# Accessibility software

## Wikipedia

In human–computer interaction, computer accessibility (also known as Accessible computing) refers to the accessibility of a computer system to all people, regardless of disability or severity of impairment, examples include [Web accessibility](http://en.wikipedia.org/wiki/Web_accessibility) guidelines.[[5]](http://en.wikipedia.org/wiki/Assistive_technology#cite_note-5) Another approach is for the user to present a token to the computer terminal, such as a smart card, that has configuration information to adjust the computer speed, text size, etc. to their particular needs. This is useful where users want to access public computer based terminals in Libraries, ATM, Information kiosks etc.

It is largely a [software](http://en.wikipedia.org/wiki/Software) concern;[[*dubious*](http://en.wikipedia.org/wiki/Wikipedia:Disputed_statement) *–* [*discuss*](http://en.wikipedia.org/wiki/Talk:Computer_accessibility#Dubious)] when software, hardware, or a combination of hardware and software, is used to enable use of a computer by a person with a disability or impairment, this is known as [Assistive Technology](http://en.wikipedia.org/wiki/Assistive_Technology).

There are numerous types of impairment that affect computer use. These include:

* [Cognitive impairments](http://en.wikipedia.org/wiki/Cognitive_disabilities), illiteracy and [learning disabilities](http://en.wikipedia.org/wiki/Learning_disability), such as [dyslexia](http://en.wikipedia.org/wiki/Dyslexia), [ADHD](http://en.wikipedia.org/wiki/ADHD) or [autism](http://en.wikipedia.org/wiki/Autism).
* [Visual impairment](http://en.wikipedia.org/wiki/Visual_impairment) such as low-vision, complete or partial [blindness](http://en.wikipedia.org/wiki/Blindness), and [color blindness](http://en.wikipedia.org/wiki/Color_blindness).
  + For individuals with mild to medium vision impairment, it is helpful to use large [fonts](http://en.wikipedia.org/wiki/Typeface), high DPI displays, high-contrast themes and [icons](http://en.wikipedia.org/wiki/Icon) supplemented with auditory feedback and screen magnifying software. In the case of severe vision impairment such as blindness, [screen reader](http://en.wikipedia.org/wiki/Screen_reader) software that provides feedback via [text to speech](http://en.wikipedia.org/wiki/Text_to_speech) or a [refreshable braille display](http://en.wikipedia.org/wiki/Refreshable_braille_display) is a necessary accommodation for interaction with a computer.
  + About 8% of people, mostly males, suffer from some form of [colour-blindness](http://en.wikipedia.org/wiki/Color_blindness). The main colour combinations that might be confused by people with visual deficiency include red/green and blue/green. However, in a well-designed user interface, color should not be the only way of distinguishing between different pieces of information.
* Hearing-related disabilities including [deafness](http://en.wikipedia.org/wiki/Deafness), being [hard of hearing](http://en.wikipedia.org/wiki/Hard_of_hearing), or [hyperacusis](http://en.wikipedia.org/wiki/Hyperacusis).
  + While [sound user interfaces](http://en.wikipedia.org/w/index.php?title=Sound_user_interface&action=edit&redlink=1) have a secondary role in common desktop computing, usually limited to [system sounds](http://en.wikipedia.org/w/index.php?title=System_sound&action=edit&redlink=1) as feedback, software producers take into account people who can't hear, either for [personal disability](http://en.wikipedia.org/wiki/Deaf), [noisy](http://en.wikipedia.org/wiki/Noise_%28environmental%29) environments, [silence](http://en.wikipedia.org/wiki/Silence) requirements or lack of [sound hardware](http://en.wikipedia.org/w/index.php?title=Sound_hardware&action=edit&redlink=1). Such system sounds like [beeps](http://en.wikipedia.org/wiki/Beep_%28sound%29) can be substituted or supplemented with visual notifications and captioned text (akin to [closed captions](http://en.wikipedia.org/wiki/Closed_captions)).
* Motor or dexterity impairment such as [paralysis](http://en.wikipedia.org/wiki/Paralysis), [cerebral palsy](http://en.wikipedia.org/wiki/Cerebral_palsy), or [carpal tunnel syndrome](http://en.wikipedia.org/wiki/Carpal_tunnel_syndrome) and [repetitive strain injury](http://en.wikipedia.org/wiki/Repetitive_strain_injury).
  + Some people may not be able to use a conventional [input device](http://en.wikipedia.org/wiki/Input_device), such as the [mouse](http://en.wikipedia.org/wiki/Computer_mouse) or the [keyboard](http://en.wikipedia.org/wiki/Computer_keyboard). Therefore it is important for software functions to be accessible using both devices; ideally, software uses a generic input [API](http://en.wikipedia.org/wiki/API) that permits the use even of highly specialized devices unheard of at the time of software development. [Keyboard shortcuts](http://en.wikipedia.org/wiki/Keyboard_shortcuts) and [mouse gestures](http://en.wikipedia.org/wiki/Mouse_gesture) are ways to achieve this. More specialized solutions like on-screen software keyboards and alternate input devices like [switches](http://en.wikipedia.org/wiki/Switch_Access), [joysticks](http://en.wikipedia.org/wiki/Joystick) and [trackballs](http://en.wikipedia.org/wiki/Trackball) are also available. Speech recognition technology is also a compelling and suitable alternative to conventional keyboard and mouse input as it simply requires a commonly available audio headset.

