

Version 1.73

Patch: 1.73.15

Date: January 31, 2024

1. **Issue:** Compatibility issues with Salesforce prevented multiple components from loading

Fix:

Updates were made to the SMS Magic application to ensure compatibility with Salesforce. The application's integration methods and internal logic were adjusted to align with the new release requirements. Extensive testing confirmed that all components now function seamlessly within the updated Salesforce environment, effectively resolving the loading issues for users.

2. **Issue:** Campaign completion without SMS history record creation

Fix:

A campaign using a media template showed as completed in the management UI, but SMS history records were not created. The issue was traced to a file sharing limit exceeded error during campaign execution. The solution involved modifying the campaign execution process to accommodate the file sharing constraints, ensuring SMS history records are properly generated for all campaign recipients.

Subsequent testing confirmed that campaigns with multimedia templates now correctly create SMS history and portal history records, effectively resolving the file sharing limitation concern and ensuring accurate campaign tracking.

3. **Issue:** OAuth revoke notifications not working for standard users

Fix:

Standard users assigned with all SMS Magic permissions were not receiving OAuth revoke notifications. This was identified when revoking OAuth for all users and checking for notifications across standard and admin profiles.

The root cause was found to be a missing permission in the SMS Converse Conversation User permission set. Adjustments were made to ensure that the OAuth notification is now visible to both standard users and admin profiles. After implementing the necessary permission updates and conducting tests, it was verified that OAuth revoke notifications are correctly received by standard users, addressing the issue and ensuring comprehensive notification delivery.

Patch: 1.73.12

Date: 7th Nov 23

Bug Fixed

1. **Issue:** Conversation Component Error While Loading Messages.

Fix:

Users encountered an error with the Conversation component when loading previous messages on the conversation view. The issue specifically occurred when attempting to load a set of messages that included Multimedia Messaging Service (MMS) messages. The spinner at the top of the component is continuously displayed, preventing the successful loading of the next set of messages.



The root cause of the problem was identified in the sanitize URL method, where an object of URL was mistakenly passed instead of the actual URL string. This error in handling URLs was preventing the successful loading of messages, particularly when MMS messages were included.

The issue has been successfully resolved with the following solution:

- **Correct URL String Handling:** The code has been updated to correctly handle URL strings in the sanitize URL method. This correction ensures that old messages (SMS/MMS) are now displayed without errors, allowing users to load previous messages seamlessly.

With this adjustment, users will no longer encounter errors when loading previous messages on the Conversation component, even when the set includes MMS messages. The patch addresses the reported issue and ensures a smoother and more reliable experience for users engaging with message history.

Patch: 1.73.11

Date: 3rd Nov 23

Bug Fixed

1. **Issue:** Component Error While Handling Incoming SMS

Fix:

Customer reported a component error in the Converse setting while receiving incoming SMS messages in the organization. The error message was identified as "Uncaught Error (Cannot read properties of undefined parameters)."



The root cause of the problem was identified in the code where the action parameter was becoming undefined, triggering the component error during the handling of incoming SMS messages.

The issue has been successfully resolved with the following solution:

- **Null Check for Action Parameter:** A null check has been added for the action parameter in the helper class. This adjustment ensures that the code can handle scenarios where the action parameter may be undefined, preventing the occurrence of the component error.

With this modification, users will no longer encounter the “Uncaught Error” component error when handling incoming SMS messages in the Converse setting.

Bug Fixed

1. **Issue:** Error on Converse Home

Fix:

Users encountered an error on the Converse home, with the displayed error message indicating “(Too many query rows: 50001).” This error was triggered when a query on the user object resulted in fetching more than 50,000 result rows.

Fix:

The root cause of the problem was identified in the code where a query on the user object was fetching a result set exceeding the Salesforce limit of 50,000 rows. This resulted in the “Too many query rows” error on the Converse home.



The issue has been successfully resolved with the following solution:

- **Query Adjustment to License Object:** The query that was originally running on the user object has been modified to run on the license object of the managed package instead. This change ensures that the result records count is limited to the licensed users only, avoiding the “Too many query rows” error.

With this adjustment, users will no longer encounter the error on the Converse home related to query row limits. The patch addresses the reported issue and ensures that the query runs within the constraints of Salesforce limits, providing a more stable and error-free experience on the Converse home.

Patch: 1.73.8

Date: 6th Sept 23

Bug Fixed

1. **Issue:** Issue with Conversation View in Utility Bar

Fix:

Users encountered issues related to the conversation view within the utility bar in the Sales app. Two specific problems were identified:

- When embedding the Conversation view in the utility bar for the Sales app with the “Enable Contextual Conversations” value set to false, the conversation did not load after opening a contact or lead record from the Sales Home.
- All conversation actions were opening in a new tab instead of a new window from the utility bar.

The root cause of these issues was traced to the absence of a check for contextual conversations when closing a conversation. This omission affected the loading of conversations in the utility bar and the behavior of conversation actions in terms of opening in a new tab.

The issues have been successfully resolved with the following solution:

- **Contextual Conversation Check for Closing:** A check for contextual conversations has been added to ensure proper handling when closing a conversation. This addition addresses the loading issue of conversations in the utility bar when the “Enable Contextual Conversations” value is set to false.
- **Consistent Opening of Conversation Actions:** All conversation actions from the utility bar now consistently open in new tabs, aligning with expected behavior.

With these adjustments, users should no longer encounter issues with the conversation view in the utility bar in the Sales app. The patch addresses the reported problems, ensuring a smoother and more reliable experience for users interacting with conversations in the specified context.

Bug Fixed

1. **Issue:** Records Count Showing Zero in Converse App Bulk Campaign.

Fix:

Users encountered a discrepancy where the records count in the Converse App’s “Send Bulk Message” window displayed as zero, even when records were present in the list view. This issue specifically occurred when creating a list view with a Campaign filter, and selecting that list view while sending a bulk campaign.



The root cause of the problem was identified as a limitation on the Salesforce side. When using an aggregate query with a Campaign filter added on the list view, it returned a count of zero, leading to an inaccurate representation of the records available for the bulk campaign.

The issue has been successfully resolved with the following solution:

- To address the limitation of the aggregate query returning zero counts, the system now hides the count display in the Converse App’s “Send Bulk Message” window when the count is zero but records are present in the list view. This ensures a more accurate representation of the available

records for the bulk campaign.

With this adjustment, users will no longer encounter misleading information about the records count when using a list view with a Campaign filter during the creation of a bulk campaign. The patch enhances the user experience by providing more reliable and transparent information in the Converse App.

Version 1.72

Patch: 1.72.65

Date: February 02, 2024

Issue: Compatibility issues with Salesforce prevented multiple components from loading

Fix:

Updates were made to the SMS Magic application to ensure compatibility with Salesforce. The application's integration methods and internal logic were adjusted to align with the new release requirements. Extensive testing confirmed that all components now function seamlessly within the updated Salesforce environment, effectively resolving the loading issues for users.

Patch: 1.72.62

Date: 22nd Jan 24

Bug Fixed

1. **Issue:** The customer's users were facing an error in data migration

Fix:

While performing data migration, the customer's projects team encountered the following error during the 17th step of the process.

"First error: smagicinteract:Too many query rows: 50001".

To solve this issue, a Static Resource was created with name SMI_DataMigrationBatchSize and description in JSON format for each batch.



Patch: 1.72.59

Date: 19th Dec 23

Bug Fixed

1. **Issue:** The non-admin users of the customer were facing a component error on the Converse desk.

Fix:

The non-admin users of the customer were facing a component error on the Converse Desk. The unused and unreferenced code was causing this problem.



2. **Issue:** Users of the customer were facing an error in the audit history for recent records.

Fix:

The customer was facing an error on all recently created non-handset consent records. They were not able to see the audit history. It was giving an error. The issue was not noticed for incoming SMS. A few older non-handset consent records also did not show any error. Wrong syntax was used in dynamic query due to which this error occurred.

To solve the query, changed the dynamic query to use a variable instead of the Map.keySet() method.



Patch: 1.72.60

Date: 27th Dec 23

Bug Fixed

1. **Issue:** The utility bar conversation view record text was overlapping for the customer

Fix:

The Utility Bar Conversation View record text was overlapping for some account records. This was not happening for all the account records. To solve this issue, the sls-truncate class was applied to the UI.



Patch: 1.72.55

Date: 8th Dec 23

Bug Fixed

1. **Issue:** The user was facing an error while upgrading the managed package.

Fix:

The users of the customer were facing an invalid definition error while upgrading the managed package. To solve the issue, created a new .design file for the component 'createTemplateComponent' as suggested in one of the workarounds.

2. **Issue:** The user was facing a name field issue while creating a shortened link via custom code.

Fix: The user was facing a name field error while creating a shortened link via a custom code. A specific value for the "Name" field was provided. Still that value was not picked up while creating the shortened link record. Value for the name field was inherited from the long URL. To solve the issue, a condition to populate the value for the name field of the ShortenedLink record was added, only if it was not provided.

Patch: 1.72.52

Date: 30th Oct 23

Bug Fixed

1. **Issue:** Scrolling Functionality Restored in Template Dropdown

Fix:

After upgrading to the latest version (1.72.51), users in the customer's organization lost the capability to scroll through templates in the 'Insert template' dropdown. This issue required users to type the title or a word to locate the desired template, impacting the user experience.

The issue has been successfully resolved with the following solution:

- **CSS Adjustment for Template Dropdown:** CSS has been added to the template dropdown component to allow scrolling of the template list. This enhancement ensures that users can now scroll through the template list when the number of templates exceeds the initial UI capacity.

With this adjustment, users will regain the ability to scroll through the template list, providing a more user-friendly experience when selecting templates. The patch addresses the reported issue and ensures that the scrolling functionality is restored in the 'Insert template' dropdown after the upgrade to version 1.72.51.

Patch: 1.72.51

Date: 24th Oct 23

Bug Fixed

1. **Issue:** Error on Converse Home

Fix:

Users encountered an error message on Converse Home, specifically a "Too many

query rows: 50001" error. This issue was associated with the execution of a query in the analytics controller that ran on the user object and executed twice. The large user count in the organization contributed to the query exceeding the Salesforce limit of 50,000 rows.

The root cause of the problem was identified in the analytics controller class, where the query was running on the user object and executing twice. The combination of a large user count and the execution of the query on the user object contributed to the "Too many query rows" error.

The issue has been successfully resolved with the following solution:

- **Query Adjustment to License Object:** The query in the analytics controller class has been modified to execute on the license object rather than the user object. This change ensures that the result records count is limited to licensed users only, preventing the "Too many query rows" error.
- **Null Check Addition:** A null check has been added in the Message List Component Server Controller class. This additional validation enhances the code's robustness and stability.

With these modifications, users will no longer encounter the "Too many query rows" error on Converse Home, and the data should be displayed correctly even in organizations with a large user count. The patch addresses the reported issue and ensures a more reliable and error-free experience for users on Converse Home.

Bug Fixed

1. **Issue:** Email Notifications not working for Unassigned Sender ID

Fix:

Users experienced a scenario where email notifications were not functioning if a user was not assigned to the sender ID associated with an incoming message. This issue was linked to the code's filtering mechanism, which was based on sender ID visibility.

The root cause of the problem was identified in the code, specifically in the filtering mechanism for sender IDs based on visibility. This mechanism was preventing email notifications from being sent if the user was not assigned to the sender ID associated with an incoming message.

The issue has been successfully resolved with the following solution:

- **Sender ID Filtering Adjustment:** The code's sender ID filtering mechanism has been adjusted to allow email notifications even when the user is not assigned to the specific sender ID associated with an incoming message.

With this modification, users will now receive email notifications for incoming messages, even if they are not assigned to the sender ID linked to the message.

Bug Fixed

1. **Issue:** Error Log Generation in Message List Component Server Controller

Fix:

An error log with the message "Attempt to de-reference a null object" was generated for all users in the Message List Component Server Controller class. This issue occurred when rendering media messages in the Conversations component, either in Converse Desk or when embedded in a record detail page.

The issue has been successfully resolved with the following solution:

- **Query Adjustment to License Object:** The query in the Analytics Controller class has been modified to execute on the license object rather than the user object. This change ensures that the result records count is limited to licensed users only, preventing the "Too many query rows" error.
- **Null Check Addition:** A null check has been added in the Message List Component Server Controller class. This additional validation enhances the code's robustness and stability, preventing the "Attempt to de-reference a null object" error.

With these modifications, users will no longer encounter the error log related to "Attempt to de-reference a null object" when rendering media messages in the Conversations component. The patch addresses the reported issue and ensures a more reliable and error-free experience for users in Converse Desk and when viewing media messages embedded in record detail pages.

Patch: 1.72.50

Date: 20th Oct 23

Bug Fixed

1. **Issue:** Conversation View Loading Issue

Fix:

Users were encountering a problem where the Conversation component at the record level was continuously loading, preventing its proper use.

The root cause of the problem was identified, the logic to hide the composer spinner was placed inside a condition checking for the existence of the old compliance. This condition was not evaluating correctly when the new compliance was enabled, leading to the spinner not hiding.

The issue has been successfully resolved with the following solution:

- **Condition Removal:** The condition checking for the existence of the old compliance has been removed. Hiding the composer spinner is required irrespective of whether the old or new compliance is active. This modification ensures that the spinner hides automatically.

With this adjustment, users should no longer experience the continuous loading issue with the Conversation component at the record level. The patch addresses the reported problem, allowing for a smoother and more reliable user experience with the Conversation view.

Bug Fixed

1. **Issue:** Mobile number display in popover bubble for individual messages

Fix:

In the previous version of SMS Magic (e.g., V1.60.xx), the recipient's phone number was displayed in the popover bubble of individual messages. However, in the newer versions, if the text message was short, the number would only show partially, and the popover bubble did not display the recipient's phone number as expected.



The root cause of the problem was identified as a potential removal of the corresponding functionality during the development of newer versions or additional functionalities.



The issue has been successfully resolved with the following solution:

- **Conditional Mark-up Addition:** Conditional mark-up has been added in the message component file to display the recipient's mobile number when the setting is enabled from Converse settings.

With this modification, the recipient's mobile number is now displayed in the popover bubble for both incoming and outgoing messages, providing users with the expected and complete information. The patch addresses the reported issue, ensuring a consistent and accurate display of mobile numbers in the popover bubble for individual messages.

Bug Fixed

1. **Issue:** Email-to-Text functionality issue for Admin (when Guest User configured with same Email ID)

Fix:

Users encountered an issue with the Email-to-Text functionality when configured for admin users. Specifically, when a guest user was configured with the same email id as the admin user, the outgoing message record was encountering an error due to the guest user being returned as the owner.

The root cause of the problem was identified as the configuration of a guest user with the same email id as the admin user. When fetching user records based on email, the guest user record was returned in the result, and as guest users cannot own records in the org, an error occurred.

The issue has been successfully resolved with the following solution:**WHERE Clause Addition in User Query:** In the user query, a WHERE clause has been added to filter the user type for 'standard' users. This modification ensures that when fetching user records based on email, only standard users are considered.

With this adjustment, incoming and outgoing message SMS history is now created in the Salesforce org without encountering the previous error. The patch addresses the reported issue, allowing for the smooth operation of the Email-to-Text functionality even in scenarios where a guest user shares the same email id as an admin user.

Patch: 1.72.47

Date: 4th Oct 23

Bug Fixed

1. **Issue:** Last Incoming Lookup and Last Incoming Time was not getting populated with encrypted phone fields.

Fix:

Users experienced an issue where the Last Incoming Lookup and Last Incoming Time fields were not getting populated on the conversation object, specifically when encryption was enabled on phone fields. This resulted in incomplete and inaccurate information being displayed on the Converse desk.

The root cause of the problem was identified in the code, which encountered errors when querying on Message Object Configuration (MOC) phone fields, especially when they were encrypted. This issue affected the accurate population of Last Incoming Lookup and Last Incoming Time fields.

The issue has been successfully resolved with the following solution:**Error Handling for Encrypted Phone Fields:** The code has been updated to handle errors related to encrypted phone fields during querying. This adjustment ensures that Last Incoming SMS Record and Last Message Time are now correctly populating on the conversation object, even when phone fields are encrypted.



With this enhancement, users will now observe accurate and complete information for the Last Incoming Lookup and Last Incoming Time on the conversation object, improving the reliability of data on the Converse desk. The patch addresses the reported issue and ensures a more robust functionality in the presence of encrypted phone fields.

Improvement

1. **Issue:** A new feature has been introduced to play a sound notification upon conversation assignment. This enhancement is designed to provide users with an audible alert when they are assigned a conversation. The feature is configurable and will only activate if a specific static resource is present with a predefined value.

Fix:

The improvement has been successfully implemented with the following solution:

- **Static Resource Configuration:** The system now checks for the presence of a specific static resource with a predefined value (e.g., a static resource named "Conversation Assignment Sound" with the value "true").
- **Play Sound on Assignment:** If the configured static resource is present with the correct value, a sound notification will be played upon conversation assignment. This feature aims to enhance user awareness and responsiveness to newly assigned conversations.
- With this enhancement, users will receive sound notifications after the assignment of conversations, providing an additional means of alert for improved workflow management. The patch introduces a customizable and user-friendly experience for users engaged in conversation assignments.

Bug Fixed

1. **Issue:** A new feature has been introduced to play a sound notification upon conversation assignment. This enhancement is designed to provide users with an audible alert when they are assigned a conversation. The feature is configurable and will only activate if a specific static resource is present with a predefined value.

Users encountered an issue where, despite removing custom permissions for specific users to restrict access to Conversations Actions such as "Close Conversation" and "Assign Conversation," the "Close Conversation" option remained enabled for those users. This discrepancy in behavior resulted in inconsistent access control.

The root cause of the problem was identified in the conversation header component. The necessary code to disable the "Close Conversation" option was missing, leading to the persistence of the option even after the removal of custom permissions.

The issue has been successfully resolved with the following solution:

- Updated the conversation header component functionality to disable the "Close Conversation" option. This ensures that users without the necessary custom permissions no longer have access to the "Close Conversation" functionality.



With this adjustment, users will experience consistent access control for Conversations Actions, and the "Close Conversation" option will be appropriately disabled for users without the required permissions. The patch addresses the reported issue and ensures accurate enforcement of access restrictions.

Fix:

Patch: 1.72.47

Date: 6th Sept 23

Bug Fixed

1. **Issue:** Customer was getting License Restriction Error on Attaching Media File in Global Message and Sidebar.

Fix:

Users faced a license restriction error when attempting to attach a media file to a global message and sidebar in the Salesforce org. The error prevented the successful attachment of MMS files, causing inconvenience for users.



The root cause of the problem was identified in the post message function where the source URL was not matching. This discrepancy in the source URL was triggering a license restriction error, preventing the smooth attachment of media files in global messages and sidebars.

The issue has been successfully resolved with the following solution:

- **Source URL Matching Fix:** The source URL in the post message function has been adjusted to ensure accurate matching. This correction eliminated the license restriction error that was occurring during the attachment of MMS files in global messages and sidebars.

With this adjustment, users will now be able to attach media files to global messages and sidebars without encountering license restriction errors.

Patch: 1.72.46

Date: 28th Aug 23

Bug Fixed

1. **Issue:** Issue with Conversation View in Utility Bar

Fix:

Users encountered issues related to the conversation view within the utility bar in the Sales app. Two specific problems were identified:

- When embedding the Conversation view in the utility bar for the Sales app with the "Enable Contextual Conversations" value set to false, the conversation did not load after opening a contact or lead record from the Sales Home.
- All conversation actions were opening in a new tab instead of a new window from the utility bar

The root cause of these issues was traced to the absence of a check for contextual conversations when closing a conversation. This omission affected the loading of conversations in the utility bar and the behavior of

conversation actions in terms of opening in a new tab.

