`

Health Level Seven Standard

Context Management Specification
User Interface: Microsoft Windows OS
Version CM-1.0

DOCUMENT ID: HL7SIGVI_5_5_99

REVISION ID: May 24, 1999

FILE NAME: hl7_sigvi_windows_cm_1_0.doc

SUPERCEDES: n/a

Contents

1. INTRODUCTION **5**

2. TECHNOLOGY NEUTRALITY **6**

3. INTERFACE APPROACHES **7**

4. ARCHITECTURE **8**

4.1 JOINING A CONTEXT 9
4.2 CHANGING THE CONTEXT 9
4.3 REJOINING THE CONTEXT 11

Figures

Figure 1: Clinical link on icon 8
Figure 2: Clinical link broken icon 8
Figure 3: Clinical link changing icon 8
Figure 4: Context Dialog When No Applications Are Busy 10
Figure 5: Context Dialog When At Least One Application is Busy 10

Tables

Table 1: Required and recommended features 5

Preface

This document was prepared by Jeff Amfahr, Component Software International., on behalf of Health Level Seven's Special Interest Group on Visual Integration (formerly the Clinical Context Object Workgroup --- CCOW). Comments about the organization or wording of the document should be directed to the author (amfahr@csi-corporate.com). Comments about technical content should be directed to the Clinical Context Object Workgroup ccow@lists.hl7.org.

1. Introduction

This document specifies the user interface when using the common clinical context component as specified by the Clinical Context Object Workgroup (CCOW). It is assumed that the reader is familiar with the architectural specification for that software. The user interface explained in this document is intended to describe only those user interface features that are directly concerned with the clinical context component itself. Since a goal of this software is to be as seamless and natural for the user as possible, only the minimum set of user interface features is specified. No attempt to standardize the overall look and feel of clinical applications is made.

It is intended that this user interface be applicable to the full breadth of user interfaces in use today. The clinical context object itself has no specified user interface. All the information specified here is implemented by the clients of the clinical context object. Below is a table that indicates which of the following user interface concepts are required in order to be considered CCOW compliant and which are recommendations that should be used when possible and appropriate.

	Required	Recommended
Context status	Continuous, consistent, representation of state of context link using either text or icons or both.	Use icons and text
Status icons	If using icon, use standard icons.	
Status text	If using text, use standard text.	
Status location		At least one of the context link indicators should be prominently displayed.
Individual context status		Some visual indicator near the most related data to context data itself.
Joining context		Show text to indicate that the application is joining the context manager. If implementing this recommendation it is a requirement that you use the specified text.
Changing context	The dialog for indicating that there are conditional, busy or mapping agent problem responses to the survey is a required element.	
Rejoining Context	There must be a mechanism for rejoining the context if there is any way to break the context link from the application.	On rejoining, the user should be given the option of using the application's current context values or the values from the global context.
Modifying link status		Clicking on the link indicator instigates reversing the state of the link.

Table 1: Required and recommended features

2. Technology Neutrality

Since user interface tends to be a technology specific detail, this document attempts to address the user interface from a generic standpoint. The primary assumption is that the application uses a graphical user interface (as opposed to a command line) that provides support for overlapping windows. This would include all flavors of Windows™ and web-based applications, for example. Because there is no desire to specify the overall look and feel of clinical applications, the guidelines use only standard constructs available in all these systems.

The ability to internationalize this specification has been considered, but this initial version contains only English strings. Some minor modifications may be necessary for other languages, especially multibyte language systems. For example, maximum string lengths may be increased.

3. Interface Approaches

The overall architecture of the clinical context component is described in the Architectural Specification, but several features described in that document are especially applicable in the user interface scope.

- An application may choose to defer applying a context change until some time in the future. For example, an application that retrieves large medical image files (that require substantial processing) might choose to not retrieve images each time a different patient is selected as part of the clinical context. Instead, the application might wait for an explicit directive or gesture from the user before actually retrieving the image. An application that behaves in this manner must be sure that it does not show data for an earlier context. Blanking-out its data displays or minimizing itself are possible ways that this can be accomplished.

- An application for which a change in the context might result in the loss of work performed by the user can request that the user explicitly decide whether to proceed with the context change anyway, or to cancel the change. The solicitation of user input is performed by the application that is being used to change the context. The solicitation includes an identification of the application for which work might be lost and a description of the work that might be lost. An application that behaves in this manner is expected to be able to discard its work in progress and apply the context changes if instructed to do so. For example, a medication ordering application might indicate that the inputs for a medication order, which has not yet been completed by the user, will be lost if the context is changed to a different patient.

- When an application is unable to respond to a context change, perhaps because the user left it waiting for user input, the user is asked to explicitly decide how to proceed. The solicitation of user input is performed by the application that is being used to change the context. The solicitation includes the identification of the non-responsive application and indicates that the application cannot respond to a context change. For patient safety reasons, when there are applications that cannot respond to the changes, context changes will not be automatically applied to the applications that share a common context.