These impairments can present themselves with variable severity; they may be acquired from [disease](http://en.wikipedia.org/wiki/Disease), [trauma](http://en.wikipedia.org/wiki/Physical_trauma) or may be [congenital](http://en.wikipedia.org/wiki/Congenital_disorder) or degeneration in nature.

**Accessibility APIs**

* [Microsoft Active Accessibility](http://en.wikipedia.org/wiki/Microsoft_Active_Accessibility) (MSAA) on [Microsoft Windows](http://en.wikipedia.org/wiki/Microsoft_Windows)
* [Microsoft UI Automation](http://en.wikipedia.org/wiki/Microsoft_UI_Automation) on [Microsoft Windows](http://en.wikipedia.org/wiki/Microsoft_Windows), replacing MSAA
* [IAccessible2](http://en.wikipedia.org/wiki/IAccessible2) on [Microsoft Windows](http://en.wikipedia.org/wiki/Microsoft_Windows), a competitor of Microsoft UI Automation also replacing MSAA
* [AT-SPI](http://en.wikipedia.org/wiki/AT-SPI) on [UNIX](http://en.wikipedia.org/wiki/UNIX) and [Linux](http://en.wikipedia.org/wiki/Linux)
* [Mac OS X](http://en.wikipedia.org/wiki/Mac_OS_X) Accessibility
* Java Accessibility and the Java Access Bridge for [Java](http://en.wikipedia.org/wiki/Java_%28programming_language%29) software.[[4]](http://en.wikipedia.org/wiki/Computer_accessibility#cite_note-4)[[5]](http://en.wikipedia.org/wiki/Computer_accessibility#cite_note-5)

Some of these APIs are being standardised in the ISO/IEC 13066 series of standards.[[6]](http://en.wikipedia.org/wiki/Computer_accessibility#cite_note-6)[[7]](http://en.wikipedia.org/wiki/Computer_accessibility#cite_note-7)

**Accessibility Features in Mainstream Software**

* [Keyboard shortcuts](http://en.wikipedia.org/wiki/Keyboard_shortcuts) and [MouseKeys](http://en.wikipedia.org/wiki/MouseKeys) allow the user to substitute keyboarding for mouse actions. [Macro recorders](http://en.wikipedia.org/wiki/Macro_recorder) can greatly extend the range and sophistication of keyboard shortcuts.
* [Sticky keys](http://en.wikipedia.org/wiki/Sticky_keys) allows characters or commands to be typed without having to hold down a modifier key (Shift, Ctrl, Alt) while pressing a second key. Similarly, ClickLock[[8]](http://en.wikipedia.org/wiki/Computer_accessibility" \l "cite_note-8) is a [Microsoft Windows](http://en.wikipedia.org/wiki/Microsoft_Windows) feature that remembers a mouse button is down so that items can be highlighted or dragged without holding the mouse button down throughout.
* Customization of mouse or mouse alternatives' responsiveness to movement, double-clicking, and so forth.
* [ToggleKeys](http://en.wikipedia.org/wiki/ToggleKeys)[[9]](http://en.wikipedia.org/wiki/Computer_accessibility#cite_note-9) is a feature of [Microsoft Windows](http://en.wikipedia.org/wiki/Microsoft_Windows) 95 onwards. A high sound is heard when the CAPS LOCK, SCROLL LOCK, or NUM LOCK key is switched on and a low sound is heard when any of those keys are switched off.
* Customization of pointer appearance, such as size, color and shape.
* [Predictive text](http://en.wikipedia.org/wiki/Predictive_text)
* [Spell checkers](http://en.wikipedia.org/wiki/Spell_checker) and [grammar checkers](http://en.wikipedia.org/wiki/Grammar_checker)

