Name of the TC:

User Interface Markup Language (UIML) Specification Technical Committee

Statement of Purpose:

The purpose of the User Interface Markup Language (UIML) Specification Technical Committee is to develop a specification for an abstract meta-language that can provide a canonical XML representation of any user interface (UI). The language should be capable of specifying the requirements, design, and implementation of any UI.

The committee uses the UIML3 specification created by Virginia Tech's Center for Human Computer Interaction, Harmonia, Inc., and other organizations on uiml.org as a starting point [1]. The owners/creators of this specification claim no intellectual property with regard to UIML and will not seek licensing reparation for its use in this TC, nor will they seek ownership of any work produced by the TC.

The TC differs in scope from other committees and working groups that address *concrete* UI implementation languages (e.g., assembly language, C, C++, CSS, Java, HTML, VoiceXML, WML, XForms, XHTML, XSL-FO, etc.). Instead, this TC focuses on an *abstract* language, UIML, which expresses UIs at a higher level of abstraction than the concrete languages. UIML's goal is to subsume in expressive power all concrete languages, and to permit efficient mapping from UIML to any concrete language.

The TC has a secondary purpose -- to use UIML to bridge UI-related fields. The TC will explore the use of UIML in collaboration with the following techniques:

- Techniques from the Human Computer Interaction (HCI) field. These include UI models, model-based design, transformational techniques, techniques for computer automation of UIs, usability engineering, generalization of the Model-View-Controller, and new UI metaphors.
- Techniques for expressing UI designs in terms of domain-specific abstractions (e.g., navigation maneuver for an automobile UI, material path for a factory automation UI), rather than at a widget level.
- Techniques for capturing author intents and mapping UIs to different devices to create portable and accessible UIs.
- Techniques to internationalize UIs.
- Techniques that help integrate UIs with Web Services (e.g., OASIS WSDL TC).

A general motivation for a canonical UI representation language is to accelerate the development of tools for UI design and development. Just as tools built for XML work for any XML vocabulary, tools built for UIML work for any UIML vocabulary, whether that vocabulary represents a concrete UI implementation language or higher level abstractions. If practitioners build tools that utilize UIML as a representation language, then the tools can interoperate. Thus the TC's work will serve to assemble the jigsaw puzzle pieces of UI and HCI technology into a coherent and interoperable set of design, development, evaluation, and deployment tools.

To create a UIML standard, the TC will perform the following activities.

- Evaluate UIML3 and other UI language and tool initiatives to glean the best practices from the field,
- Develop the specification,
- Create compliance tests and implementations, and
- Submit the specification to the OASIS membership for approval.

Relationship to Existing Activities:

Many efforts related to the use of XML to describe UIs are underway throughout the industry. The following work may be relevant to this TC:

- Many W3C Working Groups and Activities, including Accessibility, CSS, Device Independence, Voice, XForms, XHTML
- Mozilla XUL
- USI-XML (being developed at the Universite' catholique de Louvain in Belgium),
- DISL (being developed at Paderborn University in Germany)

List of Deliverables

Deliverables	***Status***	***Estimated Completion Date***
Assessment of how UIML compliments other committees and working groups addressing user interfaces	Initial draft document completed	Submitted on July-26, 2004
List of open issues in the UIML 3.x specification	Updated list posted	Revised on November 14, 2005
Revised specification based on issues list	Ongoing	~ April 1, 2006
Complete Implementation of the specification and use by at least three different parties		~ July 1, 2006
Compliance test suite		~ July 1, 2006
List of best implementation practices		~ July 1, 2006

Deliverables	***Status***	***Estimated Completion Date***
Revised Specification based on implementation experience		~ October 1, 2006
Optional deliverables		
Documents describing requirements for UIML		
Analysis of using UIML to make interfaces accessible		
Document giving UIML examples		