## Contents

1. **What is Gang Run Printing?** ................................................................................................. 5  
   1.1 Nesting................................................................................................................................. 5  
   1.2 Ganging............................................................................................................................... 5  
   1.3 Tiling.................................................................................................................................... 6  
   1.4 Manual or Automated Gang Run Printing......................................................................... 8  

2. **Working with Gang Run Printing** .......................................................................................... 9  
   2.1 Preparing and Submitting your Graphics (Input Workflow) ............................................... 9  
      2.1.1 Preparing your Graphics............................................................................................... 10  
      2.1.2 Tiling Oversize Graphics............................................................................................. 11  
      2.1.3 Submitting your Graphics............................................................................................ 11  
   2.2 Creating Nested Layouts on the Nesting Server ................................................................ 12  
      2.2.1 Working with Substrate Queues.................................................................................. 13  
      2.2.2 Substrate Sizes............................................................................................................. 15  
      2.2.3 Keeping Track of your Stock....................................................................................... 15  
   2.3 Getting Production Files out of your Nested Layouts (Output Workflow) ..................... 16  

3. **The Gang Run Printing View** ................................................................................................ 17  
   3.1 Working with Substrate Queues......................................................................................... 17  
      3.1.1 Adding a Sheet Substrate Queue.................................................................................. 18  
      3.1.2 Adding a Roll Substrate Queue.................................................................................... 23  
      3.1.3 Editing a Substrate Queue........................................................................................... 24  
      3.1.4 Duplicating a Substrate Queue.................................................................................... 25  
      3.1.5 Deleting a Substrate Queue......................................................................................... 25  
   3.2 Working with Graphics Submitted for Printing............................................................... 26  
      3.2.1 Graphics Overview....................................................................................................... 26  
      3.2.2 Controlling the Printing Flow..................................................................................... 27  

4. **Automating your Workflow Using XML** ............................................................................. 29  

5. **The Gang Run Printing Tasks** .............................................................................................. 31  
   5.1 Prepare Graphics for Nesting.............................................................................................. 31  
      5.1.1 Create Cut Path From................................................................................................... 31  
      5.1.2 Cut Path's Ink Name.................................................................................................... 33  
      5.1.3 Extend Graphics.......................................................................................................... 33  
      5.1.4 SmartMarks................................................................................................................ 35  
   5.2 Import Substrate Sizes from XML..................................................................................... 36  
   5.3 Create Tiles......................................................................................................................... 41  
      5.3.1 Example of SmartMarks for Tiles.............................................................................. 47  
   5.4 Submit to Gang Run............................................................................................................. 52  
      5.4.1 General Tab................................................................................................................. 53
7.3.6 Step 6 - Create the Output Workflow.................................................................115
7.3.7 Step 7 - Add the Output Workflow to the Substrate Queue...........................115
7.3.8 Step 8 - Test........................................................................................................115
1. What is Gang Run Printing?

Gang Run Printing enables you to efficiently match graphic designs to be printed with the available substrate (to have as little substrate waste as possible and faster turnaround), with:

- **Nesting**, 
- **Ganging**, 
- **Tiling**.

This is especially useful when printing Sign and Display material and for shorter production runs.

It takes into account the designs' sizes, shapes and ordered quantities to produce layouts on substrate sheets.

1.1 Nesting

Nesting is laying out several graphic files in the most efficient way so that waste is minimized.

When working with non-rectangular graphics, their bounding boxes may overlap in the layout.

For more information about nesting in Gang Run Printing, see *Creating Nested Layouts on the Nesting Server* on page 12.

1.2 Ganging

Ganging is combining several different graphic files, in one print job, even if those files belong to several orders (for example if one order doesn't fill a whole substrate sheet).
This helps drive production costs down and helps you offer a competitive price to your customers.

1.3 Tiling

Tiling is splitting oversize graphics into rectangular pieces (tiles) for printing. These tiles are separate files that the Gang Run Printing mode can then nest with other tiles or even other graphics.