The issues have been successfully resolved with the following solution:

- **Contextual Conversation Check for Closing:** A check for contextual conversations has been added to ensure proper handling when closing a conversation. This addition addresses the loading issue of conversations in the utility bar when the “Enable Contextual Conversations” value was set to false.
- **Consistent Opening of Conversation Actions:** All conversation actions from the utility bar now consistently open in new tabs, aligning with expected behavior.

With these adjustments, users should no longer encounter issues with the conversation view in the utility bar in the Sales app. The patch addresses the reported problems, ensuring a smoother and more reliable experience for users interacting with conversations in the specified context.

Bug Fixed

1. **Issue:** Records Count Showing Zero in Converse App Bulk Campaign.

Fix:

Users encountered a discrepancy where the records count in the Converse App’s “Send Bulk Message” window displayed as zero, even when records were present in the list view. This issue specifically occurred when creating a list view with a Campaign filter, and selecting that list view while sending a bulk campaign.



The root cause of the problem was identified as a limitation on the Salesforce side. When using an aggregate query with a Campaign filter added on the list view, it returned a count of zero, leading to an inaccurate representation of the records available for the bulk campaign.

The issue has been successfully resolved with the following solution:

- To address the limitation of the aggregate query returning zero counts, the system now hides the count display in the Converse App’s “Send Bulk Message” window when the count is zero but records are present in the list view. This ensures a more accurate representation of the available records for the bulk campaign.

With this adjustment, users will no longer encounter misleading information about the records count when using a list view with a Campaign filter during the creation of a bulk campaign. The patch enhances the user experience by providing more reliable and transparent information in the Converse App.

Patch: 1.72.45

Date: 24th Aug 23

Improvement

1. **Issue:** Users were not able to send bulk SMS

Fix:

Users reported an issue where they were unable to send bulk SMS, with the SMS getting stuck in the "submitted" status.

The root cause of the problem was identified as a lack of access to the campaign field on the batch tracker object for the affected user. This absence of access hindered the proper execution of bulk SMS sending, resulting in SMS messages being stuck in the "submitted" status.

The issue has been successfully resolved with the following solution:

- **Field Access Check on Batch Tracker Object:** A check for access to the campaign field on the batch tracker object has been implemented. Before running the query for sending bulk SMS, the system now verifies that the user has the necessary access to the campaign field. This ensures that users can send bulk SMS without encountering issues related to field access.

With this adjustment, users should no longer face obstacles in sending bulk SMS, and the problem of SMS messages getting stuck in the "submitted" status has been effectively addressed. The patch enhances the overall functionality and reliability of the bulk SMS sending feature.

Bug Fixed

1. **Issue:** Sender ID & Channel Not Mapped

Fix:

Following the upgrade of the customer Sandbox account from version 1.58.48 to version 1.72.35, automated messages were failing with the error message: "Channel and senderid not mapped."

The issue was traced to the handling of sender IDs during message processing. Specifically, the inclusion of a '+' before the sender ID was causing the error, resulting in the mentioned error message.

The issue has been successfully resolved with the following solution:

- **Sender ID Handling Adjustment:** The process of appending '+' before the sender ID has been reviewed and rectified. This adjustment ensures that sender IDs are processed correctly, eliminating the error that occurred during automated message testing post-upgrade.

This patch ensures the smooth execution of automated messages in the customer's Sandbox account following the upgrade to version 1.72.35. Users can now confidently deploy automated messages without encountering any error.

Bug Fixed

1. **Issue:** Component Error When Attaching Media Files on Conversation Component

Fix:

Users encountered a component error when attempting to attach media files to the Conversation component in Converse Desk. The error occurred during the process of saving the file, and further complications arose when attempting to cancel the operation.



The absence of the defined origin context variable in the Upload Controller file was identified as the root cause of the issue. This omission led to a component error when interacting with media files in the Conversation component.

The following solution has been implemented to resolve the issue::

- **Origin Context Variable Definition:** The origin context variable has been defined in the Upload Controller file. This addition ensures that the necessary context is established, addressing the component error encountered during media file attachment.

With this adjustment, users should now be able to seamlessly attach and save media files within the Conversation component without experiencing errors during the saving or cancellation process. The patch addresses the reported issue and enhances the overall functionality of media file handling in Converse Desk.

Patch: 1.72.44

Date: 07th Aug 23

Improvement

1. **Issue:** Improvement for Showing Seconds and Time Zone in Chat Conversation Topic Window

Fix:

This patch introduces an enhancement to the display of messages within the Chat Conversation Topic window. Specifically, it improves the presentation by including seconds in the timestamp and replacing AM/PM with the user's time zone.

Details:

- **Show Seconds in Timestamp:** Messages in the Chat Conversation Topic window now include seconds in the timestamp. This refinement offers users a more precise understanding of the timing of each message.
- **Display User Time Zone:** Instead of AM/PM, the timestamp now reflects the user's time zone. This change provides users with immediate visibility into the time context of messages based on their individual time zone

settings.

Validated the functionality across multiple browsers, including Chrome, Safari, and Firefox, to ensure consistent and reliable performance.



Users can now enjoy a more granular timestamp and a personalized time zone representation in the Chat Conversation Topic window, contributing to an improved and user-friendly messaging experience.

Patch: 1.72.43

Date: 07th Aug 23

Bug fixed

1. **Issue:** Unable to Install SMS-Magic in the Salesforce Instance.

Fix:

Users encountered difficulties installing SMS Magic version 1.72.40 in their Salesforce instance. The installation process failed, particularly when attempting to install the package.

The installation failure was traced back to encrypted fields on critical objects (account, contact, user, lead). The encrypted fields were causing the installation to fail, specifically generating an error stating, "cannot be filtered in a query call."

To resolve the installation issue related to encrypted fields, we have implemented the following solution:

- **Used Encrypted Field in WHERE Clause:** We have modified the installation process to accommodate the encrypted fields by incorporating them into the WHERE clause. This adjustment ensured that the package installation was successfully handled with encrypted fields on the specified objects (Account, Contact, User, Lead).

With these changes, users should now be able to install SMS Magic version 1.72.40 successfully, even in instances where certain fields are encrypted.

Improvement

1. **Issue:** Data Migration Improvement for Timeframe

Fix:

The Data Migration page in Salesforce lacked a crucial option to specify a migration timeframe, causing users to face challenges when attempting to migrate data selectively. To address this limitation, we have introduced an option enabling users to migrate data either from a specific date or encompassing all available data.

Details:

Enhancements have been made to the Data Migration functionality in Salesforce, specifically addressing the need for improved control over the timeframe for data migration. Previously, users encountered limitations in selecting the data migration timeframe.

To empower users with more control over their data migration process, the following solution has been implemented:

- **Selectable Data Migration Timeframe:** In the Salesforce Data Migration page, users can now choose between migrating all available data or specifying a particular date from which the migration process should commence. This option provides flexibility and precision, allowing users to tailor their data migration according to specific business needs or compliance requirements.

With this enhancement, users can experience a more streamlined and efficient data migration process, selecting the timeframe that best suits their operational requirements. This improvement is designed to enhance the overall usability and functionality of the Data Migration feature within SMS Magic and offer a more refined data migration experience for users.

Bug fixed

1. **Issue:** Default Notification Settings Saving Issue

Fix:

Users reported an issue where changes made to the “Default Notification Settings” in Converse settings were not persisting. Specifically, when selecting “Reply to address (Email ID)” and attempting to save, the changes were being reverted upon page refresh.

We have identified a duplication of the Default Email Template and a compatibility issue with the Lightning Web Security session setting. The redundancy in the Default Email Template and the error thrown by the aura modal when Lightning Web Security was enabled contributed to the inability to save Default Notification Settings successfully.

To address this issue, the following solution was implemented:

- **Default Email Template Redundancy Fix:** The duplication of the Default Email Template was rectified, eliminating any conflicting settings that might have hindered the saving of Default Notification Settings.
- **Lightning Web Security Compatibility Enhancement:** Compatibility with Lightning Web Security was improved to prevent errors in the aura modal. This ensured a seamless saving process for Default Notification Settings, even when Lightning Web Security was enabled.

With these changes, users should now be able to successfully save Default Notification Settings without encountering issues related to duplication and Lightning Web Security. The patch addresses the reported problem and enhances

the overall stability of the Converse settings.

Patch: 1.72.42

Date: 28th July 23

Bug fixed

1. **Issue:** The customer was facing a component error

Fix:

The Customer reported a component error when the Aircall was terminated from the lead. It was an intermittent problem.



The issue stemmed from the Compliance Data method of the Composer Component in the conversation component controller file, which was not returning the expected value.

To address this, we have implemented the following solutions:

- **Validation Added for Compliance Data Method:** We enhanced the validation process for the Compliance Data method in the conversation component controller file. This ensured that the method consistently returned the expected value, mitigating the component error observed during air call termination.
- **Null Check for Compliance Data Method in Composer Component Controller:** A null check was incorporated for the Compliance Data method in the Composer Component controller. This addition aimed to prevent unexpected behavior and resolve the intermittent component error observed during air call termination.

These updates collectively addressed the component error issue reported by the customer, providing a more robust and reliable experience during and after air call interactions. Users should no longer experience disruptions related to the described problem.

Patch: 1.72.40

Date: 14th July 23

Bug fixed

1. **Issue:** The customer faced an issue while creating a new task and logging a call.

Fix:

An error occurred while the customer tried to create a new task and log a call in the organization. The error stated a system null pointer exception while attempting to de-reference a null object.

We observed that the error was caused by missing the empty value condition for the subject and description fields in the creation process (create Converse task method). We have resolved this by adding the necessary 'not null' conditions for the subject and description fields in the Converse task trigger helper.

Patch: 1.72.25

Date: 15th Feb 23

Bug fixed

1. **Issue:** The customer was facing an error related to automation for community users after the post-upgrade.

Fix:

An error occurred while the community user performed automation post-upgrade.

The customer had configured 11 objects as MOC and had an automation setup on the service appointment object. This triggered the automation by changing a field value via a community user, and the automation failed with an error related to access to quote and lead objects.

It was analyzed that the community user could not access quotes and lead objects. We have checked the permissions of all the objects configured as MOC while triggering the automated message and now resolved the issue.

Patch: 1.72.26

Date: 17th Feb 23

Bug fixed

1. **Issue:** The customer faced an issue with the custom MLUI service injection point.

Fix:

An error occurred while the customer fetched the list of messages, as the custom MLUI service injection point was not working as expected. It was observed that when the custom message list UI service injection point was enabled, it was applied to the embedded conversation component on the record details page. In this component, the list of messages should have been listed as per the logic written in the custom injection class (overriding the default package behavior) only after the composer component is loaded.

It was further analyzed that the messages were displayed even before the composer component was loaded by passing the injection point during the initial component load and defeating the injection point functionality. Due to this complex component architecture, the message list was fetched from multiple methods rather than just one.

We analyzed the complex component architecture and unified the source for fetching the list of messages, and the issue is now resolved.

Patch: 1.72.27

Date: 20th Feb 23

Bug fixed

1. **Issue:** The customer was facing an error related to automation for community users after the post-upgrade.

Fix:

An error occurred while the community user performed automation post-upgrade.

It was observed that the customer had configured 11 objects as MOC and had an automation setup on the service appointment object. This triggered the automation by changing a field value via a community user, and the automation failed with an error related to access to quote and lead objects.

The community user could not access quotes and lead objects. We have checked the permissions of all the objects configured as MOC while triggering the automated message and now resolved the issue.

Patch: 1.72.30

Date: 10th Mar 23

Bug fixed

1. **Issue:** The customer was facing an error related to Client Customisation Request.

Fix:

An error occurred when the outgoing message was not reflected on the conversation component after being sent.

We observed that whenever the customer sent outgoing messages from the conversation component on the detail page, they were not reflected on the message lists. Also, when the incoming messages were received, they were later reflected in the message list along with the incoming message.

We have analyzed that the customer was using Custom Injection Service for messages, and a JS condition was missing for customization (for outgoing push topic).

We added the missing JS condition and observed that as soon as we send the outgoing message, it's reflected on the conversation component, and the issue is now resolved.

Patch: 1.72.35

Date: 24th May 23

Bug fixed

1. **Issue:** The customer was facing an error related to Sandbox upgrade

Fix:

An issue occurred while setting up and configuring the SMS-Magic application. It was throwing an error related to the Salesforce package installation.

It was observed that the installation failed when the customer was ordering query results based on the contact object's mobile or phone field. These fields were encrypted in the customer's organization and could not be used in order by clause in any query. This limitation was causing an installation failure of v1.68.14.

We have removed the order by clause in two queries referring to mobile and phone fields of the contact object and came up with a new build, v1.72.35, and installed it successfully.

Version 1.69

Patch: 1.69.38

Date: December 22, 2023

1. **Issue:** Compatibility issues with Salesforce prevented multiple components from loading

Fix:

Updates were made to the SMS Magic application to ensure compatibility with Salesforce. The application's integration methods and internal logic were adjusted to align with the new release requirements. Extensive testing confirmed that all components now function seamlessly within the updated Salesforce environment, effectively resolving the loading issues for users.

2. **Issue:** UI misalignment for "To know more about Plan details" text box

Fix:

The "To know more about Plan details" text box displayed improper UI alignment when accessing the plan details through the Converse setting. The expected layout was for the text box to be center aligned to improve readability and maintain consistency with design standards.

Adjustments were made to correct the alignment, ensuring that the "To know

more about Plan details” text box is now properly centered. These changes were rigorously tested, verifying that the UI displays correctly across various environments.

3. **Issue:** Emoticons not being added to components

Fix:

An issue was identified where emoticons (emojis) were not being added to components despite selection. This was due to a malfunction in the emoji insertion functionality within the application.

A resolution involved modifying the application’s code to correct the handling of emoji insertion into components. These changes addressed the malfunction and ensured the proper addition of emoticons.

4. **Issue:** “Type” Field Mislabeled in SMS History

Fix:

Incoming messages were incorrectly marked as “Outgoing” in the SMS History object. This was rectified by updating the default value of the ‘smagicinteract__Type__c’ field to “Incoming” for such messages to ensure they are represented accurately.

Patch: 1.69.36

Date: December 22, 2023

Issue: Intermittent Errors in Conversation Component Due to Inconsistent setComplianceData Method in Composer and Controller Files

Fix: The conversation component now operates more reliably and consistently, as new validation and null checks are introduced in the “setComplianceData” method. Previously, this component exhibited intermittent errors, originating from the *setComplianceData* method in both the *createComposerComponent.js* and *conversationComponentController.js* files not returning the expected values.

Patch: 1.69.35

Date: September 5, 2023

1. **Issue:** Utility Bar Conversation View Loading Issues in the Console App: Problems with Embedding and Contextual Conversation Functionality

Fix: The integration of the Conversation view within the Console Application utility bar is improved providing a smoother user experience. As you navigate through various contacts and leads within the Console app, the conversations now remain active and easily accessible, ensuring seamless transitions between different records without interruptions or loss of context. Earlier, the UI had loading issues with the Conversation view when transitioning from a conversation in Sales Home to a contact or lead record.

2. **Issue:** Records Count in List View with Campaign Filter Applied Erroneously Displayed Record Count as 0

Fix: Previously, when sending bulk campaigns in the Converse App with a campaign filter applied in the List view, the records count erroneously appeared as 0 despite the presence of records:

This data is sample set. The message will be sent to 0 recipients of campaign list when the conditions are met. Details.

This was due to an aggregate query limitation when a campaign filter was added. The update now intelligently hides the count display when it can incorrectly show 0, ensuring users do not get confused by a misleading message. Now the message appears as the following:

This data is sample set. The message will be sent to recipients of contact list when the conditions are met. Details.

Patch: 1.69.34

Date: 24th Aug 2023

Bug fixed

1. **Issue:** Issue with Admin User (No SMS Magic License Assigned) during User Activation/Deactivation.

Fix: Admin users without SMS Magic licenses were encountering errors when attempting to activate or deactivate users. The error prompted the admin user to grant read permissions to fields of the SMS Magic license object, even when the user being activated/deactivated did not possess SMS Magic licenses.

The root cause of the issue was identified as a permission-related challenge with the SMS Magic license object. Admin users lacking SMS Magic licenses were incorrectly prompted to grant read permissions to fields on the SMS Magic license object during user activation or deactivation.

The issue has been successfully resolved with the implementation of the following solution:

- **Permission Adjustment on License Object:** Permissions on the SMS Magic license object have been reviewed and adjusted to prevent unnecessary errors during user activation or deactivation for admin users without SMS Magic licenses. Admin users will no longer be prompted to grant read permissions to fields on the SMS Magic license object.

This patch ensures a seamless user activation and deactivation process for admin users without SMS Magic licenses, eliminating the reported error related to permissions on the SMS Magic license object. Users can now perform these actions without hindrance.

Patch: 1.69.21

Date: 25th Jan 22

Bug fixed

1. **Issue:** The default Sender ID functionality was not working as expected

Fix: The non-admin users were getting the error message on the Converse Home screen which was reported in case#00107408. (Previously it was reported for the 1.68 version). The organization object did not have permission for the field(s).

The read access in the CheckIsSandbox() method of FMAUtility.cls was checked. In this case, the user did not have access and was receiving an error on the Converse Home page.

Here, there was a read check for the PermissionSetAssignment object. If the user did not have view setup and configuration permissions, it was throwing an error for this check.

To solve this issue, removed the check for reading access on organization access for the PermissionSetAssignment object.

Patch: 1.69.23

Date: 13th Apr 22

Bug fixed

Issue: The client did not have a license to send an SMS from the SMS tab on Salesforce or even to access the Converse Desk tab of SMS-Magic

Fix: One of the SMS-Magic users who held a Power License had a problem when they wanted to send an SMS from the SMS tab on Salesforce or even to access the Converse Desk tab of SMS-Magic. They were getting an error message: "You don't have a license for this feature. This feature is part of Standard and Enterprise plans." This happened to only one user with the power license, and the rest of the power license holders were not experiencing this issue.

There was an inactive license record present for this user which was of Basic type. While checking for a license, the code was fetching that inactive license.

To solve this issue, in the getLicenseType() method of LicenseService, a filter of smagicinteract__status__c was added while querying licenses for users.

Patch: 1.69.24

Date: 27th Apr 22

Bug fixed

1. **Issue:** The customer requested customization that on the conversation component the conversation should be visible based on the mobile number

that they select.

Fix: Conversations related to the "To" number were visible on the conversation component on the record page. The Conversations were to be changed if the "To" mobile number was changed.

To fulfill this customization, an injection point was created for a custom injection service to which a selected phone number or SenderID was provided as a parameter. The data received from the custom injection service was used to display on the UI in the message list component.

Patch: 1.69.25

Date: 4th May 22

Bug fixed

Issue: Customer was not able to select a plan from converse settings as it was giving an error 'the list had no rows for assignment to SObject' while selecting a plan.

Fix: The user was unable to select the plan from Converse settings as it was giving the following error: "List had no rows for assignment to SObject"

□□ In the current version, we were fetching the active users whose profile name is 'System Administrator', which caused this issue.

To solve this issue, removed the Profile name criteria from GetActiveUserId() method.

Issue: The customer was receiving an error while adding a manual consent

Fix: The customer was facing an issue when creating manual consent.

Below is the error that the customer was receiving while creating manual consent:



For v1.68.23, the complianceNewConsentHelper.js file attribute was set to the Sender list.

To solve this issue, the profile name criteria were removed from GetActiveUser ID () method.

Patch: 1.69.26

Date: 18th May 22

Bug fixed

1. **Issue:** Record Owners Sender ID option was not available to the customer

Fix: The customer was creating a Converse App automation on the message flow

creation window and it was showing the Sender ID that was assigned to them. However, the option to select Record Owners Sender ID was not visible.

