OneShield | UX through UI: Navigating Customer Expectations

ePaper authored by:
Vivek Gujral, CTO - OneShield Software

OneShield SOFTWARE | Business Solutions. Simplified.

OneShield | UX through UI: Navigating Customer Expectations

Vivek Gujral
CTO - OneShield Software

User interfaces (UIs) and communication channels for core systems will continue to face pressure to enhance the customer experience. Furthermore, from OneShield's perspective, for a particular user profile and channel, not all UIs or channels are equal. We believe that to truly be adaptive, UI's need to be built using reusable software components accessed via scalable web services that can be leveraged seamlessly across every communication channel.

The winners and losers in the emerging wars for customer attention may well be determined by who best understands both the limits and potential of each communication channel. And, who has the most capable toolset to leverage the customer and user data within their platform.

Top 5 Communication Channels for Insurers

Our belief, based on extensive experience in building out software interfaces for financial services organizations and the evolution of UIs in similar industries, is that user interfaces will revolve around five main channels of communication:

1. **Traditional web browsers**
2. **Mobile apps**
3. **Chat**
4. **Voice**
5. **Web services called from applications or devices in the Internet of Things (IoT) ecosystem**

The applicability and priority, for each of these communication channels will largely be determined by the type of use case an internal stakeholder or external customer requires.

Broadly, OneShield foresees these interaction types will fall into the following four categories, each with its appropriate communication channels:

1. Frequent, Complex Data Visualization and Data Entry Interactions

Users will most likely turn to big-screen tablets, PC desktops or other interactive displays to access large quantities of information, such as those from image recognition apps or scanned data. Interactive graphs (pie charts, bars, etc.) and drill-down tables are best leveraged on these types of UIs. Devices such as Google Glass, the so-called "smart glasses" offering heads-up displays of Internet information — also come to mind.

2. Moderate Frequency and Complexity Interactions

When choosing between two financial services products or reviewing options for an insurance policy, this level of user interaction will most likely be dominated by mobile apps, whether native or web-based. Even more likely, as technology advances, chat application platforms will become dominant for this level of interaction, as we are beginning to see on Facebook and the fast-growing Line communication app popular in Asia.

That said, two facts need to be considered when deciding on the best UI for moderate-level interactions with customers.

i. **First,** there is the basic fact people don't like to clutter mobile devices with apps they only use occasionally. Even if they do keep them on their device, they are often relegated to obscure parts of the mobile canvas.

ii. **Second**, chat is becoming one of the most used applications on mobile devices, quickly replacing the now-notorious Interactive Voice Response (IVR) applications on landlines that many businesses have tried to get customers to use. Given this, it's clear the ability to interface via chat rather than a mobile app (no matter how well-designed or "slick" it is), is likely the best channel for certain transaction types. We discuss this again in occasional-level interactions.

A notable exception here, of course, are insurance and other financial products that actively interact with customers – for example, those that offer "pay-per-use" insurance, requiring driver feedback, or offer portfolio updates, gamification, etc. For these, an app (native or otherwise) is essential.

As well, for moderate-level interactions, another exception favoring mobile apps is communication with internal stakeholders. Unlike consumers, moderate interactions with insurance company employees and value chain partners, such as inspectors, repair shops and towing companies, require access via mobile devices.

Whether web or native applications, they permit quick and reliable tasks (approvals, reviews, repairs, etc.) while on the move. In a 24x7x365 working environment, this is a necessary adjunct to interfaces on larger format devices.

3. Occasional/Simple Interactions

For occasional/simple level interactions, we believe chat and voice (Alexa or Siri, for example) will be the most utilized. "What's the weather like this week?", "What are today's news headlines?" and "What's my portfolio worth?" are examples of interactions that are great use cases for a chat/voice interface.

We often lump chat and voice together because both are forms of chat, with voice having the additional complexity of voice to text conversion, as well as the accompanying metadata of tone and meaning.

OneShield also believes that, in time, chat and voice will evolve into being used in moderate-level interactions as well. Imagine, for example, a plumber who tells Siri: "I'm at the site and starting the job," and when finished onsite, tells Siri he's done. All that is needed are these messages and the geo-location to know a piece of claim remediation is complete.