**Software with support for learning disabilities**

* Cause and effect software[[10]](http://en.wikipedia.org/wiki/Computer_accessibility#cite_note-10)
* Switch accessible software
* Hand-eye co-ordination skills software
* Diagnostic assessment software
* [Mind mapping](http://en.wikipedia.org/wiki/Mind_map) software
* Study skills software
* Symbol-based software[[11]](http://en.wikipedia.org/wiki/Computer_accessibility#cite_note-11)
* [Text-to-speech](http://en.wikipedia.org/wiki/Text-to-speech)
* [Touch typing](http://en.wikipedia.org/wiki/Touch_typing) software

## Aegis-project – Open Accessibility framework (OAF)

<http://www.aegis-project.eu/index.php?option=com_content&view=article&id=176&Itemid=73>

The Open Accessibility Framework (OAF - [D1.2.1 AEGIS Open Accessibility Framework, version 1.2.5, CC licensed](http://www.aegis-project.eu/images/Deliverables/AEGIS_OAF_v1.2.5.pdf)) consists of two parts:

* A document describing the **framework of things needed for 3rd generation accessibility**, as validated by the prototypes and user/developer feedback in AEGIS
* A collection of largely **open source prototypes and code Deliverables** implementing various aspects of the OAF, proven in AEGIS and contributed back to the open source projects of which they are part

It contains the initial AEGIS Open Accessibility Framework (OAF) description. It is based upon:

* The **accessibility API and framework support from the existing Open Desktop** (GNOME Accessibility framework) and the **Java platform** (the Java Accessibility API, keyboard operability guidelines, and theme support);
* The **AEGIS generic accessibility framework requirements** (AEGIS ID1.2.1).

## The Java Accessibility API Interfaces and Classes

<http://docs.oracle.com/cd/E17802_01/j2se/javase/technologies/accessibility/docs/jaccess-1.3/doc/core-api.html>

<http://docs.oracle.com/javase/7/docs/technotes/guides/access/jaapi.html>

The [Java Accessibility API](http://java.sun.com/products/jfc/swingdoc-api/javax/accessibility/package-summary.html) defines a contract between individual user-interface components that make up a JavaTM application and an assistive technology that is providing access to that Java application.  If a Java application fully supports the Java Accessibility API, then it should be compatible with, and friendly toward, assistive technologies such as screen readers, screen magnifiers, etc.  With a Java application that fully supports the Java Accessibility API, no screen reader off screen model would be necessary because the API provides all of the information normally contained in an off screen model.

## US Section 508 Standards – software applications & operating systems

<http://www.epa.gov/inter508/standards/index.htm>

(a) When software is designed to run on a system that has a keyboard, product functions shall be executable from a keyboard where the function itself or the result of performing a function can be discerned textually.

(b) Applications shall not disrupt or disable activated features of other products that are identified as accessibility features, where those features are developed and documented according to industry standards. Applications also shall not disrupt or disable activated features of any operating system that are identified as accessibility features where the application programming interface for those accessibility features has been documented by the manufacturer of the operating system and is available to the product developer.

(c) A well-defined on-screen indication of the current focus shall be provided that moves among interactive interface elements as the input focus changes. The focus shall be programmatically exposed so that assistive technology can track focus and focus changes.

(d) Sufficient information about a user interface element including the identity, operation and state of the element shall be available to assistive technology. When an image represents a program element, the information conveyed by the image must also be available in text.

(e) When bitmap images are used to identify controls, status indicators, or other programmatic elements, the meaning assigned to those images shall be consistent throughout an application's performance.

(f) Textual information shall be provided through operating system functions for displaying text. The minimum information that shall be made available is text content, text input caret location, and text attributes.