In the filterSenderID'sssFromSenderVisibility method, the Record-Owner Check was not added, hence this issue occurred.

To solve this issue, added the below condition in the filterSenderID's FromSenderVisibility method of SenderService Class:

```
SenderInfoList[i].SenderID != Constants.RECORD_OWNERS_SENDERID)
```

Patch: 1.69.27

Date: 3rd June 2022

Bug fixed

1. **Issue:** The customer was facing license issues for Guest User

Fix: The customer had reported that the automated messages were not getting triggered and they were receiving an error "Required License not assigned to send SMS or MMS". Also, the org-wide license was not supported.

To solve this issue, the 'IsFeatureAccessble' method was updated in the 'LicenseManagerUtility' class to support org-wide licenses for user types like community users, and guest users.

Patch: 1.69.28

Date: 12th Oct 2022

Improvement

1. **Issue:** The customer wanted to create a provision to change the owner of the 'License Type' records via 'protectedSettings' Visualforce Page page

Fix:

The "License Type" object of the managed package is protected via a trigger interface which makes it impossible to update the records via UI. The records of such protected objects can be created/updated only via Apex.

Sometimes the owner of the records of such protected objects is a ghost user. This causes problems in the expected functionality of the package.

To resolve the problem, the owner of such records needs to be set as one of the internal users of the target org, but that is not possible via UI, thus the need for the provision to update such records of protected objects occur.

Created provision via "protectedSettings" VF page to update the owner of the records of protected objects by passing the user id of the desired user.

Patch: 1.69.32

Date: 21st Dec 2022

Bug fixed

1. **Issue:** The customer was unable to add a merge field

Fix: Whenever the user was trying to create a Converse Template with the primary object as "Service Appointment" and the "Assigned Resource" selected in the merge field, the merge field dropdown "AppAssistInfoURL" field was not getting displayed.

The managed package code was on API version 47 (Metadata PAI version). The assigned resource was a standard Salesforce object. The field AppAssistInfoURL was only available on API version 51.0 and later. Hence, this field was not populating in the merge field list.

To resolve this, the MessageUtility.cls-meta.xml and Security.cls-meta.xml was updated to API version 55.

Patch: 1.69.35

Date: 5th Sept 2023

Bug fixed

1. **Issue:** The customer was facing an issue with the conversation view in utility bar

Fix: The customer was facing an issue with the conversation view in the utility bar. The conversation actions were opening in new window from the utility bar and not in a new tab.

To solve this issue, the conversation view was embedded in the utility bar for the Sales app with {{Enable Contextual Conversations}} value as false.

The following steps were followed to resolve the issue:

- Go to Sales Home and open any conversation.
- Open any contact/lead record and verify the opened conversation in the utility bar is not getting closed.

With this, it is now verified that all conversation actions are opening in a new tab and not in a new window in the utility bar.

2. **Issue:** The records count was showing 0 in the Converse app bulk campaign

Fix: The records count was showing 0 in the Converse app bulk campaign even if records were present in the list view.

- Create a list view having a campaign filter as shown in below image:



- Select the above list view while sending a bulk campaign and you can see the records count showing as 0:



This was happening because the aggregate query was returning the count as 0 if the campaign filter was added to the list view. This is actually a limitation from the Salesforce side.

To resolve this issue, the count was hidden if it was 0 and if records were present in the list view.

Version 1.68

Patch 1.68.75

Date: January 30, 2024

Bug

- **Issue** – Compatibility issues with Salesforce prevented multiple components from loading

Fix –

Updates were made to the SMS Magic application to ensure compatibility with Salesforce. The application's integration methods and internal logic were adjusted to align with the new release requirements. Extensive testing confirmed that all components now function seamlessly within the updated Salesforce environment, effectively resolving the loading issues for users.

Patch 1.68.74

Date: 18th Jan 2024

Bug

- **Issue** – The customer was facing issues with character count and emoticons while sending message template

Fix –

The customer was facing an issue with the emoticons while using them in conversations since the patch 1.68.73 was upgraded.

Also, they were facing issues with the character count, which used to be 1400 but it was cut down to 700. When special characters were used in the conversations, the character count was 700 earlier but now it was cut down to 350. Due to this, the customer was not able to use message templates.

The markup for the emoticons was not as per the JS function and it was referring to the elements from the markup which were not returning the

expected value of the user-selected emoticons. Also, in the composer component, the character count was fetched from the templateinfo object which is part of the composer data passed on from the parent component 'conversationComponent'. The value for the character count was hard coded to 700 in the class 'MessageComposerServerController'. Due to this, the customer was facing the character count and emoticons issues.

To resolve the emoticons issue, changed the markup for rendering the emoticons and changed the JS code referring to the emoticon that the user was selecting; to identify the correct emoticon. To resolve the character count issue, replaced the hardcoded value of 700 with the character count value that was set in the org.

Patch 1.68.72

Date: 17th Nov 2023

Bug

- **Issue** – The last incoming message time was not getting populated on Conversation on Incoming message.

Fix –

Users encountered an issue where the Last Incoming Time field was not getting populated on the conversation object while receiving an incoming message. This inconsistency affected the accuracy of conversation data and user visibility.

The root cause of the problem was identified as the presence of two lookup fields for a single object on the SMS History object. This setup was causing confusion and preventing the accurate population of the Last Incoming Time field on the conversation object.

The issue has been successfully resolved with the following solution:

- **Single Lookup Field Configuration:** The configuration of the SMS History object has been adjusted to have a single lookup field for a custom object. This eliminates the confusion caused by having two lookup fields for the same object on the SMS History object.

With this adjustment, users should now observe the correct population of the Last Incoming Time field on the conversation object when receiving incoming messages. The patch addresses the reported issue and ensures accurate and reliable data representation in the conversation history.

Patch 1.68.69

Date: 31st Oct 2023

Bug

- **Issue** – Component Error on Conversations Component in Utility Bar.

Fix –

Users were encountering a component error while using the Conversations component in the Utility bar.

The root cause of the problem was identified in the code, specifically in the Conversation Controller functionality. A null condition for the workspace API variable was missing, leading to the component error.



The issue has been successfully resolved with the following solution:

- **Null Condition Addition for Workspace API Variable:** A null condition has been added for the workspace API variable in the Conversation Controller functionality. This adjustment ensures that the code can handle scenarios where the workspace API variable may be null, preventing the occurrence of the component error.

With this modification, users will no longer encounter component errors when using the Conversations component in the Utility bar.

Patch 1.68.68

Date: 29th Aug 2023

Bug

- **Issue** – Text Messages were getting auto-deleted

Fix –

Several users reported an issue where, upon accessing a new contact record and promptly typing an SMS, the entered text was automatically deleted within the first few seconds. Subsequent attempts to type the SMS resulted in the text remaining intact. This behavior caused inconvenience for users, particularly when trying to send a message quickly.

The root cause of the problem was identified as a delay in the template loading and window data validation method. During this delay, while the user was typing a message, the method processed fetched data and inadvertently removed the message text value, regardless of the channel check. This issue was particularly noticeable when typing immediately upon accessing a new contact record.

The issue has been successfully resolved with the following solution:

- The logic in the template loading and window data validation method has been updated. The method now checks the current time against a valid 24-hour window. The message text will only be removed if the current time is outside of this window. This adjustment is specifically applicable only for non-SMS channels.

With these changes, users should no longer experience the automatic deletion

of typed SMS text within the first few seconds, providing a smoother and more reliable messaging experience. The patch addresses the reported issue and enhances the usability of the Conversations component.

Bug

- **Issue** – Event Record Creation on SMS Scheduling from “Single SMS” Button

Fix –

Users encountered an issue where an Event record was automatically created when scheduling an SMS using the “Single SMS” button from the opportunity/affiliation record detail page. The automatic creation of these events caused confusion and inconvenience for users who did not find them useful.

The root cause of the problem was identified in the functionality of the manage package code. When scheduling an SMS using the “Single SMS” button, the process involved creating an SMS schedule record, followed by the creation of an Event record with the subject “Schedule SMS.”

The issue has been successfully resolved with the following solution:

- **Event Creation Control:** The patch includes a modification to stop the automatic creation of an Event with the subject “Schedule SMS” when scheduling an SMS from the custom button. The creation of the Event record is now bypassed, addressing the user’s concern about unnecessary events being generated.
- **User Override Option:** To provide users with flexibility, a new option has been introduced. Users who do not want events to be created when scheduling messages, can skip the event with a static resource which serves as a user override, allowing for customization based on individual preferences.

With these changes, users can now schedule SMS without the automatic creation of events, providing a more streamlined and user-friendly experience. The patch addresses the reported issue and introduces a configurable option for users who prefer not to have events generated during SMS scheduling.

Patch 1.68.67

Date: 7th july 2023

Bug

- **Issue** – The customer faced an error related to the message object API name, showing an inconsistent letter casing for incoming and outgoing messages.

Fix –

An error occurred when the customer viewed the object name of Contact displaying differently for Incoming and Outgoing messages.

We observed that the issue was caused by using the lowercase method in the SMS-MMS Messaging class, resulting in the object being populated in lowercase.

We have implemented a fix using the capitalization method to show the object's first letter in capital. Changes have been made in the do lookup method of the consent trigger helper class to capitalize the object name while populating the object type in the consent confirmation message.

- **Issue** – The customer was facing an error related to the Apex CPU Time Limit while loading conversations.

Fix –

When the customer activates the Conversations component on the lead record page layout (in the activity section), an Apex CPU Time Limit error occurs after a few seconds.

We observed that the error is caused by many converse templates (more than 3900) created on the lead object, which exceeds the processing time limit.

We have implemented a provision to append additional clauses to the original query, order by and limit. This implementation limits the number of records fetched via the query, eliminating the time-exceeding issue and preventing the “Apex CPU Time Limit Exceeded” error.

Patch 1.68.65

Date: 24th May 2023

Bug

- **Issue** – The customer had a concern regarding the performance of the Converse App.

Fix –

An issue occurred while the customer tried to run the bulk campaigns, as the UI was not getting loaded when trying to execute the new bulk action.

We have observed that the customer extensively used bulk campaigns and has created more than 3800 bulk campaign actions till now. When this UI was loading, the build was querying all the existing bulk actions leading to a failure. The loader keeps loading this vast data and fails to fetch it.

We have analyzed and added an infinite scroll functionality to the UI actions. Initially, only 100 activities were shown. Upon scrolling next, the set of actions will be fetched and displayed on the UI, and the functionality continues until all the records are fetched. Also, we have created a provision to override the default value of 100 records, and this can be implemented by creating a static resource with an intended number of description values.

Patch 1.68.64

Date: 28th April 2023

Bug

- **Issue** – The customer had a concern regarding Salesforce legacy API

Fix –

The customer had a concern regarding Salesforce legacy API. They found “smagicinteract__QuickReply” was using 24.0 and 30.0 versions.

The Soap API versions 24.0 and 30.0 are deprecated. These API versions have been used in our managed package code.

To resolve this, the Soap API version was changed to 54.0.

- **Issue** – The converse package upgrade was failing due to field encryption

Fix –

The customer had enabled field encryption in the target org. When they were trying to upgrade the existing version (1.48.15) to the target version (1.68.55), it was failing. This enforced certain limitations regarding the use of the encrypted fields in the SOQL queries. In this scenario, the field in the question was ‘smagicinteract__PhoneNumber__c’ of SMS History object. This field was used in the “GROUP BY” clause in the ConsentService.cls class. Due to this the apex classes compilation step was failing which was executed by SF during package installation.

To resolve the issue, the code for ConsentService.cls was updated. The query was made dynamic such that the query was declared as a string and later executed via the “Database.query()” method. Also, we ran ‘DescribeFieldResult’ on the SMS History object and checked whether the field was encrypted. Based on the result the query execution was skipped if the field was found to be encrypted.

Patch 1.68.61

Date: 22nd Mar 2023

Bug

- **Issue** – Requirement to add more details in the consent edit page search.

Fix –

The customer had raised a requirement related to the consent object as they had multiple contacts with the same name and different numbers and accounts in their organization. As per the customer’s workflow, when they created a manual consent record, they searched the record by name and ended up with a list of names without any number or related data. This made identifying the exact contact record complex and raised a requirement to show more details on the result list.

We have observed that this was the package limitation and was fetching only the names of the records.

We have analyzed the customer requirement and updated the build, fetching the mobile phone field values configured in MOC. The required record now fetches these mobile phone field values and displays them on the new consent creation page. When two records with similar names are present, the search result shows their respective name and phone/mobile number.

Patch 1.68.50

Date: 22nd Sept 2022

Improvement

- **Issue** – The user is facing an issue with the global and manual opt-out content type

Fix –

The user faced an issue operating with the global and manual opt-out based on content types. Countries with only one-way communication cannot opt-out using keywords. In such a scenario, they go for manual opt-out, enabling the “Content-type” and creating the manual consent records. It is mandatory to add a content type, as the global opt-out was not possible.

We have observed that the ‘Default’ content type is unavailable for selection on the UI while creating manual consent for the record (when the content type is configured as one of the compliance parameters).

We have updated the compliance service apex class query to include the ‘Default’ content type. Currently, the “Default” content-type option is available on UI (manual consent creation) while creating manual consent.

- **Issue** – The user was unable to search information object records on the bulk interface.

Fix –

The user gets an error message at the bottom of the page when searching for information object records on the bulk interface. It throws an error and assists the user in refreshing the page.

We have observed that the data service defined for the bulk sms component controller for search information object record failed. The reference to a service component was missing, causing the issue.

We have analyzed the missing reference of the service component and added it. The functionality worked well as expected, and the issue is now resolved.

Patch 1.68.49

Date: 21st Sept 2022

Improvement

- **Issue** – The user is facing an issue with the global and manual opt-out content type

Fix –

The user faced an issue operating with the global and manual opt-out based on Content types. Countries with only one-way communication cannot opt-out using keywords. In such a scenario, they go for manual opt-out, enabling the “Content-type” and creating the manual consent records. It is mandatory to add a Content type, as the global opt-out was not possible.

We have observed that the ‘Default’ content type is unavailable for selection on the UI while creating manual consent for the record (when the content type is configured as one of the compliance parameters).

We have updated the compliance service apex class query to include the ‘Default’ content type. Currently, the “Default” content-type option is available on UI (manual consent creation) while creating manual consent.

- **Issue** – The user was unable to search information object records on the bulk interface.

Fix –

The user gets an error message at the bottom of the page when searching for information object records on the bulk interface. It throws an error and assists the user in refreshing the page.

We have observed that the data service defined for the bulk sms component controller for search information object record failed. The reference to a service component was missing, causing the issue.

We have analyzed the missing reference of the service component and added it. The functionality worked well as expected, and the issue is now resolved.

- **Issue** – Display label for the invocable method and its fields while setting up automation for consent creation/update

Fix –

While setting up automation for creating/updating consent (via invocable method), sometimes users add the consent status value slightly different than expected. (e.g., the expected value is “Opt-Out,” but the entered value is “Opt out”). Even this slight difference could create a problem, and consent records were not created/updated.

We have added a label for this field which will be displayed on UI (in PB or Flow) while the user enters the value for this field. This way, the users will know the expected values for this field and will be less likely to make mistakes while entering the required value.



We have made the labels visible while setting up automation for creating/updating consent records. It displays the acceptable values; the user would know the value to be set for the field and is less likely to make mistakes in setting the acceptable value.

- **Issue** – Renaming the Error Logs to App System Logs

Fix –

This is the new improvement to capture all the errors during data syncing between SF org and the platform.

All errors during sync between SF org and platform (and vice-versa) will be captured and further used for debugging the issue.

This improvement involves the following categories:

1. All license failures should be logged in the App system logs.
 - a. Whenever any license is allocated or removed from user management and the operation fails due to some error, the error is logged in the App system logs object. Earlier, it was logged in the error log object and rolled back the license assignment in Salesforce.
2. All platform events are to be logged in App system logs.
 - a. There are various event types logged in a specific method of classes. The application configuration service will create an error log entry if such request types are not found.
3. All manual syncs are to be logged in App system logs.
 - a. Whenever we perform a manual sync, we get the converse subscription information method from the FMA Utility class to fetch updated plan details. Later it also calls the feature master and populates data methods from the license manager utility to fetch and update feature master data. Here we can insert a system log record upon receiving a server response.

Patch 1.68.48

Date: 21st Sept 2022

Improvement

- **Issue** – The user was facing an issue with the global and manual opt-out content type

Fix –

The user faced an issue operating with the global and manual opt-out based on Content types. Countries with only one-way communication cannot opt-out using keywords. In such a scenario, they go for manual opt-out, enabling the “Content-type” and creating the manual consent records. It is mandatory to add a Content type, as the global opt-out was not possible.

We have observed that the 'Default' content type was unavailable for selection on the UI while creating manual consent for the record (when the content type was configured as one of the compliance parameters).

We have updated the compliance service apex class query to include the 'Default' content type. Currently, the "Default" content-type option is available on UI (manual consent creation) while creating manual consent.

- **Issue** – The user was unable to search information object records on the bulk interface.

Fix –

The user was getting an error message at the bottom of the page while searching for information object records on the bulk interface. It was throwing an error.

We have observed that the data service defined for the bulk sms component controller for search information object record failed. The reference to a service component was missing, causing the issue.

We have analyzed the missing reference of the service component and added it. The functionality worked well as expected, and the issue is now resolved.

- **Issue** – The user wanted to display a label for the invocable method and its fields while setting up automation for consent creation/update.

Fix –

While setting up automation for creating/updating consent (via invocable method), sometimes users add the consent status value slightly different than expected. (e.g., the expected value is "Opt-Out," but the entered value is "Opt out"). Even this slight difference could create a problem, and consent records are not created/updated.

We have added a label for this field which will be displayed on UI (in PB or Flow) while the user enters the value for this field. This way, the users will know the expected values for this field and will be less likely to make mistakes while entering the required value.



We have made the labels visible while setting up automation for creating/updating consent records. It displays the acceptable values; the user would know the value to be set for the field and is less likely to make mistakes in setting the acceptable value.

- **Issue** – Renaming the Error Logs to App System Logs

Fix –

We have implemented a new feature to capture all the errors during data syncing between SF org and the platform.

All errors during sync between SF org and platform (and vice-versa) will be captured and further used for debugging the issue.

This improvement involves the following categories:

1. All license failures should be logged in the App system logs.
 - a. Whenever any license is allocated or removed from user management and the operation fails due to some error, the error is logged in the App system logs object. Earlier, it was logged in the error log object and rolled back the license assignment in Salesforce.
2. All platform events are to be logged in App system logs.
 - a. There are various event types logged in a specific method of classes. The application configuration service will create an error log entry if such request types are not found.
3. All manual syncs are to be logged in App system logs.
 - a. Whenever we perform a manual sync, we get the converse subscription information method from the FMA Utility class to fetch updated plan details. Later it also calls the feature master and populates data methods from the license manager utility to fetch and update feature master data. Here we can insert a system log record upon receiving a server response.

Patch 1.68.47

Date: 8th Sept 2022

Improvement

- **Issue** – The Guest user was unable to access records of the object “License Feature Matrix” and was unable to send SMS or MMS

Fix –

The issue occurred when the guest user was unable to send SMS or MMS and could not access records of the object “License Feature Matrix.”

The objects “License Type” and “Feature” are protected via a trigger handler, making it impossible to change the records outside the managed package. These records were owned by a specific user (ghost user). Due to this, the sharing rules created for the records of these objects were unable to be accessed by the guest users. Thus, the functionality of validating the feature access for the user was failing and created the sms history record via the guest user with a “Rejected” status along with an error message related to the license not being assigned (though the guest user has the appropriate license).

We have updated the global settings component and created a provision for which SMS-Magic people can update the ownership of the records to one of the valid users in the org. The records’ ownership has been updated to the new user.

