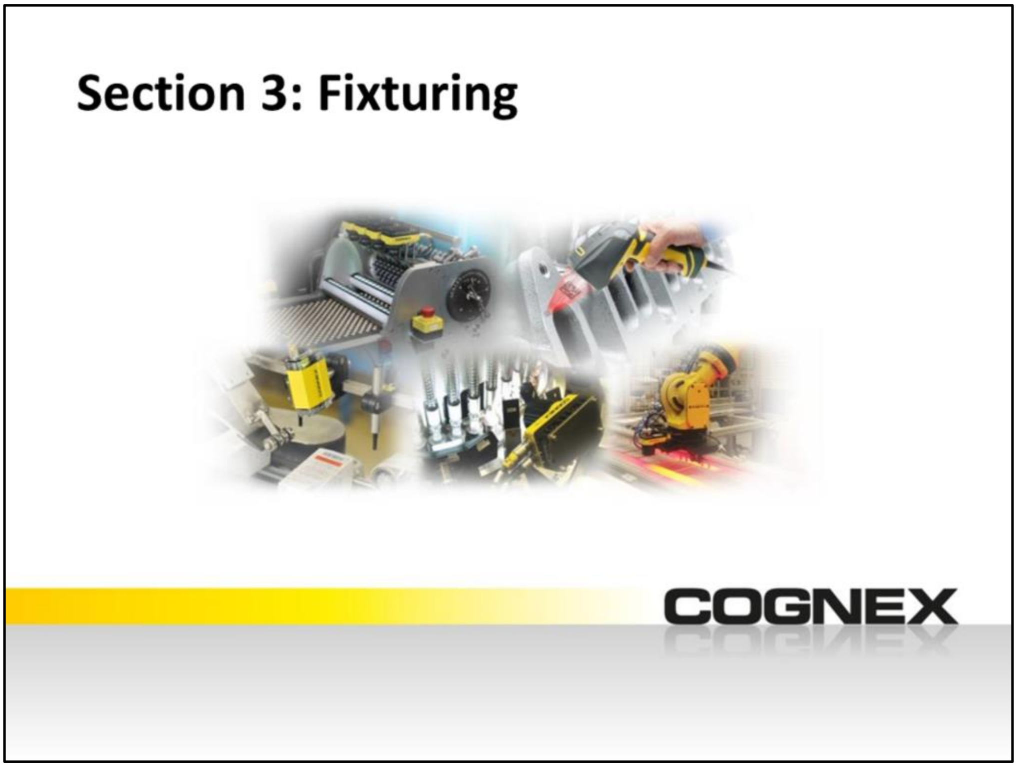
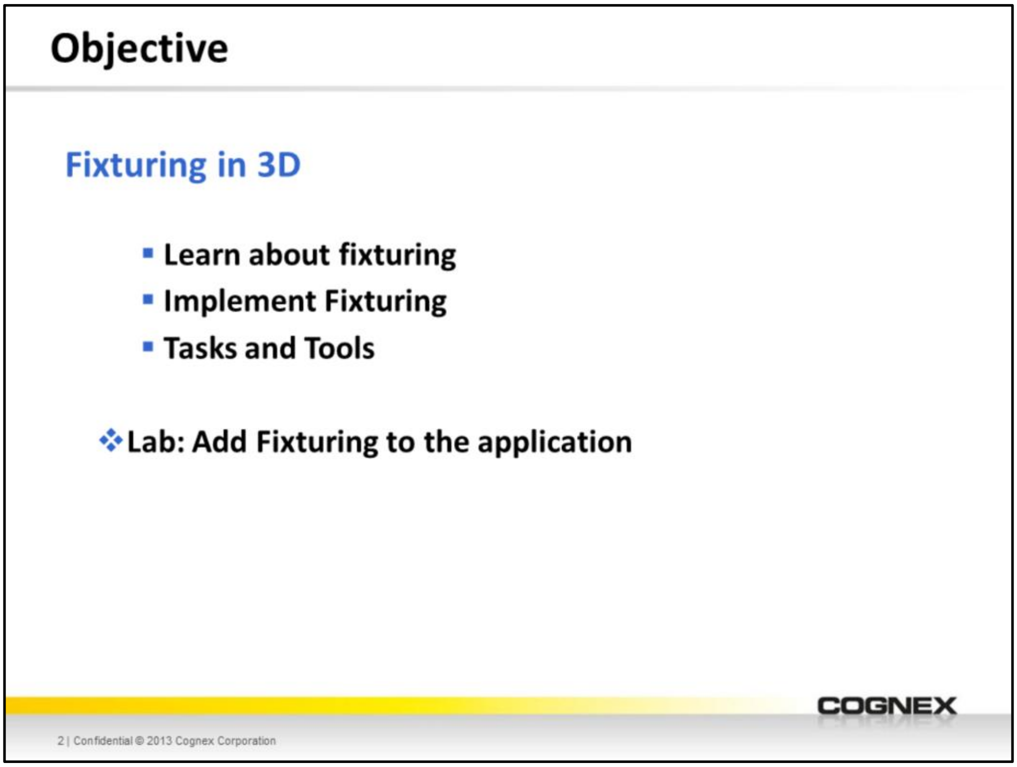


Section 3: Fixturing



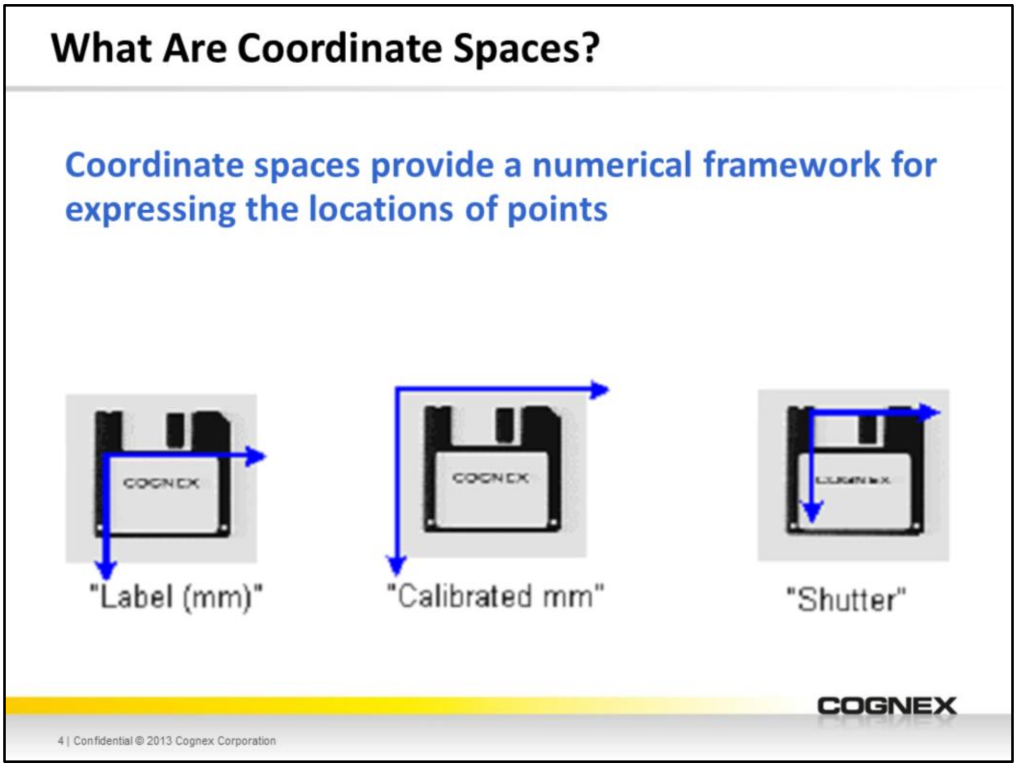
Objective

Fixturing in 3D

- Learn about fixturing
- Implement Fixturing
- Tasks and Tools

❖Lab: Add Fixturing to the application

Fixturing Your Part



What Are Coordinate Spaces?