- When it is not desirable or possible for context changes to be automatically applied, either because there are applications for which work might be lost or there are busy applications that cannot be notified about context changes, the user can explicitly interact with these applications to correct the situation, and then apply the context changes. For example, the user might complete or terminate a dialog that was left open in order to enable an application to apply the context changes.

- When it is not desirable or possible for context changes to be automatically applied, the user can also decide to apply the context change only to the application that is being used to change the context. The decision to do this is typically in response to an interruption during which the user needs to momentarily divert his attention to a different context for a specific application. The application is, in effect, disconnected from the common context, and must clearly indicate this fact to the user in a visual manner. The application can be subsequently instructed by the user to reconnect and apply the common context. The common context may have changed between the time the application was disconnected and the time it is reconnected to the common context.

4. Architecture

Although the desire is for the common context to be seamless, the user does need to be aware when the component is and is not a part of the context. A common vocabulary for the context is required so users can be familiar with the meaning of common dialogs. In all text that the user sees, the context is referred to as the "clinical link." When the component is currently using the common context, the clinical link is "on". When a component leaves the context, the clinical link is "broken".

It is vital that the current state of the link be visible and apparent to the user. An application can use two methods to indicate the status of the link. The first is textual, in which case the text should be "Clinical link on", "Clinical link changing", or "Clinical link broken". The second method is using icons which are shown below. At least one of these indicators must be used, although both are recommended. If both indicators are used, they must always show the same state for the link.

Please note: the picture objects shown are samples only. The official versions of these images can be obtained from the HL7 web site.

link48.bmp

Figure 1: Clinical link on icon

broken48.bmp

Figure 2: Clinical link broken icon

changing48.bmp

Figure 3: Clinical link changing icon

If the application provides a user interface, this overall status indicator should be located prominently and persistently on the screen. It is also recommended that the application indicate the subjects (i.e. areas of clinical context) to which it is linked. If the application indicates these linked subjects, there are two recommended formats:

- The first format is to provide the indicator via a menu item, preferably in the same menu that allows the user to re-join the context. This menu should be hierarchical, with the main menu item labeled as "Supported clinical subjects" and the submenus having labels for each subject area (for example, "Patient", "User", etc.).

- The alternative format (which may be used in conjunction with the first approach) is to provide a visual indicator near a representative piece of clinical data. For example, an icon near the patient name to indicate that the patient subject is supported, an icon near the user name to indicate that the user subject is supported, etc. This icon should be similar to the normal clinical link status icons shown above, however they may be smaller in order to accommodate the actual area of the screen used. These icons should change when the link is broken to indicate that the subject is no longer linked.

There are three stages to using the common clinical context:

- Joining a context
- Changing a context
- Rejoining the context.

4.1 Joining a Context

It is assumed that most applications will attempt to join the common context when launching, with no user intervention. When joining the context, a transaction may be in progress. No components are allowed to join the context while a transaction is in progress. If the component chooses to block on the call to join the transaction (by setting the wait parameter to true), then a dialog must be displayed informing the user that the thread is blocked. This dialog must only be displayed if the transaction takes longer than one second. This dialog should be a standard system dialog with the text "Establishing clinical link". Once the context has been joined, this dialog should close itself with no further user interaction. At this point, the visual indicator for link status should indicate that the link is on, or if the context cannot be joined for whatever reason (e.g., the TooManyParticipants exception is raised), the link status should indicate a broken link.

4.2 Changing the Context

Once the context has been joined, the component will normally be in synch with the current context data. When another application desires to change the context, the context manager informs other applications by calling their ContextChangesPending method. At this point, the other applications need to change their status to indicate the patient link is changing. This tells the user that no context data change is presently possible. At some later point, the context manager will call either the ContextChangesAccepted or ContextChangesCanceled method. The link status should then change to indicate it is on.

If the current component desires to change the current context itself, it needs to inform the other CCOW clients. This document does not specify how new context data is selected by the user. Once the data has been entered, however, the context manager calls a mapping agent (if available) to confirm and add to the data. If the mapping agent finds no inconsistencies, the context manager surveys the registered applications. If all applications can change the current context, then the context data is changed as described above. If, however, any of the applications are unable to change because they are busy or because they require user interaction, or the mapping agent finds an inconsistency in the identifying information, the user must be queried for the appropriate action.

An application that requires user interaction (i.e., conditionally accepts) sends the context manager a string stating the reason and the potential consequences if the data change occurs. The format of that string should be declarative (i.e., state what the consequences will be, rather that query the user for information) and should be less than 64 characters in length. Some examples are "The open patient record will not be saved", "Annotations for record 123 will be lost" or "Current information will be saved". If an application is busy, meaning it is unable to respond to the survey, the context manager will return the string "This application appears to be busy" for that application.