Patch 1.68.46

Date: 29th Aug 2022

Improvement

- **Issue** – Showing the confirmation popup when the user clicks on the “Send” or “Schedule” button on the bulk interface.

Fix –

As a part of the improvement, user wanted to update the confirmation pop-up message. The existing confirmation popup shows the information about selected recipients or the selected list view.



The newly created confirmation modal was applied with the click of the “Send” or “Schedule” button. It shows relevant information based on whether it is a bulk campaign or selected records from a list view, or a single record.



Implemented the confirmation popup on all three interfaces, i.e., converse bulk interface, send bulk SMS interface, and community sends bulk SMS interface.

- **Issue** – The user could not access Converse App Task, the UI in global navigation was unable to load.

Fix –

The user faced an issue loading the Converse app UI in the global navigation. It was showing an internal server error with a specific error id.

We have observed more than 40 lac records on converse app task objects in the customer’s org. Because of this huge data, the query responsible for the above mentioned functionality was failing, generating an error message from the Salesforce side.

We have added a ‘last seven days’ filter by ‘create date’ and now the query clause searches records from only the previous seven days.

Patch 1.68.45

Date: 23rd Aug 2022

Bug fixed

- **Issue** – The Non-SMS Magic user got an issue when opening the contact detail page.

Fix –

An error occurred when the Non-SMS Magic user (Did not have the license and

permission of SMS Magic) opened the contact detail page. It stated that the "object name message config does not have permission for fields (read)." As per the defined functionality, the non-SMS Magic user should not receive this error.

We have analyzed that it threw the error from the conversation header component. During the optimization process, it did not include the required data for the component in the common apex call from the parent component. Thus, when the component is loaded, though the parent component displays a license-related error, the apex calls for the child component are still executed, resulting in an error.

Removed apex calls from the component, included the required data to be fetched from the parent component and passed further.

- **Issue** – The error message generated from the custom injection class should be displayed on bulk UI.

Fix –

The user could not see the message while using an injection point with the custom class to enable their custom logic. When they validate in the custom class, the required parameters are not met, and the custom class throws an error. That error message should be displayed on the UI as an error notification.

Analyzed that the error handling code was different for one of the methods in which the injection point was called. Due to this, it did not pass the error generated from the custom class to the package apex class.

We have updated the package apex class's error handling method so that the custom class's error will be passed on to the packaged apex class and to the component UI. The custom validate message generated from the custom injection class is now displayed on bulk UI.

- **Issue** – The user was unable to schedule any new automated SMS jobs in Salesforce.

Fix –

The issue occurred when the user scheduled new automated SMS jobs in Salesforce. The error states that the message suggests that we've reached our limit of scheduled batch jobs. The SF default is 100; however, the account was permitted up to 125.

We have observed that the users were not receiving this error when trying to schedule non-SMS jobs (e.g., scheduled emails).

Checked the specific static resource in the target org and updated the description field value limit of the static resource for the scheduled apex jobs. If the static resource is present and the description value of this static resource is greater than 100, then the value from the description field is considered a limit for scheduled apex jobs in the system. This will

override the default of 100, which was considered in the package code.

Patch 1.68.44

Date: 19th Aug 2022

Bug fixed

- **Issue** – The Non-SMS Magic user got an issue when opening the contact detail page.

Fix –

An error occurred when the Non-SMS Magic user (Did not have the license and permission of SMS Magic) opened the contact detail page. It stated that the “object name message config does not have permission for fields (read).” As per the defined functionality, the non-SMS Magic user should not receive this error.

We have analyzed that it threw the error from the conversation header component. During the optimization process, it did not include the required data for the component in the common apex call from the parent component. Thus, when the component is loaded, though the parent component displays a license-related error, the apex calls for the child component are still executed, resulting in an error.

Removed apex calls from the component, included the required data to be fetched from the parent component and passed further.

- **Issue** – The error message generated from the custom injection class should be displayed on bulk UI.

Fix –

The user could not see the message while using an injection point with the custom class to enable their custom logic. When they validate in the custom class, the required parameters are not met, and the custom class throws an error. The user wanted that error message to be displayed on the UI as an error notification.

Analyzed that the error handling code was different for one of the methods in which the injection point was called. Due to this, it did not pass the error generated from the custom class to the package apex class.

We have updated the package apex class’s error handling method so that the custom class’s error will be passed on to the packaged apex class and to the component UI. The custom validate message generated from the custom injection class is now displayed on bulk UI.

- **Issue** – The user was unable to schedule any new automated SMS jobs in Salesforce.

Fix –

The issue occurred when the user scheduled new automated SMS jobs in Salesforce. The error states that the message suggests that we've reached our limit of scheduled batch jobs. The SF default is 100; however, the account was permitted up to 125.

We have observed that the users were not receiving this error when trying to schedule non-SMS jobs (e.g., scheduled emails).

Checked the specific static resource in the target org and updated the description field value limit of the static resource for the scheduled apex jobs. If the static resource is present and the description value of this static resource is greater than 100, then the value from the description field is considered a limit for scheduled apex jobs in the system. This will override the default of 100, which was considered in the package code.

Patch 1.68.43

Date: 4th Aug 2022

Bug fixed

- **Issue** – The Non-SMS Magic user got an error while opening the contact detail page.

Fix –

An error occurred when the Non-SMS Magic user (Did not have the license and permission of SMS Magic) opened the contact detail page. It stated that the "object name message config does not have permission for fields (read)." As per the defined functionality, the non-SMS Magic user should not receive this error.

We have analyzed that it threw the error from the conversation header component. During the optimization process, it did not include the required data for the component in the common apex call from the parent component. Thus, when the component is loaded, though the parent component displays a license-related error, the apex calls for the child component are still executed, resulting in an error.

Removed apex calls from the component, included the required data to be fetched from the parent component and passed further.

- **Issue** – The user wanted the error message generated from the custom injection class to be displayed on bulk UI.

Fix –

The user could not see the message while using an injection point with the custom class to enable their custom logic. When they validate in the custom class, the required parameters are not met, and the custom class throws an error. That error message should be displayed on the UI as an error notification.

Analyzed that the error handling code was different for one of the methods in which the injection point was called. Due to this, it did not pass the error generated from the custom class to the package apex class.

We have updated the package apex class's error handling method so that the custom class's error will be passed on to the packaged apex class and to the component UI. The custom validate message generated from the custom injection class is now displayed on bulk UI.

Patch 1.68.42

Date: 2nd Aug 2022

Bug fixed

- **Issue** – The Non-SMS Magic user got an issue while opening the contact detail page.

Fix –

An error occurred when the Non-SMS Magic user (Did not have the license and permission of SMS Magic) opened the contact detail page. It stated that the "object name message config does not have permission for fields (read)." As per the defined functionality, the non-SMS Magic user should not receive this error.

We have analyzed that it threw the error from the conversation header component. During the optimization process, it did not include the required data for the component in the common apex call from the parent component. Thus, when the component is loaded, though the parent component displays a license-related error, the apex calls for the child component are still executed, resulting in an error.

Removed apex calls from the component, included the required data to be fetched from the parent component and passed further.

- **Issue** – The error message generated from the custom injection class should be displayed on bulk UI.

Fix –

The user could not see the message while using an injection point with the custom class to enable their custom logic. When they were validating the message in the custom class, the required parameters were not getting met, so the custom class threw an error.

Analyzed that the error handling code was different for one of the methods in which the injection point was called. Due to this, it did not pass the error generated from the custom class to the package apex class.

We have updated the package apex class's error handling method so that the custom class's error will be passed on to the packaged apex class and to the component UI. The custom validate message generated from the custom injection

class is now displayed on bulk UI.

Patch 1.68.41

Date: 21st July 2022

Bug fixed

- **Issue** – The user was getting an error while attempting to create a New Consent record manually

Fix –

The users who have non-admin access were unable to create Consent Records manually. It was throwing an error stating that the “Procedure was only valid for System.DMLException and System.MailException”, as shown below:



Analyzed the debug logs and observed that a few components didn't have the update permission. We have provided the update permission to the specific components at the user level and resolved the issue.

Patch 1.68.40

Date: 11th July 2022

Bug fixed

- **Issue** – The user was getting a component error when they opened the Home page.

Fix –

An error occurred when the user tried to open the Home page. This issue occurred when they were in classic and were switching to lightning on the Home page.



We have analyzed the missing validation for the recipients list component instance in the conversation view helper class and added a validation. We have added the missing validation component, and the issue is now resolved.

- **Issue** – The user wanted to enable a custom injection point (Sender ID & Template) for Bulk interface

Fix –

The user faced an error in the Conversation tab on the detail page when they upgraded the application to the latest version 1.68.37 from 1.60.75.

Error: “SObject row was retrieved via SOQL without querying the requested field: smagicinteract__SMS_SenderId__c.smagicinteract__Channel_Code__c”
Enabled injection point for bulk interface based on the flags for SenderId

and Template.

The issue related to retrieving SObject rows via SOQL without querying the request in the Conversation tab on the detail page after upgrading to version 1.60.75 is now resolved.

Patch 1.68.37

Date: 19th May 2022

Bug fixed

- **Issue** – The user was facing an invalid field account name issue while creating the case/opportunity under the error logs.

Fix –

An error occurred when the user configured “Case” as a MOC. The error states: “invalid field Account.Name for case/opportunity” under the error logs.

As per the functionality, when any incoming message record is created, the package sends an email to notify the user about the incoming message. The subject of that email contains the name of the record. This value is resolved from the name field configuration in the MOC for the record. In this case, the Account name was not resolved. But the system only parses the static values and cannot resolve relationships. Thus when the system encounters the relationship data instead of static value, a corresponding error log is generated.

We have updated the receipt name map method of the SMS utility class and resolved the field name as a relationship.

We have updated the receipt name map method of the SMS utility class and resolved the field name as a relationship. No error logs were created upon receiving the incoming message, and also received the notification email with an account name in the subject line.

- **Issue** – The user was facing an issue with the SMS Format while adding a Short link.

Fix –

An error occurred when the user tried to send messages with a short link, the outgoing message showed an improper format of the message.

Outgoing messages containing shortened links added via the “++ “prefix showed “++” in the shortened link.

We have updated the functionality to check the condition for a line break in the shortening service.

- **Issue** – The user was facing an issue with the merge field where the time format was not correct.

Fix –

An error occurred when the user sent an SMS from the “Student application” object. They passed a merge field for a time, but the time format was incorrect.

In the specific record, the interview time field was 1:00 PM, but in the SMS text, it showed as “13:00:00.000Z,” which was not the correct format. We have analyzed the dateTime string converter in the system, which calls to convert the time field and was not working as expected.

We have added an internal class in the SMS Utility class to convert the time field, which converts the time value into a user-readable value as displayed on the UI. e.g., 1:00 PM.

Currently, the time value in the final message text is in a human-readable format (e.g., if the time value is set at 1:00 PM, in the message text, it should be populated accordingly).

Patch 1.68.35

Date: 8th March 2022

Bug fixed

- **Issue** – The customer was facing the Converse app bulk action issue; it was showing the incorrect date on the UI for the scheduled jobs.

Fix –

The user had a use case to send bulk recurring campaigns weekly. When scheduling a campaign for any future date, for example, a recurring campaign starting from 22nd February 2022 to 8th April every Tuesday at 7 pm, 22nd was the date selected as the start date but on saving the details section and scheduled jobs, the start date was shown as 15th February 2022, i.e. one week before the actual scheduled date.



The user also wanted to pause these scheduled bulk campaigns on the Easter break on 17th April and then restart on 19th April. The user tried to pause a campaign but it still triggered messages.

The program action controller was analyzed (action next run map functionality) and it was observed that it was fetching only the next fire time; hence, it was showing an incorrect date on the UI.

To solve this issue, the start date-time conditions were modified in the program action controller, which then resolved the error and displayed the correct date on the UI.

- **Issue** – SMS was not getting sent via Bulk Button, and no history was getting created at the record level.

Fix –

The user was trying to send SMS to 150-200 customers of Lead's list view via Bulk SMS Button. Upon clicking the send button, the message "SMS sent successfully" was appearing, but SMS History was not getting created at the record level or on the SMS History object.

Analyzed the debug logs and found that one consent record had empty values for compliance status and external key, causing the error.

To resolve this issue, added value for Compliance status as 'Expired' for this record. Also, the consent service method added the null checks for name and compliance status fields, which exempted the issue, causing the "Attempt to dereference a null object."

Patch 1.68.34

Date: 10th Feb 2022

Enhancement

- **Issue** – Issue with the SMS template when the user had a self-lookup relationship

Fix –

The user uses a third-party Mortgage App in Salesforce org. It has a Mortgage's Contact object (Field Name: Referred By), which has a self-lookup relationship with the contact object where, the mortgage's contact is the parent object, and a contact is a child object.

The user had set up the flows to send the messages from the Child record to the Parent Contact record. When a criteria is met on the child record, it created the SMS history record on the Parent Contact record. The SMS template was created on the Contact object using the merge fields of Parent and Child Contact records.

The user faced an issue while sending these outgoing SMS; the SMS template could not fetch the information from the Child's contact record.

The user had checked this with the SMS 360 App and confirmed that this feature is already implemented in their managed package, where the SMS template can fetch the Parent and Child Contact record information while sending automated SMS. They have a related object ID field on the SMS history object, which can be used to pass the ID value of the 2nd Contact lookup field.

The user required customization to achieve this use case as they needed to address more than 15 customers and needed the functionality to be added to the package.

As per the enhancement request, the user configured a self-lookup on the object, i.e., contact (field name – Referred By). As per the analysis, the

configured Flows on the Contact object acts on contact and sends a message referred by a contact. The user used the template for sending msg and merging fields of both primary contacts and was referred by a contact. But as they are adding lookups of referred by a contact, fields were not resolved correctly in the template for both contacts.

As per the user requirement, we have created a custom field for related objects on the Converse App Task object and populated the value of this field (record Id) while configuring the flows for Converse App Task. The process first checks whether the field is present, then checks if the value is present in this field, and then uses the lookup and resolves the merge fields.

The enhancement related to the self-lookup relationship for the SMS template is now implemented.

Patch 1.68.33

Date: 12th Jan 2022

Bug fixed

Issue – The non-admin users of the customer were getting an error on the Converse home page

Fix– When a non-admin user of the customer was logging in on Salesforce and navigating to the Converse Home page, they were getting the following error: 'Organization object does not have permission for field(s) Read'

In the checkIsSandbox() method of FMAUtility.cls the 'Read' access was being checked. And if the user was not having the 'Read' access, they were getting the said error on the Converse Home page.

To solve this issue, the checkIsSandbox() method of FMAUtility.cls the 'Read' access was removed.

Patch 1.68.31

Date: 7th Jan 2022

Bug fixed

1.Issue -Non-admin users were getting an error on the converse home page

Fix–

When the non-admin user was trying to turn off the notification, it was saying, "Your plan details are changed. Please click here to get the updated plan details". And when the non-admin users tried to click on "Click Here," no operation took place.

The pop-up was triggered only when the user had customized application permission. Also, the check for read access on organization access was removed.

The issue related to the converse home page for the non-admin users is now resolved.

Patch 1.68.30

Date: 23rd Dec 2021

Bug fixed

1. **Issue** – The guest user of the customer was not able to send messages as the required license was not assigned but an org-wide license was present

Fix- Guest users should be able to send messages even if the required license was not assigned but an org-wide license was present with a feature to send messages. However, the guest user of the customer was getting a license error and was not able to send messages even after the org-wide license was present. Messages were failing with an error- 'You don't seem to have the required license to send SMS or MMS'.

One 'if condition' was present in the isFeatureAccessibleForUsers() method of LicenseManagerUtility, where only if any license was assigned to a user, the features of the org-wide license were getting checked. If any license was not assigned to the user then this 'if condition' was not pulling any features of the org-wide license.

To solve this issue, added condition if orgWideFeaturesSet is present, and if current userType is present in category field of License Type or category field is empty then add those features to the user.

2. **Issue** – The non-admin users of the customer were getting a component error while they were navigating to the contact record.

Fix- In the getConversations method of LightningUtilityMessageNotification.helper, an undefined value was found, due to which this issue was occurring.

To fix this issue, added one condition (not null and isValid()) for component.find method in helper class.



3. **Issue** – SMS-Magic mobile users were able to send SMS to leads even before obtaining their consent.

Fix- SMS-Magic mobile users were able to send SMS to leads even before obtaining their consent.

The customer had created a test user and provided the consent for testing purposes. The customer was using CustomConsentService class to validate consents. In SendSMSTriggerHelper.sendSMSBeforeTrigger(), the ConsentService.getConsentValidatedRecords() method was called to fail records on the basis of consent. The injection point was provided in this method. Messages were getting failed in the custom class and were returning to the

list mobileApplicationSMSList in SendSMSTriggerHelper. As CustomConsentService was used, the list was sent and processed in a serialized form. In the process of serialization and deserialization, the reference was not followed and mobileApplicationSMSList was treated as a different list from smsList which is the main list. This main list was not getting updated.

To solve this issue, the main list was updated as per the updated mobileApplicationSMSList in SendSMSTriggerHelper.sendSMSBeforeTrigger().

Patch 1.68.29

Date: 14th Dec 21

Improvement

1. **Issue:** Customer wanted to exceed the limit of objects to be added to MOC from 10 to more.

Fix:

The user had a requirement to text to more than 10 objects. While trying to add the 11th object on MOC, they were getting an error that the limit is exceeded.

We had a limitation of Max 10 MOC records in converse settings. But as per the user requirement, we have addressed the below solution details:

1. Added configurable Static Resource with the name as- SMIExceedMOCLimit.
2. The current value > 10 was modified and allowed the user to add more than 10 MOC objects but not more than 20 Objects in MOC, without any error.
3. If the user wants to allow 20 available objects in MOC then they need to disable the Add Object button message should be displayed like 'Max limit of 20 objects reached. Please see attached screenshot.



The requirement to add more than 10 objects to the MOC is now attained, and the error of exceeded limit is now resolved.

2. **Issue:** Custom Map integration created by SMS Magic was not working

Fix:

The user was facing an error while using the "Send Bulk SMS" button on the maps app.

Analyzed the issue and found out that it occurred because the package code tried to trim the list of IDs, assuming that the list has square brackets at the first and last positions. But in the real scenario, we observed that the square brackets were not present; thus, the trim function was causing the issue.

Modified the substring function and eliminated the truncating effect for the first and last ID. Used the substring function on the 'idList' string only if square brackets were present in the string.

3.Issue: The user was facing Bulk SMS Component Error and Bulk button issue on LWC.

Fix:

When the user opened a custom widget and clicked on the Send SMS button, no list was shown, and no record was selected.

It was observed that after selecting a record from the listview to send Bulk SMS, sometimes the records were passed without the '[' ']' brackets. Also, it was observed that as the '[' & ']' brackets were unavailable, removing characters from record ids caused the issue. We analyzed and removed the first & last characters from the array.

The normal bulk SMS from any List view of any object and the bulk button issue on LWC is now resolved.

4.Issue: The customer was facing an issue with the consent component and getting an error message when there is no phone number on the record

Fix:

An error occurred when the phone number was not present on the record. Error message: "No phone numbers are available on this record."



If the user tried to add a consent component to the record on the same record, an error was prompted: "You do not have a License that provides access to this feature. Raise a request to your Admin by clicking the 'Request For License' button below if you need this feature."

Analyzed that if the phone number was not present on any record, then from the APEX method, an exception was thrown, which was shown as a notification error on the component. Subsequently, the component UI was hidden, showing only the access denied UI.

Updated the internal class with two new attributes, which use a return datatype for the apex method class. The exception thrown from the apex method was removed, and updated the new attributes with the error message. In the component, this error message is referenced to identify if there is an issue fetching the data. An error message is displayed on the UI instead of showing an error notification.

