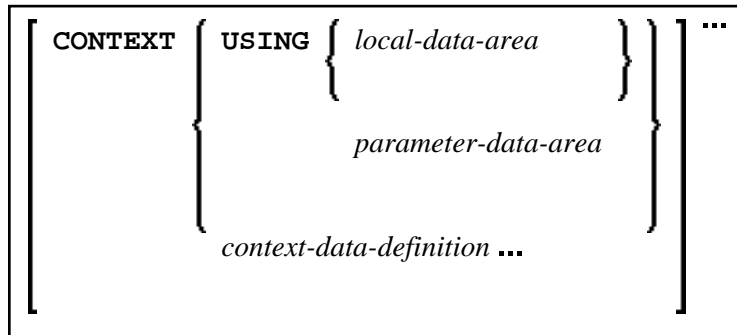


Definition von Kontext-Variablen für den Natural RPC

Allgemeine Syntax von DEFINE DATA CONTEXT:



Dieses Kapitel behandelt folgende Themen:

- Funktion
- Einschränkungen
- Syntax-Beschreibung

Eine Erläuterung der in dem Syntax-Diagramm verwendeten Symbole entnehmen Sie dem Abschnitt *Syntax-Symbole*.

Gehört zur Funktionsgruppe: *Natural Remote Procedure Call*

Funktion

Das DEFINE DATA CONTEXT-Statement wird im Zusammenhang mit dem Natural Remote Procedure Call (RPC) verwendet. Es dient dort zur Definition von als Kontext-Variablen bekannte Variablen, die für mehrere Subprogramme auf einem externen (Remote-)Rechner innerhalb einer Konversation zur Verfügung stehen sollen, ohne dass Sie die Variablen explizit als Parameter mit den entsprechenden CALLNAT-Statements übergeben müssen.

Eine Kontext-Variablen wird über ihren Namen referenziert, und ihr Inhalt wird von allen in einer Konversation ausgeführten Programmierobjekten gemeinsam benutzt, die auf diesen Namen verweisen. Die Variable wird vom ersten ausgeführten Programmierobjekt zugewiesen, das die Definition der Variable enthält, und wird freigegeben, wenn die Konversation beendet ist.

Kontext-Variablen können auch in einem nicht-konversationellen CALLNAT benutzt werden. In diesem Fall sind die Kontext-Variablen nur während eines einzigen Aufrufs dieses CALLNAT vorhanden, aber die Variablen können mit allen ihren aufgerufenen Calleees gemeinsam benutzt werden.

Eine Kontext-Variablen wird nicht mit Subprogrammen gemeinsam benutzt, die innerhalb der Konversation aufgerufen werden. Wenn ein solches Subprogramm oder eines seiner Calleees eine Kontext-Variablen referenziert, wird ein separater Speicherbereich für diese Variable zugewiesen.

Die optionale `INIT`-Klausel wird bei jedem ausgeführten Programmierobjekt ausgewertet, das diese Klausel enthält (nicht nur im Programmierobjekt, das die Variable zuweist). Bei globalen Variablen funktioniert `INIT` anders.

Weitere Informationen siehe *Defining a Conversation Context* in der *Natural Remote Procedure Call (RPC)*-Dokumentation.

Einschränkungen

Eine Kontext-Variablen muss auf Level 01 definiert werden. Andere Levels werden nur bei einer Redefinition benutzt.

Syntax-Beschreibung

USING <i>local-data-area</i>	<p>Eine Local Data Area (LDA) enthält Datenelemente, die in einem einzelnen Natural-Modul benutzt werden sollen. Sie können mehr als eine Data Area referenzieren; in diesem Fall müssen Sie die reservierten Wörter <code>CONTEXT</code> und <code>USING</code> wiederholen, zum Beispiel:</p> <pre>DEFINE DATA CONTEXT USING DATX_L CONTEXT USING DATX_P ... END-DEFINE ;</pre> <p>Weitere Informationen siehe <i>Felder in einer separaten Data Area definieren</i> im Leitfaden zur Programmierung.</p>
USING <i>parameter-data-area</i>	Eine Parameter Data Area enthält Datenelemente, die als Parameter in einem Subprogramm, einer externen Subroutine oder in einem Dialog benutzt werden.
<i>context-data-definition</i>	Kontext-Daten können auch direkt innerhalb eines Programms oder einer Routine definiert werden. Bei einer direkten Daten-Definition gilt die weiter unten gezeigte Syntax.
END-DEFINE	Das für Natural reservierte Wort <code>END-DEFINE</code> muss zum Beenden des <code>DEFINE DATA</code> -Statements benutzt werden.

Kontextdaten-Definition

Kontext-Daten können direkt innerhalb eines Programms oder einer Routine definiert werden. Bei einer direkten Daten-Definition gilt die folgende Syntax:

$\text{level} \left\{ \begin{array}{l} \text{variable-definition} \\ \text{redefinition} \\ \text{handle-definition} \end{array} \right\}$

Weitere Informationen siehe *Felddefinitionen im DEFINE DATA-Statement* im Leitfaden zur Programmierung.

<i>level</i>	Eine anwendungsunabhängige Variable muss auf Stufe (Level) 01 definiert werden. Andere Levels werden nur bei einer Redefinition benutzt.
<i>variable-definition</i>	Eine Variablen-Definition wird zur Definition eines einzelnen Feldes oder einer einzelnen Variable benutzt, die einen Wert (Skalar) oder mehrere Werte (Array) haben kann. Siehe <i>Definition von Variablen</i> . Anmerkung: Die CONSTANT-Klausel darf in diesem Zusammenhang nicht benutzt werden.
<i>redefinition</i>	Mit einer Redefinition redefinieren Sie eine Gruppe, ein View, ein DDM-Feld oder ein einzelnes Feld oder eine einzelne Variable (d.h. ein Skalar oder ein Array). Siehe <i>Redefinition</i> .
<i>handle-definition</i>	Eine Handle identifiziert ein Dialog-Element im Code und wird in Handle-Variablen gespeichert. Siehe <i>Handle-Definition</i> .

Anmerkung:

Die sich aus einer Redefinition ergebenden Felder werden nicht als eine Kontext-Variable angesehen. Diese Felder werden als lokale Variablen behandelt.