Coordinate spaces provide a numerical framework for expressing the locations of points

"Label (mm)"

"Calibrated mm"

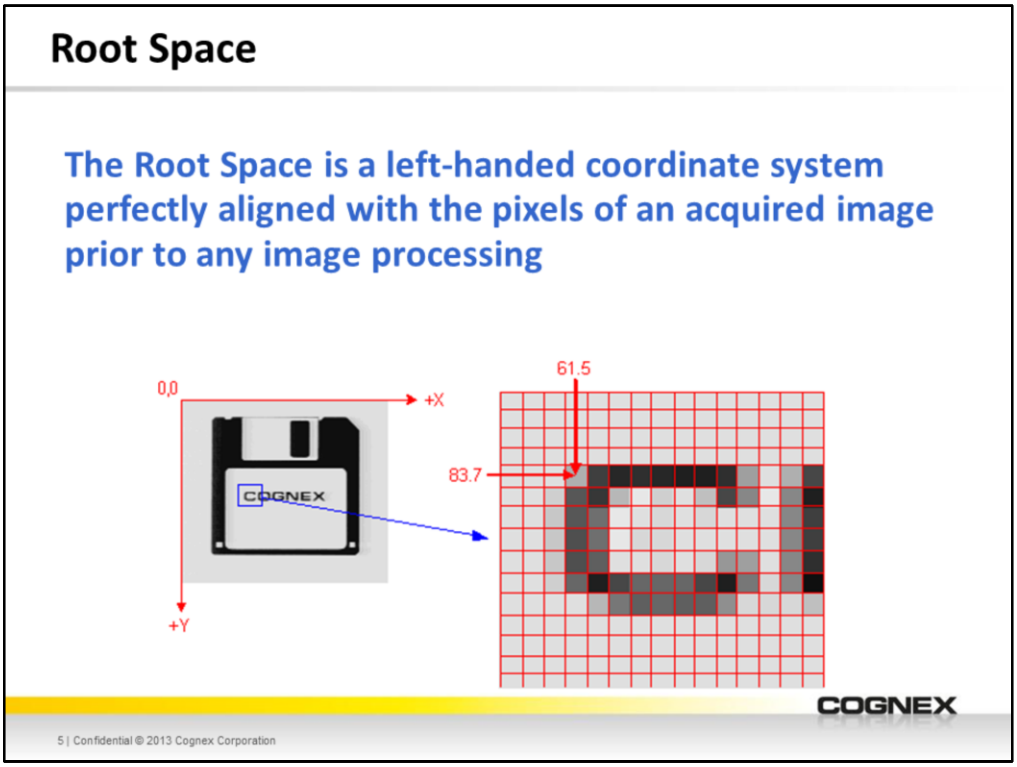
"Shutter"

COGNEX

4 | Confidential © 2013 Cognex Corporation

Coordinate spaces are a way for the account for fixturing and calibration changes to the pixel image.

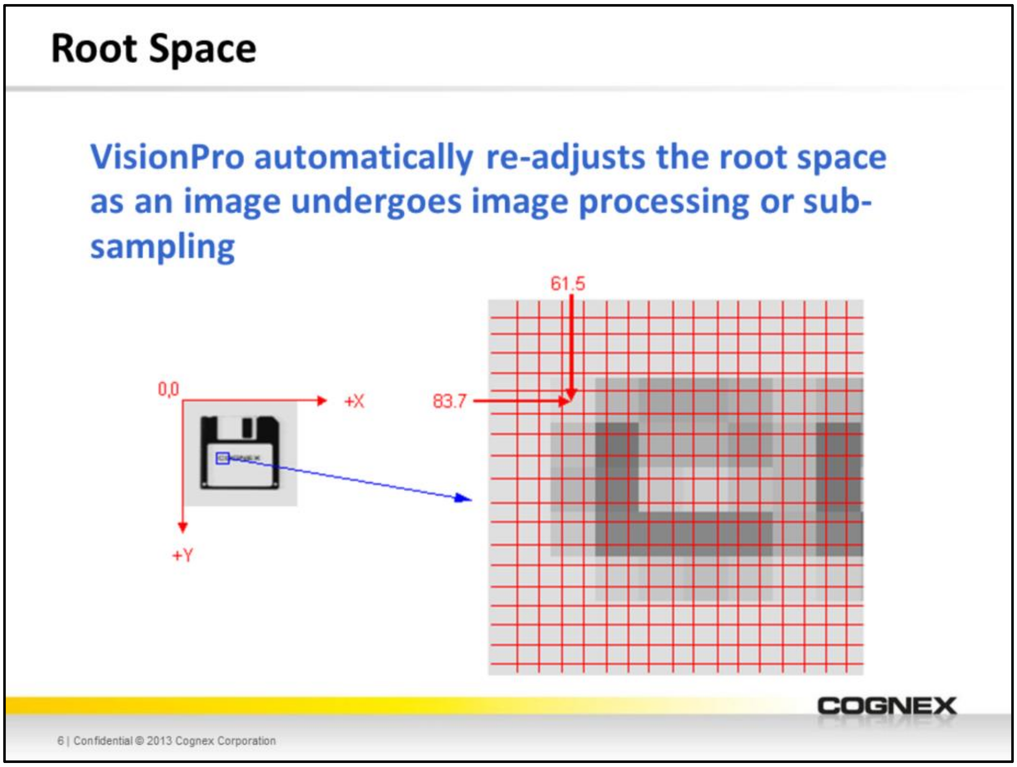
It allows for subsequent tools to move with the image or the ability to use real world measurements. Since the DS1000 series is already using calibrated coordinates (built into the range image), the calibration section is not needed.



Root Space

The Root Space is a left-handed coordinate system perfectly aligned with the pixels of an acquired image prior to any image processing

Think of the Root space as the pixel image as it is acquired by the camera.



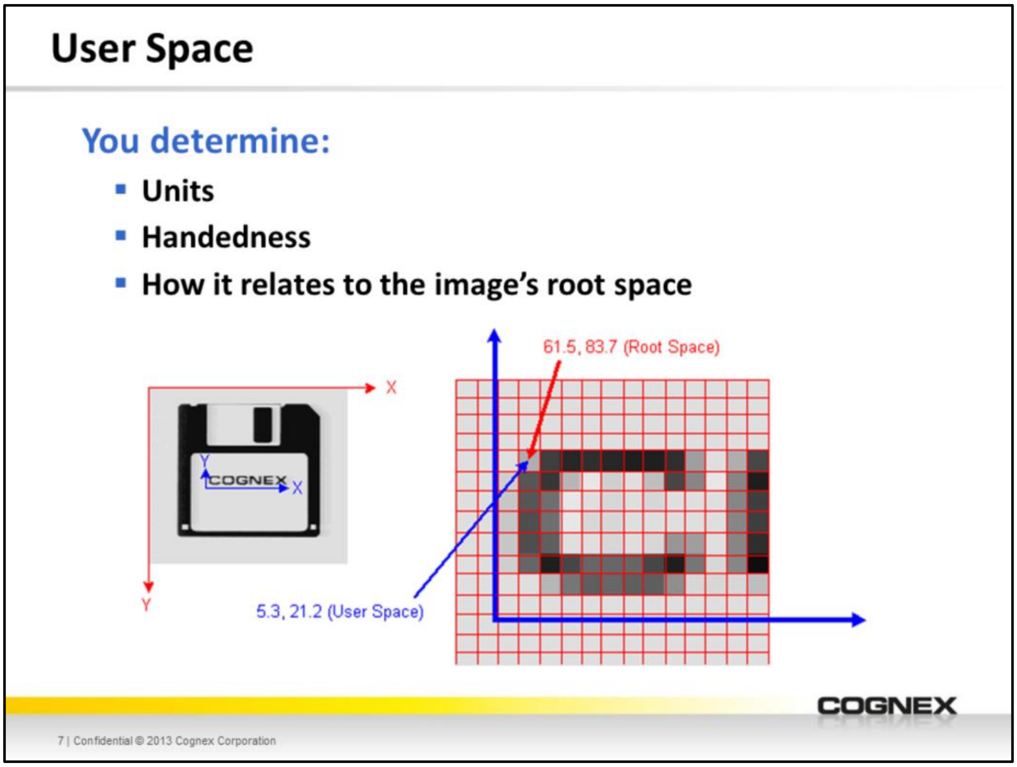
Root Space

VisionPro automatically re-adjusts the root space as an image undergoes image processing or sub-sampling

