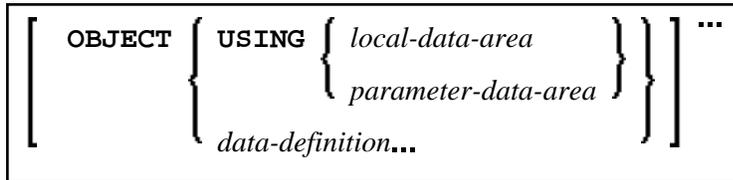


Definition von NaturalX-Objekten

Allgemeine Syntax von DEFINE DATA OBJECT:



Dieses Kapitel behandelt folgende Themen:

- Funktion
- Syntax-Beschreibung

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

Funktion

Das DEFINE DATA OBJECT-Statement wird benutzt in einem Subprogramm oder einer Klasse in Zusammenhang mit NaturalX. Weitere Informationen, siehe *NaturalX-Dokumentation im Leitfaden zur Programmierung*.

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 OBJECT und USING wiederholen, zum Beispiel:</p> <pre>DEFINE DATA OBJECT USING DATX_L OBJECT USING DATX_P ... END-DEFINE ;</pre> <p>Weitere Informationen siehe <i>Felddefinitionen im DEFINE DATA-Statement im Leitfaden zur Programmierung.</i></p>
USING <i>parameter-data-area</i>	<p>Eine mit DEFINE DATA OBJECT definierte Data Area kann eine Parameter Data Area (PDA) sein. Wenn Sie eine PDA als eine Object Data Area benutzen, können Sie sich die zusätzliche Mühe ersparen, eine Object Data Area zu erstellen, die dieselbe Struktur wie die PDA hat.</p>
<i>data-definition</i>	<p>Daten können auch direkt mit der im Abschnitt <i>Direkte Datendefinition</i> weiter unten angegebenen Syntax definiert werden.</p>
END-DEFINE	<p>Das für Natural reservierte Wort END-DEFINE muss zum Beenden des DEFINE DATA-Statements benutzt werden.</p>

Direkte Datendefinition

Daten können auch mit der folgenden Syntax direkt definiert werden:

$\left\{ \begin{array}{l} \textit{level} \left\{ \begin{array}{l} \textit{group-name} [(\textit{array-definition})] \\ \textit{variable-definition} \\ \textit{view-definition} \\ \textit{redefinition} \\ \textit{handle-definition} \end{array} \right\} \end{array} \right\}$

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

<i>level</i>	<p>Dies ist eine ein- oder zweistellige Zahl im Bereich von 01 bis 99 (die vorangestellte 0 ist nicht erforderlich), die in Verbindung mit der Gruppierung von Feldern verwendet wird. Felder mit einer Level-Nummer von 02 an aufwärts werden als Teil einer unmittelbar vorangehenden Gruppe mit einer jeweils nächstniedrigeren Level-Nummer betrachtet.</p> <p>Durch die Definition einer Gruppe (die auch nur aus einem Feld bestehen kann) ist es möglich, durch Angabe lediglich des Gruppennamens eine ganze Reihe von aufeinanderfolgenden Feldern gleichzeitig zu referenzieren. Bei manchen Statements (CALL, CALLNAT, RESET, WRITE usw.) können Sie den Gruppennamen als Aufrufnamen angeben, um die in der Gruppe enthaltenen Felder zu referenzieren.</p> <p>Eine Gruppe kann aus weiteren Gruppen bestehen. Bei der Vergabe von Level-Nummern für eine Gruppe darf kein Level ausgelassen werden. Eine View-Definition muss immer auf Level 1 definiert werden.</p>
<i>group-name</i>	<p>Der Name einer Gruppe. Der Name muss den Regeln zur Definition eines Natural-Variablenamens entsprechen. Siehe auch die folgenden Abschnitte:</p> <ul style="list-style-type: none"> ● <i>Namen von Benutzervariablen</i> in der Dokumentation <i>Natural Studio</i> benutzen. ● <i>Datenstrukturen qualifizieren</i> im Leitfaden zur Programmierung.
<i>array-definition</i>	<p>Mit <i>array-definition</i> definieren Sie die Unter- und Obergrenze einer Dimension in einer Array-Definition. Siehe auch <i>Definition einer Array-Dimension</i>.</p>
<i>variable-definition</i>	<p>Eine <i>variable-definition</i> wird zur Definition eines einzelnen Feldes (Skalar) oder einer einzelnen Variablen (Array) verwendet. Siehe <i>Variablen-Definition</i>.</p>
<i>view-definition</i>	<p>Eine <i>view-definition</i> wird benutzt, um ein View mit Bestandteilen aus einem Datendefinitionsmodul (DDM) zu definieren. Siehe <i>View Definition</i>.</p>
<i>redefinition</i>	<p>Eine <i>redefinition</i> kann zur Redefinition einer Gruppe, eines Views, eines DDM-Feldes oder eines einzelnen Feldes oder einer einzelnen Variable benutzt werden (d.h. Skalar oder Array). Siehe <i>Redefinition</i>.</p>
<i>handle-definition</i>	<p>Eine Handle identifiziert ein Dialog-Element im Code und wird in Handle-Variablen gespeichert. Siehe <i>Handle-Definition</i>.</p>