# Building Highly Customized Communications by Using Dynamic Data Access Methodologies in Communication Management Tools

**Renuka Kulkarni**

Independent Researcher
USA
Renukak12@gmail.com

**Abstract**
**In industry, communicating the right message to the right audience at the right time is critical. Customer communication management (CCM) tools can be implemented as a service per customers' requirements for efficient communication systems. CCM tools provide a solution to create communication templates by integrating with enterprise data, and they can scale from supporting small departments to multi-departmental organizations by building billions of communications. It creates highly personalized communication by integrating with systems like ERP, CRM, or any upstream system connecting to customer data.**

**This article takes a deep dive into Building highly personalized templates (for any industry, such as healthcare, banking, or utility) by interacting or utilizing data residing outside the code base or outside the organization. CCM tool's dynamic data access module makes importing and embedding external contents (text, images, PDFs, etc.) into customer communication possible.**

**Keywords: Dynamic Data access, Placeholder routines, Customer communication**

## Introduction

Dynamic data access refers to the ability to access and manipulate data in a flexible and adaptable manner, especially in environments where the data or source can change dynamically during runtime. Dynamic data access is commonly used in computing, primarily within databases, APIs, and web services, to access data that is not static and may vary due tofactors like system conditions or external sources.

In Customer communication management (CCM),dynamic data access plays a vital role. Implementing dynamic data access in CCM allows businesses to retrieve and use real-time customer data to personalize communications. It also can provide real-time updates and information to customers, improving responsiveness and satisfaction; thesystem can generate messages or documents on the fly based on current information, ensuring that content is not outdated or irrelevant.

## Need for dynamic importing.

Customer Communication Management solution follows a typical software development project lifecycle for developing communication for an organization. It startswithrequirements and convertsthem into technical specifications, code, tests, and user acceptance testing for deployment and post-production support.

In the code phase, CCM developers create specific templates per the requirement by converting business logic into code, integrating upstream data, and adding user-specific content (communication text/letter verbiage) by designing templates in the template database (e.g., Oracle or DB2). In today'sdynamic software industry, requirement changes are not uncommon. In case of a change request during the development process or post-deployment, the template database must be updated to incorporate changes requested for business needs or compliance requirements. These requirement changes are segregated into two categories in the Customer communication domain.

1)Content change requirement, where letter/fax/email in general communication template paragraphs are updated to incorporate different verbiage (examples: Customer-specific verbiage,Generic verbiage in the communication)

2)Business logic changesthat change the flow or logic: In the case of alteration of business logic,it is inevitable to update the code, make changes to the database, and complete the entire software cycle to implement the change. For events where generic content is frequentlyupdated,we can leverage dynamic data access methodologies offered by CCM tools to import changed contents dynamically without code change.Instead of constantly updating Library objects, we can leverage the Dynamic Import functionalities provided by the CCM tool to dynamically import information stored in a centralized repository or folder on the network. Dynamic data access modules can access data from various sources and systems and create dynamic communications. This module is handy if content/Text changes often or must stay in a specific format. Some common examples of generic content updates are:

- Bank statements containing check images.
- Insurance policies with frequently updated provisions.
- Reports or documents featuring static images with large file sizes.
- Financial reports displaying daily stock prices or sales data.
- Newsletters incorporating articles and images from various departments.
- Newsletter or statement pages generated as HTML output.
- Marketing materials that require a fixed format.

Using a dynamic import method to access data residing in the Enterprise / or outside the Enterprise

Flow diagram of input-output, processing engine, and integration of external data with typical Application set up

**Dynamically Importing Content**

Customer communication tools help to manage frequently changing content or file formats not natively supported in customer communication. These modules allow external storage of images and text files, enabling their on-demand import during runtime.

Content Import imports images and text files into documents at runtime or design time. When referencing, the file import process allows the import of entire PDF and RTF files into output from the location specified by placeholder variables that contain the paths to the files. Content is imported in one of two ways:

