Notes by Kristy Davis ... Research based on developing a tool found in use studies and examined the lifecycle of software-based art from acquisition to installation to display to long-term storage. Looked at risks involved such as possible and expected loss of value in physical form, inherent decay, custodial neglect and information disassociation. Software-based risk is similar to time-basked risk – both systems are bespoke, both systems are easily changed and both technical environments rapidly decay. Magnifying factors influence consequence of risk, which includes anything that can be described as a lack of something, a lack of technical expertise, lack of information on emerging procedures and a lack of trained staff. The significance refers to values and meanings that items have for people and the community. A statement of significance should be a reasoned, clear summary of values, meaning and importance of an object or collection. The recovery and significance for traditional art objects that are damages equals a loss of value and for time-based medial there is recovery. Recoverability means is an element replaceable and can one find or produce an appropriate replacement and can one afford to acquire a replacement. Obsolescence and cost of recovery is expensive so it is best to act before the work starts to decline. Noted the importance of interviewing the artists and programmers of software and time-based works in order to define display specifications and parameters and to identify what can and cannot be changed and to discover how an artist wants their work preserved. Some ways to secure the work are to make clones immediately, create exhibition copies, gather operation manuals and specifications for software and hardware and test the system throughout the installation. Other issues with the display of software and time-based work is the maintenance of the device throughout the exhibition since long-term display is a magnifying factor with obsolescence and recoverability, but one can, over the life of a work, document any change to the system. Suggests that we should develop procedures for acquiring software-based work and to identify tools and test recovery strategies.

See full presentaion

{jcomments on}