The issue related to the consent component throwing an error when observed no phone number on the record is now resolved.

Patch 1.68.28

Date: 1st Dec 21

Improvement

1. **Improvement:** The user was unwilling to assign "Customize Application" permission to Non-Admin users.

Fix:

The user wanted Non-Admin licensed users to view and make changes in Converse settings, but they were not ready to assign "Customize Application" permission to these users.

Implemented the solution by creating one static resource and a new method in the common utility controller. Checked the preferred access for the present resource in the org and made the access flags available.

The functionality related to the Non-Admin licensed users viewing and making changes in Converse settings and not assigning the "Customize Application" permission to these users is now addressed.

Patch 1.68.27

Date: 15th Nov 21

Bug fixed

1. **Issue:** Customer was facing an error while creating consent record manually

Fix: The customer was getting an error '**Invalid Phone Field**' while creating the consent record manually. The customer tried to create a consent record for the Case Role object and the phone field which the customer had selected was a lookup field that was fetching data from Accounts Object.

The phone field was added under the Message Object Configuration and the user was having all the required permissions still they were getting the same error.

Created a custom object named '**Case Role**' and a lookup field that was fetching phone numbers from Accounts Object. Customer was getting an error '**Invalid Phone Field**' while they were adding the object under Message Object Configuration and trying to create the consent.

The same issue was occurring for the standard objects as well.

In the Phonenummer() method of the ComplianceService.cls class, the phone field was not getting resolved for lookup values.

Created similar configurations as mentioned in the description and tried creating consents manually. It created consent records successfully.

To Solve this issue Used the existing method SMSUtility.resolveFieldValue() to resolve the lookup values.

2. **Issue:** Many users were complaining that they were seeing SMS messages in their conversation view, while in 'My Inbox' that were not for them.

Fix: Whenever a push topic subscription fails, An alternative method called get Latest Conversations After Given Time () in Conversation Service. cls from where the conversation records were fetched. This query did not have a filter for the owner.

To solve this issue, added the Owner ID filter in the query while fetching conversation records.

3. **Issue:** All the mobile numbers were not showing when tried to send SMS from Send Bulk SMS Button

Fix: All the numbers were present in the conversation lightning component. However, the numbers were not showing when the user was trying to send SMS from the Send Bulk SMS Button. After debugging the issue, it was analyzed that in LightningBulkSMSController the variable Allowable_Phone_Fields was initialized to 3. Hence it was showing only 3 phone numbers on the SendBulkSMS UI.

To solve this issue removed the Allowable_Phone_Fields variable from LightningBulkSMSController wherever it was used and populated the number of phone fields on SendBulkSMS UI which had been configured in the Message Object Configuration.

Patch: 1.68.26

Date: 3rd Nov 21

Bug fixed

1. **Issue:** Customer was facing Component Error.

Fix: The user had reported the issue, which was not regular but intermittent, and occurred when the user opened any contact record. It couldn't capture any logs as this was an intermittent issue.

The user was having permissions as well as licenses assigned, still was facing this issue.

Component Error occurred because push topic subscription was not successful for the affected user of the customer.

To solve this issue, added the Null check-in 'ProcessMessages' method of conversationComponentController.

Below is the link attached to the chatter and the component error that the customer was facing:

https://docs.google.com/spreadsheets/d/16EQVM_MDKgvsbB0uD9a94GZn031518UjjzzJg6XxZgE/edit#gid=0

Patch: 1.68.25

Date: 21st Oct 21

Bug fixed

1. **Issue:** The user was facing an issue where the converse home was not loading after upgrading Converse App from version 1.54 to 1.68.23 version

Fix:

The user was facing an issue while loading the converse home page after upgrading the Converse App from version 1.54 to 1.68.23 version. This issue occurred after upgrading the PROD account and syncing 1k+ licenses.

The debug logs in the analytics controller class were analyzed and active users' select id, Name, and ManagerId were fetched from the user and more than 50000 records were returned. The fatal error was observed with limited system exception with too many query rows: 50001.

To solve this issue, queried the active users and license on the license object with the select id, name, user component, and manager id from the license component with active status and active users. Configured these parameters and loaded the converse home with accurate data.

Patch: 1.68.24

Date: 18th Oct 21

Bug fixed

1. **Issue:** Health cloud instance was facing issues with a conversation on Referral object

Fix: The customer was getting the following error while adding Account as an object from Message Object Configuration.

1. The failed update was: First exception on row 0 with ID 0IF8s0000001ddGAA; first error: INVALID_FIELD, No such column 'CareProgramEnrollee_c' on entity 'smagicinteratsmsMagic'. The team was attempting to use a custom field, to be sure to append the '_c' after the custom field name. Please reference your WSDL or the describe call for the appropriate names.: [Query]
2. Also, when the customer was trying to send the SMS from the Referrals object through the conversation component was getting the below error –

Object Name Referral did not have a field(s) Owner ID.

There was a cloned permission set created and assigned to the user.

The customer was using HealthCloud instance org.

The object CareProgramEnrollee_c did not contain the Owner ID field. It was a master-detail relationship where the customer got the Owner ID from the master object. In this case, a condition field. Is cascade delete) was added

on fields of this object. If the object was a Master-detailed relationship, it returned true and the Owner ID of the parent object was used. But for some objects, including 'Careprogramenrollee_c', it was returning false and we were not able to add parent Owner ID.

To solve this issue, added one flag Owner ID Present. It's set false if the object doesn't contain the Owner ID field and true if it contains Owner ID and if the 'Is Cascade Delete' is also true. Later we used this flag to remove the Owner ID field if it's not in an object from the query where we're getting the error.

Patch: 1.68.23

Date: 13th Oct 21

Bug fixed

1. **Issue:** The customer was not able to reply to his customers on WhatsApp during the 24hour session.

Fix: When a reply was sent from our mobile app to a WhatsApp message it was not coming through and giving the error 'WhatsApp 24 hr active window is closed for this recipient'. You can send a message using a registered template only'. But a response from Salesforce and a reply were coming through.

For sources 1124, 1150, 1105, 1495, 4001, the 24-hour window was being validated by using the checkForMultiChannelSupport() method which was called from SendSMSTriggerHelper.cls in BEFORE context. From this method, the getQueryResults() method was called to query on SMS History Object to fetch incoming records within the last 24 hrs. Here '**PrimaryLookupField**' was used in the 'Where' clause. There was an object lookup in case the SMS record contained any object lookup, like contact/lead or a conversation lookup.

In the case of orphan record, there was no lookup of any object on the SMS record so the conversation lookup field was being used in the 'Where' clause while querying. But in the previous context, there was no conversation lookup on an SMS record and that query was returning 0 rows. That's why the valid duration was false.

To solve this issue instead of querying on the basis of any object lookup, querying on Phone number, Sender ID, and Channel code was done. So it was not affecting even if the Conversation lookup was empty.

Patch 1.68.21

Date: 8th Sept 21

Bug fixed

1.Issue— User was facing an issue with the Conversation Component in Utility Menu

Fix-

The user was facing an issue with the Conversation component in the Utility bar. The user created two different contacts, and when they tried to navigate the individual conversation component, it still showed the earlier messages of previously loaded contacts.

The conversation component controller adds a call for the toggle to display all the messages through the handle record id functionality and passes the record id to the message list component.

The issue related to the conversation component in the utility menu is now resolved.

2.Issue- User was facing License Sync and Plan Issue

Fix-

In v1.68, there was an intermittent issue where the license was not getting synced, and the user was getting locked out since the SMS-Magic license was not assigned to that user. Some fields of "Global Setting" and "Security Setting" were unavailable to edit on the given VF page. These fields need to be made available on the VF page for editing.

Certain fields of protected custom settings were available for editing via the VF page. This page can be used to update the available fields of protected settings in the orgs where subscriber access was not available. Especially in government cloud orgs. These fields need to be updated to reset the registration step and simulate the registration again.

The global setting component was updated to show the fields required to reset the registration step. It was useful in the orgs where SA was unavailable and if the user was getting locked out due to license sync issues or other possible errors.

The new fields will be useful to reset registration steps in the orgs where subscriber access is unavailable. With this, the auto-assignment of the license for the logged-in user was implemented, and the issue got resolved. The issue related to the license sync is now resolved.

3.Issue- SMS-Magic license error received by all non-SMS users when they were accessing the record page

Fix-

The customer had added an SMS-Magic component to the record page layout. Some of the non-SMS users of the customer did not have an SMS-Magic license assigned, and were not supposed to have an SMS-Magic license. When these non-SMS users were accessing the record page, they were getting an error notifications about not being able to access SMS-Magic-related files or licenses. The customer did not want these non-SMS users to see such notifications. The customer was neither ready to create different page layouts for such non-SMS users nor were they ready to assign SMS-Magic

permission set to such non-SMS users.

To solve this, we updated all the conversation components and data service streaming components to consume (not to display) the error messages related to non-access to SMS-Magic apex classes. It displays the no access-related message on the component UI for non-licenses users and suppresses any further apex calls from the components.

Patch 1.68.19

Date: 3rd Sept 21

Bug fixed

1. **Issue:** Get domain API of the portal, called though URL shortening was not enabled

Fix:

Get domain API of the portal, called though URL shortening was not enabled. To solve the issue, checks were added for restricting the API if the URL shortening was not enabled.

The issue related to enabling API portals is now resolved.

Patch 1.68.18

Date: 31st Aug 21

Bug fixed

1. **Issue:** Automation Key Reference was wrong

Fix: The customer was facing an issue while sending automated messages to Prospects(lead). The flows was set up for the 1st SMS message to new Prospect (Lead) records created. Any Lead created by the following user "Lender Site Guest User" was receiving an SMS.

Earlier, a customer was facing the following issue '**Automation key reference was wrong**'. This issue was resolved after sharing the settings assigned on the Converse App and App task.

The SMS history record was not getting created for guest users, the team was running a process in the queue in MessageAutomationHelper.queueablePostprocess() method, but in this method, a condition was applied, which was checked if it is Asynchronous or not. This condition was failing. That's why it was not running in a queueable context.

To solve this issue, the check MessageUtility.isAsynchronousContext() was removed from MessageAutomationHelovper.queueablePostprocess().

Patch 1.68.17

Date: 19th Aug

Bug fixed

1. **Issue:** Upgrade failed in Sandbox as the customer was trying to upgrade from 1.55 to 1.68.14

Fix: Customer was trying to upgrade from 1.55 to 1.68.14 version and was receiving the below error: The error was seen on the Deployment status page: Error: Unexpected Error and the details were that the package installation failed. 1.68 Error Message: The post-install script failed.

The error that the customer received in the email was: The request to install package 'SMS-Magic Converse Spring 2021' was unsuccessful. None of the data or setup information in the Salesforce.com organization was affected. In AnalyticsControllerTest.cls the customer was ordering query results based on the Mobile Phone field of Contact Object. In the customer's org the field which was encrypted was Salesforce limited and was used in ORDER BY clause in any query. The limited field was causing the installation of the package (v1.68.14) to fail in the customer's org. The salesforce.com organization was affected

To solve this issue, the ORDER BY clause was removed in two queries referring to the MobilePhone field of Contact Object.

Patch 1.68.15

Date: 17th August 2021

Bug fixed

1. **Issue:** The customer was receiving a component error on the case record page

Fix: A component error was appearing every time the customer was trying to send a message from a case record. The package was upgraded from 1.68.9 to 1.68.14 but the component error was still appearing.

Although the component error was appearing, the customer was able to send SMS but for this, they had to close 4 to 5 error windows. The customer was getting an error as shown in the image.

There was a reference to the attribute "slashTagBackUp" in the JS helper file. This attribute did not exist in the component which was causing the issue.

To solve this issue, the reference of this attribute was removed from the JS helper file.



Patch 1.68.14

Date: 14th July 2021

Bug fixed

1. **Issue:** Customer was not able to install SMS-Magic package on Salesforce essential

Fix –

While trying to install the SMS-Magic package on Salesforce essential, the customer was getting an error message, 'Your request to install package "SMS-Magic Converse Winter2021" was unsuccessful. None of the data or setup information in your salesforce.com organization was affected.'

In CampaignManageControllerTest.cls an operation was being run on a Product object which was a Salesforce-provided object. This was causing issues in Salesforce essentials as this object was not present in Salesforce essentials.

To solve this issue, the Product object was replaced with Contact & respective fields

2. **Issue:** Error during an attempt to de-reference a null object

Fix –

The user was creating a Consent record by calling the Invocable method from the flows. For Phone type compliance, if the user does not add Sender ID in the flows & tries to run that flows, it was throwing an error in the error log object, "Attempt to dereference Null object".

In the consent service, the Sender ID was being searched & if Sender ID was available then only the channel code was being checked.

In this case, Sender ID was always blank and that's why it was pushing channel code as null while creating the consent.

To solve this issue, after Sender ID & channel code was checked, a check was added to see if the channel code is null. If yes, then '1' was assigned as a default channel code.

Patch 1.68.13

Date: 1st July 2021

Bug fixed

- **Issue:** The user was facing an error while attempting to create a consent from the flows

Fix:

An error occurred when the user created a consent record by calling the Invocable method from the flows. If the user was trying to run the flows without adding the Sender ID for phone type compliance, it was throwing an error: error log object "Attempt to dereference Null object."

To resolve this, the Sender ID & channel code was checked, and a value was added as a default channel code if it was empty.

The issue related to the consent creation record for dereferencing a null object is now resolved.

- **Issue:** The customer was facing a Mobile app licensing issue on the base license

Fix:

The functionality needed to handle the new response values that were returned, based on the validations added for mobile app licensing, based on the base license. But the customer was facing the mobile app licensing issue on the base license.

To resolve this, the functionality was updated to handle the response that was returned from the sync license portal callout, the error log was analyzed, and the license status was returned in case of a failure.

It addressed the below three cases:

1. Adding a mobile license if the user doesn't access the converse desk, marks the license as inactive.
Failure: "You do not have any license having a converse desk feature for user {} cannot assign mobile licenses."
2. Removed any license if the user lost access to the converse desk and then activated the license again.
Failure: "You cannot remove the license as the mobile license is allocated to the user {actual user name}."
3. If the platform returns any error response while syncing the license allocation change, that operation will be reverted, and the error log will be added with the failure reason.

Patch 1.68.12

Date: 17th June 2021

Bug fixed

1. **Issue:** User was receiving a Sound Notification for Incoming SMS even when the notification was disabled from the SMS-Magic Converse settings

Fix –

The default value of attribute 'isIncomingVoicePermission' in conversationView component was set to true. In controller.js the value of this attribute was set only if the corresponding flag value in global settings was set to true. Due to this, the value of the sound notification setting configured in the converse setting was properly populating in the Converse desk component. Thus even if it was set to false, the sound notification was getting played on incoming messages.

To solve this issue, the default value of the sound notification attribute in

the conversation view was set to false. With this, the sound notification was not received on incoming messages.

2. **Issue:** Admin was not able to access the SMS-Magic Converse Settings

Fix –

The customer was not able to access the Converse Settings and was receiving an error, “You do not have a License which provides access to this feature.” in spite of having the required license.



The code was checking for the profile name of the logged-in user to be “System Administrator”. Thus even if the user had admin privileges, as the profile was not “System Administrator”, the user was not able to access the Converse settings.

To solve this issue, the unnecessary “System Administrator” profile check was removed.

[Version 1.64](#)

Patch 1.64.45

Date: 8th Aug 2022

Bug fixed

1. Issue: The user was facing an error related to the consent object for an opt-out functionality in the error log

Fix:

The Admin user was receiving an error “Procedure is only valid for System.DMLException and System.MailException” in the error log object. This occurred when the “WhatsApp opt-out” checkbox was enabled from the record page and a consent record was updated for the mobile number to Opt-out. The consent record was getting created; however, this also generated the error log.

The issue related to the consent object for an opt-out functionality is now resolved.



Patch 1.64.44

Date: 16th Dec 2021

Bug fixed

1. Issue: SMS-Magic mobile users were able to send SMS to leads even before obtaining their consent.

The customer had created a test user and provided the consent for testing purposes. The customer was using CustomConsentService class to validate consents. In SendSMSTriggerHelper.sendSMSBeforeTrigger(), the ConsentService.getConsentValidatedRecords() method was called to fail records on the basis of consent. The injection point was provided in this method. Messages were getting failed in the custom class and were returning to the list mobileApplicationSMSList in SendSMSTriggerHelper. As CustomConsentService was used, the list was sent and processed in a serialized form. In the process of serialization and deserialization, the reference was not followed and mobileApplicationSMSList was treated as a different list from smsList which is the main list. This main list was not getting updated.

To solve this issue, main list was updated as per the updated mobileApplicationSMSList in SendSMSTriggerHelper.sendSMSBeforeTrigger().

Patch: 1.64.43

Date: 3rd Dec 21

Bug fixed

1. Issue: Channel ID and Sender ID were not mapped.

Fix: While sending the message, the customer user was getting the following error, "**Channel and Sender ID are not mapped**". The Sender ID was added to their production instance and the customer had created a record of that Sender ID in their Sandbox instance. The issue was occurring randomly. Some of the messages that were sent through the Sender ID 447418344628 were delivered and some of them were failing.

While checking the debug logs it was found that the MessageDetailsUtility.PreparedUnifiedPayload method was creating a payload. This method was called MessageDetailsUtility.GetMessageDetailsList method where the channel code was being worked on, depending on the '**Channel Name template ID**'.

Here, the MessageDetailsUtility.getTemplateMetaMap method was called and inside this method, it was executing 1 query on smagicinteract__Template_Channel_Map__c & inside the clause it was fetching records from template ID.

This query was fetching 1 record, that tried to run the same query in the dev console island and it was returning 0 records, which was correct. It was suspected that the one record-receiving of type was WhatsApp channel Type.

To solve this issue, Added (smagicinteract__Channel_Code__c ==2 ||

smagicinteract__Channel_Code__c ==3) condition and thus it was a part of MessageDetailsUtility.getMessageDetailsList method.

Patch: 1.64.42

Date: 16th Nov 21

Bug fixed

1. **Issue:** Customer was facing Component Error

Fix: The customer had reported that their users were getting the component error while opening the lead record. This error was intermittent but it was affecting the user experience.

License and permission sets were already assigned to the users. Component Error occurred because push topic subscription was not successful for the affected user.



To solve this issue a Null check-in '**ProcessMessages**' method of conversationComponentController was created.

2. The Customer was facing an issue with incoming notification with an email setting

Fix: In the Sender ID step of converse settings, '**Email notification on incoming (Default Setting)**' was configured.



The logic in the '**SendEmailBySender ID**' method of SMSUtility.cls was not correct. If any specific Sender ID record did not have the '**Send Notification From Email**' value populated, the code was not picking up the default configured email address. Thus the email was sent from the Admin user's email address.

To solve this issue, the logic in the '**Send Email By Sender ID**' method of SMSUtility was updated to populate default values from '**Global Setting**'.

Patch 1.64.41

Date: 03rd Nov 21

Bug fixed

1. **Issue:** Whatsapp 24 hours window was not working

Fix: The client was sending MTM templates to their international number. While receiving incoming SMS, the conversation window for WhatsApp SMS was still showing as blocked for the user. Once we received the incoming SMS, it allowed the users to send the messages in the next 24 hours.

Whenever we tried to send a message from the conversation view on the record

detail page. We checked for the valid duration and if it was applicable we call this method '**ChannelUtility.checkForValidDuration()**'. Inside this method, we checked if the Record ID is not null, and we fetched the incoming messages. To fetch records from SMS History we had a method **ChannelUtility.getLastIncomingTimeIntervalForSMSHistory()**. Here, we executed 1 query on SMS History which fetched incoming records from SMS history using the phone field.