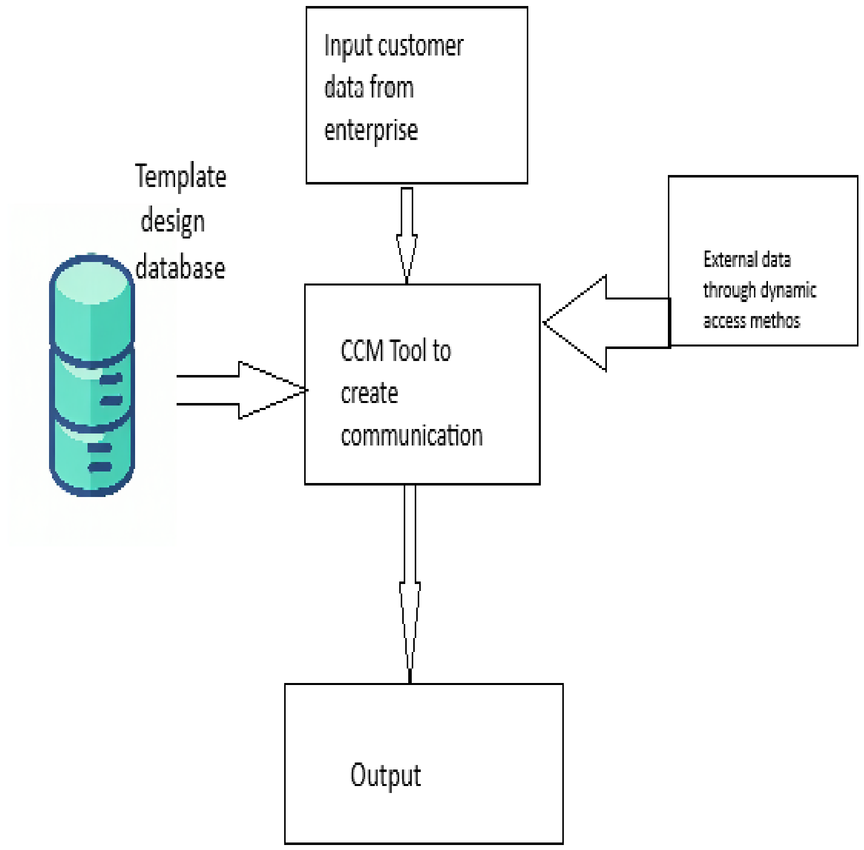

Input customer data from enterprise
Template design database
External data through dynamic access methos
CCM Tool to create communication
Output

**Figure1:Pictorial interpretation of connecting to external data sources**

One way to import external data is to directly place imported text, RTF files, and images into the application, converting unsupported formats into compatible ones. Another method is referencing text files and images without adding actual data in the code /application. In this method, files, images, and PDFs are referenced on the final communication without adding them to the code base, making the size of the executable (Codebase for creating communication) lighter. This method also allows conditional text selection and incorporates logical processing so only qualified text or documents are populated on the final communication.

## Placeholder concept

The placeholder concept refers to a temporary construct representing a value, object, or piece of content intended to be defined or replaced in development or execution. Acting as stand-ins, placeholders facilitate program flow, UI design, and data processing without requiring the final details upfront.

Communication composing tools use the placeholder concept to access the data residing outside the code base or in different locations to be added to the customer letter, email, or fax communication. These placeholders are critical for personalizing documents and ensuring the content is populated correctly at runtime.

Placeholder methods playa critical role in Dynamic data access implementation. The Dynamic Content Import module allows the creation of placeholder variables. This variable is called a placeholder as its primary purpose is to hold the path of resources (Text, Image, or any file type).Placeholders can be a string, date, or a path to access the external file.Placeholder variables can be used to import extensive text data, and the PDF Import can be used as an image module to hold a place in anobject for external content to be imported when the engine is run. For example, to dynamically import a TIFF file into a communication, a placeholder variable needs to be created supporting the TIFF file type and placedon design objectsto get the TIFF image imported at run time. Data that can be generally imported using placeholder is as follows:

1. Plain text
2. RTF text
3. B&W TIFF
4. Image resource
5. Overlay resource
6. TIFF
7. JPEG
8. PNG

**Summary of workflow:**

-Analyze the requirement to understand what type of data needs to be imported
-Create a placeholder variable data type based on the need of imports (text, PDF, TIFF)
-Provide the details of the path where the dynamic data is accessed
-Check the accessibility if the data is residing in any external folders and set up the connectivity if needed
-Decide what type of access method needs to be used:Direct File Access or an API.
-Configure the placeholder variable with path details
-Place the variable on the object where the content is supposed to be populated.

**Case Study to import dynamic text content from the business site**

Consumer Domain: Banking customer HSBC Bank

Project: Integrated customer communication Letters project

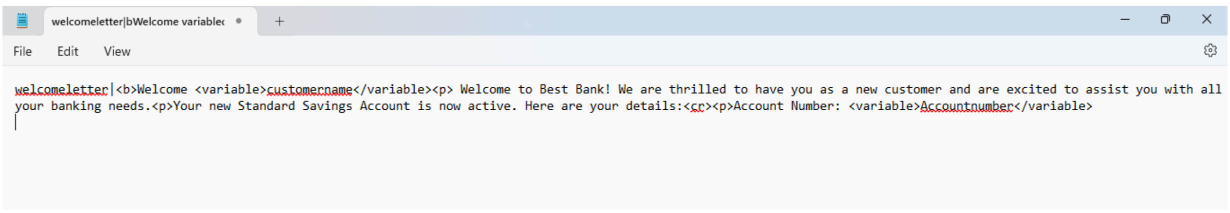
Business Problem Statement: As a part of unified solution implementation,the business requested to keep the letter verbiage dynamic except for customer data (names and addresses, etc. details). The entire Letter verbiage, except Customer-specific information, was supposed to be dynamically provided by the business development team. No static contents were to be added to the Letter/design database. The aim was to keep the letter content outside the code base to keep a lightweight template design and make code changes to a minimum level, keeping letter verbiage outside the code.