6 | Confidential © 2013 Cognex Corporation

The image has been subsampled; automatically adjusted the root so that image features (such as the "C" in "COGNEX") retained the same locations

The image now has fewer pixels; note that the root grid lines no longer correspond to the pixel boundaries.



User Space

You determine:

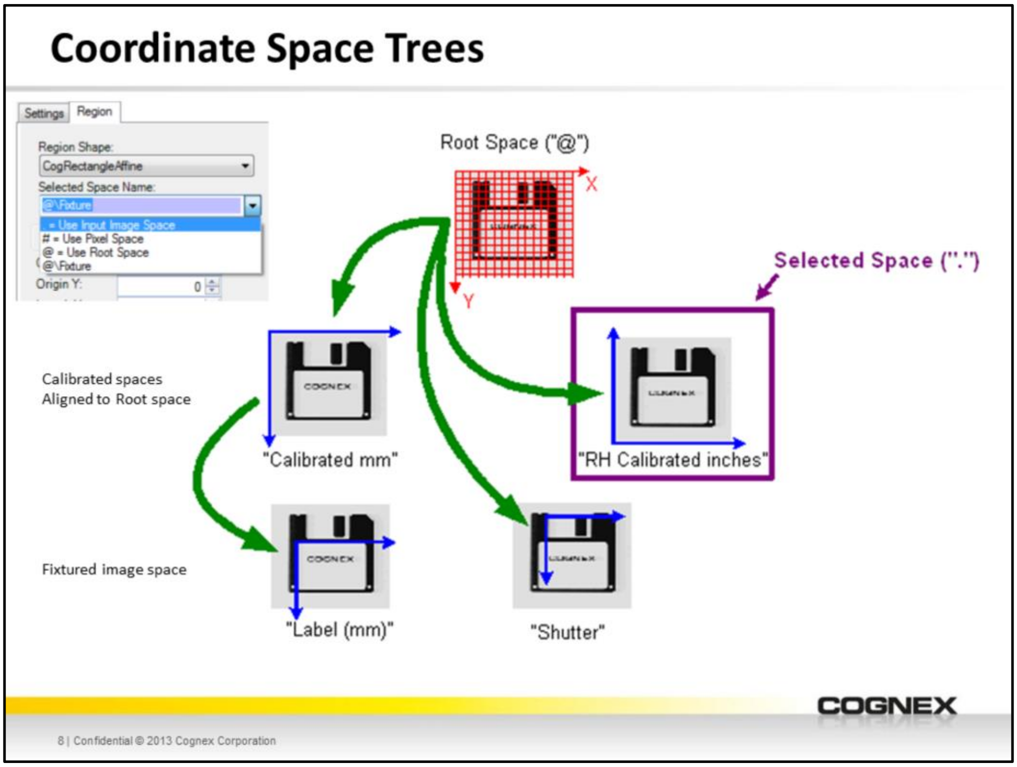
- Units
- Handedness
- How it relates to the image's root space

COGNEX

7 | Confidential © 2013 Cognex Corporation

VisionPro lets you define any number of additional coordinate systems

Typically, user spaces are used to create and manipulate calibrated spaces and fixtures



Coordinate Space Trees

Settings | Region

Region Shape: CogRectangleAffine

Selected Space Name: @\Fixture

. = Use Input Image Space
= Use Pixel Space
@ = Use Root Space
@\Fixture

Origin Y: 0

Root Space ("@")

Selected Space (".")

Calibrated spaces
Aligned to Root space

Fixtured image space

"Calibrated mm"

"RH Calibrated inches"

"Label (mm)"

"Shutter"

8 | Confidential © 2013 Cognex Corporation

Coordinate space trees contain

- An image's root space
- All user spaces you created
- How all the spaces are related to each other
  - a.k.a. Transformation

At all times, one space within the tree is the *Selected Space* for the image

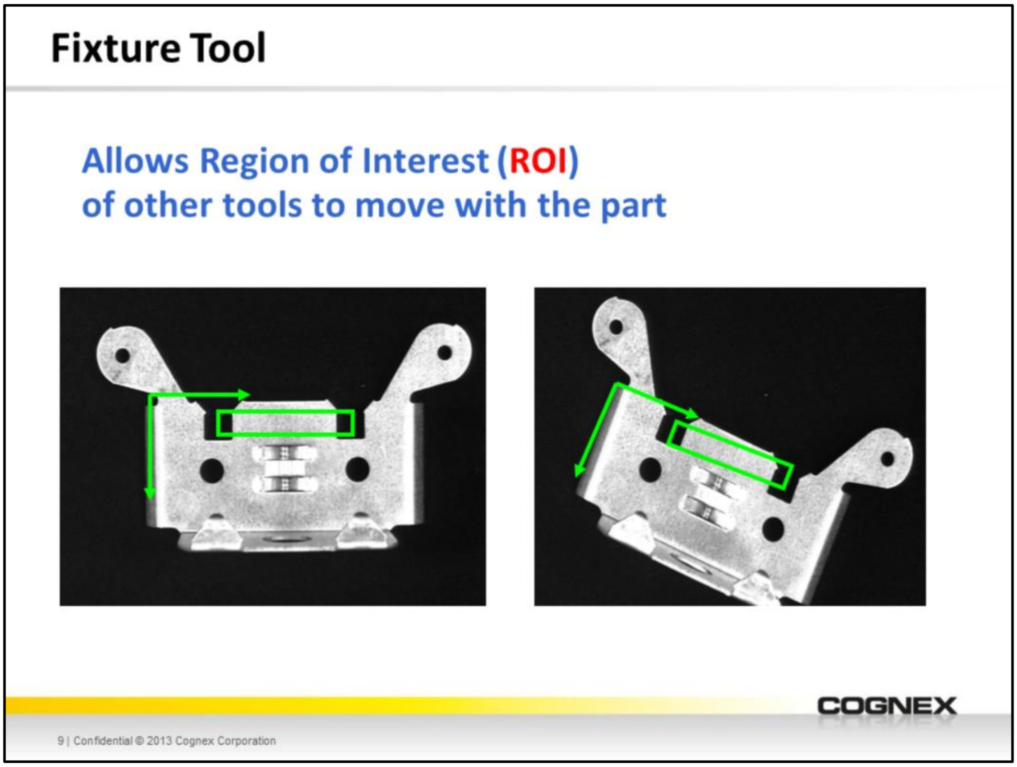
The coordinate system in which all VisionPro tools that operate on an image

- Return results
- Interpret input data
  - i.e. regions of interest

Creating a new image through some transformation adds a new coordinate space to the coordinate space tree