A typical example are billboards:

This example could be tiled into these 10 smaller parts:
Typical for tiling are the overlaps. We distinguish 3 classic concepts:

- **Standard overlap**: a side without overlap is placed on top of the overlap of the neighbouring tile:

- When producing wallpaper or murals that are applied with a **double cut**, you need an **overlap on both sides**:

  See a video of an example [here](#).

- When producing wall prints in thick material or tiles that are inserted in a framework, you might not need any overlap. This could be an example:
1.4 Manual or Automated Gang Run Printing

- In the Esko editor i-cut Layout, you can perform manual Gang Run Printing: selecting and preparing each graphic individually, then finding layouts (the most efficient layouts are calculated automatically).
  
  When working in i-cut Layout, you can tweak the results if necessary. Learn more in the i-cut Layout manual.

- In Automation Engine, you can automate all parts of the workflow: the preparing of graphics, the submitting to Substrate Queues, their nesting settings and the output of the resulting layouts.
  
  When working automatically in Automation Engine, you can process more graphics in parallel, and you don't need an operator to keep checking when to generate layouts (this is done automatically).
2. Working with Gang Run Printing

Here is an overview of how you can work with the Pilot’s Gang Run Printing mode.

1. You first **prepare** your graphics and **submit** them to be nested on a particular substrate, in your chosen quantity.

   You do this through an **input** workflow, that should contain the **Prepare Graphics for Nesting** task and the **Submit to Gang Run** task.

   Learn more in **Preparing and Submitting your Graphics (Input Workflow)** on page 9.

   Optionally, you can add the **Create Tiles** task to split the prepared graphics into tiles before submitting them.

2. Once the graphics are submitted, they go in a **substrate queue** and the **nesting server** calculates the best possible **nested layouts**, based on the **sheet or roll sizes** and their **nesting settings**. All this is set up and can be monitored in the Pilot’s Gang Run Printing view.

   Learn more in **Creating Nested Layouts on the Nesting Server** on page 12.

3. Once the nesting server has decided one or more **optimal nested layouts**, it delivers them to your **output workflow**, that creates **production files** (layout PDFs, cut files and reports). The output workflow must contain the **Export Nested Layouts** task. You can make this workflow copy the production files to a specific folder where they are picked up by the press.

   Learn more in **Getting Production Files out of your Nested Layouts (Output Workflow)** on page 16.

2.1 Preparing and Submitting your Graphics (Input Workflow)
2.1.1 Preparing your Graphics

The nesting is based on the graphic's cut path, so before you submit graphics to a substrate queue you need to make sure they have a cut path. You may also want to add bleed and SmartMarks. This what the Prepare Graphics for Nesting task is for.

Learn more about the task's options in Prepare Graphics for Nesting on page 31.

Identifying or Extracting a Cut Path

As mentioned, the cut path defines how graphics can be nested.

For example, cardboard cut-outs of a tree (as below left) will be nested and cut differently than postcards of a tree (as below right). Depending on the case, your tree graphic will need a different cut path (shown in red).

The cut path is typically provided in the file as a contour line in a spot color.

For graphics that don't already contain a cut path, you can generate one from the white background, the trim box, or use a separate file.
Adding Bleed
If your file doesn't already have bleed outside the cut path or trim box, you can generate some using the Prepare Graphics for Nesting task.

Adding SmartMarks
SmartMarks are marks that automatically update based on changes to the sheet size, the number of inks on the sheet, etc. You can use them as regular marks or as production controls. There are many types of SmartMarks: registration marks, barcodes, text, placeholders for eyelets, etc.

You typically add SmartMarks to your nested layout in your output workflow (mostly for cutting the sheet), but in some cases you need to add some SmartMarks to your graphic in your input workflow.

For example:
• You can add eyelet marks to a graphic that will be used for a large eyeleted banner. You need to add the eyelet marks before tiling the graphic to fit the substrate.
• When defining the cut path in the input workflow, you can add text on this cut path for easy processing after cutting (for example to indicate the job-order and customer reference, or the order the tiles must be glued in...).