Problem resolution:

As the project required no static content for the letter body to be added to the database, a Dynamic data access module was implemented to resolve the problem. As text content/letter body text was stored in the business repository and provided by the business development team, the project team decided to utilize the pass-through method available in dynamic data access to solve the problem.

As the external text contents were supposed to be referenced, a file Reference file (a text file mapped as a key and value pair was created). The Key was a condition to choose the required letter template, and the value field in the file was the actual contents of the letter.For the contents of the letter, a placeholder variable was used to hold dynamic content provided by the business team. Placeholder variables can populate tag-based<b>,<u>,<variable> text to import formatted static content to achieve the required branding and presentment of the letter.

Data map of the reference file and placeholder variable.



```
welcomeletter|<b>Welcome <variable>customername</variable><p> Welcome to Best Bank! We are thrilled to have you as a new customer and are excited to assist you with all
your banking needs.<p>Your new Standard Savings Account is now active. Here are your details:<cr><p>Account Number: <variable>Accountnumber</variable>
```

This Solution worked for the customer. It met the requirements, and project showed in the rapid developmentin each development phase.As the data was dynamically accessed, development and testing time was reduced drastically.

**Advantages of implementing dynamic data access methods:**

Advantages:
As explained in this article,dynamic data access methods are the techniques used to retrieve data at runtime rather than being hard-codedand offer several advantages. These methods enhance flexibility, scalability, and adaptability in software systems.
-Dynamic methods allow applications to adapt to changing data structures or formats without needing extensive code changes.
-Applications can work with varying data sources, such as databases, APIs, or files, without rigid constraints.
-Eliminates the need for predefined access paths or rigid schema bindings.
Promotes cleaner, modular code by abstracting data retrieval logic.
-Supports growing datasets or evolving database structures. Enables seamless integration with additional data sources without rewriting core logic.
-Enables real-time data retrieval based on user interactions, context, or environmental conditions.
-Generic, dynamic access methods can be reused across multiple components, reducing development time and redundancy.
-Encourages consistent patterns for data access across the application.
-Changes to a data structure (e.g., adding a new field to a database) require fewer modifications, typically only in configuration or mapping layers.
-Centralized access logic ensures easier debugging and updates.
-Support for Diverse Data Formats,inclusion text, images and XMLs
-Enables fetching live data instead of relying on precompiled or cached information.

-Useful in scenarios requiring up-to-date information, such as financial dashboards or live monitoring systems.
-Dynamic methods support custom queries and on-the-fly aggregations, enabling personalized reports and analytics.
Drawbacks of Dynamic Data Access
While dynamic data access methods offer significant advantages, they also have potential drawbacks. These disadvantages should be carefully considered to avoid performance, security, and maintenance issues.

-Dynamic data access methods are more complex to design, implement, and debug than static methods.
-Requires additional logic to handle unexpected inputs or varying data structures.This can lead to higher development and maintenance costs.
-Improper handling leads to vulnerabilities.
-Dynamic data access requires careful implementation of input validation and sanitization.
-Many dynamic systems use metadata (e.g., schema definitions and data dictionaries) to operate correctly.
If metadata is missing or inconsistent, it might lead to incorrect design function.

**Conclusion:**

Implementing Dynamic access solutions is efficient and can provide an efficient architecture by providing flexibility and reducing development time. As the data resides outside the database or codebase,the size of the code package is lightweight, and development efforts are minimized. However, it adds dependency and responsibility on an outside system to provide up-to-date and specifically formatted data. Creating a dynamic data access setup also involves implementing other software for delivery channels. Overall, this solution has many advantages, but implementation decision solely depends on project requirements and client licensing agreements.

**References**

[1]"What is CCM? A guide to Customer Communication Management (CCM)".sendbird.https://sendbird.com/blog/what-is-customer-communication-management-and-how-it-benefits-your-business.(accesses Dec 19,2018)

{2}Jason."OpenText: OpenText™ Exstream 9.5 – Product overview." OpenText.https://www.OpenText.com/file_source/OpenText/en_US/PDF/OpenText-po-exstream-9.5.pdf".(accesses Sep 19,2016)

[3] Jason Rault. "OpenText: OpenText™ Exstream – ProductOverview".OpenTextm.https://www.OpenText.com/file_source/OpenText/en_US/PDF/OpenText-po-exstream-16-2.pdf. (accesses Apr 25,2017)

[4]"OpenText Communications".https://www.opentext.com/products/customer-communications-management.(accesses Dec 19,2018)

[5]Antora-Fani Dima,"Insight into Customer Communications Management: Current state and future projections",researchgate,https://www.researchgate.net/publication/316191532_Insight_into_Customer_Communications_Management_Current_state_and_future_projections(accesses Dec 11,2018)