Challenges of Integrating LLMs like ChatGPT with Enterprise Software and Solving it with Object Messaging and Intelligent Objects as a new Software Design Paradigm

Massoud Alibakhsh
Co-Founder & CEO
Xeba Technologies, LLC
Atlanta, USA
massoud@xeba.tech

Abstract—The proliferation of AI tools, such as ChatGPT and other Large Language Models (LLMs), has raised expectations regarding their widespread adoption in the workplace. However, this paper delves into the limitations that hinder the deep integration of LLMs with existing software applications used in business environments. While LLMs excel in processing natural language, conventional software applications mostly communicate structured information in a rigid fashion. To bridge the gap between human usage of natural language and the structured data utilized by existing applications, communication tools such as email and Slack emerged. While these tools filled the void by accommodating the informality of chat-like interactions, they struggled to incorporate the applications' structured data. The prevalent form-based models employed by existing software applications present a fundamental obstacle for a meaningful AI integration. This paper examines the challenges of integrating LLMs with form-based software, shedding light on the associated complexities and introduces a new design paradigm, Object Messaging Model and Intelligent Objects (OMIO).

By leveraging this model, natural language and structured data become logically combined while enabling seamless integration of LLM capabilities within the software application. This paper provides insights into the potential of the OMIO and paves the way for unlocking the transformative power of LLMs, resulting in highly enhanced productivity within the contemporary business landscape.

The Object Messaging paradigm offers an elegant path for deep integration of LLMs and software applications with structured data, while optimizing communication among humans in the workplace producing humans and AI collective intelligence.

Keywords—AI, Object Messaging Design Paradigm, Intelligent Objects,, OMIO, ChatGPT, LLM, Communication, Business Applications, Collective Intelligence, Neural Nets, Optimizing Links, Deep AI Integration, AI Fusion

I. BACKGROUND

In the beginning, there were paper forms, then entered the computers.

All or most business applications are essentially communication tools. Traditionally, they were based on the form automation model which was inherited from paper forms. As computers found their way into the business world, programmers would cooperate with business managers to understand existing paper based processes in the business world. They would usually interview key employees and review the paper forms that were used in the established business process to communicate information from one person or group to another. These forms usually contained captions along with some structured data filled by different people and handed to the next person or group to use and/or fill out their designated section on the form. The process of creating automation usually started with some type of flow charting which later was formalized within a data flow and/or control flow diagram as the first step toward the automation design.

The resulting program automation usually resembled all or parts of the paper form on the computer screen. Each user would have his own part of the form on his screen to review or enter his part of the data. The data was usually stored in a database for later querying and generating reports. This provided a significant improvement in terms of speed of information transfer, more accuracy in data collection as the computer program could include syntactical editing rules as well as enforcing business rules that would apply to the data collection. Additionally, the storage of the data in a computer database allowed for flexibility and improvement of report generation with much higher speed.

II. TRANSITIONAL MODELS OR PLATFORMS: FIRST GUI AND THEN THE CLOUD

With the emergence of Personal Computers running on Local Area Networks (LANs) and the advent of Graphical User Interface (GUI) in the mid 80's up to early 90's, most of these programs were upgraded or completely rewritten to take advantage of this new modern form of interface [10]. Software companies rushed to provide business applications offerings with the new GUI as this became the dominant competitive advantage in the business world[11]. Companies such as Apple ushered in the first GUI operating systems and later

Microsoft followed suit. The GUI improved the basic human and machine interaction using the icon paradigm along with new data entry gadgets such as the now ubiquitous mouse on every desktop. But the fundamental model underlying the process design for all these applications was still the paper form. Programmers continued to use the paper process as the starting point for creating business automation. In fact, the new colorful screens with icons and images further motivated the perpetuation of the paper process paradigm due to its apparent fidelity with the physical world. The unconscious assumption was the existing familiarity of the physical forms reflected on the computer screens eased the users' anxiety about this new business tool, namely the computer, which had up to that time a formidable and intimidating reputation for many users in the business world unfamiliar with such sophisticated tools.

The rise of the Internet and the emergence of cloud computing[2] forced most of these business applications to leave the world of LANs and simple client/server architecture and be redesigned for the new public network, the Internet or the Web[12][13], and its emerging multi-tier distributed platform, the cloud[3][14]. But the form based paradigm of application design remained intact. These business applications take in structured data and operate like basic state machines. But humans communicate with natural language. That's where the shortcomings of the traditional form-based software applications lie. The natural language type of communication continued in the form of memos that were usually instructions from above relayed by mid level managers all the way down to the rank and file or summary reports compiled by mid-level managers to flow to the top informing high level executives of the goings on in the production environment.