- *And automatically selects the space as the new image's selected space name*

Allows you to automatically map coordinates from a processed image back to the original image or vice-versa



Fixture Tool

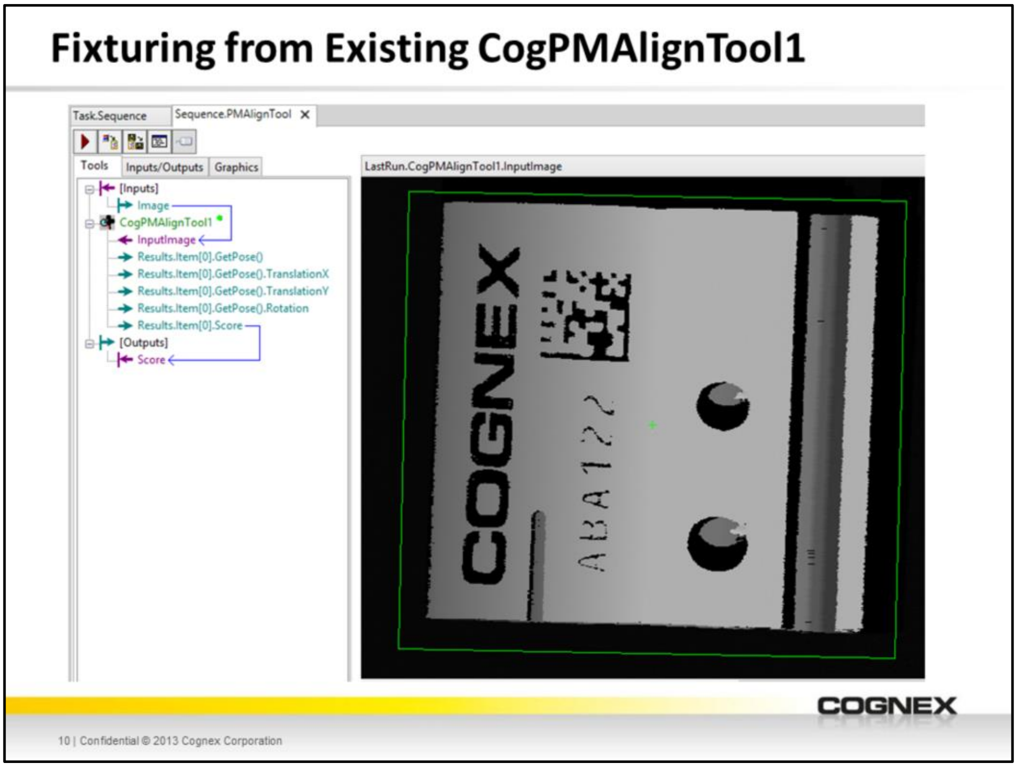
Allows Region of Interest (ROI)
of other tools to move with the part

COGNEX

9 | Confidential © 2013 Cognex Corporation

The Fixture Tool is used to create a fixture coordinate system when you already have a coordinate transform calculated

In our example, we'll find our part using PMAlign; it produces a transform in its results



Fixturing from Existing CogPMAlignTool1

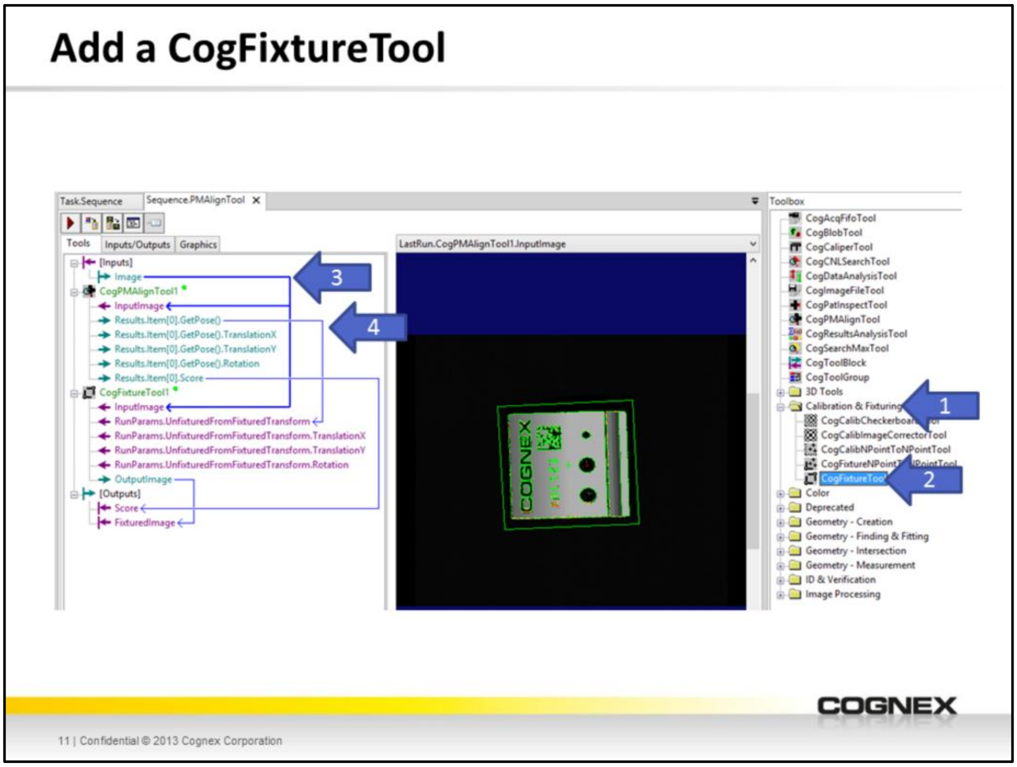
```
[Inputs]
  Image
CogPMAlignTool1
  InputImage
  Results.Item[0].GetPose()
  Results.Item[0].GetPose().TranslationX
  Results.Item[0].GetPose().TranslationY
  Results.Item[0].GetPose().Rotation
  Results.Item[0].Score
[Outputs]
  Score
```

COGNEX

10 | Confidential © 2013 Cognex Corporation

The first part of creating a fixture is to identify a portion of the part to be able to get translation and rotation information.

Use the existing CogPMAlignTool1 to perform the fixturing step.

