

Linkedlist of products that represent children dependencies.

</TD>

</TR>

<TR>

<TD ALIGN="right" VALIGN="top" WIDTH="1%">

EXECUTED

<NOBR>

byte

</NOBR>

</TD>

<TD VALIGN="top">

Indicates if the product has been generated.

</TD>

</TR>

<TR>

<TD ALIGN="right" VALIGN="top" WIDTH="1%">

NAME

<NOBR>

string

</NOBR>

</TD>

<TD VALIGN="top">

Name of the product.

</TD>

</TR>

<TR>

<TD ALIGN="right" VALIGN="top" WIDTH="1%">

DESCRIPTION

<NOBR>

string

</NOBR>

</TD>

<TD VALIGN="top">

Description of the product.

</TD>

</TR>

<TR>

<TD ALIGN="right" VALIGN="top" WIDTH="1%">

PLUGIN

<NOBR>

object reference

</NOBR>

</TD>

<TD VALIGN="top">

Plugin object that implements the product.

</TD>

</TR>

```

<TR>
<TD ALIGN="right" VALIGN="top" WIDTH="1%">
<FONT CLASS="param_name">PLUGIN_INFO</FONT><BR>
<FONT CLASS="param_attrib"><NOBR>
object reference
</NOBR></FONT>
</TD>
<TD VALIGN="top">
  PluginInfo object that has the plugin
  information.

```

```

</TD>
</TR>
</TABLE>

```

```

<P>
<TABLE CELLPADDING="3" CELLSPACING="0" CLASS="listing">
<TR><TD CLASS="title">
<A NAME="_routine_summary">Routine Summary</A>
&nbsp;<FONT CLASS="list_tagline">15 routines</FONT>
</TD></TR>
<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" STYLE="BORDER-BOTTOM:
1px groove" WIDTH="100%"><TR><TD ALIGN="left" VALIGN="top"><P
CLASS="code_header">procedure <A
HREF="#_Product__define">Product__define</A></P></TD><TD ALIGN="right"
VALIGN="top"></TD><TR><TD><P class="first_line"></P></TD></TR></TABLE>
<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" STYLE="BORDER-BOTTOM:
1px groove" WIDTH="100%"><TR><TD ALIGN="left" VALIGN="top"><P
CLASS="code_header">function <A HREF="#_Product::Init">Product::Init</A>()</P></TD><TD
ALIGN="right" VALIGN="top"><FONT CLASS="param_attrib">private
</FONT></TD><TR><TD><P class="first_line"> Creates and initializes the
object.</P></TD></TR></TABLE>
<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" STYLE="BORDER-BOTTOM:
1px groove" WIDTH="100%"><TR><TD ALIGN="left" VALIGN="top"><P
CLASS="code_header">procedure <A
HREF="#_Product::Cleanup">Product::Cleanup</A></P></TD> <TD ALIGN="right"
VALIGN="top"><FONT CLASS="param_attrib">private </FONT></TD><TR><TD><P
class="first_line"> Frees the resources used by the object.</P></TD></TR></TABLE>
<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" STYLE="BORDER-BOTTOM:
1px groove" WIDTH="100%"><TR><TD ALIGN="left" VALIGN="top"><P
CLASS="code_header">procedure <A
HREF="#_Product::SetProperty">Product::SetProperty</A>, [NAME=<EM>string</EM>],
[DESCRIPTION=<EM>string</EM>]</P></TD><TD ALIGN="right"
VALIGN="top"></TD><TR><TD><P class="first_line"> Set individual property
values.</P></TD></TR></TABLE>
<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" STYLE="BORDER-BOTTOM:
1px groove" WIDTH="100%"><TR><TD ALIGN="left" VALIGN="top"><P
CLASS="code_header">procedure <A
HREF="#_Product::GetProperty">Product::GetProperty</A>, NAME=<EM>NAME</EM>,

```

DESCRIPTION=DESCRIPTION</P></TD><TD ALIGN="right" VALIGN="top"></TD><TR><TD><P class="first_line"> Get individual properties.</P></TD></TR></TABLE>

