ISO/IEC TR 13066-2:2016-02 (E)

Information technology - Interoperability with assistive technology (AT) - Part 2: Windows accessibility application programming interface (API)

Contents

Page

Foreword v

Introduction vi

1 Scope 1

2 Termsanddefinitions 1

3 General description and architecture of the Microsoft Windows Automation API 7
3.1 General description 7
3.1.1 Microsoft Active Accessibility overview 7
3.1.2 UI Automation overview 8
3.1.3 IAccessibleEx interface 9
3.2 Architecture 10

4 Using the API 12
4.1 Using the Microsoft Active Accessibility API 12
4.1.1 Types of Microsoft Active Accessibility support 13
4.1.2 Retrieving an accessible object 13
4.1.3 The WM_GETOBJECT message 13
4.1.4 Special values of Object Identifier 14
4.2 Using the UI Automation API 15
4.2.1 UI Automation model 15
4.2.2 UI Automation tree 16
4.2.3 UI Automation control patterns, control types, properties, and events 16
4.3 Using the IAccessibleEx interface 21
4.3.1 The IAccessibleEx interface implementation 21

5 Exposing User Interface Element Information 24
5.1 General 24
5.2 Exposing UI Elements with Microsoft Active Accessibility 25
5.2.1 How an MSAA Server exposes relevant properties 25
5.2.2 Provide support for the Accessible Object structure 26
5.2.3 Support hit testing 27
5.2.4 Generate appropriate WinEvents 27
5.2.5 Object identifier 27
5.2.6 How MSAA clients access exposed UI elements 28
5.3 Exposing UI Elements with UI Automation 28
5.3.1 Types of providers 28
5.3.2 UI Automation provider concepts 28
5.3.3 Provider interfaces 29
5.3.4 Property values 30
5.3.5 Provider navigation 30
5.3.6 Provider reparenting 31
5.3.7 Provider repositioning 31
5.3.8 How UI Automation clients access exposed UI Elements 32

6 Exposing UI Element actions 33
6.1 Exposing UI Element actions in MSAA 33
6.2 Exposing UI Element actions in UI Automation 33
6.2.1 UI Automation control pattern components 33

6.2.2 Control patterns in providers and clients 34
6.2.3 Dynamic control patterns 34
6.2.4 Control patterns and related interfaces 34

7 Keyboard focus 36
7.1 MSAA keyboard focus and selection 36
7.1.1 Focus and selection properties and methods 36
7.1.2 Events triggered in menus 37
7.2 UI Automation keyboard focus and selection 37

7.2.1 Focus 37
7.2.2 Selection 38

8 Events 44
8.1 WinEvents 44
8.1.1 USER's role in WinEvents 44
8.1.2 Receiving event notifications 45
8.1.3 Sending events 45
8.1.4 The allocation of WinEvent IDs 45
8.2 UI Automation events 46
8.2.1 How providers raise events 47
8.2.2 How clients register for and process events 48

9 Programmaticmodificationsofstates,properties,values,andtext 48
9.1 UI Automation specifications 48
9.1.1 Introduction 48
9.1.2 UI Automation elements 49
9.1.3 UI Automation tree 49
9.1.4 UI Automation properties 50
9.1.5 UI Automation control patterns 50
9.1.6 UI Automation control types 50
9.1.7 UI Automation events 50

10 Design considerations 51
10.1 UI Automation design considerations 51
10.1.1 UI Automation clients 51
10.1.2 UI Automation providers 54
10.1.3 Coexistence and interoperability with Microsoft Active Accessibility 57
10.2 IAccessibleEx design considerations 58
10.2.1 Design consideration for providers before implementing the IAccessibleEx interface 58
10.2.2 IAccessibleEx interface for providers 58
10.2.3 IAccessibleEx interface for clients 59

11 Further Information 63
11.1 Microsoft Active Accessibility and Extensibility 63
11.2 UI Automation extensibility features 63
11.2.1 Registration of custom UI Automation properties, events, and control patterns.63 11.2.2
How clients and providers support custom control patterns 64

Annex A (informative) Microsoft Active Accessibility to Automation Proxy 65

Annex B (informative) UI Automation to Microsoft Active Accessibility Bridge 72

Annex C (informative) UI Automation for W3C Accessible Rich Internet Applications
(ARIA)Specification 77

Annex D (informative) Other Useful APIs for Development and Support of Assistive Technologies . 81

Bibliography 88