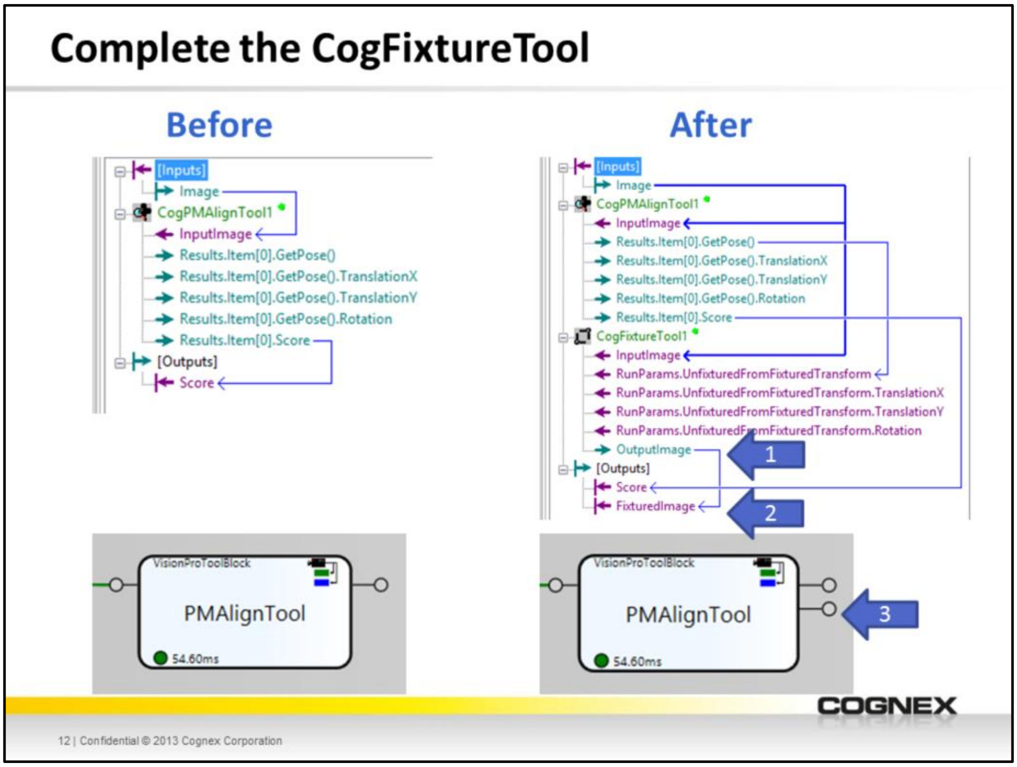


Add a CogFixtureTool

11 | Confidential © 2013 Cognex Corporation

To implement a CogFixtureTool:

1) Expand the Calibration & Fituring category.
2) Select the CogFixtureTool and drag it under the CogPMAlignTool in the block’s tool listing.
3) Drag the Image to the InputImage input terminal of the CogFixtureTool.
4) Drag the Results.Item[0].Pose() to the RunParams.UnfixturedFromFixturedTransform input terminal of the CogFixtureTool.



Complete the CogFixtureTool

Before

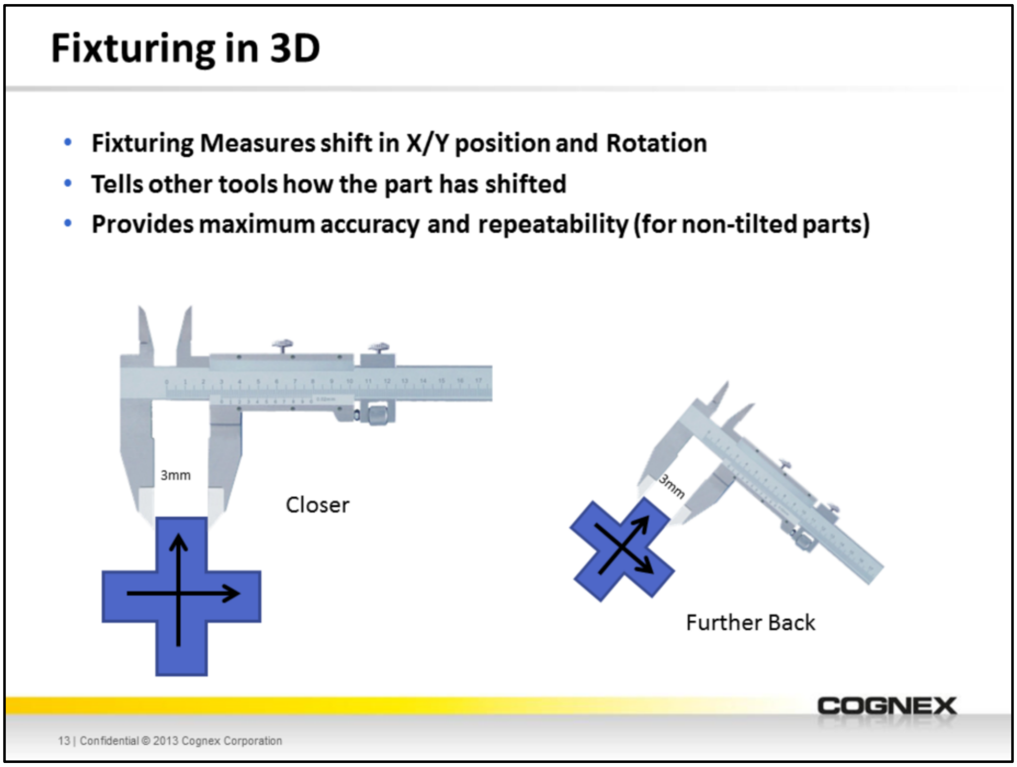
After

[Inputs]
Image
CogPMAlignTool1
InputImage
Results.Item[0].GetPose()
Results.Item[0].GetPose().TranslationX
Results.Item[0].GetPose().TranslationY
Results.Item[0].GetPose().Rotation
Results.Item[0].Score
[Outputs]
Score
[Inputs]
Image
CogPMAlignTool1
InputImage
Results.Item[0].GetPose()
Results.Item[0].GetPose().TranslationX
Results.Item[0].GetPose().TranslationY
Results.Item[0].GetPose().Rotation
Results.Item[0].Score
CogFixtureTool1
InputImage
RunParams.UnfixturedFromFixturedTransform
RunParams.UnfixturedFromFixturedTransform.TranslationX
RunParams.UnfixturedFromFixturedTransform.TranslationY
RunParams.UnfixturedFromFixturedTransform.Rotation
OutputImage
[Outputs]
Score
FixturedImage
1
2
VisionProToolBlock
PMAlignTool
54.60ms
VisionProToolBlock
PMAlignTool
54.60ms
3
COGNEX

12 | Confidential © 2013 Cognex Corporation

Perform the last bit of housekeeping:

1) Drag the OutputImage generated by the CogFixtureTool to the [Outputs] collection
2) Rename the default output name to FixturedImage
3) Verify that a new outgoing score terminal is coming out of the right side of the Vision Tool Block



Fixturing in 3D

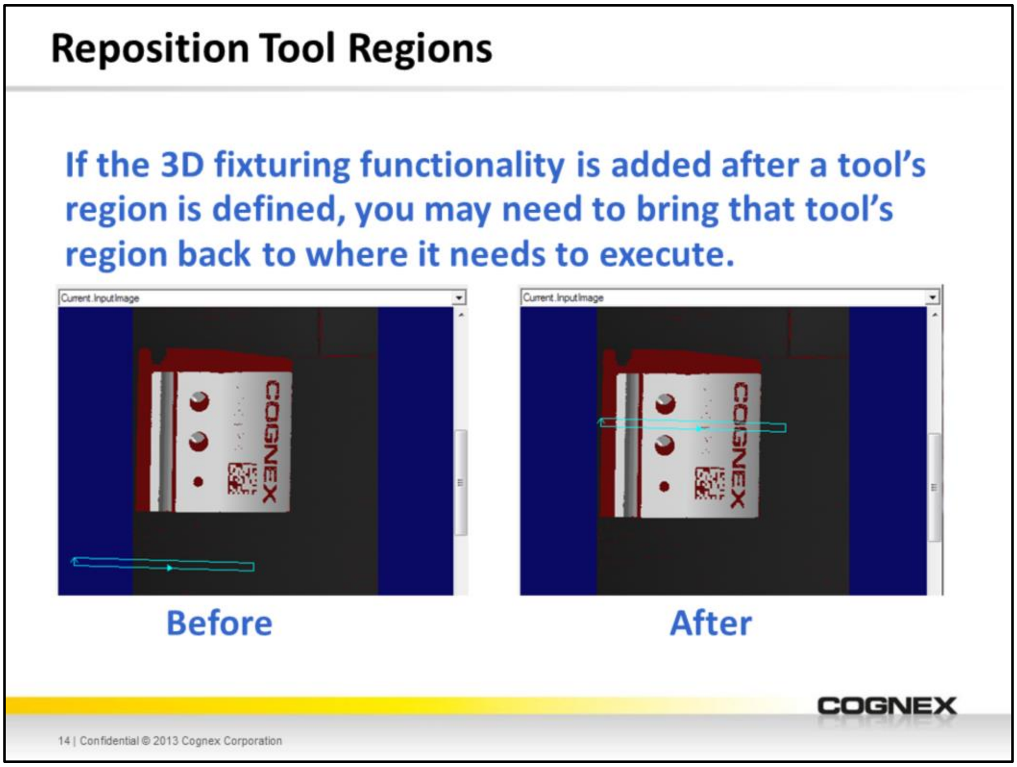
- Fixturing Measures shift in X/Y position and Rotation
- Tells other tools how the part has shifted
- Provides maximum accuracy and repeatability (for non-tilted parts)

13 | Confidential © 2013 Cognex Corporation

Since the height image data is calibrated, it is actually quite simple.