2.1.2 Tiling Oversize Graphics

When the final printed object is too big to fit on your largest substrate size (for example when printing billboards), you can split the graphic into rectangular tiles (fitting the substrate size) that can later be glued back together.

You can do this with the Create Tiles task. This task allows adding overlaps to be able to glue the tiles on top of each other.

2.1.3 Submitting your Graphics

Once your graphics are ready, you can send them to the 'nesting server' using the Submit to Gang Run task.

The most important things to define in that task are:
• how many copies you want of that graphic,
• on which substrate (and optionally on which substrate size) you want it printed,
• if you have a specific due date to get that graphic printed.

Note:
Substrates are organized into substrate queues, which not only define the substrate to use but also how the graphics will be nested.

Learn more in Working with Substrate Queues on page 13.
2.2 Creating Nested Layouts on the Nesting Server

**Automatic Calculation of Nested Layouts**

Once your graphic(s) arrive in the *substrate queue*, the nesting server calculates possible layouts to nest those graphics, together with any graphic(s) already in the queue.

The nesting server tries to find the most optimal layout(s) possible, based on these settings:

- the *nesting settings* you set up,
- the different possible *substrate sizes* (if applicable),
- the *minimum fill* percentage for that substrate queue,
- the use of any *grouping keys* (custom labels added to graphics when you submit them, to control the grouping of graphics for nesting).

When the minimum fill is reached, the nesting server sends the layout(s) to the output workflow to generate production files. All graphics that have been nested in these layouts are then removed from the substrate queue.

If it isn't reached, the graphics stay in the substrate queue until new graphics are submitted to the queue, at which point the nesting server calculates possible layouts again.

**Outputting a Partial Quantity of a Graphic’s Order**

Sometimes the nesting server cannot fit all of the ordered copies of a graphic on layout(s) that satisfy the substrate queue settings.

In this case, rather than generating an extra layout with the remaining copies, the nesting server only outputs the quantity that does fit on the optimal layouts.

For example, if 10 copies of a graphic are ordered but only 8 fit on a layout satisfying the minimum fill, the nesting server does not generate a layout with only the remaining 2 copies. These 2 will stay in the substrate queue.
Once new graphics are submitted to that queue, the nesting server will again try to generate a nested layout with those remaining 2 copies and the new incoming graphics.

When Time becomes More Important Than the Minimum Fill

As your customers expect their printed goods to be delivered on a certain date, sometimes you cannot wait for the minimum fill to be reached. In this case, you can either:

- manually force a layout to be generated from the remaining urgent graphics,
- enter a due date when submitting these graphics. The graphics will then be force-nested in time for that due date.

In both cases the nesting server generates a layout ignoring the minimum fill. However, it may also use other graphics from the queue in the layout to still raise the substrate's fill percentage.

2.2.1 Working with Substrate Queues

A substrate queue acts like a printer queue to which you can send jobs you want to nest and print on a particular substrate.

Substrate Queues Based on Material Types

For each type of substrate material you have in your stock (and still plan to use), you should have a substrate queue.

For example:
In your stock you have 2 piles of PVC substrates, one with thickness 3 mm and one with thickness 5 mm. For these you should set up 2 substrate queues with for example these names:

- PVC 3mm
- PVC 5mm
Substrate Queues Based on Nesting Settings

It can also be useful to define substrate queues based on how the graphics will be nested and printed.

- If you have an expensive substrate, you can set the nesting server to always nest the graphics so that the substrate waste is minimal.
- If your graphics need to be nested on a material that will be printed in offset, you could aim for less plates (so less layouts) and accept the higher waste factor.

So if some graphics need to be nested differently (even if it is on the same material), you might need to create a different substrate queue for each type of printing/processing.

For example:

In your stock you have a pile of 250 g/m cardboard both for business cards and door hanger orders.

Business cards and door hangers will be cut differently:

- business cards will be cut by a paper guillotine,
- door hangers will be cut on a cutting table.

The way graphics are cut influences the way they need to be nested. Since nesting settings are attached to a substrate queue, you need different substrate queues.