Here, the phone number we were using consists of all the characters other than numbers (Like +, -, etc). We were not removing those unwanted characters. That's the reason why it was not fetching any record in the query. That's why the 24hrs window was not getting enabled.

To solve this issue Added a call to MessageUtility and removed the **LeadingZeroes()** method which will remove all the unnecessary characters other than a number.

Patch: 1.64.40

Date: 14th Oct 21

Bug fixed

1. **Issue:** The customer was facing an issue with Automated Messages as they were getting failed

Fix: Most of the Automated messages were getting failed and the customer was receiving multiple error log records. They were getting an error 'Selected Sender ID is not valid. Select another Sender ID and try again.'

Also, there was an error when Converse settings was clicked. Whenever a user tried to send a message using Record Owner's Sender ID, and some of the records owners didn't have Sender ID assigned then it was throwing an error 'Selected Sender ID is not valid. Select another Sender ID and try again.' SMS History record was also not getting created.

To solve this issue, try-catch block in **Senderservice.resolveandAttach()** method was added.

In the try-catch block, the issue regarding invalid Sender ID was resolved and thus the customer was able to send automated messages properly.

Patch: 1.64.39

Date: 22nd Sept 21

Bug fixed

1. **Issue:** Converse Bulk Campaign was stuck in the running state

Fix: Whenever the customer was trying to fetch the Listview records that were created, a query in the **Metadata Service.ParseSelectCondition()** popped up, which split into 2 parts.

Part 1: On the right side of the keyword '**FROM**'

Part 2: On the left side of the keyword '**FROM**'

Here, whenever the user created a list view with criteria that contained a lookup of another object, the query contained 2 '**FROM**' keywords in it. The code was taking 2nd FROM keyword and was getting split into 2 parts. This caused a returned query that was running on the lookup object and was causing the issue.

To solve this issue the object name was fetched and divided the query from keyword '**FROM**' + object name into 2 parts. 1st part consisted of values FROM Object name & 2nd part consisted of the where clause.

2. **Issue:** Customer was facing future Callout from flows

Fix: An IncomingService.pushNotificationsToMobile() was called in the conversationTriggerhelper class after a conversation was created. A callout was created while the conversation was in the same callout IncomingService.pushNotificationsToMobile().

To solve this issue, temporary changes regarding pushNotificationsToMobile() were not certified due to platform issues. The changes were reverted and final changes were made once all the issues from the platform side were resolved.

Patch 1.64.38

Date: 13th Sep 21

Bug fixed

1. **Issue:** Customer needed a Patch Version for 1.64

Fix: The client had the requirement to implement the custom component created for Intake, Matter, and Party. For this purpose, the Composer Component needed to initiate independently fetching the required data. To solve this, the condition in the init of the composer component was updated to fetch the required data when it did not pass from the parent component and added the '**had access**' attribute while creating the composer component and passed its value as true.

Patch: 1.64.36

Date: 1st Sept 21

Bug fixed

Issue: Post upgrade issues from 1.58 to 1.64.30. (Customer was facing the conversation component error on the opportunity record and also the browser was running slow)

Fix –

The user faced the below issues after upgrading from 1.58 to 1.64.30.

1.Admin user was getting an error when he opened the Conversation component on the opportunity record.



2.The users faced issues where their browsers ran very slow and sometimes froze.

The user had a custom push topic event implemented on the record details page of the opportunity object. The subscription error of that custom event was captured in the streaming data service component of the package. Because of the error, the connection flag value of the component was set to false, and the component fallback to a continuous polling mechanism instead of a push topic subscription. The Apex calls are part of continuous polling, causing the page to hang until the apex call is resolved, thus hampering the user's interaction with the page.

Added check-in callback handling of the error and checked whether it's a subscription error of the push topics created via package and not any other push topic event that would have been present in the target org. Also, we partially removed the conversation utility bar to lower the freezing time.

Patch 1.64.35

Date: 25th Aug 21

Bug fixed

1. **Issue:** The customer was facing an issue after the upgrade from 1.58 to 1.64.30

Fix:

1. Admin user was getting an error when they open the Conversation component on the opportunity record.



2. Users were facing issues where the browser was running very slow and it was freezing sometimes.

The issue was fixed after removing the Conversation utility bar.

The customer wanted to use the Conversation utility bar, however, didn't want the system to run slowly.

The customer had some custom push topic events implemented on the record details page of the opportunity object. Thus the subscription error of that custom event was captured in the "StreamingDataService" component of our package. Because of that error, the connection flag value of the component was set to false and the component was falling back to the continuous polling mechanism instead of push topic subscription. Apex calls, which were part of

continuous polling, were causing the page to hang until the apex call was resolved. This was hampering the user's interaction with the page.

To solve this issue, a check-in was added to check whether the error was for the subscription of the PushTopics created via package and not for any other PushTopic.

2. **Issue:** The customer was facing a Non-selective query against a large object error

Fix: The customer was facing an error non-selective query even after indexing the fields.

This query was arising when they were trying to send automated SMSs through the converse app.

While querying on SMS history records in fetchMonthlyMessages method of ConsentService class, it was throwing an exception because the team was using 'status != and template!=null' in the where clause.

Whenever the customer was using P+S compliance, messages regarding content type were getting fetched. This was fetching multiple records using queries & was causing unnecessary code execution for P+S compliance.

To solve this issue, changes were made in the query. The 'status in (status set) and Is_SMS_Template_Empty__c = false' condition was removed from both places and added a check to see if the customer was using Content-Type in compliance. If the customer was using Content-Type then added a check & called fetchMonthlyMessages() method.

Further, in v1.65.35. The particular query was supposed to be executed only if the content type was a part of compliance parameters. Thus added the check-in ConsentService class if content-type was enabled for compliance and only then executed the query.

Patch 1.64.34

Date: 20th Aug 2021

Bug fixed

1. **Issue** – Unnecessary continuous debug logs were getting created in the Dev Console

Fix –

In the developer console, there were unnecessary debug logs getting generated from SMS-Magic.

The periodic polling mechanism was generating continuous logs. This call should not have been made for non-licensed users.

To solve this issue, a license check was added in the method 'conversationsWrapper' of MessageComposerServreController.cls. and access

check was also added in conversationsComponent before calling apex method to fetch the latest messages.



Patch: 1.64.33

Date: 06th Aug 2021

1. **Issue:** Non-selective query against large object error.

Fix –

The customer was receiving the error '**Non-selective Query**' even after indexing the fields. The query was arising when they were trying to send automated SMSs in the converse app. While querying on SMS history records in the '**FetchMonthlyMessages**' method of ConsentService class, it was throwing an exception because 'status != and template!=null' was used in the clause and to solve the issue. Changes were made in the query. To solve this issue, the method 'status in (status set) and Is_SMS_Template_Empty__c = false' was added. != condition was removed from both places.

Patch: 1.64.32

Date: 02nd Aug 2021

Bug fixed

1. **Issue:** The fallback Sender ID set for action was not consistent with the UI of the Converse App action

Fix –

When a customer was trying to update the Sender ID on the UI of Converse App Action, it was getting auto-updated to the old one. Existing functionality was that while fetching Sender IDs based on SenderId value rather than Record Id, we were assigned the first element of the list to that Sender ID.

If an org contains Sender IDs with the same alphanumeric value, it was producing a list of Sender IDs with more than one value, and each time the first Sender ID was getting assigned to Converse app action.

The proposed solution was that instead of fetching Sender IDs based on values, now it would be fetched based on Record Ids, so that every time a unique Sender ID will get produced.



2. **Issue:** Unable to Send MMS Via Bulk SMS button

Fix –

The customer was facing an issue where the users were unable to attach media while sending messages via the Bulk SMS Button.

To solve this issue, implemented negation for the '**AllowToChangeTemplateSelection**' variable. Negation for attribute

'AllowToChangeTemplateSelection' was missing in the 'class' attribute of the button element. A class with the name 'slds-is-disabled' was getting assigned to the button element thus the button was clickable yet shown disabled.



Patch 1.64.30

Date: 2nd July 2021

Bug fixed

1. Issue – Unnecessary Continuous Debug logs were getting generated in the dev console

Fix –

In the developer console, unnecessary debug logs were getting generated from SMS Magic. These continuous logs generated by the SMS Managic managed package were observed and prompted by the customer.

It was analyzed that the periodic polling mechanism generated these continuous debug logs in the dev console.

Resolved this issue by adding a license check in the conversations wrapper method of the message composer server controller class and by adding an access check-in conversations component before calling the apex method to fetch the latest messages.

This issue of continuous debug logs in the dev console is now resolved.

2. Issue – The email-to-text feature was not working for the customer.

Fix-

The user was facing an issue where their email-to-text configuration was not working as expected. Previously when they were replying to the email notification, they could leave the Reply-To address field blank, which caused the individual users' notification reply service email address to populate when they replied to an email.

Now, when the user hits the Reply on the email button, the notification reply service email address is that of CRMAdmin, which is set up only to receive emails from assigned mail (CRMAdmin). If we add all users to this notification reply service, the text messages created by the incoming email will be owned by CRMAdmin, and all further notifications will go to CRM Admin.

Executed the DML operation in the admin user's context; by default, the current user was the owner of the SMS history record. The user id is set to the owner id for the missing user for a newly created SMS history record. Queried the user id based on the from address attribute of the incoming email

and ensured the user who replied to the email would be the owner of the SMS history record.

The error related to the functionality of email to text is now resolved.

3. Issue – Salesforce campaign issue occurred when the user ran the campaigns with more than 5000 contacts

Fix –

The user was facing an issue while running large Salesforce campaigns. The user created a salesforce campaign, sending messages to 5000 (approx) recipients; however, when they were clicking the Run SMS Campaign button on the Campaign record, it was throwing the Apex CPU time limit exceeded error.

The issue occurred when the user ran the campaigns with more than 5000 contacts. Previously (before upgrading to 1.64), the user was able to schedule SMS for 25,000+ contacts in one campaign.

The enabled new compliance message utility functionality was inside the 'for' loop. Internally, this method calls Security.hasAccess(); if there are 22k campaign member records, then both methods will get called 22k times, which is time-consuming.

We have stored the result in one variable and would use that variable wherever necessary.

The issue related to the Salesforce campaign is now resolved.

4. Issue – Customer was facing bulk SMS component error and bulk button issue on LWC.

Fix –

When the user was opening a custom widget and clicked on the Send SMS button, no list was shown, and no record was selected.

After selecting a record from the listview to send Bulk SMS, sometimes the records were passed without the '[' ']' brackets. Also, it was observed that as the '[' & ']' brackets were unavailable, removing characters from record ids caused the issue.

We analyzed and removed the first & last characters from the array.

The normal bulk SMS from any List view of any object and the bulk button issue on LWC is now resolved.

Patch 1.64.29

Date: 14th June 2021

Bug fixed

1. **Issue** – Issue of SMS getting sent to duplicate number in Converse App >>

Bulk Campaign

Fix –

ID of the converse app task was not passed to batch before execution. Thus the required check for duplicate numbers was not happening.

To solve this issue, the ID of the Converse App task was passed to batch ensuring check for duplicate numbers. With this, sending messages to duplicate numbers was avoided.

2. **Issue** – Need to pass the record ID parameter to the injection service:CustomConsentService

Fix –

The client wanted to check for field on the record and based on that they wanted to manage the Consent record. For customization, the case record ID was required to be passed to the custom class (“CustomConsentService”) at the injection point.

At the injection point, the record ID was passed to the custom class (“CustomConsentService”) to support the customization case.

Patch 1.64.28

Date: 2nd June 2021

Bug fixes

1. **Issue:** The SMSs were not getting pushed in SF due to excess SOQL queries.

Fix:

The incoming SMSs were not getting pushed in SF due to too many SOQL queries.

It was analyzed that when the toggle for notification on incoming SMS was Off, SMS messages were getting pushed, and if it was On, the user was receiving an error.

The issue related to the error while pushing the toggle for notification is now resolved.

Patch 1.64.27

Date: 24th May 2021

Bug fixes

1. **Issue:** Customer was getting an error while sending SMS from Bulk Campaign from the Converse App

Fix –

When any object from parent-child relationship was configured in the Message Configuration Object (MOC), the package code was not able to resolve the

relationship to get the value of the opt-out field. Due to this, the customer was getting an error while sending SMS from Bulk Campaign from the Converse App.

To resolve this issue, the code was updated to resolve the relationship and get the value of the opt-out field as expected.

2. **Issue:** Customer was facing an issue with lookup object on SMS consent record

Fix –

The last updated record ID was getting attached to each consent record that was created. This was causing the mentioned error.

To solve the issue, the unique record ID which is fetched from input consent data was updated.

Patch 1.64.24

Date: 17th May 2021

Bug fixed

Fix: Conversations Lightning component was not loading messages properly. The customer was facing an issue where the messages before 16th march were not loading on the conversation component on the record page layout. In message.cmp there was an unnecessary use of the method `sanitizeUrl()` that was causing an issue while parsing the object instead of a string.

To solve this issue, we removed the unnecessary use of the method. Used it at a relevant place from a security perspective.



Patch 1.64.23

Date: 5th May 2021

Bug fixed

1. **Issue:** Customer asked for customization for SenderId Injection point

Fix: Customer wanted Sender IDs to be available in a specific sequence.

In the injection point, the data was run through the “transformData” method which was causing an issue in implementing the desired customization for the customer.

To address this customization requirement from the customer, an ‘IF’ condition was added where a deprecated flag ‘smagicinteract__Canned_Responses_Only__c’ from the ‘globalSetting’ was used. If it was true then the data was not transformed, if false, the functionality was executed as usual.

Patch 1.64.17

Bug fixed

1. **Issue** – Consent was not getting created for SMS channel via flows

Fix -Customer had configured a flows named “Lead Driven Processes on Lead” to create consent for SMS and WhatsApp channel when the conditions were fulfilled. This was done by the apex class coming with the managed package.

It was working fine, the customer could see that consent entries were getting created for the SMS channel till 17th March 2021.

Later, it only created consent for the WhatsApp channel.

The customer could see an entry created under the error log at the same time with an error “Attempt to de-reference a null object”.

2. **Issue** – Consent was not getting created even when the criteria were met

Fix – A flows was created for consent creation that calls apex” smagicinteract__ConsentService”, when the criteria of the checkbox “Consent to contact via Whatsapp” getting checked was met. But the number in the WhatsApp number field was not opted-in (Condition 2 and 3 in the flows)

3. **Issue** –

Issue #1:

When any user was logging in and selecting any Sales or Service Path where the utility bar component was assigned, the tab at the top showed “Converse Desk|SMS-Magic”. The client users were getting confused with this and so the client wanted this to be resolved.

Issue #2:

Misspelling in the error message that was getting displayed.

If the user does not have any license then they were getting an error message, “You don’t seems to have required license to send SMS or MMS”.

Here, in the error message, there was a spelling mistake; it should have been ‘SEEM’ and not ‘SEEMS’.

Issue #3:

The ‘Send SMS’ button was configured on Lead, Opportunity, Account, and Case object. When the customer user was clicking on the button, it was not fetching the mobile number to send the SMS.

If the VisualForce pages assigned to the Send SMS button were edited and the phone fields were added to it, it was fetching the phone info. However, the client did not want to add any manual steps to update the VisualForce page script.

As per logic, the Send SMS VisualForce pages should have displayed the phone fields configured in the MOC, however it was not working.

Fix –

Issue 1 – Added a condition in ConversationView to skip the execution of setting title if SF1 was not found in the context

Issue 2 – Custom Label was changed

Issue 3 – Removed the condition from LightningBulkSMSController which was limiting the selection of Phone Field from MOC configured.

4. Issue – Issue with Lightning Tab and with Send SMS button

Fix –

Issue 1:

When any user logs in then selects any Sales or Service Path where we have assigned the utility bar component, the tab at the top shows “Converse Desk|SMS-Magic”.The user are getting confused with this and client wants this be resolved asap.

Lightning app name: Sales Path (Sales Management app for Lead Gen)

Issue#2:

We have configured Send SMS button on Lead, Opportunity, Account and Case object. When we click on the button, it is not fetching the mobile number to send the SMS.

Patch 1.64.12

Date: 24th Feb 2021

Bug fixed

Issue – The 24hr Conversation Window was not working properly on Facebook and WhatsApp

Fix –

The 24hr Conversation Window was available on One record with the same mobile number and Sender ID. However, it was not available on another Object. The system was checking for the 24hr window on the basis of recordID, phone, Sender ID, and channel. The recordID was causing the issue.

To solve this issue, recordID was removed from the key. With this, the 24hr Conversation Window started functioning properly on WhatsApp and Facebook both.

Patch 1.64.10

Date: 20th Jan 2021

Bug fixed

Issue – The consent records were not getting created with country code prefix

Fix –

Consent records were not getting created with country code prefix for those who have set a specific country as messaging destination (other than the 'All' option). Consent records and compliance enforcement functionality was not working as expected.

To solve this issue, security enhancements were done in the country-specific consent recording.

Version 1.60

Patch: 1.60.89

Date: February 02, 2024

Issue: Intermittent Component Error for Incoming Messages

Fix:

Users experienced an intermittent error within the component responsible for handling incoming messages. The underlying cause was identified as a missing parameter in the onInboundMessage function. The issue has been resolved by updating the function to include the required parameter, thereby restoring consistent and error-free operation for all users.

Patch: 1.60.86

Date: 22nd July 2022

Improvement

- **Issue:** The customer had a customization request to setup the validation rule for the user

Fix:

Implementation:

As per the customer's requirement, the validation rule was added with the click of the send button. Implemented the validation rule for all three interfaces: Converse Bulk interface, Send Bulk SMS interface, and Community

Send Bulk SMS interface.

Implemented the rule with the name of the contact list and the name of the list view from contacts, followed by a continuous and cancel button.

The requirement of validation pop-up while sending a bulk Campaign from the converse app is now implemented.



Patch: 1.60.82

Date: 9th March 2022

Bug fixed

- **Issue:** Feature license error for the power license users

Fix:

The SMS-Magic user who holds a Power License faced an issue while sending an SMS from the SMS tab on Salesforce and accessing the Converse Desk tab of SMS-Magic. It prompted an error message that the user didn't have a license for the feature as it was a part of Standard and Enterprise plans.

Added a filter while querying licenses for users in the get license method of License Service.

The issue related to the feature license for the power-licensed users is now resolved.

- **Issue:** Improper working Template for the SMS-History records created via business hours functionality

Fix:

When the SMS-History records were created from the business hours with source "1495" and were scheduled for the next business hours, the SMS text was not getting populated.

In the send SMS trigger helper class, while adding the automation message record with source value as "1495," if the code checks the disable SMS on-trigger field value as 1, the message record was not added to the list of automated messages. The template is not resolved for such records, and SMS text is not populated.

Removed the check for the disable SMS on-trigger field while adding the message record. Thus, the message record will be added to the automated message list, and the associated template will be resolved.

The issue related to the SMS-History records template for the non-business hours is now resolved.

Patch: 1.60.80

Date: 30th Oct 21

Bug fixed

Issue:

1. The customer was facing an issue while using the Converse App

Fix: The following error message was detected while the customer was using the Converse App:

Error Message:**Invalid field OwnerId for Training__c.**

The user was trying to send an SMS using the Detail record from Converse App, and the Owner ID did not exist for the detailed record. Hence the issue encountered.

To solve this issue, fixed the get RecordsSender ID() method of SenderService.cls, and explicitly added "OwnerId" in the fieldName. Also fixed the code, and checked if the object was detailed which took the Owner ID from the master object as well as used the existing method MessageUtility.personObjectOwnerFldMap().

2. The customer wanted to exceed the limit of objects from 10 to more, to the Message Object Configuration

Fix: There was a requirement from the customer to exceed the limit of objects from 10 to more so as to be able to text to more than 10 objects. While the customer users were trying to add the 11th object on Message Object Configuration, they were receiving an error regarding the limit of objects being exceeded.