III. EMAIL AND THE PROMISE OF PANACEA: THE FIRST WAVE

The lack of natural language support by software applications maintained the necessity of paper memos. And in the mid 90's, a new "Killer Application", as they were referred to then, entered the business realm with fanfare and lots of excitement[4]. Out goes the memos and the expectation was that the new electronic form of the memo would speed up the communication using natural language. With fancy inboxes and the ability of organizing these 'electronic memos', each user can create their own database of information about their particular work or project. It didn't take long for many to understand that this new wunder tool opened a Pandora's box of its own[5][6]. Aside from the external spam emails, even the number and volume of the internal emails were increasing on a daily basis due to the inherent flaw in the email model, not understood at the time[1]. The 'cc' and 'reply all' features only exacerbated the problem. The email model places the burden on each individual to process and understand the information, and based on his own understanding of the organization's workflow and relevant stakeholders, to figure out how to route that information to the right stakeholders at the right time[1]! The routing problem may be less critical for small organizations where the number of individuals are no more than a handful and the workflow is simple. But as the size of the organization grows, it becomes an impossible task

for any human. And the timeliness of information exchange remains a challenge even for small organizations since that is also dependent on humans!

That's how social networking tools for business such as Yammer[7] and later Slack[8] found their way into the corporate world to deal with email problems. They changed the communication model by shifting the pivot from the individual to the subject line [1]. The assumption here was that users need not identify other users for routing information. Using these new tools, a manager with some knowledge of the workflow creates a series of communication channels with names that closely resemble aspects of the workflow, then the users simply subscribe to relevant channels and enter or receive the information from these channels. While this was an improvement over email for internal communication, it not only failed to address the root of the problem, it created some of its own[9]. Let's also keep in mind that the new channel communication was still an addition to the existing business applications which consumed structured data on forms displayed on computer screens. And now you have company information piling up and scattered within emails and Slack channels as well as the existing business applications.

Form based applications communicate their structured information based on user roles and access rights. When the user is entering data, she usually has no choice in determining the routing of that data. The job of routing that information has been predetermined by the designers of the application based on user roles and a predetermined dataflow. For example, in an accounting system, a user entering data in the accounts receivable screen (form) saves her data. Later, some other user with higher access rights requests a balance sheet or income statement or views the general ledger screen (form). Between these two events, there may be lots of conversations amongst relevant stakeholders embedded in emails and channel chatters. That information is usually filed away in emails and channel silos and for all practical purposes forgotten. One can intuitively surmise the need to merge all the relevant conversations with the structured data that exist in the database.

IV. EUREKA: OBJECT MESSAGING AND INTELLIGENT OBJECTS (OMIO) THE 3RD WAVE [PATENTS PENDING]

The Problem:

Before we delve into the solution and discuss the characteristics and benefits of the Object Messaging Model, let's first describe the problem in an abstract way using a biological cell as a metaphor which some may rightfully view as closer to reality than just a metaphor.

Modern organizations are composed of humans and workflows. They are mostly organized hierarchically in a pyramid like structure. Workflows are essentially abstract versions of assembly lines invented in the early 1900s. Human members are organized in specialized groups with specific functions (Production, Support, Sales, Marketing, Admin, etc.) around the proverbial assembly line we call workflow. They are gathered in this superorganism[15] for a common purpose.

And that is to produce specific products and services. The pyramid structure and the assembly line or the workflow can be viewed as the skeleton and the heart of this organism respectively. Various specialty groups are analogous to its organelles. The workflow has a theoretical optimal tempo to produce the final products as fast as possible with the existing capacity of all the participants. The capacity of each group on one hand depends on the expertise and competence of each member of the group. But more importantly, the capacity of each group, 'the organelles', and the overall competence of the entire organization is dependent on the timely exchange of relevant information between each human member and their synchronization with the workflow process or the objects moving about within this process. Not all members have stakeholdership with all the parts or each other for that matter.

Since products and even services are composed of parts that are assembled together within the workflow process, one may intuit that there are myriads of connections between these parts and members of each group.

So we can conclude that the real glue that binds everyone in the organization is the components, parts or the final product or products. And the conversations that matter in this setting are the ones related to the objects or virtual objects that are moving around within the workflows. And these conversations belong to the object itself. As described in the 3rd wave of corporate communication, these objects are the real pivots of human communication in the workplace [1].