In this case you should set up 2 substrate queues:

- Cardboard for Business Cards
- Cardboard for Door Hangers

Grouping Certain Graphics Together

Even when graphics are to be printed on the same substrate or using the same printer, you may still want control with which other graphics they are nested. Using grouping keys allows you to do this without the need to set up separate substrate queues for this.

When submitting the graphics, you can add grouping keys to help control which group of graphics can be considered to end up on the same layout. To the nesting server, these key words act like limitations in its freedom to decide nesting groups.

Even when submitting them to the same substrate queue, the nesting server will only nest graphics together that have the same value for a same grouping key.

Some examples:

- In your stock you have 1 pile of photo paper type "220 gsm". Depending on the order, some graphics printed on this paper need to be laminated, some don't. You can use a grouping key called "Lamination", with the values "Yes" or "No", to prevent laminated graphics being mixed with normal graphics.
  
  In this case you only need to set up 1 substrate queue, for both regular and laminated graphics.

- A similar case is when you do not want to mix graphics that need varnishing with graphics that don't.

- Some customers do not want to mix graphics from different customers. In that case they use the customer ID as grouping key.
2.2.2 Substrate Sizes

If you have a substrate material in several sizes (sheet or roll), you can add these sizes to the substrate queue for that material.

For example:
In your stock you have 3 piles of substrates:
• 1 pile of cardboard 2000x1000mm
• 1 pile of cardboard 750x1200mm
• 1 pile of polystyrene 750x1200mm

For this stock you should set up 2 substrate queues:
• **Cardboard** (with 2 substrate sizes)
• **Polystyrene** (with 1 substrate size)

2.2.3 Keeping Track of your Stock

**Keeping Track of Stock or Not**
Per substrate size, you can decide to keep track of its stock. For sheets this is done in units and for rolls in length.

When you select this option for a substrate size, you first manually enter its current quantity. Then, every time the nesting server uses that size for a nested layout, it will be subtracted from the available quantity. In this workflow, Automation Engine will refuse to create a layout for a size that is out of stock.

When you choose to not have the Gang Run Printing mode keep track of your stock, the nesting server then always considers that substrate size 'available'.

Learn more about this setting in *Adding Sheet Sizes* or *Adding Roll Sizes*.

**Substrate Offcuts**
The nesting server always tries to create layouts with as little waste as possible. However, an approaching due date might force you to produce a sheet or roll that has a significant empty area that could actually be re-used.

Such rectangular left-overs are often called "offcuts". When the substrate is not that cheap, many customers store those offcuts in stock and manually add them as a smaller sheet or roll size of that substrate queue, together with their current quantity.

**Manual or Automatic way to Keep Track of Stock**
- Manual means that a person checks your stock and manually types in those latest amounts directly in the Automation Engine Pilot, in the definition of each substrate size.
• Automatic means that a person checks your stock and manually types in those latest amounts in your stock management system, typically a part of your business system. This business system then updates Automation Engine by sending an XML that is picked up by the dedicated Import Substrate Sizes from XML task.

Note: Learn more about using XML in Automating your Workflow Using XML on page 29.

2.3 Getting Production Files out of your Nested Layouts (Output Workflow)

Every substrate queue is linked to an output workflow. You can build this workflow to your own needs, but it should at least contain the Export Nested Layouts task.

As soon as the nesting server has found one or more optimal nested layouts for your substrate queue, it delivers them to its output workflow, which is started automatically.

The output workflow then generates production files:

• A PDF file for the press, containing the nested layout including marks for printing and cutting, but without technical inks.
• A CUT file for the cutting table, containing the combined cut path of the nested graphics. You can choose from many different formats.
• (Optionally) a nesting report, in PDF format (which is useful for a cutting table operator) and/or in XML format (which is useful for a business system). Learn more in the Export Nested Layouts task.
3. The Gang Run Printing View

The **Gang Run Printing** view in the Pilot shows an overview of all graphics that have been submitted for nesting on each of the available substrate queues.