In either of these cases, the component that requested the context change should then present the user with a dialog (Figure 4) containing the following text: "There may be a problem changing the current clinical data. The following application(s) reported a problem. Would you like to continue with the change?" followed by a list showing the application name(s) and reason(s) returned. This dialog contains three action buttons that must be labeled "OK", "Cancel", and "Break link". All of these choices dismiss the dialog box. The default choice is "Cancel". If at least one of the responses indicates an application is busy, then the "OK" choice must be disabled. If an application does not support the choice of "Cancel" (for example, because choosing a patient is a difficult process), then that choice should be disabled and "Break link" should be the default choice (Figure 5). Even if no other choice is possible other than "Break link", the dialog should be presented so that the user is aware they are breaking the patient link.

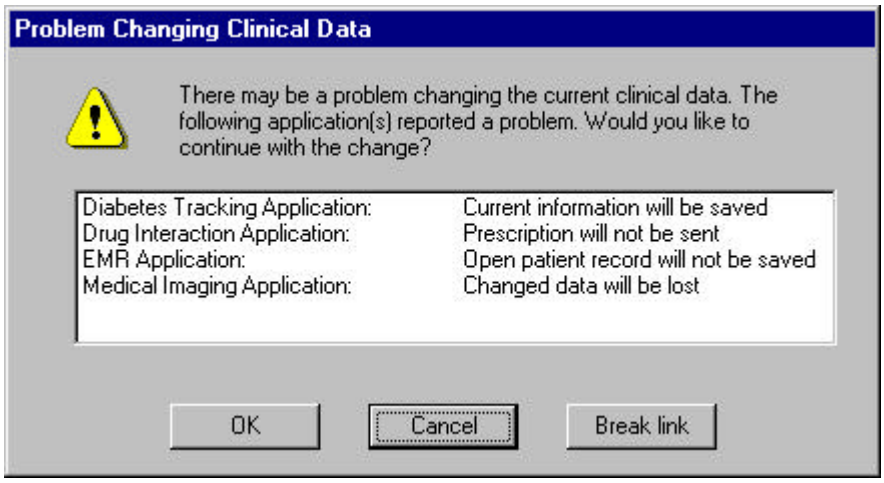

Figure 4: Context Dialog When No Applications Are Busy

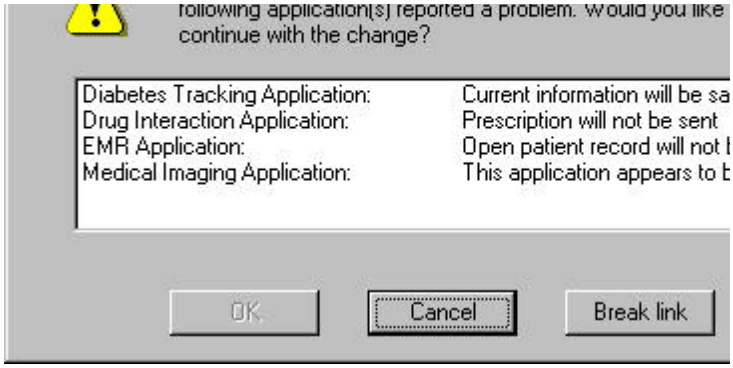

Figure 5: Context Dialog When At Least One Application is Busy

If the mapping agent finds an inconsistency in the identifying data set by the instigating application, it should send the context manager a string stating the reason for the inconsistency. The format of that string should be declarative and should be less than 64 characters in length. Some examples are “The IDs map to two different patients”, or “Patient identifiers are not consistent with data recorded”. The context manager should log these for future reporting as appropriate. The requesting application will receive a programmatic response identical to that received in the case of a busy application. The application name returned will be “Data mapping System” and the string returned by the context manager will be “Some of the IDs used were conflicting”. This result should be reported as described above for a busy application.

An application may be unable to interpret some or all of the context data that is established when another application changes the common context. For example, the selected patient might not be a patient known to all of the applications. In this case, the application should clearly indicate that it is unable to apply the context data. It should either minimize itself or blank-out its data displays. It is also acceptable for an application to do this if it is not the current application and accessing its data is a lengthy process. It would begin this process when it became the front most application. Note in either case that the application is still part of the common context and its status should indicate this.

4.3 Rejoining the Context

When an application is currently not a part of the common context (because the user has explicitly broken the link), there must be a mechanism for the user to rejoin the context (e.g. a menu item). If a menu item or equivalent is used it must be labeled “Rejoin Clinical Link”. This menu will include a sub-menu with two choices: “Use this applications’ data” and “Use global data”. When an application is part of the context this menu item or equivalent should be disabled.

Index

A

Architecture, 8

C

Clinical link
  broken icon, 8
  changing icon, 8
  on icon, 8
common clinical context, 8
  changing, 9
  joining, 9
  rejoining, 9

F

Figure 1: Clinical link on icon, 8
Figure 2: Clinical link broken icon, 8
Figure 3: Clinical link changing icon, 8
Figure 4: Context Dialog When No Applications Are Busy, 10
Figure 5: Context Dialog When At Least One Application is Busy, 10

I

Interface Approaches, 7
Introduction, 5
Introduction to Context ManagementSpecification
  Table 1 Required and recommended features, 1

T

Table 1 Required and recommended features, 5
Technology Neutrality, 6

