

Modifications of ESL script execution within Active Picture

Modifications of ESL script execution within active picture

Description of modifications of ESL script behaviour within active picture.

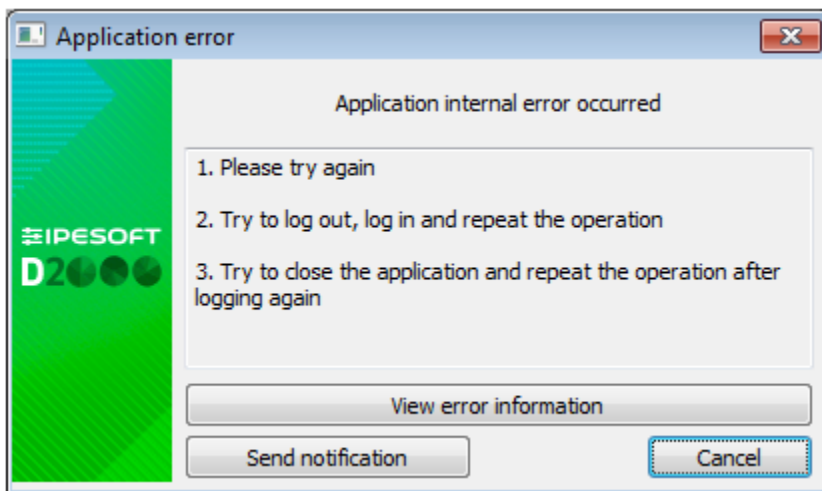
END action or terminating the script execution

Script within active picture can not be terminated. After the execution of the initialization part activated immediately after opening the picture (before its displaying), or after execution [END](#) action, the script is waiting for user-generated events. Incoming events are handled by using [ENTRY](#) or by object value change handler ([ON CHANGE](#)).

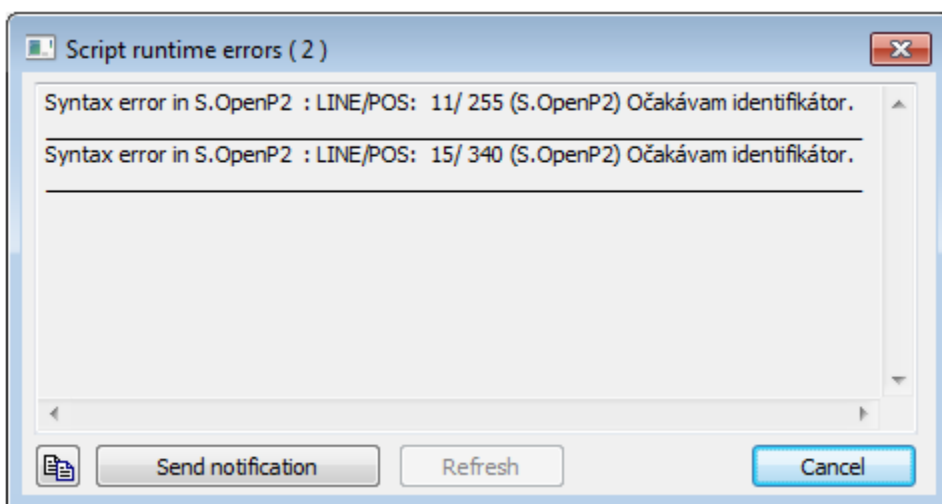
Error state handling

When error status occurs, the ESL standard behaviour is extended with displaying the message in the process [D2000 HI](#). As the script cannot be terminated, the current event processing is aborted and the script is waiting for an event.

Information about that situation appears in the dialog box. It offers several options such as solving an error, displaying the error messages or prepare message with error warnings for sending to email address defined in the application parameters (parameter [MailingAddress](#)).



After clicking "View error information", dialog box with the most recent error messages opens. Message count is displayed in header. The messages are separated by line and their structure may vary over time.



A user may copy this message to a clipboard by clicking the button  and send to an administrator. Or he may send it on defined email address (mentioned above). If additional error messages occur in dialog box, the **Refresh** button become available. It allows you update the displayed contents.

The button **OK** confirms the error and closes the dialog box.

Assignment to indexed variable

Indexed variable declared in active picture is bound (its declaration) with an object of [Structured variable](#) type. It determines a row (in term of value on the row) of the structure that is continuously updated (alteration method). Therefore a value change of such a local variable is executed in the special way:

`_INDEX` - indexed variable for the structure variable `SV.Struct`. Current value is 3.
Assignment:

```
_INDEX := 2
```

closes (terminates updating) the row `SV.Struct[3]` and opens the row `SV.Struct[2]`.

Because closing and opening the row may take some time in dependence on the system load, the assignment is implemented in the following steps:

```
_INDEXPREV    ; current value of indexed variable (in the example it is 3)
```

```
_INDEXNEW     ; new value of indexed variable (in the example it is 2)
```

```
%HI_EnablePicture(@FALSE)

_INDEX := _INDEXNEW           ; with setting the Transient state
IF %HI_ChangeRow(SV.Struct\HBJ, _INDEXPREV, _INDEXNEW) = _ERR_NO_ERROR THEN
  _INDEX := _INDEXNEW         ; without Transient state
ELSE ; unsuccessful index change
  _INDEX := invalid value     ; without Transient state
ENDIF

%HI_EnablePicture(@TRUE)      ; only if the picture is enabled before assignment
```

The sequence of actions mentioned above is executed internally within the assignment.