The customer had a limitation of Max 10 Message Object Configuration records in converse settings. That's why this error was occurring.

To solve this issue, added Configurable Static Resource with name – SMIExceed Message Object Configuration Limit. It contained a value > 10, and thus allowed users to add that number of Message Object Configuration records without any error.

Patch: 1.60.79

Date: 26th Oct 21

Bug fixed

1. **Issue:** Customer was facing Error while sending outgoing SMS through Non-Admin Users

Fix: When a non-admin user Cody Woody tried to send a text from the Conversation component it added the Contact object and was getting the following error message:

(magic interact.SendSMSTrigger: execution of AfterInsert)

It was caused by: System.NullPointerException: Attempt to de-reference a null object

The customer was using CustomLookupService and adding Conversation lookup on the SMS record before inserting. The Conversation ID that was associated with the SMS record before inserting was owned by the Admin user and sharing setting on the Conversation object. This user did not have access to that conversation record. In our code we use this conversation ID and create a map of ID and conversation record to access conversation records in the updateConversationFields() method of ConversationService.cls class. Here conversation value was null therefore an error was thrown.

To solve this issue Added a null check so that if the conversation is null then skip that record.

2. Issues with Auto Append Text In Compliance

Fix: The customer was facing issues with the Auto Append option in the compliance section.

They have configured automation in their system, where once the specific response is received on the basis of criteria and outgoing message and was sent out automatically.

When the auto-append option was enabled, the response was getting triggered correctly but the Compliance response that was supposed to go along with it was getting triggered separately creating a record. Also, a NULL value was added in the text.

When the customer was trying to create an SMS History record via flows. In sendSMSTriggerhelper.sendSMSBeforeTrigger(), we were checking for sources. In this case, the source was an automated message & for the automated message, we were appending the opt-out instruction message. In the sendSMSAfterTrigger() method, we were resolving the template bypassing automatedSmsList() to resolve the template method. While adding values in automatedSmsList we check if SMS Text is null & that condition was falling because we were appending opt-out instructions before trigger. That's why it was sending a null text in SMS Text.

Patch: 1.60.78

Date: 28th Sept 21

Bug fixed

1. Issue: A new conversation was getting created for every message

Fix: The customer had reported for all the users who had the non-admin profile, that every message created a new conversation, whereas all the incoming and outgoing messages were getting attached to a single conversation from the admin profile user.

To solve this issue, the unnecessary field(Lead) was queried. Conversation creation and lookup fields were set on receiving an incoming message(Single/dual object) for the user who did not have permission to lead the object.

2. Issue: UI Injection point for Conversations component | Customisation Use case requirement

Fix: Customer's requirement: (To be implemented via Customisation)

On the Conversation Component, the conversations were visible based on the mobile number selected. Conversations related to the "To" number were visible on the conversation component on the record page. Thus the Conversations were changed if the "To" mobile number was changed.

An injection point was required in the Interact package using which this requirement was achieved via customization.

To achieve this customization, an injection point was created for a custom injection service to which a selected phone number or Sender ID was provided as a parameter, and the data received from the custom injection service was used to display on the UI in the message list component.

Patch: 1.60.77

Date: 1st Sept 21

Bug fixed

1. Issue: Customer was not able to access custom Inbox in Converse Desk

Fix: The issue was with Safari 14.1.1 and above. On the Converse Desk, when the customer was trying to open a Custom inbox, it was throwing an error. The issue was reproduced in Test org as well.

In the SMSUtility class, the BaseUrl method was not returning to the correct org domain URL and hence the issue was occurring.

The screenshot is attached below with the error:



To resolve this issue, the debug logs of affected users and the non-affected users were compared. Different Endpoints for those users were found out.

Further, SMSMagicConfigurationRemoteURL Remote Site Setting was updated and set within the correct Remote Site URL. This was working fine on the Chrome

browser but the issue was reproducible on the Safari browser. In SMSUtility class, the getBaseUrl method was not returning the correct org domain URL hence this issue was occurring. The getBaseUrl method of the SMSUtility class was then updated to return the correct org domain URL.

Patch: 1.60.76

Date: 23rd Aug 21

Bug fixed

1. **Issue:** Multiple Issues were reported with converse desk

In the Converse Desk, 'My Inbox' page should get auto-refreshed for the Conversation owner, however, when the client went to the Converse Desk >> All sections, this page was not getting refreshed automatically for all the users.

The client wanted the All list view to be refreshed automatically for all their users. The user had to click on the refresh icon or sometimes complete page refresh was required to see the latest incoming SMS.

Issue 2:

When the customers opened the Conversation on the Converse desk and clicked on Lookup filters of the latest outgoing SMS, the details of the lookup information were not displaying properly on the screen.

Issue 3:

The OAuth user was continuously getting governor limit emails from Salesforce when the user went to Converse Desk and clicked on any list view in the Converse desk. The client had requested to look into this issue on priority.

There was a check only for Inbox whenever a new incoming message on the Converse Desk was received. If the user tried to open any custom inbox & if any incoming message was received, as the user was in the custom inbox, the inbox check condition was failing.

To solve this issue the check for Inbox was removed.

Patch: 1.60.75

Date: 17th Aug 2021

Bug fixed

1. **Issue:** The customer was facing an issue while using the Injection code and the code was not working.

Fix –

Sender Id Allocation – It allows students from multiple Accounts to send

texts to – which was considered brand bleed and the aspect was unique sender number by Account / Partner and separate/unique content by Account / Partner. It was showing all Sender Numbers that were purchased under the Wiley Account as Sender Numbers was restricted to one sender number that applied to the Account associated with the students chosen for bulk SMS.

Therefore, the customer received the underlying code/page/component (from package code) and applied these 2 restrictions, and used the same Bulk SMS Sending Page. The design injection points were not enabled for the bulk components.

To solve this issue the backward compatibility (upgrade scenarios) and use cases of other customers was kept, the record id check was not removed and an alternative flag was added. The injection point was functional for the bulk interface as well and a certain flag was enabled.

Patch: 1.60.74

Date: 29th July 2021

Bug fixed

1. **Issue:** Customer was facing Issues with Auto Append text in compliance

Fix –

When the customer was trying to create an SMS History record via flows, in `sendSMSTriggerhelper .sendSMSBeforeTrigger()`, the source was checked in both fields. In this case, the source had an automated message & for the automated message the opt-out instruction message was appending. In the `sendSMSAfterTrigger()` method, the template was resolved by passing `automatedSmsList()` to resolve the template method. While adding values in `automatedSmsList`, it was checked if SMS Text is null & that condition was falling before triggering the appending opt-out instruction. That's the reason it was sending a null text in SMS Text.

To solve the issue this method was removed, call to the `ConsentService.append Opt-Out Instruction()` method from before trigger & added in after trigger & passed the list which gets after resolving the template. The values were upserting in the SMS History record.



1. **Issue:** Name field from SMS History object was blank and Automation was triggered from Converse App

Fix: Here, the name field was auto-populating. To solve this issue added a name field in the `messageAutomationHelper` class



Patch 1.60.73

Date: 24th June 2021

Bug fixed

1. **Issue** – Last Incoming SMS lookup was not working

Fix –

When a new conversation was created by the first incoming message, the code was populating the lookup of the last incoming message record. But if the conversation existed (created due to the first outgoing message), the code populated lookup of the last outgoing message only.

To solve this issue, the code and logic in the 'ConversationService' class were updated to populate both fields 'Last Incoming Message' and 'Last Outgoing Message' for every corresponding incoming and outgoing message.

2. **Issue** – Customer was getting a component error while accessing utility bar item

Fix –

This was an intermittent issue, occurring when a method of a service component 'conversationDataService' was called. The issue was occurring because intermittently, the code was not able to confirm the availability of component 'conversationDataService'; and when its method was called the component error was thrown.

To solve this, a check for the availability of the service component 'conversationDataService' was added before calling its method. That way, for any reason, if the service component was not available, its method would not be called and the component error would not be thrown.

Patch 1.60.71

Date: 9th June 2021

Bug fixed

Issue: Injection code not working as expected

Fix:

Customer had implemented customization to show specific sender IDs in SenderID dropdown while sending messages. This worked as expected for interactive interfaces. The customer wanted the same functionality to be available on the bulk interface as well (bulkSMSComponent).

By design, the injection point was made available only for interactive interfaces. For that reason, there was a check for the record ID added at the injection point. In the case of bulk interface, record ID was not available, thus the logic of injection point did not work and the implementation customization did not reflect.

To solve this, two specific configurations were made to make the injection point work for bulk interfaces and to maintain backward compatibility:

1] Enabled the injection point for Sender ID customization: In globalsetting (custom setting) the value of field "Max SMS Text Length" was set to "101"

2] Enabled the injection point for Template customization: Enabled the flag "Enable Internal Note"

Patch 1.60.70

Date: 8th June 2021

Bug fixed

1. **Issue:** Incoming SMS were not getting pushed to Salesforce

Fix –

The incoming SMS were not getting pushed to Salesforce. On incoming SMS, a user from the previous outgoing message was identified, and that user was assigned as the owner of the new incoming message. Since the user was inactive, Salesforce was not allowing to create the SMS History record with the inactive user as the record owner. On incoming message, while doing a lookup of the previous outgoing message, if it was found that the owner of that outgoing message is inactive, the owner of the new incoming message was set to the OAuth user. This solved the issue.

2. **Issue:** Uncaught Error in \$A.getCallback() [Cannot read property 'getInboxFilter UnreadCount' of undefined]

Fix –

A data service component was passed as a parameter while calling the helper function in the AURA component. With the Summer '21 update that was deprecated, it was causing the error, where the property 'getInboxFilter UnreadCount' was not getting read.

To solve the issue, the parameter was removed and the respective data service component was identified in the corresponding helper method.

Patch 1.60.69

Date: 19th May 2021

Bug fixed

1. **Issue:** In the SMS-Magic Converse App, the Bulk Campaign was getting stuck in a running state

Fix –

In the SMS-Magic Converse App, the Bulk Campaign was getting stuck in a running state. In the Details section, List View details were not getting displayed for both the Bulk Campaigns. This was due to the wrong formation of the List View clause query.

To solve this issue, the List View query was modified by adding 'from' + 'object name' in the 'where' clause of the list view.

2. **Issue:** Conversations were not getting created for WhatsApp automated messages

Fix –

Conversations were not getting created for WhatsApp automated messages. The automated messages triggered through the customer's WhatsApp number were not creating conversations because of which they were not showing on the Converse Desk. There were entries created in the error log with the error 'Empty SMS List'.

To solve the issue, the condition where we were checking SuccessList size for callout, was removed. Now Conversation Association is happening as expected in all cases and in the SMS-Magic Converse App, the Bulk Campaign is not getting stuck in the running state.

Patch 1.60.68

Date: 26th April 2021

Bug fixed

Issue – Messages were reaching the SMS-Magic server even when the 'Disable SMS' option on Trigger was set to '1' by the client.

Fix –

Messages were reaching the SMS-Magic server even when the 'Disable SMS' option on Trigger was set to '1' by the client.

SMS records with the 'Disable SMS' option on Trigger was set to '1' and were added to the failed SMS list. But they were also added to the SMS list which we are sending to a callout in the sendMessage method of SMSUtility class. That's why they were reaching the server.

To solve the issue, the if-else condition was added: 'If disable on the trigger is 1 then add to the failed list else add to the callout list'. With this, the messages stopped reaching the SMS-Magic server when the 'Disable SMS' option on Trigger was set to '1' by the client.

Patch 1.60.67

Date: 26th April 2021

Bug fixed

1. **Issue:** Get conversations function was calling for update/insert of the recently used filter on the converse desk.

Fix –

In the conversation controller class, get conversations function was calling

for update/insert of the recently used filter. Due to this, Apex job was getting created for every filter selection on Converse Desk. That was unnecessary.

To solve this issue, the logic was changed so that the new apex job enqueued only when required.

2. **Issue:** Incoming text notification was not working properly.

Fix –

Incoming text notification was not working. In the Lightning environment, pushtopic subscription was supposed to happen via “empApi” which is integrated into the “streamingDataService” component. But, empApi was forced-disabled due to which the incoming text notification was not working.

To solve this issue, the forced disabled state of empApi was removed so that it would function as expected in the Lightning environment.

Patch 1.60.62

Date: 12th March 2021

Bug fixed

1. **Issue –** Unable to create manual consent record from consent object, when contact is entered

Fix –

Unable to create manual consent record from consent object, when contact is entered. In complianceService class, the GROUP BY clause is used to fetch the phone fields. In this case, the customer used the FORMULA field as a phone field to fetch the phone fields. Salesforce doesn't support the FORMULA field to be grouped. To solve this issue, the GROUP BY clause was removed.



Patch 1.60.60

Date: 23rd Feb 2021

Bug fixed

1. **Issue –** MMS messages were getting delivered even after they were Opted Out by the customer

Fix –

The customer had opted out of MMS messages in the contact object, yet they were receiving them.

In the MMSDetail trigger, SMS History records with MMS Type and failed status was not checked, and so the corresponding MMS records were triggered even after customers had opted out. To solve this issue, a check in MMSDetail trigger for MMS Type was added.

2. **Issue** – When the User was trying to write an Internal Note from the Conversation View on the Record Detail page, it was throwing an error message

Fix –

When the User was trying to write an Internal Note from the Conversation View on the Record Detail page, it was throwing an error message: Conversation: id of incorrect value: 000.

To solve this issue, the Internal Notes tab in the composer component in the Message Notification component was disabled, based on the origin Name value in requestContext.

For “Conversations” component changes are made in the ‘composerWrapper’ method of ‘MessageComposerServerController.cls’. Value of flag for ‘isInternalNoteEnabled’ is set to false if the the object name retrieved from available record id is not ‘smagicinteract__Conversation__c’



3. **Issue** – After an upgrade to the package from 1.46 to 1.60.56, the Converse App tab was opening in full-screen view.

Fix –

After an upgrade to the package from 1.46 to 1.60.56, the Converse App tab was opening in full-screen view. The tab notification panel was not accessible once the user navigated to Templates in the converse App. This was because the wrong event was used to replace the deprecated ones. To solve the issue, the “lightning:navigation” component was used to navigate from one component to another component in the lightning environment.



4. **Issue** – Send Bulk SMS button on maps was throwing an error, ‘Invalid ID’

Fix –

When the customer was using the “Send Bulk SMS” button on the maps app, they were getting an error, “Invalid id: 034A00003Dp7yzQAB”.

This issue was occurring because the package code was trying to trim the list of ids, assuming that the list has square brackets at first and last position. But in the real scenario, the square brackets are not present, thus the trim function was causing an issue. To solve this issue, the list of ids were trimmed based on the presence of square brackets. Used Substring function on ‘idList’ string only if square brackets are present in the string. Bulk messaging functionality now works flawlessly for normal list view and communities.

Patch 1.60.58

Date: 11th Feb2021

Bug fixed

Issue: The customer was receiving too many SOQL queries: 101

Fix:

The customer was getting too many SOQL queries from the SMS Magic Interact package.

Migrated the changes and resolved the issue related to SOQL queries.

Patch 1.60.57

Date: 2nd Feb 2021

Bug fixed

1. **Issue** – While creating a template, the customer noticed that 2 SMS credits were getting consumed for a message that was less than 160 characters in length. (158 characters with space). Ideally, if a message crosses 160 characters, it should consume 2 SMS credits.

Fix –

While trying to send the same template from the detail page, it was correctly showing that 1 SMS credit will be consumed.

This was happening because of the wrong syntax that was used where the text length was compared with a single message length (160 characters).

To solve this issue, the syntax was corrected and hence the comparison was correctly handled. Now, while creating a template, if the message length is below 160 characters, only 1 SMS credit is being consumed.

Patch 1.60.56

Date: 6th January 2021

Bug fixed

1. **Issue** – A component error was occurring while creating a template in the Converse App or while accessing the Converse App.

Fix –

Initially “force:navigateToComponent” was used to navigate from 1 Lightning component to another Lightning component. It was throwing an error while navigating to the Converse app.



To solve this issue, we changed “force:navigateToComponent” to “lightning:

navigation” as suggested by Salesforce Support.

Patch 1.60.55

Date: 4th Dec 2020

Bug fixed

1.Issue – The system was throwing a ‘Component error’ when the Converse App was accessed with any console other than the SMS-Magic app

Fix –



When the “Converse Apps” tab was accessed with any console app other than SMS-Magic and the user was trying to create a template, the system was throwing a component error. This was hampering the user experience and functionality. This was NOT observed when the Converse Apps tab was accessed within the SMS-Magic app. Create template action was navigating the user from one component to another. For this, we used the “force:navigateToComponent” event. This event was not handled properly in other console apps, since Salesforce was advocating the use of the “lightning:navigation” event instead of the “force:navigateToComponent” event.



To solve this issue, the “lightning:navigation” event was used instead of the “force:navigateToComponent” event to navigate from one component to another.

2.Issue – Not able to open PDF from converse desk via WhatsApp

Fix –



The issue was observed with WhatsApp Addon. When a customer received any incoming message with a PDF file, a message was displayed as ‘Broken UI’ and the user was not able to access the PDF file in the message. We embedded the PDF file into an iFrame to enable the preview of the file and to allow the user to click on it. With recent security enforcements, Salesforce deprecated embedding external URLs into iFrame.



To solve this issue, we changed the implementation. Instead of embedding the PDF in an iFrame, we are now showing a download button to let the user click on it and download the PDF.

3.Issue – Exception in Migration Batch; 11th step in the migration failed

Fix –

After upgrading customers from any version below 1.5 to 1.60, migration steps

were required to populate phone number and sender ID values in the newly created conversation records. The 11th step in the migration failed with an error. An external script was required to be run to achieve the required migration.

To solve this issue, the code of the migration step was updated to properly resolve fields and proceed with the data population required for migration.

Patch 1.60.54

Date: 12th Nov 2020

Bug fixed

1.Issue –Callout Exception error – when multiple jobs were queued, after the first job callout, the next jobs were failing due to DML being in progress.

Fix –

To address the CPU timeout issue during the bulk campaign, the approach to break the records count into multiple batches was changed. With this, each batch was getting processed separately. While doing this, after the first batch callout, DML was attempted to update the lookups for the records. At the same time, the callout for the next batch was getting processed, which failed because of the DML.

To solve this issue, the DML setting is changed; now the DML for updating lookup is attempted only after completion of all callouts.

2.Issue – Automated SMS not being sent as the system was throwing an error of 'available jobs limit consumed'

Fix –

The customer was not able to schedule a bulk campaign (converse app) as the system was throwing an error of 'available jobs limit consumed'. For instance, although the customer had consumed only 97 out of 100 available scheduled jobs in his org, the system was throwing an error of 'available jobs limit consumed'.

This issue was due to the check for the max scheduled jobs in the org, which was 95 instead of 100.

To solve this issue, the check for existing scheduled jobs was changed from 95 to 100. Now, even if there are 99 scheduled jobs present in the org, the user is able to consume the last one to schedule the bulk campaign.

3.Issue – Bulk campaign from the Converse App was not running properly

Fix –

The customer had created a lookup of a custom object (say object A) with another object (say object B). Object A was configured on the MOC, and the SMS opt out field of Object B was configured as the opt out field for Object A via lookup. While identifying the opt out value of the record, the error was logged in the debug log. There was no visual indicator for the error.

Because of incompatible syntax, the code was not able to resolve the SMS opt out field configured via lookup. This was throwing the error.

To solve this issue, we updated the syntax/code to properly resolve the lookup field value and obtain the boolean value of the opt out field.