The Solution:

Our proposed new software application design approach starts with a comprehensive semantic analysis within the domain of the process to be automated. With this approach, we identify the critical entities that are created and move along the 'assembly line' or workflow. These entities are any real or virtual objects that humans may have a conversation about. We then associate a unique communication channel to each entity for each class of stakeholders in the organization. That way, each type of communication is held in its own separate channel. These channels are not named nor created by any users. They simply are attributes of the object itself and are created by the object as needed.

In this manner, the communication channels are innate attributes of the objects similar to the structured data elements that define and describe the object. We then create the necessary algorithms to give a simple awareness or intelligence to the object as to its state within the workflow as well as the knowledge of all its various classes of stakeholders interested or associated with the entity. Stakeholders become associated with their objects of interests in a variety of purposeful ways as part of the workflow. For example, they can be the creator of the object or are assigned as a resource to work on the object, or simply find the object in a search and 'follow' the object. All relevant information is stored with the object itself via these channels. The object uses its LLM to read and understand any information submitted into these channels as attachments

VI. Self Aware Intelligent Objects

In the Object Messaging & Intelligent Objects (OMIO), objects or entities are not only the pivot of connection for the relevant stakeholders but are aware of their own states at all times. When people interact with an object and affect any of its attributes or when the object changes state moving within the workflow, it converts these events into messages and records them in their appropriate chat channels on behalf of the person or process responsible for the event. Messages or other types of information such as video/audio files, links or documents can be inserted into the object's channel. The object can utilize AI to read, understand and remember the information from these files as well as keeping them organized for later access by stakeholders. There is no need for users to come up with their own elaborate ways of creating folders for organizing files. That's the job of the object as the information belongs to the object and so it is processed, understood, remembered and organized by the object itself. Any message insertion in the object's chat channel causes the routing of the object to all the stakeholders' Inboxes connected to this object. In this way, the stakeholders are informed in real time of important events affecting their objects of interest and can react and intervene if necessary by posting their reactions, thoughts or concerns in the same channel. One can imagine that the object is calling for a virtual asynchronous meeting amongst the stakeholders.

In this fashion, it's the object's responsibility to communicate the event information to the stakeholders as soon as the event occurs. This also solves the synchronization problem with the workflow. All the stakeholders are thus tuned to the heartbeat of the organization which is the tempo of the workflow. The events are usually changes in status or one of its attributes or they can be messages inserted into one of its channels by stakeholders. All discussions and occurrences of events are stored in the appropriate channel of the object and is maintained by the object itself. And since the object is responsible to communicate any changes to its state in the form of natural language message insertion into its channels, the communication happens in real time and only to the appropriate stakeholders.

This is a significant feature of this model as it maintains a totally optimized communication in the form of relevant recipients only and precise timing for the transmission of these objects to human nodes within the organization. The precise timing is in synchronicity with the resonating pulse or tempo of the workflow as the heartbeat of the organization. The significance of this timing is usually overlooked in the current automated forms of communication in the workplace as the routing and timing of information transfer is dependent on the least reliable agents within the network; the humans!

Object Messaging relegates the routing and the timing of all significant events about the object to the object itself. As a result, the emerging automated world is composed of a collection of intelligent objects, or super objects as they get assembled together, while they maintain all the conversations and event messages about themselves within appropriate channels as well as the list of all relevant stakeholders of various types. Even the insertion of a message is considered an

event that causes the object to deliver a digital representation of itself along with its entire entourage (media, files, links, etc.) to the appropriate stakeholders inboxes.

In Object Messaging, the users' inboxes receive objects or super objects that have been associated with or self identified themselves as relevant to a stakeholder or resource. The stakeholders can then examine the relevant chat channel to get informed of the latest events that have occurred to the object since it was last examined.

VII. Integration, Deep Integration, And

AI Fusion:

COMPREHENSIVE AND MEANINGFUL INTEGRATION WITH AI IN ONE STEP

Some existing applications may find ways to do some integration with AI LLMs such as ChatGPT, which may provide some useful functions such as summarization of emails or writing emails , document generation for legal or other professions and code generation for programmers. All if not most business applications will struggle to find any place to plug in the AI in a really useful or meaningful way. How would you connect ChatGPT to Quickbooks in its current form, for example? Where would you start?

