STEP 0 :

- Create an instance of FESTIVAL_SELECTION_GUI object (oFestivalSelectionGUI).
- The FESTIVAL_SELECTION_GUI object holds :
  - **1 FESTIVAL_QUERY object :**
    - It is used to query the catalogs from user defined query parameters (_oQuery).
  - **3 FESTIVAL_CONTAINER result containers, one per probe :**
    - (_oSOHOResultContainer, _oSTEREOAResultContainer, _oSTEREOBResultContainer)
    - They are used to store the FESTIVAL_SEARCH_RESULT objects produced by the FESTIVAL_QUERY object.
    - A FESTIVAL_SEARCH_RESULT is identified by a probe name, an instrument name and an image index. Image index corresponds to the nth image found for that instrument.
    - Typical FESTIVAL_CONTAINER contents may be a follows :
      - Left column gives the FESTIVAL_SEARCH_RESULT absolute position.
      - Right column shows the FESTIVAL_SEARCH_RESULT objects with identification parameters :

0	SOHO, C2, 2
1	SOHO, C2, 1
2	SOHO, C2, 0
3	SOHO, C1, 3
4	SOHO, C1, 2
5	SOHO, C1, 1
6	SOHO, C1, 0
7	SOHO, EIT, 3
8	SOHO, EIT, 1
9	SOHO, EIT, 0

STEP 1 :

- User runs a search with the SEARCH button from the SOHO tab :
  - Get the user query parameters :
    oFestivalSelectionGUI->GETPROPERTY, SOHO_SEARCH_CRITERIA = sohoSearchCriteria
  - Passes sohoSearchCriteria to the query object :
    result = _oQuery->SEARCH, CRITERIA_STRUCTURE = sohoSearchCriteria
    result is an array of FESTIVAL_SEARCH_RESULT_OBJECT

STEP 2 :

- If results were found, fill-in the SOHO FESTIVAL_CONTAINER object with the results :
  _oSOHOResultContainer->ADD, result
  Note:
  The FESTIVAL_CONTAINER::ADD method sets the FESTIVAL_SEARCH_RESULT:_absoluteIndexProperty

STEP 3 :

- Reads the contents of _oSOHOResultContainer :
  - Look for "SOHO_EIT_0" search result and get its absolute image position l0.
    l0 = oSOHOResultContainer->GETIMAGEPOSITION(INSTRUMENT="EIT", IMAGE_INDEX=0)
  - Look for "SOHO_C1_0" search result and get its absolute image position l1.
    l1= oSOHOResultContainer->GETIMAGEPOSITION(INSTRUMENT="C1", IMAGE_INDEX=0)

- Dishighlight any highlighted lines in the SOHO tab table widget.
  oFestivalSelectionGUI->CLEAR, /SOHO
- Highlight [l0, l1] (if not equal -1) in the SOHO tab table widget :
  oFestivalSelectionGUI->HIGHLIGHT, /SOHO, LINES = [l0, l1]
- Set the oFestivalSelectionGUI SELECTION property :
  oFestivalSelectionGUI->SETPROPERTY, SOHO_SELECTION = [l0, l1]

STEP 4 :

- Create an instance of a FESTIVAL_VISUALISATION_GUI (oFestivalVisualisationGUI), and pass it a reference to the FESTIVAL_SELECTION_GUI object.
- Sets the respective FESTIVAL_SELECTION_GUI object property :
  oFestivalSelectionGUI->SETPROPERTY, SOHO_VISUALISATION_GUI = oSOHOFestivalVisualisationGUI
- Sets the respective FESTIVAL_VISUALISATION_GUI property :
  oFestivalVisualisationGUI->SETPROPERTY, DATA_SELECTION_GUI = oFestivalSelectionGUI
- Reads the contents of _oSOHOResultContainer :
  oFestivalSelectionGUI->GETPROPERTY, SOHO_RESULT_CONTAINER = oSohoResultContainer
- Get the selected images in the SOHO tabke tab widget :
  oFestivalSelectionGUI->GETPROPERTY, SELECTION = selection
  oFestivalSearchResult = _oSOHOResultContainer->GETNTHFESTIVALSEARCHOBJECT(ABSOLUTE_INDEX = selection)
- Loop through the oFestivalSearchResult array, and create the FESTIVAL_IMAGE objects :**oFestivalEITImage = OBJ_NEW("FESTIVAL_EIT", festivalSearchResult[i])**

  - This image is constructed using all the user defined parameters :
    oFestivalSelectionGUI->GETPROPERTY, MASKING=oMasking, DISPLAY_ORDER=oDisplayOrder, PROJ_COORD=oProjCoord, COLOUR=oColour, DATA_CALIB=oDataCalib|[FILTER=oFilter, ENHANCEMENT=oEnhancement
  - oMasking, oDisplayOrder, oProjCoord, oColour, oDataCalib, oFilter and oEnhancement are the user parameters objects. From them, it is possible to get all of the user settings.
    oFestivalEITImage->BUILDIMAGE(PARAMS = userParams) **TBD**

- Add the FESTIVAL_IMAGE images to the FESTIVAL_VISUALISATION_GUI visualization stack (FESTIVAL_VISUALIZATION_STACK object). The images are displayed in FESTIVAL_VISUALISATION_GUI object.
- Initially, no image is highlighted in the SOHO visualization GUI. The data visualization GUI

displays the first found image per instrument.