This means no matter if you are close to an object or far away, the number of pixels used to show it is the same.

For this reason the Fixture tool requires only translation and rotation.



Reposition Tool Regions

If the 3D fixturing functionality is added after a tool's region is defined, you may need to bring that tool's region back to where it needs to execute.

Before

After

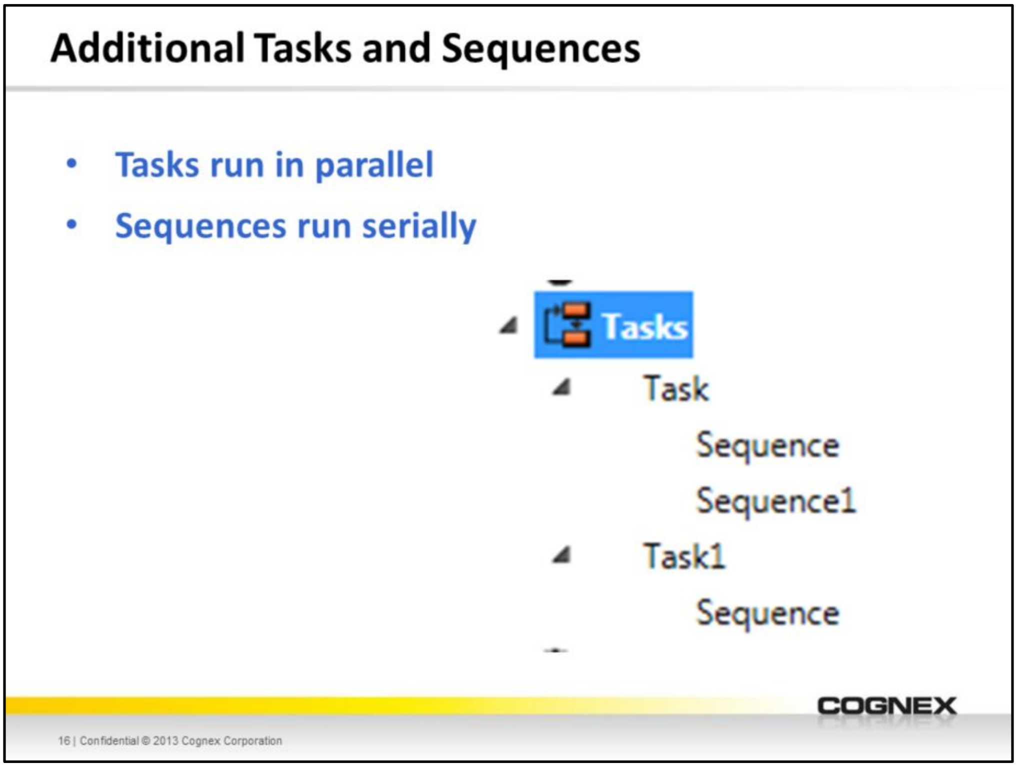
14 | Confidential © 2013 Cognex Corporation

If we perform this fixturing step AFTER other tools have already been placed, then you may have to reposition those tools just once more since they will receive the reposition information from their current location and will be shifted.

Just open up those tools and reposition their region. When we execute the sequence, we can see the region will continue to fall in the correct position.

Tasks and Tools

COGNEX



Additional Tasks and Sequences

- Tasks run in parallel
- Sequences run serially

Tasks
- Task
  - Sequence
  - Sequence1
- Task1
  - Sequence

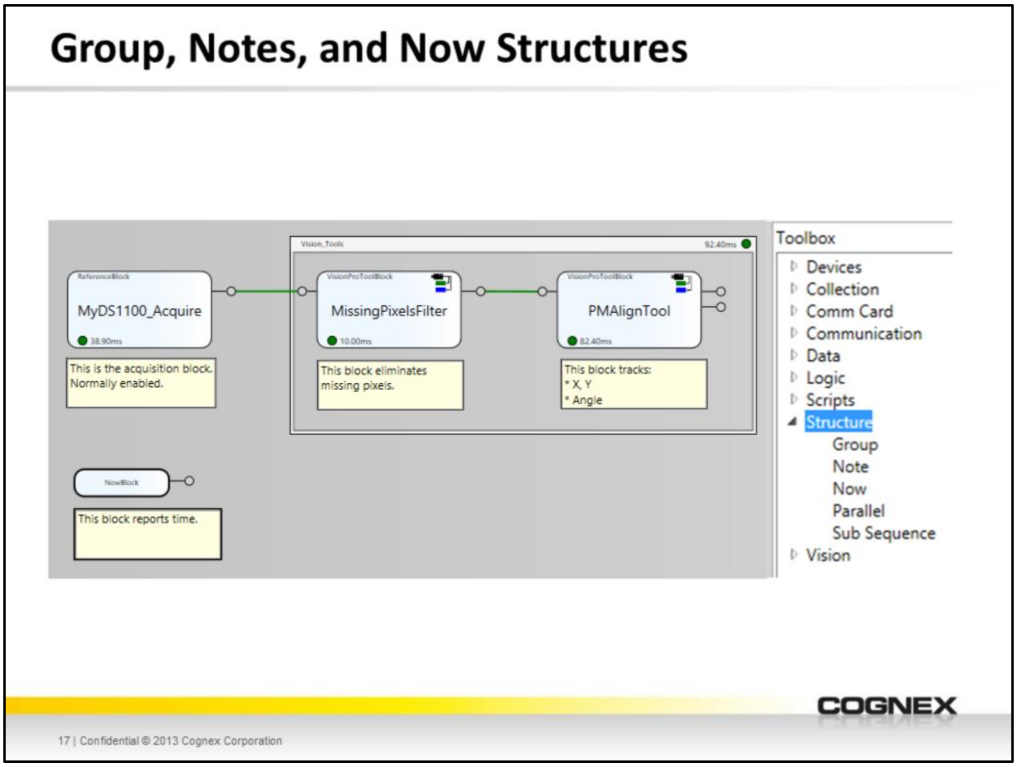
16 | Confidential © 2013 Cognex Corporation

Additional Tasks and Sequences can be added through right-clicking “Tasks” (for another Task) or “Task” (for another sequence).

Multiple Tasks will run in parallel to each other. They could be used as a different path of inspection for a given part.

Multiple Sequences will run in serial to each other – Sequence first, then Sequence1, Sequence2…

If the sequence is renamed, there is no guarantee as to how it would run. For this first release of Cognex Designer, it is suggested to use only one sequence per task.



Group, Notes, and Now Structures

This is the acquisition block. Normally enabled.

This block eliminates missing pixels.

This block tracks:
* X, Y
* Angle

This block reports time.

17 | Confidential © 2013 Cognex Corporation

Groups:

Multiple objects can be combined within a group. This will run the objects "together" and the time for that group will be reported in the upper left corner of the Group box.