<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" STYLE="BORDER-BOTTOM: 1px groove" WIDTH="100%"><TR><TD ALIGN="left" VALIGN="top"><P CLASS="code_header">procedure Product::AddParent, product</P></TD><TD ALIGN="right" VALIGN="top"></TD><TR><TD><P class="first_line"> Sets a parent dependency for the object and automatically sets a reference from parent to child.</P></TD></TR></TABLE>

<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" STYLE="BORDER-BOTTOM: 1px groove" WIDTH="100%"><TR><TD ALIGN="left" VALIGN="top"><P CLASS="code_header">procedure Product::AddChild, product</P></TD><TD ALIGN="right" VALIGN="top"></TD><TR><TD><P class="first_line"> Sets a child dependency for the object and set a reference from child to parent.</P></TD></TR></TABLE>

<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" STYLE="BORDER-BOTTOM: 1px groove" WIDTH="100%"><TR><TD ALIGN="left" VALIGN="top"><P CLASS="code_header">function Product::GetParents()</P ></TD><TD ALIGN="right" VALIGN="top"></TD><TR><TD><P class="first_line"> Gets the parent dependencies of the product </P></TD></TR></TABLE>

<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" STYLE="BORDER-BOTTOM: 1px groove" WIDTH="100%"><TR><TD ALIGN="left" VALIGN="top"><P CLASS="code_header">function Product::GetChildren() </P></TD><TD ALIGN="right" VALIGN="top"></TD><TR><TD><P class="first_line"> Gets the children referencies.</P></TD></TR></TABLE>

<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" STYLE="BORDER-BOTTOM: 1px groove" WIDTH="100%"><TR><TD ALIGN="left" VALIGN="top"><P CLASS="code_header">procedure Product::MakeProduct</P ></TD><TD ALIGN="right" VALIGN="top"></TD><TR><TD><P class="first_line"> Executes the plugin associated with the Product to generate the output data.</P></TD></TR></TABLE>

<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" STYLE="BORDER-BOTTOM: 1px groove" WIDTH="100%"><TR><TD ALIGN="left" VALIGN="top"><P CLASS="code_header">procedure Product::SetPlugin, plugin</P></TD><TD ALIGN="right" VALIGN="top"></TD><TR><TD><P class="first_line"> Sets the Plugin object on which is based the Product.</P></TD></TR></TABLE>

<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" STYLE="BORDER-BOTTOM: 1px groove" WIDTH="100%"><TR><TD ALIGN="left" VALIGN="top"><P CLASS="code_header">procedure Product::SetPluginInfo, plugin_info</P></TD><TD ALIGN="right" VALIGN="top"></TD><TR><TD><P class="first_line"> Sets the PluginInfo object that stores the Plugin information.</P></TD></TR></TABLE>

<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" STYLE="BORDER-BOTTOM: 1px groove" WIDTH="100%"><TR><TD ALIGN="left" VALIGN="top"><P CLASS="code_header">procedure Product::AssocPluginInfo2Plugin

```

</A></P></TD><TD ALIGN="right" VALIGN="top"></TD><TR><TD><P class="first_line">
Creates an association from PluginInfo to Plugin object of the
Product.</P></TD></TR></TABLE>
<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" STYLE="BORDER-BOTTOM:
1px groove" WIDTH="100%"><TR><TD ALIGN="left" VALIGN="top"><P
CLASS="code_header">function <A HREF="#_Product::GetPlugin">Product::GetPlugin</A>()</P
></TD><TD ALIGN="right" VALIGN="top"></TD><TR><TD><P class="first_line"> Gets the
Plugin object on which the Product is based.</P></TD></TR></TABLE>
<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" STYLE="BORDER-BOTTOM:
1px groove" WIDTH="100%"><TR><TD ALIGN="left" VALIGN="top"><P
CLASS="code_header">function <A
HREF="#_Product::GetPluginInfo">Product::GetPluginInfo</A >()</P></TD><TD ALIGN="right"
VALIGN="top"></TD><TR><TD><P class="first_line"> Gets the PluginInfo object with the
information of the Plugin on which the Product is based.</P></TD></TR></TABLE>
</TABLE>
<P>
<TABLE CELLPADDING="3" CELLSPACING="0" CLASS="listing">
<TR><TD CLASS="title"><A NAME="_routine_details">Routine Details</A></TD></TR>
</TABLE>
<P>
<TABLE WIDTH="100%" BORDER="0"><TR><TD>
<H3 CLASS="routine"><A NAME="_Product__define">
Product__define
</A></H3>
</TD><TD ALIGN="right">

</TD></TR></TABLE>
<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" WIDTH="100%"><TR><TD
ALIGN="left" VALIGN="top"><P CLASS="code_header">procedure
Product__define</P></TD></TR></TABLE>
<P>

<P>
<DL>
<DT CLASS="attribute">Requires
<DD CLASS="attribute_item">
linkedlist logmsg

</DD></DT>
<DT CLASS="attribute">History
<DD CLASS="attribute_item">

Fri Jan 14 17:08:04 2005, Antonio Santiago
(santiago@grahi.upc.edu - http://asantiago.gentelibre.org)
</DD></DT>
<DT CLASS="attribute">Author
<DD CLASS="attribute_item">

