Documentation page - Genesys Web Engagement - Load Balancing](#).

D.1.4. List of variables (UPDATE FOR GWE 8.5)

(DataPost, QueryString, Cookie...)

Name	Type (DataPost, QueryString, Cookie)	Format	Description
com.genesyslab.wme.tracker.globalVisitId	Cookie (3 years)	6888573e-163e-4a70-97b4-bbefa6c0c1d9	
com.genesyslab.wme.tracker.visitId	Cookie (session)	24800c38-2fa5-481a-b90b-34316dbac402	
com.genesyslab.wme.tracker.serverAlias	Cookie (session)	.Web_Engagement_Frontend_Server_01	
com.genesyslab.wme.tracker.transport	Cookie (session)	ajax	
FRONTEND_ROUTEID	Cookie (session)	"." + Frontend_Server_Application_Name_In_Genesys_CME	See details here: https://docs.genesys.com/Documentation/GWE/8.1.2/Deployment/LoadBalancing#Sticky_Sessions
ROUTEID	Cookie (session)	:%{BALANCER_WORKER_ROUTE}e	See details here: https://docs.genesys.com/Documentation/GWE/8.1.2/Deployment/LoadBalancing#Architecture

D.1.5. List of static resources (UPDATE FOR GWE 8.5)

(example: .css, .js, .gif, .html...)

Url: /frontend/resources/*

Documentation: https://docs.genesys.com/Documentation/GWE/8.1.2/Developer/Architecture#Hosting_Static_Resources

Items marked with **green** are publically available (like monitoring API), **red** items – for internal access only (like SCXML strategies) and **blue** items (Historical REST API) are internal one, but can be publically available for the case when Agent Desktop is located in public network.

Resource	Description
invite.html	Invitation widget
chatWidget.html	Chat widget
chatTemplates.html	Default chat templates
callback.html	Callback widget
ads.html	Ads widget
dsl/domain-model.xml	DSL to define browser events
locale/callback-en.json locale/callback-fr.json	Callback default localization
js/chatAPI.min.js js/chatAPI.min.js.gz js/chatAPI-noDeps.min.js js/chatAPI-noDeps.min.js.gz js/chatAPI-noTransport.min.js js/chatAPI-noTransport.min.js.gz js/chatWidget.min.js js/chatWidget.min.js.gz js/chatWidget-noDeps.min.js js/chatWidget-noDeps.min.js.gz	Chat API Documentation: https://docs.genesys.com/Documentation/GCB/8.5.0/API/JSChatAPI#Accessing_the_Chat_Service_API_using_the_JavaScript_Bundle
js/build/GPE.min.js js/build/GPE.min.js.gz js/build/GT.min.js js/build/GT.min.js.gz	GWE packages

<p>js/build/GTC.min.js js/build/GTC.min.js.gz js/build/GTCJ.min.js js/build/GTCJ.min.js.gz js/build/GTJ.min.js js/build/GTJ.min.js.gz js/build/GWC.min.js js/build/GWC.min.js.gz js/build/genesys.min.js js/build/genesys.min.js.gz</p>	<p>Documentation: https://docs.genesys.com/Documentation/GWE/8.1.2/Developer/CustomizeMonitoringScript#Configuring_the_Instrumentation_Script</p>
<p>scxml (backend/data/resources)</p>	<p>Folder with SCXML strategies</p>

D.1.6.

D.1.7. List of URIs and methods (UPDATE for GWE 8.5)

...