Patch 1.60.53

Date: 26th October 2020

Bug fixed

1.Issue – Apex CPU Time Limit was getting exceeded

Fix:

Due to heavy processing on each outgoing record, the Apex CPU time limit was getting extended. So while sending bulk messages it threw a CPU time limit exception. To solve this issue, we processed the bulk messages asynchronously in batches of 50 if the count exceeded 100 records.

2.Issue – Permission error was getting displayed in Utility Bar and Converse Desk

Fix:

The users have access to the case object but not to the lead object. So, while reading messages of the case object, users were facing permission error of the lead object.

When the Utility bar is opened we check for conversations and while checking conversations we fetch all the MOC records. We were not checking whether the MOC record is having permission or not. To solve this, we added a check for permission of objects configured in the MOC.

3.Issue – The “Do not send Confirmation” checkbox was misbehaving; a confirmation message was being sent.

Fix:

The user was able to save the consent record without changing the previous consent status. Since the status was not changed, the old value of the “Do not send confirmation message” checkbox was referred and a confirmation message was sent to the record. To solve this, we added a validation on the consent edit UI. The “Save” button is disabled until the consent status is

changed.

4.Issue – Issue with Bulk SMS for Partner Community

Fix:

The processing time required for each message was high. To solve this, we processed the message asynchronously if the bulk count exceeds 100 records.

Patch 1.60.52

Date: 6th October 2020

Bug fixed

1.Issue – The label name in the 'To' field of the Conversation Component was not getting displayed properly.

Fix:

The user was not able to see the Lookup values in the 'To' drop-down field in the Conversation view while sending a message. To solve this issue, we showcased the label of the lookup field created on the MOC object instead of showing the label of the lookup object name.

e.g.

MOC Object = Contact

Lookup field on Contact = AccRelatedToContact (lookup of account)

Before fix : Contact > account > phone field

After fix : Contact > AccRelatedToContact > phone field

2.Issue – Record count under Campaign was not getting reflected as expected

Fix:

The record count under campaign was not getting reflected as expected, i.e., as per the actual field selected. Even if the custom field created by the user was selected instead of the default field in MOC, while running the campaign it displayed record opt-out count based on the default field & sent messages based on the custom field. Ideally, it should show the count based on the field that is used in the MOC, i.e., the custom field. It was referring to the default field and not the custom field because the hardcoded value of the default opt-out field was getting passed to record the count. To solve this, instead of passing the default value, we started fetching value from MOC & passed it for accurate record count.

Improvements

N/A

Patch 1.60.51

Date: September 16, 2020

Bug fixes

1. Issue: Bulk Action from Converse App remains in Running state for Object with Spaces in the name

Fix:

When users run a bulk action in the Converse App on an object, which has multiple words in the name field (space between the object name), the status would always remain in Running state. e.g., Student_Section__c. To solve this issue, we used a substring function to split the query that we receive from Salesforce.

Improvements

N/A

Patch 1.60.50

Date: September 08, 2020

Bug fixes

1. Issue: Issue with the SF campaign.

Fix:

Whenever customers tried to send messages through Campaign with both, contact and a lead member, the SMS history for contact reached the portal but not the lead. We executed Campaign Batch jobs differently for both, Lead and Contact, to solve this issue of multiple primary objects.

2. Issue: Issue with SMS Scheduled Record.

Fix:

When the message is scheduled through a Single or Bulk SMS button, SMS Scheduled record is created with all the field values such as SenderID, Text, etc. But when we scheduled the message through the Converse App, it created an SMS scheduled record without the SenderID. To solve this, we populated the fields required by the customer so that they can create a correct report on the SMS schedule object.

Improvements

N/A

Patch 1.60.49

Date: August 25, 2020

Bug fixes

1. Issue: Primary object lookup happening intermittently for incoming messages.

Fix:

Removed the unwanted, unformatted mobile number check from the logic of the primary object lookup which was causing this issue. With this fix, the primary object lookup should happen for each incoming message.

2. Issue: While sending outgoing messages from the record detail page conversation view, the direction field was not getting updated correctly in the conversation object.

Fix:

As a part of performance enhancement, we had modified conversation assignment logic because of which the last message direction field in the conversation object was not getting updated for the record detail page. As a part of this fix, we have corrected the logic so that it updates the last message direction field in the conversation object only after the conversation assignment is done.

Improvements

N/A

Patch 1.60.48

Date: August 19, 2020

Bug fixes

1. Issue: Component error for the non-SMS-Magic users. This is due to the latest critical update imposed by Salesforce on 9th August.

Fix:

The issue was on the Conversation View on the Utility Bar and Conversation View on the record detail page. Made changes in the notification manager component to handle the exception thrown by Salesforce and show the correct notification to the users.



2. Issue: Utility bar back button was not working properly.

Fix:

Here the back button of the utility bar was not functioning properly. In the

case of a closed conversation's component, the selected RecordId was coming intermittently as null. So removed the condition and replaced the logic which will include the button name to decide whether to close the tab or take the back action.

3. Issue: New conversation created for every incoming SMS.

Fix:

The issue happened because the conversation logic code was reverted to 1.58 implementation from the routing logic. The root cause for the failure of older conversation logic implementation was that 1.58 did not have the country prefix in the SMS History whereas versions 1.59 and above have the country prefix in the SMS History. We implemented the logic of 1.58 to solve the issue.

4. Issue: Converse bulk issue.

Fix:

When we call the API while running the bulk campaign, Salesforce returns fields of the object in Query. So, we were getting a few unnecessary fields in the query. This was throwing an error in Apex Job & status was not changing. To solve this, we split the query into 2 parts (Left & right side of the "FROM" field). With this, the unnecessary fields from the query from Salesforce were ignored and the issue was resolved.

5. Issue: Error while sending SMS from the Converse app.

Fix:

This issue was there because a static variable was used to scan the depth of the master-detail relationship. When multiple records got edited at the same time, this static variable overflowed to scan the fixed depth of the master-detail relationship, and thus the issue occurred. We replaced the static variable with a normal variable.

Improvements

N/A

Patch 1.60.47

Date: August 04, 2020

Bug fixes

1. Issue: Utility Bar Issue- On load dropdown, the selected value is not getting displayed in Chrome, it's working in Firefox.

Fix:

When the Utility bar on sales console is enabled, read/unread drop-down appears. That dropdown was blank for Chrome, it did not display the required values. The utility bar showed blank values on the initialization of filter

dropdown in the Chrome browser. We fixed this by making HTML changes.

2. Issue: Utility Bar issue- User got redirected to the Conversation Inbox when clicked on CRM action (i.e view contact, view lead, etc.)

Fix:

Corrected the conditions to check whether the utility bar is working in contextual scenario and implemented fail-safe 'try:catch' in 'StreamingAPIController.cls'

Improvements

N/A

Patch 1.60.46

Date: July 28, 2020

Bug fixes

1. Issue: Users Getting Error "Object Name: Organisation"

Fix:

Fixed the issue where users got an error "Object Name: Organisation" does not have permission to read fields.

The Error Occured when users click on Conversation Record.

2. Issue:Related conversations are not loading on Converse Desk.

Fix:

Fixed the issue where related conversations did not load in the Converse Desk.

Improvements

N/A

Patch 1.60.45

Date: July 27, 2020

Bug fixes

Issue: Utility Bar Notification Not Working.

Fix:

Fixed the issue where the users did not get notified when an incoming message was received. The users needed to refresh the page to see the notification.

Improvements

N/A

Patch 1.60.44

Date: July 23, 2020

Bug fixes

1. Issue: ApexCPU Time out error.

Fix:

Fixed the issue when

- Customer tried to send messages to bulk records. (He has created a list view and tried to send messages to all records.)
- Flows triggered bulk messages (190+) for given conditions.

2. Issue: Unable to use 'Time' datatype fields while sending messages using templates.

Fix:

Fixed the issue where the customer was not able to use "Time fields" while sending the messages. For a custom created template with a time field, the user received an error message "Invalid date/time: 00:00:00:000z".

3. Issue: Error message "Attempt to de-reference a null object".

Fix:

Fixed the issue where the customer was getting "Attempt to de-reference a null object" when a template is selected while sending a message from Send Bulk SMS or Send SMS button, if Auto Append interval was enabled.

Improvements

N/A

Patch 1.60.41

Date: July 06, 2020

Bug fixes

1. Issue: Some conversations not visible under Converse Desk.

Messages for particular records were not visible in Converse Desk.

Fix:

Added check for Access check for Push Topic in professional org.

Note: Users need to add the domain URL manually in remote site settings. Then lookups and list views will be accessible and everything work as expected.

Patch 1.60.40

Date: July 01, 2020

Bug fixes

1. Issue: Error while accessing Converse app.

All the users faced the below error while accessing the converse app.

“Aggregate query has too many rows for direct assignment, use FOR loop” The issue occurs when customers have more than 200 converse app actions for each converse app.

Fix:

The count will be displayed as 199+ if the number of actions exceeds 200 records. Although, users can always open the converse app to view the exact amount of actions present in the app.

Improvements

N/A

Patch 1.60.38

Date: June 30, 2020

Bug fixes

1. Issue: Component error.

Fix:

Fixed the intermittent issue where the customer was facing a component error while accessing a contact record (any contact record).

2. Issue: Unable to schedule SMS from the Partner Community license.

Fix:

Fixed the issue for scheduling SMS the users with community licenses.

3. Issue: Spell Check not working.

When the Customer Opens a New Message Window in Utility Bar & Typed Text, customer corrects the spelling using standard spell check utility however when customer posts the text, the words corrected in spell check are not updated. It sends incorrectly spelled words.

Fix:

Users can send an SMS from the utility bar after changing the incorrect word.

Improvements

N/A

Patch 1.60.37

Date: June 26, 2020

Bug fixes

Issue: Intermittent issue for notification in Utility bar.

Fix:

Fixed the issue with the utility bar as it is not getting blinks automatically on incoming SMS.

Improvements

Changes in SMS Magic Interact Package package to adopt the changes for Australian server addition on the portal.

Patch 1.60.36

Date: June 17, 2020

Bug fixes

1. Issue: ComplianceConsentList component is not showing the changes when we refresh the Lightning record tab.

Records were not visible in conversation view in Converse app.

Fix:

In the Converse App, conversation component, only the messages sent from Converse App will be displayed

2. Issue: Receiving an error message on clicking Send SMS Button.

When the customer clicks the Schedule/ send SMS button on the Enquiry object. The customer was getting the following error message:

“This page has an error. You might just need to refresh it. Error in \$A.getCallback() [Cannot read property ‘setListViewData’ of undefined]

Callback failed:

apex://smagicinteract.LightningBulkSMSController/ACTION\$lightningBulkWrapper

Failing descriptor:



Fix:

After clicking on the button the error message will not appear.

3. Issue: Validation rule on SMS history does not display the error message on the UI.

A customer had a validation rule on SMS history. This will not allow users to send SMS to any mobile numbers. The mobile number should be associated with a contact record in Salesforce to send out the SMS.

When users do send SMS to mobile numbers, they get a message on UI saying the 'Message has been sent' while the message has not been sent when checked in the error logs object.

Fix:

The validation rule error message is displayed.

4. Issue: Messages did not appear in the Embedded Desk.

The Owner of Conversation & SMS History gets updated to the Owner of Sender ID Correctly. The Notification in the Embedded Desk (Utility Bar)also blinks to the user but the conversation does not appear in the unread section.

Fix:

Messages will appear in the Embedded Desk.

5. Issue: Record owner sender did not work.

This is a coding bug. This was modified when we released a patch for a customer, as they wanted to send SMS to the user object. They wanted User as a MoC object.

Fix:

The Record owner sender id will be displayed.

6. Issue: Error message during upgrade on sandbox.

During Sandbox org upgrade from 1.55.7 to 1.60.24, we got below error message:

```
The user license doesn't allow the permission: Manage All
smagicinteract__Consent_Audit_History__c, Details:
SMS_Interact_Permission_Set: The user license doesn't allow the permission:
Manage All smagicinteract__Consent_Audit_History__c
```

Fix:

We removed the View all, and Modify All permissions for the following objects: Compliance configuration audit, Consent Audit History, and Consents.

Patch 1.60.34

Date: June 03, 2020

Bug fixes

Issue: Non-admin users are unable to send SMS. Non-admin users are unable to send SMS. They see the error on the conversation component and also when using the 'Send SMS' button.

Error – 'ObjObject name Organization does not have the permission to the

field: read'.

Fix:

Commented security check for Organization object while checking for the namespace, so that non-admin users can send SMS.

Improvements

N/A

Patch 1.60.28

Date: May 12, 2020

Bug fixes

1. Issue: No automatic update of 'complianceConsentList' component.

For users with the compliance consent list component, embedded on their record detail page of contact, when a new record is created or an existing record is updated, it does not automatically reflect on the record detail page. They had to refresh the browser so that the new /updated record will be reflected on the record delta page.

Fix:

Compliance consent list components can be updated without a browser refresh. An additional button on the UI will refresh the list.

2. Issue: Consent record alignment.

Fixed consent record alignment issue due to additional refresh button on the record detail page.



Additional refresh button & fixed alignment issue

3. Issue: Plan upgrade with existing active license.

Users were unable to change to a new plan when their existing license is active.

Fix:

Users can change to a new plan when their existing license is active. ie. If the user currently has a 'Start your conversation' plan then they can change it to "Grow Your Conversation Plan".

4. Issue: Trial plans (Both) are reflecting in SF org, even after setting non-trial plan (Grow plan) from back-end.

Trial plans(Start your conversation, and Grow your Conversation plan), reflected in SF org, even after setting a non-trial plan(Grow plan) from the back-end.

Because of this issue, customers could only see the SMS Credits specific to the trial plan in their Salesforce org.

On the portal, the total number of available SMS credits as per the plan was displayed.

Fix:

If the plans are updated from the backend then that information will be updated to display the active plan on UI.

E.g. A 'Grow your conversation' trial plan is updated to any full plan, then it will be updated in the Plan manager section of Custom settings as well as on UI.

The following is the screen prior to fixing the bug:



5. Issue: Trial plans (Both) are reflecting in SF org, even after setting non-trial plan (Grow plan) from back-end.

The Client users were getting error messages when they clicked on the consent list view component, added on their lightning page.

Error messages:

- I. No phone numbers are available on this record.
- II. Component error message

The issue occurred when cross-referenced objects were using the consent list component.

In this case, while adding consent list component list on the Case object and then tried to set consent for the contact these error messages occurred.



Fix:

Fixed the issue with Consent lightning component, when fetching mobile number from lookup relation. No error message will be displayed, and the 'Add Consent' button will be clickable.

6. Issue: Image upload size limitation on SF Mobile app.

For SF mobile app, allowed images size is up to 2.5 MB. An error message should be shown if image size >2.5 MB.



(Error message for file size >2.5MB, *before fix*)



(*Error Message before the fix*)

7. Issue: 'Callout Exception' error message for some of the messages.

Few of our customers have reported that their users intermittently faced an issue where a few messages were failed with an error message "Rejected from portal" or "Callout Exception error".

Also, there was not enough information in error logs related to this issue,

making the tracing of the cause difficult.

Workaround:

For any callout exception error message, an error log will capture more data using the system provided a function for troubleshooting than before so that the tracing actual reason(s) for the "Callout Exception Error" will be easier.

The customer would need to set up a debug log of appropriate level in their salesforce org to get these extra logs.

ie. We are providing users with the enhanced error logging that will help us to trace the root cause. With this patch, a stack trace only for "Callout exception error " is implemented when we sync with Portal API.



8. Issue: Unresolved templates for messages from SMS Magic Mobile app.

Templates are not resolving properly when the messages are sent from the SMS Magic Mobile application. When the user tries to send a message with a template attached to it, that includes the merge field, no data was available in the merge field.

Fix:

When messages are sent from any source, the template should resolve itself ie. merge field data will be available.

Security review changes

1. Removed jquery/angular libraries from the package as we are no longer using them.
2. Fixed Injection and CRUD/FLS bugs./[SMI-6105](#)

Improvements

N/A

Patch 1.60.24

Date: April 20, 2020

Bug fixes

1. Issue: Non-conversion of a few outgoing and incoming records.

Migration scripts for certain steps failed with an error message " First error: Attempt to de-reference a null object" in debug log. That resulted in non-conversion of a few outgoing and incoming records. i.e leads were not converted into contact/Opportunity/Account.

Fix:

Fixed Data Migration issue for leads which were missed for being converted into contact/Opportunity/Account.

2. Issue: Converse app tasks failure with the message 'No response from MessagingService'.

When the name of a custom object does not match with the name of the lookup relationship on the SMS history object, the converse app tasks fail with the message 'No response from MessagingService'.

Eg. For a custom object named 'xyz', if the user creates a lookup on an SMS history object, with a different name 'abc_xyz'. And, also creates a custom builder to create a converse app task whenever any field of the custom object is changed.

The Converse app task will be created with the error, message "No response from MessagingService".

Fix:

No error message for the mismatch in the name of a custom object and that of used to create the lookup relationship on SMS history object.

3. Issue: Converse app task failure.

In the Master-Detail relationship between Salesforce objects, when a Standard Salesforce object was the Master, and a Detail object was a Custom object, the flows showed an error message.

The automation failed when messages were sent from the detail record as a Detail custom object did not contain the field for the Master object.

This issue was only observed when the Master object was a Standard Salesforce object, not custom objects.

Fix:

The Master-Detail relationship will now work for standard objects as well to send the messages using Detail – custom objects.

Improvements

Supporting messaging to internal users with User Object(Limitation – SMS History records won't be created)

Patch 1.60.22

Date: April 09, 2020

Bug fixes

1. Issue: A intermittently error message on UI

Fixed a UI intermittently error message:

"This page has an error. You might just need to refresh it [Action failed: aura: iteration\$controller\$itemsChange [Maximum call stack size exceeded]]"

on Converse setting on the selection of plan.



2. Issue: Wrong Consent mode was shown when the user sends SMS from a bulk SMS source.

Wrong Consent mode was shown when the user sends SMS from a bulk SMS source. i.e. Global consent mode was shown when the user sent SMS from a bulk SMS source.

Eg. In the converse setting, users selected "Consent not required " in the Consent method & Content section(as global consent) for their Marketing content. Also, selected " Consent mode – Required " for their Marketing content.

In the template for Marketing content the consent type appears as – required. But when the same template is selected for sending bulk messages, appears as " Not applicable" i.e. same as global consent.

Fix:

Appropriate consent mode is now shown when the user sends SMS from Bulk SMS source. The consent mode appears as per the selection made only for the content-specific consent /SMI-5661

3. Issue: A blank space appears when MMS is received in the Facebook integration.

For Facebook integration, a blank space appears when MMS is received in the conversation window.



Fix:

Incoming images from Facebook are displayed in SF. (Converse desk, Conversation component).



4. Issue: Auto append not working for messages from templates and Phone, and Phone +senderid combinations.

If a user sends messages using the template with opting for an Auto append, for Phone, and Phone +senderid combinations, the auto append messages were not included in the template messages.

This issue was observed only for the messages from templates and Phone, and Phone +senderid combinations.

Fix:

Auto append message will work for phone, and Phone+ sender id for the selected template while sending SMS. This can be confirmed from SMS history.

Improvements

1. Issue: Failed license allocation due to permission issues.

For a few customers, Due to some permission issues – License allocation is getting failed while installation of the package. But because of such restrictions, licenses can't be added from an external interface. The only option remaining is un-installing & installing packages.

Solution:

Removed the restriction for support of license handling from the external interfaces and restricted the user from assigning multiple licenses to one user.

2. Issue: Guest users unable to send messages despite granting permission(s).

The converse app task is getting created with success status but the history record is not getting created.

Guest user from customer's org was unable to send automated messages despite granting him following permissions:

- Object and field-level permissions
- created sharing rules on required objects
- View All Custom settings
- SMS License
- custom settings, apex classes and custom metadata on at profile level