(g) Applications shall not override user selected contrast and color selections and other individual display attributes.

(h) When animation is displayed, the information shall be displayable in at least one non-animated presentation mode at the option of the user.

(i) Color coding shall not be used as the only means of conveying information, indicating an action, prompting a response, or distinguishing a visual element.

(j) When a product permits a user to adjust color and contrast settings, a variety of color selections capable of producing a range of contrast levels shall be provided.

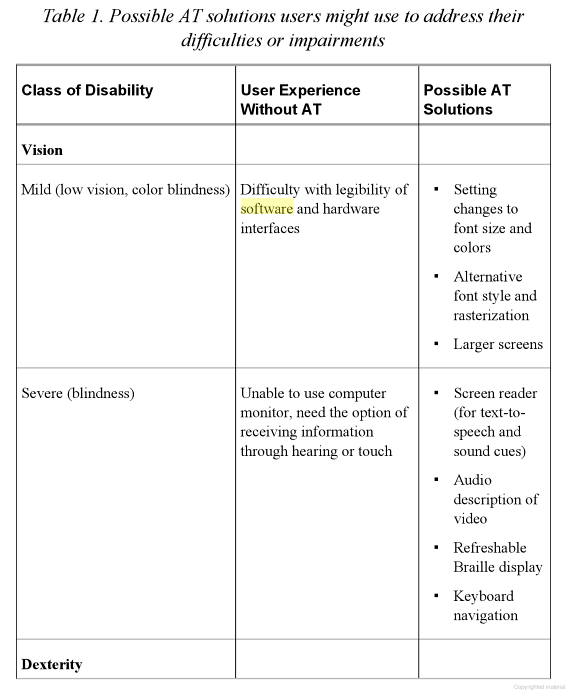
(k) Software shall not use flashing or blinking text, objects, or other elements having a flash or blink frequency greater than 2 Hz and lower than 55 Hz.

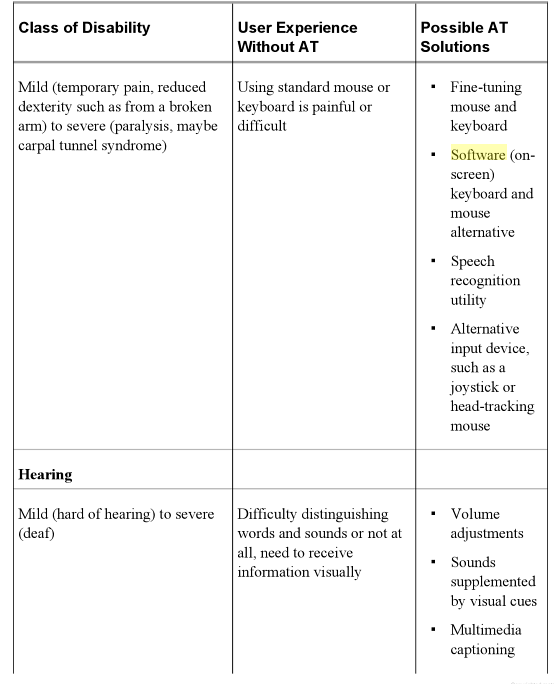
(l) When electronic forms are used, the form shall allow people using assistive technology to access the information, field elements, and functionality required for completion and submission of the form, including all directions and cues.

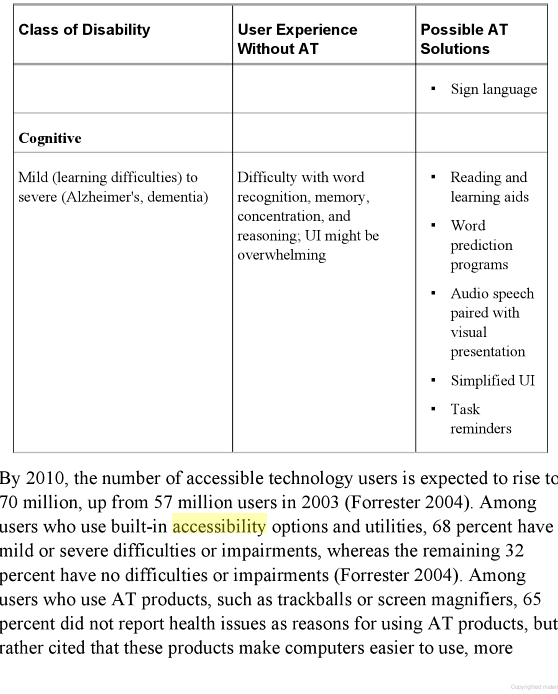
## “Engineering Software for Accessibility” – Microsoft Corporation