```

Antonio Santiago
(santiago@grahi.upc.edu - <http://asantiago.gentelibre.org>)

```
</DD></DT>
</DL>
<P>
<P>
<P>
<HR CLASS="divider">
<TABLE WIDTH="100%" BORDER="0"><TR><TD>
<H3 CLASS="routine"><A NAME="_Product::Init">
Product::Init
</A></H3>
</TD><TD ALIGN="right">
<FONT CLASS="param_attrib">private </FONT>
</TD></TR></TABLE>
<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" WIDTH="100%"><TR><TD
ALIGN="left" VALIGN="top"><P CLASS="code_header">function
Product::Init()</P></TD></TR></TABLE>
<P>
```

Creates and initializes the object.

```
<P>
<DL>
<DT CLASS="attribute">Returns
<DD CLASS="attribute_item">
1 if successful; 0 otherwise
```

```
</DD></DT>
</DL>
<P>
<P>
<P>
<HR CLASS="divider">
<TABLE WIDTH="100%" BORDER="0"><TR><TD>
<H3 CLASS="routine"><A NAME="_Product::Cleanup">
Product::Cleanup
</A></H3>
</TD><TD ALIGN="right">
<FONT CLASS="param_attrib">private </FONT>
</TD></TR></TABLE>
<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" WIDTH="100%"><TR><TD
ALIGN="left" VALIGN="top"><P CLASS="code_header">procedure
Product::Cleanup</P></TD></TR></TABLE>
<P>
```

Frees the resources used by the object.

```

<P>
<P>
<P>
<HR CLASS="divider">
<TABLE WIDTH="100%" BORDER="0"><TR><TD>
<H3 CLASS="routine"><A NAME="_Product:: SetProperty">
Product:: SetProperty
</A></H3>
</TD><TD ALIGN="right">

</TD></TR></TABLE>
<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" WIDTH="100%"><TR><TD
ALIGN="left" VALIGN="top"><P CLASS="code_header">procedure Product:: SetProperty, [<A
HREF=#_Product:: SetProperty_keyword_NAME>NAME</A>=<EM>string </EM>], [<A
HREF=#_Product:: SetProperty_keyword_DESCRIPTION>DESCRIPTION
</A>=<EM>string</EM>]</P></TD></TR></TABLE>
<P>

```

Set individual property values.

```

<P>
<P>
<TABLE CELLPADDING="3" CELLSPACING="0" CLASS="parameters">
<TR BGCOLOR="#EEEEFF" CLASS="small_title">
<TD COLSPAN=2>Keywords</TD>
</TR>
<TR BGCOLOR="white" CLASS="parameters">
<TD ALIGN="right" VALIGN="top" WIDTH="1%">
<A NAME="_Product:: SetProperty_keyword_NAME"/>
<FONT CLASS="param_name">NAME</FONT><BR>
<FONT CLASS="param_attrib">
<NOBR>in, optional</NOBR><BR>
<NOBR>string</NOBR>
</FONT>
</TD>
<TD VALIGN="top">
Name of the product.
</TD>
</TR>
<TR BGCOLOR="white" CLASS="parameters">
<TD ALIGN="right" VALIGN="top" WIDTH="1%">
<A NAME="_Product:: SetProperty_keyword_DESCRIPTION"/>
<FONT CLASS="param_name">DESCRIPTION</FONT><BR>
<FONT CLASS="param_attrib">
<NOBR>in, optional</NOBR><BR>
<NOBR>string</NOBR>
</FONT>