STEP 5 :

- User clicks the EIT image that becomes “highlighted”.
- Update the PREVIOUS/NEXT arrows, depending on :
  - The clicked image (or instrument).
  - The number of available images for that instrument. If that number is 1, the previous/next arrows remain greyed.
- Get the FESTIVAL_IMAGE object reference from the click. This object can be a FESTIVAL_EIT and FESTIVAL_LASCO, a FESTIVAL_C1, a FESTIVAL_C2, a FESTIVAL_C3 object because user clicked into a SOHO FESTIVAL_VISUALISATION_GUI object reference (oFestivalEITImage).

STEP 6 :

- Get probe name, instrument name and index of the currently displayed EIT image. **oFestivalEITImage->GETPROPERTY, PROBE=probe, INSTRUMENT=instrument, IMAGE_INDEX=imageIndex**
- User presses the next button.
- Increment imageIndex -> indexNextImage

STEP 7 :

- Look in the history stack to check if the required image is available OR a similar image is available. **oFestivalDataSelectionGUI->GETPROPERTY, HISTORY_STACK = oHistoryStack**
- Loop into the history stack and search for the FESTIVAL_IMAGE objects having indexNextImage as an index. Several images may be found.
- **Result = oHistoryStack->Search(PROBE=”SOHO”, INSTRUMENT=”EIT”, IMAGE_INDEX=indexNextImage, FILTER=Targetfilter, PROJECTION_NAME=TargetProjectionName, PROJECTION_CENTER=TargetProjectionCenter, CALIBRATION=TargetCalibration, COORDINATE_SYSTEM=TargetCoorSystem, SCIENTIFIC_UNFILTERED=scientificUnfiltered, SCIENTIFIC_FILTERED_BADPROJ=scientificFilteredBadProj**
- Result may be one of the following :

  1/ A valid FESTIVAL_IMAGE oFestivalEITNextImage if the exact image was found.

  In this case, oFestivalEITImage is pulled out the visualization stack and pushed into the history stack (if not already in). oFestivalEITNextImage is pulled out the history stack and pushed into the visualization stack.

  2/ A null object if no object was found. Check scientificFilteredBadProj buffer.

  2.1/ scientificFilteredBadProj is NOT null.

  If a valid buffer is found, use this buffer to build a new FESTIVAL_IMAGE object oFestivalEITNextImage. Pull oFestivalEITImage from the visualization stack, and push it into the history stack (if not already inside). Push oFestivalEITNextImage to the visualization stack.

  2.2/ scientificFilteredBadProj is null.

  Check scientificUnfiltered buffer.

2.2.1/ scientificUnfiltered is NOT null.

If a valid buffer is found, use this buffer to build a new FESTIVAL_IMAGE object oFestivalEITNextImage. Pull oFestivalEITImage from the visualization stack, and push it into the history stack (if not already inside). Push oFestivalEITNextImage to the visualization stack.

2.2.2/ scientificUnfiltered is null.

In this case, build a new FESTIVAL_IMAGE oFestivalEITNextImage object from scratch using the user defined parameters. Pull oFestivalEITImage from the visualization stack, and push it into the history stack (if not already inside). Push oFestivalEITNextImage to the visualization stack.

STEP 8 :

- Update the data selection GUI.
- Get the visualization stack contents.

  result = oVisualisationStack->GET(/ALL, OBJ_CLASS = "FESTIVAL_IMAGE")
- Loop on the FESTIVAL_IMAGE result array and get the respective absolute positions for each image :

  result[i]->GETPROPERTY, ABSOLUTE_INDEX = absoluteIndex

  l[i] = absoluteIndex
- Dishighlight any highlighted lines in the SOHO tab table widget.
  oFestivalSelectionGUI->CLEAR, /SOHO
- Highlight l lines (if valid) in the SOHO tab table widget :
  oFestivalSelectionGUI->HIGHLIGHT, /SOHO, LINES = l

STEP 9 :

- User clicks a line (USER_LINE) in a probe tab table widget. Get the corresponding FESTIVAL_SEARCH_RESULT object :

  resultClicked = _oSOHOResultContainer->
  GetNthFestivalSearchObject(ABSOLUTE_INDEX = USER_LINE)
- Get instrument associated to that line :

  resultClicked->GETPROPERTY, INSTRUMENT = instrumentClicked
- Get the current selection in the SOHO tab table widget :

  oFestivalSelectionGUI->GETPROPERTY, SOHO_SELECTION = sohoSelection
- Get the associated FESTIVAL_SEARCH_RESULT objects :

  resultDisplayed = _oSOHOResultContainer->
  GetNthFestivalSearchObject(ABSOLUTE_INDEX = sohoSelection)
- Get the instruments associated to the sohoSelection lines :
- **result[i]->GETPROPERTY, INSTRUMENT = instrumentDisplayed**
- **instrumentsDisplayed = [instrumentsDisplayed, instrumentDisplayed]**

- Perform a WHERE operation :

  v**ec = WHERE(instrumentsDisplayed EQ instrumentClicked, count)**
- Replaces with the good line :

  sohoSelection[vec] = USER_LINE