More than abstract.
Use cases for an integrated 3D visualization

Boris Schuster, 3S Smart Software Solutions

CODESYS Users Conference 2016
# Agenda

<table>
<thead>
<tr>
<th></th>
<th>Term definition: Integrated 3D visualization</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Develop a 3D visualization in CODESYS</td>
</tr>
<tr>
<td>3</td>
<td>Typical use cases</td>
</tr>
<tr>
<td>4</td>
<td>Sales model</td>
</tr>
<tr>
<td>5</td>
<td>Conclusion / outlook</td>
</tr>
</tbody>
</table>
What we understand by it …

- **Integrated** visualization: Visualization displayed in the CODESYS editor

- **3D** or **3-D** is a widely used abbreviation for **three-dimensional** or **three dimensions** and a synonym for the spatial representation of objects. (Wikipedia)

- **CODESYS Store:**
  The CODESYS _________ enables users to
  - create 3D models in CODESYS and to directly link them to CODESYS applications.
  - Thus, the most different elements ranging from machine parts up to complete production lines can be depicted and animated as 3D scenes within the CODESYS Development System.
Term definition: Integrated 3D visualization

... how we call the product ...

CODESYS Depictor

The CODESYS Depictor allows creating and displaying 3D models and linking their geometrical relations (i.e. axis angles) to CODESYS applications. It is possible to display and animate machine parts (e.g. robots) or entire production lines in CODESYS.

...and what it literally means:

**depict**

[dih-pikt]

verb (used with object)
1. to represent by or as if by painting; portray; delineate.
2. to represent or characterize in words; describe.
Use cases for an integrated 3D visualization

Agenda

1. Term definition: Integrated 3D visualization
2. Develop a 3D visualization in CODESYS
3. Typical use cases
4. Sales model
5. Conclusion / outlook
Geometrical tree

- **Tree**
  - From poses and elements linked with hierarchic dependencies

- **Element**
  - Description of a geometric object with graphical information
  - Child of a pose

- **Pose:**
  - Definition of a coordinate system in relation to the original pose
  - Creation top-level or as child of another pose
Poses (coordinate systems)

- **Pose**: Definition of the relative position in relation to the original pose by means of mathematical **transformations**

- **Available transformations**:
  - Translation (X, Y, Z)
  - Rotation (X, Y, Z)
  - Scaling (X, Y, Z)
  - Combined translation and rotation

- Linkage of transformation **property values** with IEC 61131-3 **application variables** (for dynamic online display)
How to create elements

- **By referencing other Depictor objects:**
  - Set of basic elements (cylinder, sphere, …) in the „DepictorBase“ library
  - Definition as template in the POU pool of the project or of libraries
  - Instances of Depictor templates are inserted as elements and assigned to their interface variables.

- **By embedding or referencing of external geometrical data. Supported geometric file formats:**
  - *.obj (Wavefront)
  - *.dae (COLLADA)
  - *.3ds (3ds Max)

Full support only for *.obj files (e.g. no full support of COLLADA data specifications)
Tracking shots: How to control the camera position

Via implicit variables from the application by means of:

- Direct position definition
- Use of positions pre-defined as array in the editor
- Interpolation via the POU „DepictorBase.InterpolateCameraPosition“
Summary

- Integration of complex 3D models (.obj, .dae, 3ds) including basic model library
- Modeling of geometric dependencies and movements in a simple tree structure
- Object orientation: Link between Depictor template and FB via interface definition
Summary

- Use of IEC 61131-3 variables within the definition of Depictor models
  - Direct connection of the depictor-scene to motion tasks in online operation
- Re-usage of Depictor models within other depictor objects via instance creation
- Application-guided tracking shots
Bitte englischen Screenshot einsetzen.

Birgit Hess; 8.2.2016
<table>
<thead>
<tr>
<th></th>
<th>Term definition: Integrated 3D visualization</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Develop a 3D visualization in CODESYS</td>
</tr>
<tr>
<td>3</td>
<td>Typical use cases</td>
</tr>
<tr>
<td>4</td>
<td>Sales model</td>
</tr>
<tr>
<td>5</td>
<td>Conclusion / outlook</td>
</tr>
</tbody>
</table>
Use cases for an integrated 3D visualization

Typical use cases

**Basic evaluation in operational mode:**
- Basic movement
- Collision detection

**Appealing demonstration of a plant/machine in 3D:**
- Presentations
- Explanations

**Application independent machine concepts for:**
- Development
- Visual diagnosis
- Training
Basic evaluations, presentation

- First visual impression of the working machine
- Possibility to model complete machine architectures to get a glimpse of the work together of the different machine parts
- Conceptual visualization of machines for marketing and sales purposes
Typical use cases

Concept / presentation / training / virtual commissioning

- Virtual machines for testing during application development
- Virtual machines for training
- Ability to adjust specific movements (of i.e. robot arms) before having the real hardware
- First check to avoid collisions
Use cases for an integrated 3D visualization

Typical use cases

Integrated diagnosis

- Appealing, modern display of machines and machine parts
- Visual diagnosis, e.g. to detect defective machine components
- Visualization of inaccessible, hidden machine parts
1. Term definition: Integrated 3D visualization
2. Develop a 3D visualization in CODESYS
3. Typical use cases
4. Sales model
5. Conclusion / outlook
- Free viewing of depictor scenes
- Workstation license necessary to **create** Depictor objects (storage on CODESYS Security Key)
Howto get

Use cases for an integrated 3D visualization

Sales model
Use cases for an integrated 3D visualization

Agenda

1. Term definition: Integrated 3D visualization
2. Develop a 3D visualization in CODESYS
3. Typical use cases
4. Sales model
5. Conclusion / outlook
- Using the CODESYS Depictor is fun!
- The sample applications really make sense.
- But there is more to it!
  - Currently in the making
    - Integration into visualization
    - Interaction with CODESYS Application Composer: Composer generates 3D machine depiction

- Background information
  - Development based on the specifications of the OPAK project
    - "Offene Engineering-Plattform für autonome, mechatronische Automatisierungskomponenten in funktionsorientierter Architektur“ (Open engineering platform for autonomous mechatronic automation components in a function-oriented architecture)
      - [http://www.opak-projekt.de/](http://www.opak-projekt.de/)
Thank you for your attention.
Korrekturen Layout // SBa // 18.01.2016
Sprachcheck // BH // 18.01.2016
Korrekturen / Screenshots English: RW 01.03.2016