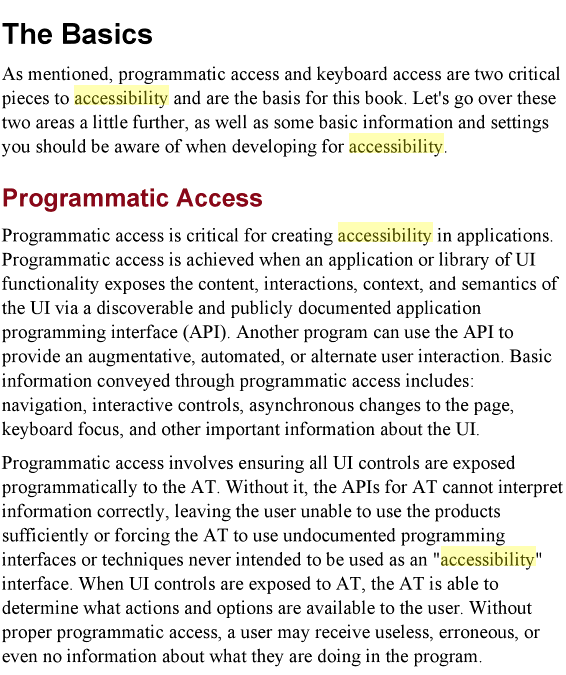
<http://books.google.co.uk/books?id=LVnfMYuHaoYC&printsec=frontcover&dq=accessibility+software&hl=en&sa=X&ei=fABYUqbRE8rT0QXz6IGoDg&ved=0CEoQ6AEwAA#v=onepage&q=accessibility%20software&f=false>

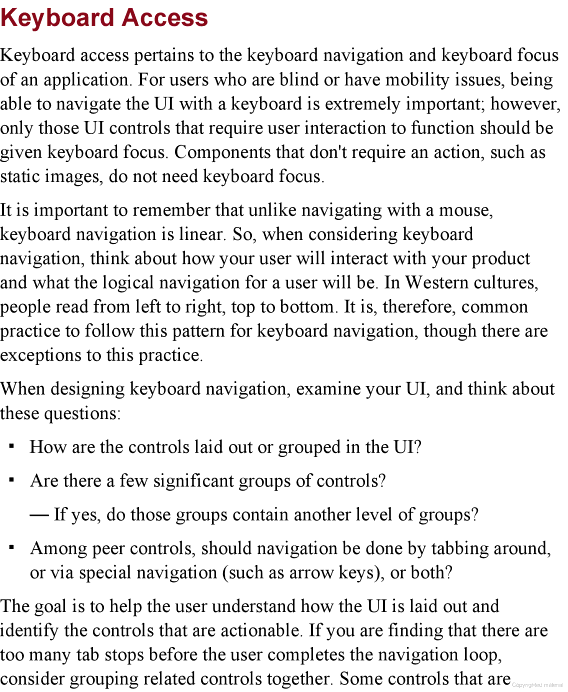
* When it comes to technology, accessibility pertains to a wide range of people with a range of abilities, not just the folks with disabilities.
* Accessible technology is technology that users can adapt to meet their visual, hearing, dexterity, cognitive, and speech needs and interaction preferences. Accessible technology includes accessibility options and utilities built into products, as well as specialty hardware and software add-ons called assistive technology (AT) that help individuals interact with a computer.
* There are essentially two types of users of accessible technology:
  1. Those who need it, because of disabilities or impairments, age-related conditions, or temporary conditions (such as limited mobility from a broken arm)
  2. Those who use it out of preference, for a more comfortable or convenient computing experience.
* A 2003-2004 study commissioned by Microsoft and conducted by Forrester Research found that over half – 57% – of computer users in the US between the ages of 18 and 64 benefit from accessible technology. Most of these users do not identify themselves as having a disability or impaired but expressed certain task-related difficulties or impairments when using a computer.

