

## **CommoditiesL**

COLLABORATORS

	TITLE : CommoditiesL		
ACTION	NAME	DATE	SIGNATURE
WRITTEN BY		July 22, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>CommoditiesL</b>	<b>1</b>
1.1	CommoditiesL . . . . .	1
1.2	TMP:Modula-2/CommoditiesL.def . . . . .	1

## Chapter 1

# CommoditiesL

### 1.1 CommoditiesL

#### Prozeduren

ActivateCxCObj	AddIEvents	AttachCxCObj
ClearCxCObjError	CreateCxCObj	CxBroker
CxMsgData	CxMsgID	CxMsgType
CxCObjError	CxCObjType	DeleteCxCObj
DeleteCxCObjAll	DisposeCxMsg	DivertCxMsg
EnqueueCxCObj	InsertCxCObj	InvertKeyMap
MatchIX	ParseIX	RemoveCxCObj
RouteCxMsg	SetCxCObjPri	SetFilter
SetFilterIX	SetTranslate	

#### Konstanten

commoditiesMinVersion	commoditiesName
-----------------------	-----------------

#### Variablen

commoditiesBase	commoditiesVersion
-----------------	--------------------

#### Typ-Deklarationen

DescrPtr

### 1.2 TMP:Modula-2/CommoditiesL.def

```
DEFINITION MODULE CommoditiesL{"commodities.library",36};
(* 04-Sep-92/bp *)
```

```
FROM SYSTEM IMPORT ADDRESS;
```

```
FROM InputEvent IMPORT
  InputEvent , InputEventPtr ;
```

```
FROM KeyMapD IMPORT
```

```
    KeyMapPtr ;

IMPORT
    d:CommoditiesD,R,ExecD;

TYPE
    DescrPtr =ADDRESS;

VAR
    commoditiesVersion:INTEGER;
    commoditiesBase:ExecD. LibraryPtr ;

CONST
    commoditiesMinVersion=36;
    commoditiesName="commodities.library";

PROCEDURE  ActivateCxObj (
    co{R.A0}:d. CxObjPtr ;
    true{R.D0}:BOOLEAN):BOOLEAN; CODE -42;

PROCEDURE  AddIEvents (ie{R.A0}: InputEventPtr ); CODE -180;

PROCEDURE  AttachCxObj (
    headobj{R.A0}:d. CxObjPtr ;
    co{R.A1}:d. CxObjPtr ); CODE -84;

PROCEDURE  ClearCxObjError (co{R.A0}:d. CxObjPtr ); CODE -72;

PROCEDURE  CreateCxObj (
    type{R.D0}:LONGCARD;
    arg1{R.A0}:ADDRESS;
    arg2{R.A1}:ADDRESS):d. CxObjPtr ; CODE -30;

PROCEDURE  CxBroker (
    VAR nb{R.A0}:d. NewBroker ;
    VAR error{R.D0}:LONGCARD):d. CxObjPtr ; CODE -36;

PROCEDURE  CxMsgData (cxm{R.A0}:d. CxMsgPtr ):ADDRESS; CODE -144;

PROCEDURE  CxMsgID (cxm{R.A0}:d. CxMsgPtr ):LONGINT; CODE -150;

PROCEDURE  CxMsgType (cxm{R.A0}:d. CxMsgPtr ):LONGCARD; CODE -138;

PROCEDURE  CxObjError (co{R.A0}:d. CxObjPtr ):d. CoErrorSet ; CODE -66;

PROCEDURE  CxObjType (co{R.A0}:d. CxObjPtr ):LONGCARD; CODE -60;

PROCEDURE  DeleteCxObj (co{R.A0}:d. CxObjPtr ); CODE -48;

PROCEDURE  DeleteCxObjAll (co{R.A0}:d. CxObjPtr ); CODE -54;

PROCEDURE  DisposeCxMsg (cxm{R.A0}:d. CxMsgPtr ); CODE -168;

PROCEDURE  DivertCxMsg (
    cxm{R.A0}:d. CxMsgPtr ;
    headobj{R.A1}:d. CxObjPtr ;
    ret{R.A2}:d. CxObjPtr ); CODE -156;
```

---

```
PROCEDURE EnqueueCxObj (
    headobj{R.A0}:d. CxObjPtr ;
    co{R.A1}:d. CxObjPtr ); CODE -90;

PROCEDURE InsertCxObj (
    headobj{R.A0}:d. CxObjPtr ;
    co{R.A1}:d. CxObjPtr ;
    pred{R.A2}:d. CxObjPtr ); CODE -96;

PROCEDURE InvertKeyMap (
    ansicode{R.D0}:LONGCARD;
    VAR ie{R.A0}: InputEvent ;
    km{R.A1}: KeyMapPtr ):BOOLEAN; CODE -174;

PROCEDURE (*38*) MatchIX(
    event{R.A0}: InputEventPtr ;
    ix{R.A1}:d. IXPtr ):BOOLEAN; CODE -204;

PROCEDURE ParseIX (
    descr{R.A0}: DescrPtr ;
    VAR ix{R.A1}:d. IX ):LONGINT; CODE -132;

PROCEDURE RemoveCxObj (co{R.A0}:d. CxObjPtr ); CODE -102;

PROCEDURE RouteCxMsg (
    cxm{R.A0}:d. CxMsgPtr ;
    co{R.A1}:d. CxObjPtr ); CODE -162;

PROCEDURE SetCxObjPri (
    co{R.A0}:d. CxObjPtr ;
    pri{R.D0}:LONGINT); CODE -78;

PROCEDURE SetFilter (
    filter{R.A0}:d. CxObjPtr ;
    text{R.A1}: DescrPtr ); CODE -120;

PROCEDURE SetFilterIX (
    filter{R.A0}:d. CxObjPtr ;
    ix{R.A1}:d. IXPtr ); CODE -126;

PROCEDURE SetTranslate (
    translator{R.A0}:d. CxObjPtr ;
    ie{R.A1}: InputEventPtr ); CODE -114;

END CommoditiesL.lib37
```

---