# Feedback for Widex WG at Vancouver IETF

Dave Raggett, <dsr@w3.org>

Explanation of W3C MMI Architecture as context for IETF work on related protocols (e.g. Widex WG)

Comments specifically relating to Widex Requirements

# Multimodal Architecture

Intended to define requirements for what is needed, but doesn't specify how these are to be fulfilled.

See http://www.w3.org/TR/mmi-arch/

Architectural Components

- Run-time framework
- Interaction Manager
- Data component
- Modality Interface components

These communicate via events (asynchronous messages) and public draft identified several life cycle events (run, running, halt, halted) together with additional events for sessions (add session, session added, end session and session ended) as well as pause/resume operations (pause, paused, resume, resumed). On going work is looking at further kinds of events.

The run-time manager provides the basic infrastructure and event loop. The interaction manager (IM) responds to events from the modality components and data component. Control messages sent by the IM are considered as kind of events. Modality components don't talk to each other directly. Any such communication take place via the run-time framework, or the optional interaction manager (if it is present).

## Design goals for the MMI Architecture include

Encapsulation. The architecture should make no assumptions about the internal implementation of components, which will be treated as black boxes.

Distribution. The architecture should support both distributed and co-hosted implementations.

Extensibility. The architecture should facilitate the integration of new modality components. For example, given an existing implementation with voice and graphics components, it should be possible to add a new component (for example, a biomentric security component) without modifying the existing components.

Recursiveness. The architecture should allow for nesting, so that an instance of the framework consisting of several components can be packaged up to appear as a single component to a higher-level instance of the architecture.

Modularity. The architecture should provide for the separation of data, control, and presentation.

MMI WG expects to publish an updated version of its MMI Architecture by time of its next face to face in March.

# Comments on Draft Widex Requirements

Lack of consistency in use of "Widex Client" versus "Widex Renderer".

The use of the term "reliable" needs to be clarified.

> The MMI Architecture requires reliable delivery of messages in the order they were sent, with the means for applications to detect when such guarantees have failed. In practice, latency and time outs may need to handled at the application level.

What's the difference between update, mutation and event?

> In general Widex is independent of the nature of the events. There needs to be an agreed means to serialize events as XML, e.g. on how serialize IDL definitions of events as XML, but Widex shouldn't know about the semantics of specific events. An exception is a set of events designed to act as commands to modify or replace DOM trees, or to notify such changes. Presumably there is also a general need to indicate the target node and the DOM document involved in events.

The application is responsible for synchronization, e.g. across modalities. Widex just provides a basic transport mechanism.

There are multiple proposals for remote DOM updates and further discussion may be needed to clarify the requirements.

It might be worth noting the relation to interaction modalities. Thus a Widex Server can have multiple renderers, and each renderer might support multiple modes of interaction (visual, tactile, aural).