Another example would be an Electronic Medical Record which has lots of text within each encounter. Perhaps, you can summarize the text but that is of very limited value. But there are better ways which may sound more like science fiction. Imagine an intelligent Electronic Medical Record where at every point of user interaction, the care provider can call up the AI for assistance and the LLM is fully aware of the situation and provides meaningful information about some aspect of the patient such as an x-ray or some lab result or an illness. And all of that interaction is using natural language and recorded within the xray's private channel reserved just for care providers' discussions. We call that deep integration because the AI is passive and awaits patiently until it is called upon to review the records and then communicate its opinion to the provider

One can also imagine yet another scenario where a patient's x-ray has just arrived in the provider's inbox and the AI within the x-ray itself has already examined it, read through the patient's history and relevant lab results, alerted the specialist, researched the latest on the issue on authorized pre-designated sections of the Internet, sent a follow up appointment reminder email to the patient, inserted its comments into the appropriate private channel and is ready to have a discussion with the care provider and the specialist in the private channel all using natural language. We call that AI fusion. And that is what OMIO makes possible today.

How it's done:

Using the Object Messaging, designers identify these entities or objects and associate one or more communication channels for the objects. In this way, human communication becomes part of the object as its innate attribute. Via these channels, the objects can also hold more information in a variety of formats such as video, audio or files as attachments. Now, the objects are ready to be fed into an AI since they contain not only their own structured data but all the conversations by all of the relevant stakeholders about them and all the attachments. The AI can use the structured data of the object to gain a more accurate context about the stakeholders conversation in its chat channels and the attached media files.

Since the entire application is based on this model, every critical object contains its own chat channel attribute as well as its structured data, the LLM AI need only be connected to the object in one place in the application under the hood. From that point on, the LLM AI is deeply integrated with the application and is present in every chat room for all the objects in the application. Since software applications based on OMIO consist of collections of intelligent objects or super components (objects assembled together), the LLM AI can be actively present literally at every point the user interacts with the system. That is what we call deep fusion AI. The AI fully understands the object since it has access to its structured data but is also kept informed via the chat channels about the object's changes of states or occurring events during the production process. It can read the conversations in the channel and act as an expert advisor to the stakeholders using natural language!

Stakeholders are associated with these objects based on their relevance or interaction with the workflow process to form literally thousands of networks with the objects as the centerpiece or pivot for the connection. This new pivot also guarantees the optimal communication amongst the stakeholders associated with the object. When various objects come together to make a super component, the stakeholders association is inherited from the original components accordingly. In this way, the stakeholders are informed of the status of the super component via the inherited association link.

VIII. Migrating to the new LLM AI Platform

As with the previous transitional platforms, namely, the GUI and the Cloud, the emergence of AI and the increasing popularity of LLMs such as ChatGPT represent a new virtual platform for most existing business applications to transition onto. Just like the previous transitions, this one requires not only a rearchitecting of existing software applications but a new way of thinking about software design approach in general. In short, we need a new paradigm and OMIO represents such a paradigm shift. The Object Messaging Model not only optimizes communication among human nodes but also facilitates a seamless and deep Integration of Artificial Intelligence with any business software application. We believe it will prove to be the required architecture for a comprehensive and meaningful AI integration or an omnipresent AI at every point of human interaction with the application.

An object containing people's chatter about itself within its own appropriate channel can provide detailed context to an

LLM for processing the conversation about the object. The structured data associated with the object is utilized as constraints for the processing of human chatter about the object. The object's structured data may also be used in 'Prompt Engineering' fashion to create the context for the LLM before feeding it the dialogs or channel chatters about the object. The object's chat channels include stakeholders dialogs as well as the object's critical events and changes of state in the form of natural language. The OMIO provides the identification of all critical objects and extends one or more dialog channels as an innate attribute of the object. This streamlines a deep AI integration since there exists only few localized points for this purpose within the application and those points of connections are the actual objects . And once the AI is connected to the object, it has access to the structured data and natural language in its chat channels as well as any attachments such as documents, media files, and hyperlinks. Traditional form based applications combine multiple data elements in one screen (form) and were never provisioned to hold natural language dialogs for each critical element within the form! The LLM AI cannot be integrated to Form Based applications as the human dialog about the data fields do not exist within the application. That dialog is chaotically scattered all over emails and Slack or Teams' various channels with arbitrary chosen names for channels, or email subjects. The OMIO requires identification of the business process elements as critical entities and refers to the smallest element as a nugget. It provides a chat channel to keep the human dialog, workflow events, changes of state and attachments together with the nugget as its extended attributes so it can easily be fed into the AI. These events happen to the nuggets as humans or processes interact with the nugget during the workflow process. The OMIO translates these events into natural language and inserts them into the appropriate chat channel belonging to the nugget.

The nugget can have more than one chat channel for different classes of stakeholders. These stakeholders usually have different functions within the business environment and have access to different chat channels of the nugget based on their roles.