4. Non-User Interactions

Finally, there is the matter of the term "user" in user interfaces becoming broader and less about the interaction of the "user" but about "servicing" the user without human interaction. An example is a water or fire alarm in a house or a car whose IoT device tells us the car has flipped over in an accident. As the insurance industry moves towards loss prevention using smart, connected devices, this is likely to become an increasingly large part of the "user" interface.

Integrating User Interfaces on the OneShield Platform

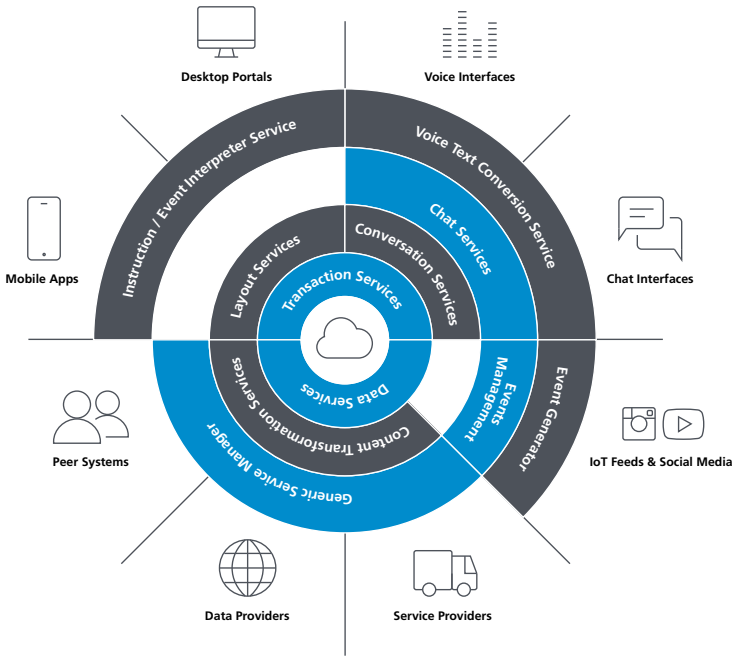
So, what can be learned from how OneShield is staying ahead of the curve in designing and implementing UIs? First, it is important to understand OneShield's platform began and continues its journey with the intent to remain as open and adaptable to technology evolution as possible. With this mindset, we achieve inter-operability via OneShield's Services Designer (formerly known as TESL) which uses a collection of vendor and product-independent SOAP or REST Web Services.

These services allow external applications to remotely and easily execute business transactions on OneShield's platform. Processing that had traditionally been tied to the UI or only available within the platform was de-coupled and is now available purely as-a-service, such as exchanging data or creating custom web portals.

This means our clients can expedite integrations with a wide range of third-party applications and with lower implementation costs.

The diagram below shows how we build and implement UIs. The blue semi-circles are OneShield's internal UI assets and communication layers, while the ones in gray are those provided by third-parties.

For discussion purposes here, we will focus on the channels above the horizontal line — mobile apps, desktop portals, voice and chat interfaces, and how your organization can leverage their capabilities throughout your various user interface journeys.



A. Mobile Apps & Desktop Portals

For this class of interface, the first layer one leverages in the OneShield platform is the Instruction/Event Interpreter Service Layer. This particular layer formalizes the interactions between a user-interface (native or web) with the platform's back end. As an example, the figure below represents the layer provided by the OneShield Portal component.

OneShield provides an implementation of this layer using the Sencha Ext-JS library. However, customers are free to implement their own within our highly adaptable platform … using Adaptive or other frameworks if they want.

Workflow Gateway	Dynamic UI Handler				List/Grid Management			File I/O	
	Rulers	Validation	Format	Domain	Pagination	Sorting	Filtering	Download	Upload

Working from left to right, the workflow gateway provides support for calling actions such as quote, bind, buy and sell. OneShield's dynamic UI handler allows for the server-side execution of any number of rules, validations, formatting instructions and domain switches. This means one can hide or show UI fields, validate fields when a user leaves them, and change the contents of a list based on context

(States or Provinces, depending on a country). OneShield's List or Grid Management also provides support for paginating or displaying chunks of lists, as well as sorting and filtering. Finally, the File I/O services allow for files to be uploaded or downloaded from the client.