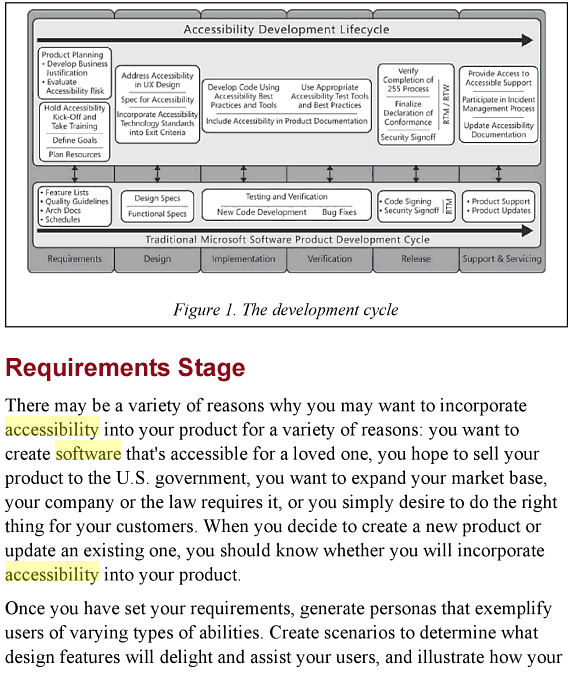


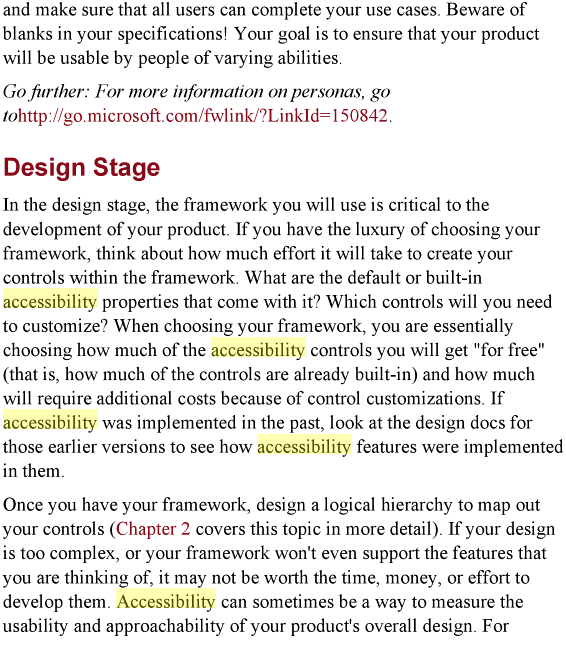












# 

# 

# 

# Research about accessibility by Microsoft

<http://www.microsoft.com/enable/research/default.aspx>

Microsoft commissioned Forrester Research to measure the market for accessible technology in the United States and to better understand how accessible technology is being used.

Findings are represented in the following two reports:

* [**The Market for Accessible Technology—The Wide Range of Abilities and Its Impact on Computer Use**](http://www.microsoft.com/enable/research/phase1.aspx) presents findings about individuals who are likely to benefit from the use of accessible technology.
* [**Accessible Technology in Computing—Examining Awareness, Use, and Future Potential**](http://www.microsoft.com/enable/research/phase2.aspx) presents findings about the use of computers among individuals with difficulties/impairments.

# Oxford Dictionaries – def accessibility

<http://oxforddictionaries.com/definition/english/accessible?q=accessibility#accessible__8>

### ***adjective***

* **1**(of a place) able to be reached or entered:the town is***accessible by***busthis room is not***accessible to***elderly people
* able to be easily obtained or used:making learning opportunities more***accessible to***adults
* easily understood or appreciated:an accessible account of his theories
* able to be reached, entered, or used by people who have a disability:features such as non-slip floors and accessible entrances
* **2**(of a person, especially one in a position of authority) friendly and easy to talk to; approachable:he is more accessible than most tycoons

# IBM – def software accessibility

<http://pic.dhe.ibm.com/infocenter/jviewmap/v8r8/index.jsp?topic=%2Fcom.ibm.ilog.jviews.defense.doc%2FContent%2FVisualization%2FDocumentation%2FJViews%2FJViews_Defense%2F_pubskel%2Fps_usradvfwork1429.html>

An accessible software product is software that can be used effectively by users with certain kinds of disabilities. Accessibility involves converting software that can only be used by people without disabilities into software that can be used by people with and without disabilities alike.