```

```

</TD>
<TD VALIGN="top">
  Description of the
    products.
</TD>
</TR>
</TABLE>
<P>
<HR CLASS="divider">
<TABLE WIDTH="100%" BORDER="0"><TR><TD>
<H3 CLASS="routine"><A NAME="_Product::GetProperty">
Product::GetProperty
</A></H3>
</TD><TD ALIGN="right">

</TD></TR></TABLE>
<TABLE CELLPACING="2" CELLPADDING="0" BORDER="0" WIDTH="100%"><TR><TD
ALIGN="left" VALIGN="top"><P CLASS="code_header">procedure Product::GetProperty, <A
HREF=#_Product::GetProperty_keyword_NAME>NAME</A>=<EM>NAME </EM>, <A
HREF=#_Product::GetProperty_keyword_DESCRIPTION>DESCRIPTION
</A>=<EM>DESCRIPTION</EM></P></TD></TR></TABLE>
<P>

Get individual properties.
<i>(See SetProperty methos)</i>
<P>
<P>
<TABLE CELLPADDING="3" CELLPACING="0" CLASS="parameters">
<TR BGCOLOR="#EEEEFF" CLASS="small_title">
<TD COLSPAN=2>Keywords</TD>
</TR>
<TR BGCOLOR="white" CLASS="parameters">
<TD ALIGN="right" VALIGN="top" WIDTH="1%">
<A NAME="_Product::GetProperty_keyword_NAME"/>
<FONT CLASS="param_name">NAME</FONT><BR>
<FONT CLASS="param_attrib">
</FONT>
</TD>
<TD VALIGN="top">
.
</TD>
</TR>
<TR BGCOLOR="white" CLASS="parameters">
<TD ALIGN="right" VALIGN="top" WIDTH="1%">
<A NAME="_Product::GetProperty_keyword_DESCRIPTION"/>
<FONT CLASS="param_name">DESCRIPTION</FONT><BR>
<FONT CLASS="param_attrib">
</FONT>

```

```

</TD>
<TD VALIGN="top">
.
</TD>
</TR>
</TABLE>
<P>
<HR CLASS="divider">
<TABLE WIDTH="100%" BORDER="0"><TR><TD>
<H3 CLASS="routine"><A NAME="_Product::AddParent">
Product::AddParent
</A></H3>
</TD><TD ALIGN="right">

</TD></TR></TABLE>
<TABLE CELLPADDING="2" CELLSPACING="0" BORDER="0" WIDTH="100%"><TR><TD
ALIGN="left" VALIGN="top"><P CLASS="code_header">procedure Product::AddParent, <A
HREF=#_Product::AddParent_param_product>product</A></P></TD ></TR></TABLE>
<P>

```

Sets a parent dependency for the object and automatically sets a reference from parent to child.

```

<P>
<TABLE CELLPADDING="3" CELLSPACING="0" CLASS="parameters">
<TR BGCOLOR="#EEEEFF" CLASS="small_title">
<TD COLSPAN=2>Parameters</TD>
</TR>
<TR BGCOLOR="white" CLASS="parameters">
<TD ALIGN="right" VALIGN="top" WIDTH="1%">
<A NAME="_Product::AddParent_param_product"/>
<FONT CLASS="param_name">product</FONT><BR>
<FONT CLASS="param_attrib">
<NOBR>in, required</NOBR><BR>
<NOBR>Product</NOBR>
</FONT>
</TD>
<TD VALIGN="top">
Product object that
        represents a parent dependency.
</TD>
</TR>
</TABLE>
<P>
<P>
<HR CLASS="divider">
<TABLE WIDTH="100%" BORDER="0"><TR><TD>
<H3 CLASS="routine"><A NAME="_Product::AddChild">