Whether building a custom UI or using a third-party application development tool, you can call OneShield's web services and quickly implement innovations using this layer. You can use the existing OneShield layer, allowing you to extend the OneShield platform or application by the following layer of services:

Custom Content			
Get Context Data	Render	Submit Request	Trigger Event

This allows a "foreign" block to co-exist within the UI by building out the four services listed. A good example of this approach is the embedded Pentaho Report creation block within the OneShield interface.

Next, we come to the **Layout, Data and Transaction Service Layers,** which provide a more highly abstracted set of services that are useful for building out an interface if one wants to depart completely from any OneShield Portal assets.

Session	Object Access	Product Metadata	Execute Rule Sets	Execute Transaction	Object Locking	Page Layout

Moving from left to right, these allow the creation or destruction of sessions, access to and search for objects, financial product metadata, rules processing, and transaction processing, etc. Not unsurprisingly, these services are shared amongst ALL the interface types depicted in the Instruction/Event Interpreter Service Layer.

What these latter services allow is for an insurer to use the OneShield portal internally, but build a completely custom set of mobile, voice, etc. interfaces for customers and partners — while still sharing the bulk of the non-UI infrastructure between the two and maintaining consistency between channels. This is an absolute must today.

B. Voice, Chat & IVR Interfaces

When designing these interfaces, the first problem to be solved in the case of voice is how to convert the user's voice into text, and vice versa. While OneShield currently uses IBM's Watson product today, this is an eminently swappable service. Furthermore, all other things being the same, the more "generic" the provider — that is, the one exposed to the most data — the better it will perform.

Turning to chat services, we find this includes not only the ubiquitous SMS/Text Messages but, more importantly, external chat applications such as WhatsApp, Line, and FB Messenger, to name a few.

As long as these services provide an API, OneShield customers have pretty much all they need to know to "connect" to the user, or vice versa. Increasingly, platforms such as a FB Messenger and Line, are providing more than adequate mechanisms to build in-chat applications. Again, these chat interfaces can use many of the same services described under Layout, Data and Transaction Service Layers — sending informed text messages back and forth to a customer.

C. Conversation Services

These services, which in OneShield's case are also built on the IBM stack, offer increasingly complex interaction graphs that allow customers or other users to have almost "realistic" conversations with your app or system. Note that a "realistic" conversation is NOT a "directed" conversation, or as we prefer to call it, an interrogation. A "conversation" in this instance is one where the user can go off on a tangent, and the software will need to be able to follow.

D. Data and OneShield Services Designer

There are among the innermost layers of OneShield's technology platform, providing Create, Update, Delete, and Search services to the common persistence for all channels. It also provides implementations of transactions. Note that a transaction need not be the classic new business, endorsement, cancellation or reinstatement, but also customer transactions, such as a change in location, sold a location, bought a car, swapped a car, lost a phone, etc. This type of interaction needs only to be built once in the OneShield platform, but then is re-usable across all communication channels.

Conclusion

The rising expectations of the customer and user interfaces are gathering momentum. For insurers looking to offer the most compelling consumer-driven or focused technology experiences, new and improved interfaces will play a key role in ensuring their success.

At OneShield, our long-standing achievement has been ensuring we are offering a uniform, common and reusable set of computing assets that can be used by our clients to seamlessly engage the customer across any communication channel, including those yet to emerge.

In part, we have accomplished this high degree of core system flexibility by combining our technological advances with those of specialist firms we partner with to provide specific know-how or tools — whether it is eliciting "events" from social networks or transforming voice into text. Working closely with innovative clients and with an eye to the future, OneShield will continue to build upon its full-spectrum suite of UI and communication assets that ultimately can be used to create real competitive advantages both for us and most importantly our clients.

To learn more about OneShield and to book a demo, connect with us at info@oneshield.com or call us at 1 888 663 2565.