Note: For Timing, the group us treated as one entity. That means that since the left edge of the group is before the left edge of the Database, the entire group will run first. Then the Database will get its next image and finally the ToolBlock "BracketInspection" will run.

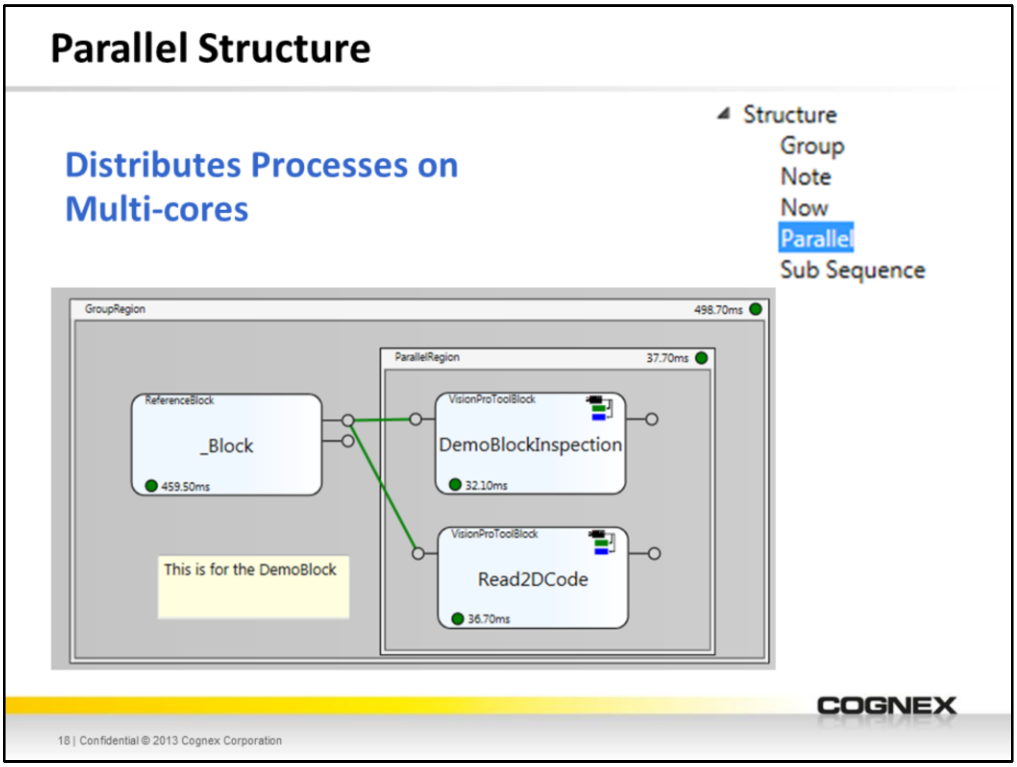
When placing object in a group, make sure the box is entirely around the objects of interest. Otherwise they will not be part of the group.

Notes:

Notes can be created on the Sequence to detail information. It could be used to explain the functionality of code or be created to dynamically give information about part of the process.

Now:

Inserting the Now structure allows access to the current date and time information of the system at a given point.



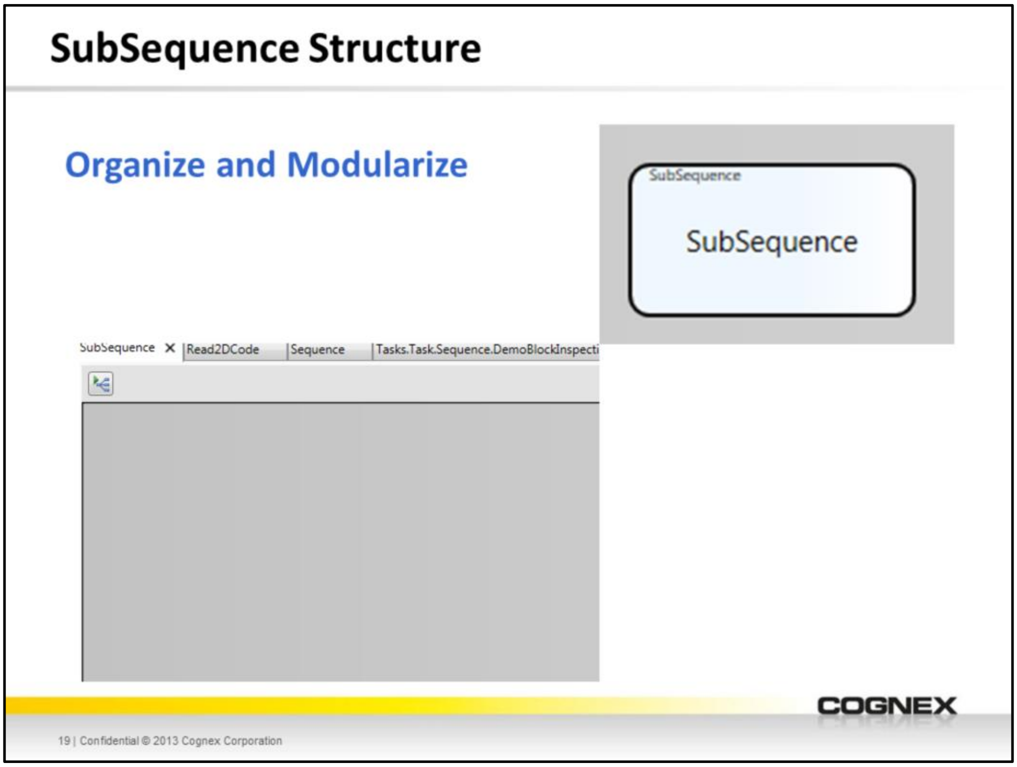
Parallel Structure

Distributes Processes on Multi-cores

18 | Confidential © 2013 Cognex Corporation

The parallel processing structure allows us to put a “box” around the processes that we would like to run in parallel with each other. It will place each process on its own thread so that execution time is reduced and the application will run faster. This allows the processing of intensive applications to be optimized for fast execution.

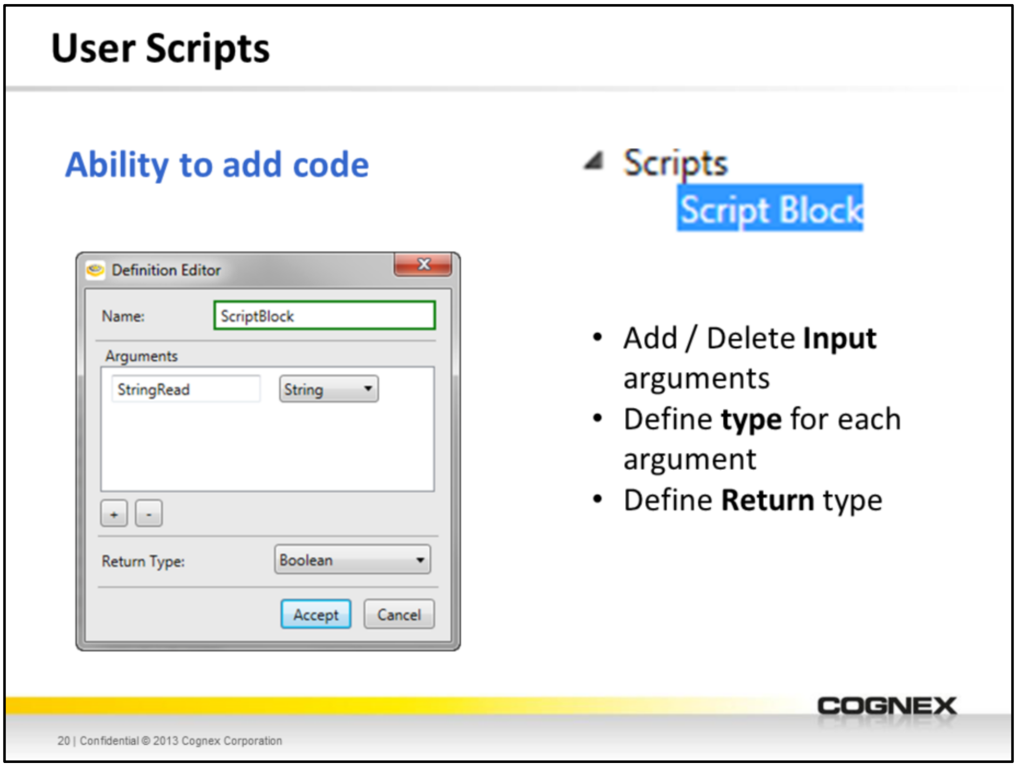
Tip: Disable hyper-threading, as the system might perform the process in parallel, but on the same physical core, hence negating any performance gains.