```

```
Product::AddChild
</A></H3>
</TD><TD ALIGN="right">
```

```
</TD></TR></TABLE>
<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" WIDTH="100%"><TR><TD
ALIGN="left" VALIGN="top"><P CLASS="code_header">procedure Product::AddChild, <A
HREF=#_Product::AddChild_param_product>product</A></P></TD ></TR></TABLE>
<P>
```

Sets a child dependency for the object and set a reference from child to parent.

```
<P>
<TABLE CELLPADDING="3" CELLSPACING="0" CLASS="parameters">
<TR BGCOLOR="#EEEEFF" CLASS="small_title">
<TD COLSPAN=2>Parameters</TD>
</TR>
<TR BGCOLOR="white" CLASS="parameters">
<TD ALIGN="right" VALIGN="top" WIDTH="1%">
<A NAME="_Product::AddChild_param_product"/>
<FONT CLASS="param_name">product</FONT><BR>
<FONT CLASS="param_attrib">
<NOBR>in, required</NOBR><BR>
<NOBR>Product</NOBR>
</FONT>
</TD>
<TD VALIGN="top">
Product object that
        represents a child dependency.
```

```
</TD>
</TR>
</TABLE>
<P>
<P>
<HR CLASS="divider">
<TABLE WIDTH="100%" BORDER="0"><TR><TD>
<H3 CLASS="routine"><A NAME="_Product::GetParents">
Product::GetParents
</A></H3>
</TD><TD ALIGN="right">
```

```
</TD></TR></TABLE>
<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" WIDTH="100%"><TR><TD
ALIGN="left" VALIGN="top"><P CLASS="code_header">function
Product::GetParents()</P></TD></TR></TABLE>
<P>
```

Gets the parent dependencies of the product

```
<P>
<DL>
<DT CLASS="attribute">Returns
<DD CLASS="attribute_item">
  Linkedlist of Product.
</DD></DT>
</DL>
<P>
<P>
<P>
<HR CLASS="divider">
<TABLE WIDTH="100%" BORDER="0"><TR><TD>
<H3 CLASS="routine"><A NAME="_Product::GetChildren">
Product::GetChildren
</A></H3>
</TD><TD ALIGN="right">

</TD></TR></TABLE>
<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" WIDTH="100%"><TR><TD
ALIGN="left" VALIGN="top"><P CLASS="code_header">function
Product::GetChildren()</P></TD></TR></TABLE>
<P>
```

Gets the children referencies.

```
<P>
<DL>
<DT CLASS="attribute">Returns
<DD CLASS="attribute_item">
  Linkedlist of Products
</DD></DT>
</DL>
<P>
<P>
<P>
<HR CLASS="divider">
<TABLE WIDTH="100%" BORDER="0"><TR><TD>
<H3 CLASS="routine"><A NAME="_Product::MakeProduct">
Product::MakeProduct
</A></H3>
</TD><TD ALIGN="right">

</TD></TR></TABLE>
<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" WIDTH="100%"><TR><TD
ALIGN="left" VALIGN="top"><P CLASS="code_header">procedure
Product::MakeProduct</P></TD></TR></TABLE>
```

<P>

Executes the plugin associated with the Product to generate the output data.

<P>

<P>

<P>

<HR CLASS="divider">

<TABLE WIDTH="100%" BORDER="0"><TR><TD>

<H3 CLASS="routine">

Product::SetPlugin

</H3>

</TD><TD ALIGN="right">

</TD></TR></TABLE>

<TABLE CELLPADDING="2" CELLSPACING="0" BORDER="0" WIDTH="100%"><TR><TD ALIGN="left" VALIGN="top"><P CLASS="code_header">procedure Product::SetPlugin, plugin</P></TD ></TR></TABLE>

<P>

Sets the Plugin object on which is based the Product.