JViews Enterprise includes support for users with several kinds of disabilities and each kind of disability is handled by one or more software techniques. Some techniques are implemented by JViews Enterprise, whereas others, as in the case of thin-client applications, are implemented by the browser; operating system, and support software.

**Note**

Applications that use JViews Enterprise are not automatically accessible. They must use at least some of the APIs and techniques described here to become accessible. The design tools, such as the designers, and most of the JViews samples are not accessible. They are not part of the applications that you deploy to your users. Rather, the techniques for accessibility are exemplified by specific samples for each product.

The following types of disabilities can hinder a user's ability to work with visualization products created using JViews Enterprise.

* Physical disabilities that prevent the user from holding or controlling a mouse. For these users, JViews Enterprise provides keyboard navigation across a diagram or chart. See the section "Keyboard navigation" for your product.
* Physical disabilities that prevent the user from typing and using a keyboard with the same ease and speed as normal users. For these users, the operating system provides keyboard accessibility features. These features are accessible across Java/Swing, browsers, and JViews. See the section "Keyboard operation modes" for your product.
* Photosensitive epilepsy. For users with this type of disability, the blinking frequency of blinking objects must be limited. Only blink periods longer than 0.5 seconds are acceptable. This can be implemented using the JViews[IlvBlinkingColor](http://pic.dhe.ibm.com/infocenter/jviewmap/v8r8/topic/com.ibm.ilog.jviews.defense.doc/html/refjava/html/ilog/views/util/java2d/IlvBlinkingColor.html) and [IlvBlinkingPaint](http://pic.dhe.ibm.com/infocenter/jviewmap/v8r8/topic/com.ibm.ilog.jviews.defense.doc/html/refjava/html/ilog/views/util/java2d/IlvBlinkingPaint.html) classes.
* Low vision. For users with low vision, it might be necessary to enable “zoomed” or “high contrast” display. See the sections "Zoomed display" and "High contrast mode" for your product.
* Color blind users. For color blind users, you must present information in a form other than colors. You can let the application auto-select the colors from a palette that avoids combinations of colors that would be indistinguishable to a color blind user. See the section "Use of colors in accessible applications" for your product.
* Blind users. Use screen readers for blind users. A screen reader is a software application that reads portions of the screen aloud. Screen readers are available for Windows, either built into the operating system, or available from third-party vendors, and for Linux/Gtk. The built-in screen reader is a simpler version with less features. JViews Enterprise does not currently support these devices.

Also includes info on Java APi for keyboard navigation etc.

# References

Java accessibility APi

* <http://docs.oracle.com/cd/E17802_01/j2se/javase/technologies/accessibility/docs/jaccess-1.3/doc/core-api.html>
* <http://docs.oracle.com/javase/7/docs/technotes/guides/access/jaapi.html>

Books

* Engineering Software for Accessibility by Microsoft Corporation. Microsoft Press, 2009
* Sarah Horton, Access by Design : A Guide to Universal Usability for Web Designers , New Riders, 2005
* Clarkson et al (Ed), Designing Accessible Technology, Springer, 2006

OAF

* <http://www.aegis-project.eu/index.php?option=com_content&view=article&id=176&Itemid=73>

Web accessibility

* initiative
  + <http://www.w3.org/WAI/>
* Evaluation tool
  + <http://www.webaim.org/>

Research about accessibility by Microsoft

* <http://www.microsoft.com/enable/research/default.aspx>

US section 508 standards

* <http://www.epa.gov/inter508/standards/index.htm>

Oxford Dictionaries – definition of accessibility

* <http://oxforddictionaries.com/definition/english/accessible?q=accessibility#accessible__8>

IBM – definition of software accessibility

* <http://pic.dhe.ibm.com/infocenter/jviewmap/v8r8/index.jsp?topic=%2Fcom.ibm.ilog.jviews.defense.doc%2FContent%2FVisualization%2FDocumentation%2FJViews%2FJViews_Defense%2F_pubskel%2Fps_usradvfwork1429.html>

IBM ILOG Visualization for Java (JViews Enterprise Features)

* <http://www-01.ibm.com/software/integration/visualization/java/>