SubSequence Structure

Organize and Modularize

Allow users to create an embedded sequence to assist with organization and modularization of the code. It will run serially as the left hand side of the SubSequence object is tapped within the original sequence.



User Scripts

Ability to add code

Definition Editor

Name: ScriptBlock

Arguments

StringRead String

Return Type: Boolean

Scripts

Script Block

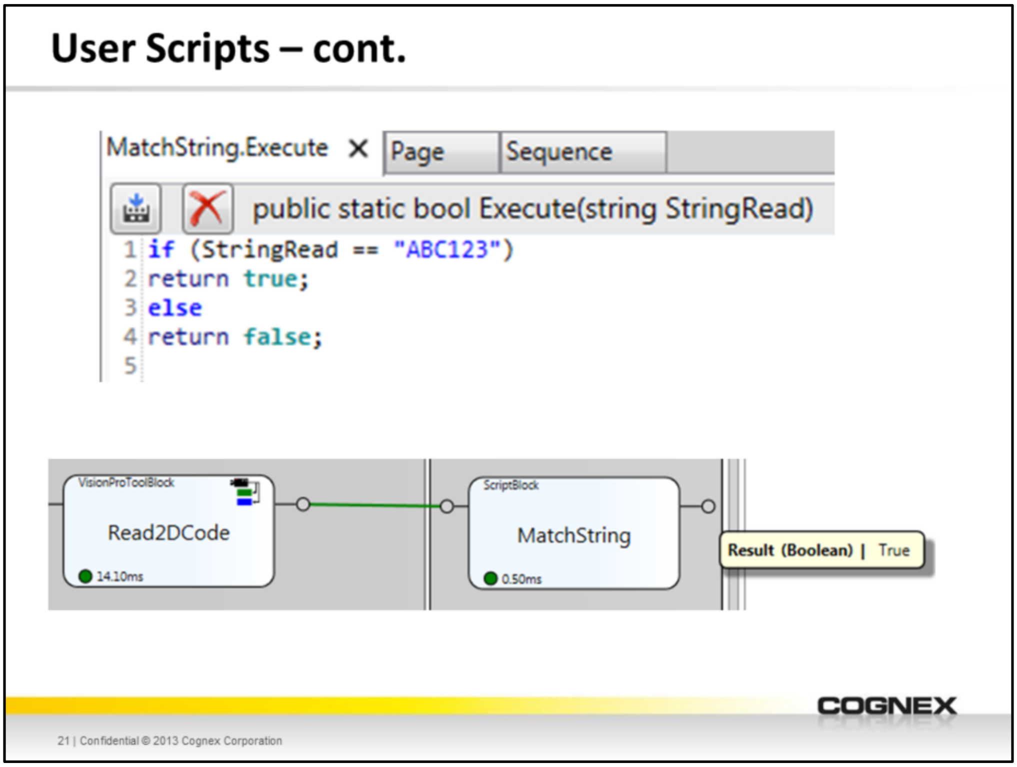
- Add / Delete **Input** arguments
- Define **type** for each argument
- Define **Return** type

20 | Confidential © 2013 Cognex Corporation

The ScriptBlock allows the user to write C# code to work upon the inputs given and to create an output.
Each input name and type is defined. Also the Return type needs to be set.

You can only have a single return value. Its type can be:

- Boolean
- Byte[]
- Integer
- Double (Real)
- String
- Datetime
- Object
- VisionPro Image
- VisionPro Record



User Scripts – cont.

```
public static bool Execute(string StringRead)
if (StringRead == "ABC123")
return true;
else
return false;
```

COGNEX

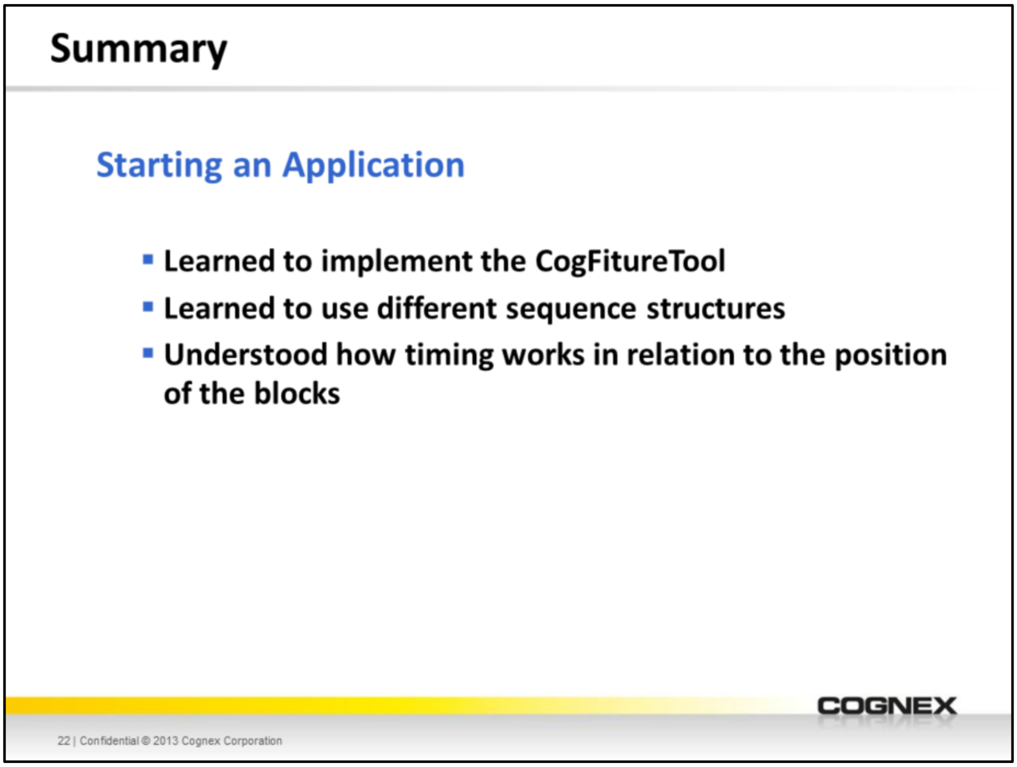
21 | Confidential © 2013 Cognex Corporation

After the script block is created, the script can be edited. You must use C# when writing code for the scripts. Then it is a matter of attaching values to the input and the result can be seen on the output tag as a fly-over value.

After a user script is created, it can also be called programmatically.

button.enabled = MatchString(“ABA122”)

This would disable a button as the inputted value is not the expected value



Summary

Starting an Application

- Learned to implement the CogFitureTool
- Learned to use different sequence structures
- Understood how timing works in relation to the position of the blocks