<P>

<TABLE CELLPADDING="3" CELLSPACING="0" CLASS="parameters">

<TR BGCOLOR="#EEEEFF" CLASS="small_title">

<TD COLSPAN=2>Parameters</TD>

</TR>

<TR BGCOLOR="white" CLASS="parameters">

<TD ALIGN="right" VALIGN="top" WIDTH="1%">

plugin

<NOBR>in, required</NOBR>

<NOBR>Plugin</NOBR>

</TD>

<TD VALIGN="top">

Plugin on which is
related.

</TD>

</TR>

</TABLE>

<P>

<P>

<HR CLASS="divider">

<TABLE WIDTH="100%" BORDER="0"><TR><TD>

<H3 CLASS="routine">

Product::SetPluginInfo

</H3>

</TD><TD ALIGN="right">

</TD></TR></TABLE>

<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" WIDTH="100%"><TR><TD ALIGN="left" VALIGN="top"><P CLASS="code_header">procedure Product::SetPluginInfo, plugin_info </P></TD></TR></TABLE><P>

Sets the PluginInfo object that stores the Plugin information.

<P>

<TABLE CELLPADDING="3" CELLSPACING="0" CLASS="parameters">

<TR BGCOLOR="#EEEEFF" CLASS="small_title">

<TD COLSPAN=2>Parameters</TD>

</TR>

<TR BGCOLOR="white" CLASS="parameters">

<TD ALIGN="right" VALIGN="top" WIDTH="1%">

plugin_info

<NOBR>in, required</NOBR>

<NOBR>PluginInfo</NOBR>

</TD>

<TD VALIGN="top">

PluginInfo object.

</TD>

</TR>

</TABLE>

<P>

<P>

<HR CLASS="divider">

<TABLE WIDTH="100%" BORDER="0"><TR><TD>

<H3 CLASS="routine">

Product::AssocPluginInfo2Plugin

</H3>

</TD><TD ALIGN="right">

</TD></TR></TABLE>

<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" WIDTH="100%"><TR><TD

ALIGN="left" VALIGN="top"><P CLASS="code_header">procedure

Product::AssocPluginInfo2Plugin</P></TD></TR></TABLE>

<P>

Creates an association from PluginInfo to Plugin object of the Product.

```

<P>
<DL>
<DT CLASS="attribute">Pre-condition
<DD CLASS="attribute_item">
  Product object must has references to a valid Plugin and
  PluginInfo objects.
</DD></DT>
</DL>
<P>
<P>
<P>
<HR CLASS="divider">
<TABLE WIDTH="100%" BORDER="0"><TR><TD>
<H3 CLASS="routine"><A NAME="_Product::GetPlugin">
Product::GetPlugin
</A></H3>
</TD><TD ALIGN="right">

</TD></TR></TABLE>
<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" WIDTH="100%"><TR><TD
ALIGN="left" VALIGN="top"><P CLASS="code_header">function
Product::GetPlugin()</P></TD></TR></TABLE>
<P>

```

Gets the Plugin object on which the Product is based.

```

<P>
<DL>
<DT CLASS="attribute">Returns
<DD CLASS="attribute_item">
  Plugin
</DD></DT>
</DL>
<P>
<P>
<P>
<HR CLASS="divider">
<TABLE WIDTH="100%" BORDER="0"><TR><TD>
<H3 CLASS="routine"><A NAME="_Product::GetPluginInfo">
Product::GetPluginInfo
</A></H3>
</TD><TD ALIGN="right">

</TD></TR></TABLE>
<TABLE CELLSPACING="2" CELLPADDING="0" BORDER="0" WIDTH="100%"><TR><TD
ALIGN="left" VALIGN="top"><P CLASS="code_header">function
Product::GetPluginInfo()</P></TD></TR></TABLE>
<P>

```

Gets the PluginInfo object with the information of the Plugin on which the Product is based.

<P>

<DL>

<DT CLASS="attribute">Returns

<DD CLASS="attribute_item">

PluginInfo

</DD></DT>

</DL>

<P>

<P>

<P>

<HR CLASS="divider">

Produced by IDLdoc 1.5 on Tue Mar 15 15:50:42 2005

</BODY></HTML>

File Attachments

1) [product__define.html](#), downloaded 144 times