The insertion of any message including translated events is a trigger for the nugget to immediately transmit pointers of itself to all the relevant stakeholders who are associated with that nugget's chat channel. These pointers, for non-programmers, can be imagined as holograms of the nugget which are exact replicas and are all one and the same as the nugget. That means a change to any of these holograms directly affects the nugget itself and is simultaneously reflected in all the hologram copies.

Conclusion:

As one can see, the Object Messaging Model and Intelligent Objects can easily combine structured data, natural language communications along with other related documents and media resources such as video, audio, Internet links, as well as the identities of relevant stakeholders and human resources within the object itself.

The final application is a collection of these intelligent objects or super objects with self-awareness about their own state and a deeply embedded LLM AI which has already processed the object, its associated attachments and its relevant conversations and is ready to answer any of the human stakeholders' questions about itself when it shows up in their inbox! These Intelligent Objects know how to navigate the workflow and calculate their own tempo and cadence in each stage or phase and report critical information to their stakeholders in real time. These objects are contained within a messaging framework with a higher level of understanding about the nature and the context of the application in such a way to properly route these entities into the 'proper inbox' appropriately named for the type and class of application. This high level framework can act as a high level 'brain' for the entire application managing, interacting with intelligent objects or super components.

IX. PROJECT OMADEUS:

PROJECT MANAGEMENT AND COLLABORATION BUILT WITH OMIO (PATENTS PENDING)

To demonstrate the benefits and practical implementations of such a model, we are making available a project management software designed with this approach. This is a first of its kind OMIO based project management and collaboration tool that guarantees optimal and real time intelligent communication amongst the participants including all the stakeholders. It is available for a free trial on www.xeba.tech to give you the experience and demonstrate the clear benefits of the OMIO model and the new world to come, a world without email frenzies and frivolous channel chatters. With this new model, intelligent objects, and not humans, manage the routing and on time delivery of events and messages to the right stakeholders, and the AI is omnipresent at every point of user interaction fully informed about the issue at hand and ready to assist, teach and learn from you!

Disclosure: The author has an executive position with Xeba Technologies with multiple patents pending for this technology and the manufacturer of the Project AMaiDEUS software product based on OMIO.

REFERENCES

[1] M. Alibakhsh, "System and methods for optimal and synchronized workflow based communication," 2021 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, USA, 2021, pp. 1451-1453.

[2] Qian, Ling, et al. "Cloud computing: An overview." Cloud Computing: First International Conference, CloudCom 2009, Beijing, China, December 1-4, 2009. Proceedings 1. Springer Berlin Heidelberg, 2009.

[3] Hayes, Brian. "Cloud computing." Communications of the ACM 51.7 (2008): 9-11.

[4] Venolia, Gina, et al. "Supporting email workflow." microsoft.com (2001).

[5] McMurtry, Kim. "Managing email overload in the workplace." Performance Improvement 53.7 (2014): 31-37.

[6] Dabbish, Laura A., and Robert E. Kraut. "Email overload at work: An analysis of factors associated with email strain." Proceedings of the 2006 20th Anniversary Conference on Computer Supported Cooperative Work. 2006.

[7] Riemer, Kai, and Asin Tavakoli. "The role of groups as local context in large Enterprise Social Networks: A Case Study of Yammer at Deloitte Australia." http://hdl.handle.net/2123/9279 (2013).

[8]   Wang, Dakuo, et al. "Slack channels ecology in enterprises: How employees collaborate through group chat." (2021).

[9]   Hargrave, Sean. "How Slack Ruined Work." WIRED UK, 13 Jan. 2020, www.wired.co.uk/article/slack-ruining-work.

[10]   Ives, Blake. "Graphical User Interfaces for Business Information Systems." *MIS Quarterly* (1982): 15-47.

[11]   Myers, Brad A. "51. Graphical User Interface Programming." (2004).

[12]   Standing, Craig. "Methodologies for developing Web applications." *Information and Software Technology* 44.3 (2002): 151-159.

[13]   Lu, Ming‑te, and Wing‑lok Yeung. "A framework for effective commercial web application development." Internet Research 8.2 (1998): 166-173.

[14]   Rossi, Gustavo, et al., eds. Web Engineering: Modeling and Implementing Web Applications. Springer Science & Business Media, 2007.

[15]   Heylighen, Francis, and Marta Lenartowicz. "The Global Brain as a model of the future information society: An introduction to the special issue." *Technological Forecasting and Social Change* 114 (2017): 1-6.

Notes: The words entity and object are used interchangeably