# Multimodality in Mobile Force Applications

## Position Paper on Architecture & Implications


Raj Tumuluri

Openstream Inc
Principal
raj@openstream.com

Vinayak Patilkulkarni

Openstream Technologies (India)
Director ( Product Planning)
vinayak@openstream.com



### Abstract

*In this position paper, we describe the ongoing product-research and study of multimodal mobile-force applications at Openstream and describe the architectural approach taken to meet the requirements of such field-force applications. We further discuss the case for incorporation & extension of W3C MMI Architecture & Interfaces in the implementation of Openstream's Smart Messaging Platform.*



***Keywords*** *Multimodal, Mobile, MMI, Speech Recognition, INKML*

**Acknowledgements:** This work includes the study on Multimodal Interaction in Mobile Force Automation done by Mr.Saagar Kaul at Openstream Inc USA and Mr.Keyur Patel at Openstream Technologies (India).


## 1. Introduction

In a typical mobile force (non-desktop) workplace like a customer-site, a construction site or a hospital ward, a field-user is often required to perform multiple tasks in addition to the primary task of "doing the job" ( e.g. measuring pressure readings, or pipe-length or patient's vital readings ), the additional task of interacting with his/her mobile computing device to input the artifacts from the primary-task (readings etc). Multimodal Interaction is a natural fit in such scenarios, as it allows the user hands-free or eyes-free or speech-free modes of interaction with the mobile computer to input or access information.

While, the user could always choose the mode that is most appropriate for the situation, based on our study, having mixed-initiative, mixed-mode interaction has been found to reduce considerably, the amount of time taken to complete the i/o operations in the field.

Openstream's Mobile Force Automation ( MoFA) solution provides one such optimal way to complete field-transactions without the clutter of options that would distract the user from the primary task..



Since, there are not many established methods to evaluate the non-desktop workplace scenarios, we adopted the user-centric approach that would take user-experience, task-completion time and accuracy of the data captured as parameters in determining the efficacy of the application interaction modes. We have also collected additional information on the user's age and computer-literacy to further qualify our observations, although this was not used for ranking.

Based on our study, mixed-mode, mixed-initiative method of interaction reduced the average time to complete the user i/o for transaction by 8% and increased the accuracy of data collected by upto 6.4% for most field-applications deployed.

## 2. Design of the system

Openstream's multimodal Smart Messaging Platform (SMP) is based on asynchronous event-driven architecture with its design principles analogous to those of the W3C MMI architecture [1], viz. modularity, extensibility, encapsulation, distribution and recursiveness. The platform enables single authoring of multimodal applications for multimodality and device independence [7].

Most field-applications implemented on SMP have a central-system that hosts the Interaction Manager and other control-plane logic while instances of the modality components run on the mobile-devices.

The solutions can adapt to the capabilities of the devices, allowing components to spread between mobile-device and central system. For example, Speech Recognition & synthesis could happen either on the mobile-device or on the central-system depending on the capability and availability of resources on the mobile-devices.

Thus, one could have a variety of configurations for a solution in the field, which may involve some devices performing remote speech-recognition, while some others doing local speech-synthesis.

As is true for many software applications, these multimodal application components also get authored not all at the same time, often permitting re-use, nesting and integration of modality components as the application evolves. We therefore augmented the current W3C MMI architecture model by devising a scheme for registration of Modality Components as they come up, with the Interaction Manager, whereby the Interaction Manager could maintain and query the capabilities and existence of the instance of a Modality Component.

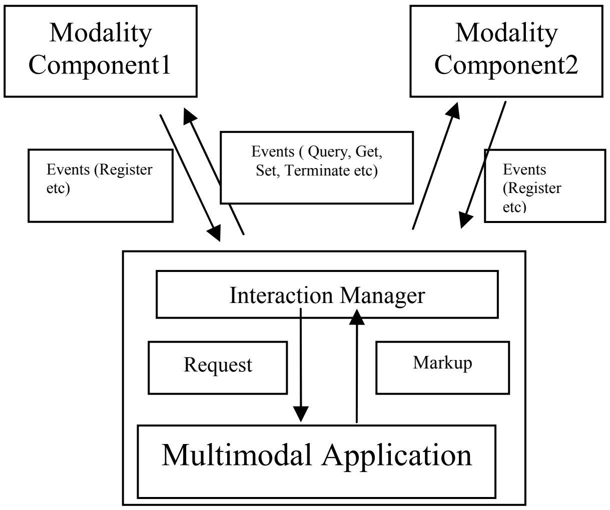


***Figure 1: MMI-Extensions provided by SMP***

Further a Device-Context component is implemented separately or as part of primary modality component on the mobile device that would enable the Interaction Manager or Modality Component to get notifications on the changes in the device-context

While this may appear to be in deviation from the principle of the W3C MMI Architecture, it solves the problem of not having a standardized way to get and set device-context-properties in the W3C MMI architecture.

Our approach was to adopt a declarative model-based user-interface development capturing the syntactic & semantic structure of the interface.

The authoring model allows for integration of newer modalities such as INK (which have thus far been incorporated through proprietary implementation in SMP for visual annotations on image-content) to standards-based W3C INKML markup as the platform support for integration of such integration becomes available for MMI Architecture.

## 3. Implementation Support needed

- Currently, XHTML MP as implemented by many mobile browsers doesn't fully support XHTML Events. Event capture and bubbling are not supported.
- Mobile Device Browsers have limited capabilities for DOM Navigation. Thus, event propagation and processing capabilities are limited with respect to visual modality component.
- While encapsulation is important, in real-life scenarios, it would be important to have a mechanism to allow modality components to expose their internal methods & sometimes the datamodel to external or nested components. Further, there should be a way for authors to specify scope rules while exporting internal methods of embedded ( nested) components. e.g Way to aggregate and resolve conflicts with speech-grammars from the top-level component that has embedded speech-dialogue-components.
- Single-authoring model while convenient, cannot prepare the author for all possible interactions through various modalities in the field.

## 4. Conclusion

In this paper we presented an overview of our multimodal application development platform and described our user-centric study to evaluate mobile-force application interfaces and interaction and the proposed extensions to the MMI architecture. Our immediate focus is to work on extension of the platform to incorporate common Device-Context-Interface layer to manage various device-properties that enrich the user experience with multimodal applications.

## References


[1] W3C MMI Architecture & Interfaces (http://www.w3.org/TR/mmi-arch)

[2] Sharon Oviatt “Taming Recognition Errors with a Multimodal Interface” (Communication of ACM September 2000 Vol. 43 No.9).

[3] C.Coutrix & LNigay “Mixed Reality: A model of Mixed Interaction” AVI’06

[4] M.Melichar & P.Cenek “From Vocal to Multimodal Dialogue Management” ICMI’06

[5] B.Suhm & A.Waibel “Multimodal Error Correction for Speech User Interfaces” ACM March 2001, Transactions on Computer-Human Interaction, p60-98.

[6] S.Prammanee, K.Moessner & R.Tafazolli “Discovering Modalities for Multimodal Interfaces”ACM Interactions (May+June 2006)

[7] R.Simon, F.Wegscheider & K.Tolar “Tool-supported Single-Authoring for Device-Independence & Multimodality” Mobile-HCI’05

[8] F.Flippo, A.Krebs & I.Marsic “ A framework for rapid development of multimodal interfaces” ICMI’03


.