1. Substrate queues.
2. Graphics submitted for nesting on the selected substrate queue. Once graphics have been nested (and are no longer needed for other layouts), the output workflow starts for the layout they belong to and they disappear from this list.
3. Details of the selected graphic.

From this view, you can:
- define substrate queues and their characteristics,
- have an overview of all graphics submitted to each substrate queue, and their state,
- control the printing flow (for example you can put some graphics on hold, or force-print some of them).

### 3.1 Working with Substrate Queues

When submitting graphics to be nested on a substrate, the graphics arrive in the corresponding substrate queue in the **Gang Run Printing** view.

A number indicates how many graphics have been submitted to that specific **Substrate Queue**. For example: Polystyrene
The nesting server then calculates the best possible nested layout(s) for these graphics, based on:

- the graphics quantity ordered,
- the sheet sizes and nesting settings you defined for that substrate queue.

The substrate queue icon ![icon] becomes animated during this calculation.

**Important:** We strongly recommend that you define your substrate queues before submitting graphics to them.

If you submit graphics to a substrate queue that is not defined yet in the list of **Substrate Queues** (by entering an unknown substrate in the **Submit to Gang Run** task), then this substrate queue will be created automatically in the **Substrate Queues** list, but it will miss crucial parameters (such as sheet sizes and a link to an output workflow).

In such a case, the nesting server is not able to calculate layouts for it and a warning sign on the substrate icon ![warning] indicates that the substrate queue isn't properly configured. In this case, make sure to **add the missing parameters**.

You can also manage your **Substrate Queues** (edit, duplicate or delete them) from the **Gang Run Printing** view.

### 3.1.1 Adding a Sheet Substrate Queue

1. Click the action button ![action] at the bottom of the **Substrate Queues** list and select **New Substrate Queue**...

   You can also go to **File > New Substrate Queue**... or press **Ctrl + N** on Windows / **Cmd + N** on Mac.

2. In the **New Substrate Queue** dialog, select the type **Sheet** click **OK**.

   **Attention:** In this page, we continue explaining how to set up a Sheet based substrate queue. For specifics on adding a Roll substrate queue, go to **Adding a Roll Substrate Queue** on page 23.

3. In the **New Substrate Queue** dialog, enter a **Name** for this substrate queue.

   **Note:** Substrate queue names (and also names of their sizes, see below) are often used in names of output files. That is why it is advised to avoid too special characters in these names.

4. Click **Add** to add a **Sheet Size** for this queue. Learn more in **Adding Sheet Sizes** on page 19.

   **Attention:** You need to define at least one sheet size per substrate queue, otherwise graphics sent to this substrate queue will not be nested and printed.

5. Click **Nesting Settings**... to define how this queue will create a nested layout. Learn more in **nesting settings**.

6. In **Output Workflow**, **select the output workflow** that will be launched on the nested layout(s).
Attention: You need to define an output workflow here, otherwise you won’t get production files for the graphics you submitted to this substrate queue!

Note: A workflow will be launched as soon as a nested layout is made.

7. Click OK to save your substrate queue.

Note: It is possible to save your substrate queue without having defined all the necessary settings (for example if you haven't linked a workflow to your substrate), but you will get a warning message and the substrate queue icon will show a warning sign.

If this is the case, you need to edit your substrate queue to define the missing settings. When you are done, you should see this regular substrate icon:

Adding Sheet Sizes

For every sheet size you want to add:

1. Click Add (on Windows) or + (on Mac).
2. In the Add Sheet Size dialog, enter a Name for the sheet size.
3. In the Size parameters, enter the sheet's width and height.
   
   The units used here are those you defined in the Pilot Preferences (in Edit > Preferences > Units on Windows or Automation Engine > Preferences > Units on Mac).
4. Enter the Margins to use.
   
   The margins won't contain nested graphics, but can contain marks.
5. If you want, you can keep track of the stock you have of that sheet size.
   
   To do this, select Keep track of stock and enter the number of sheets you currently have Available in Stock. Every time the nesting server uses that sheet size for a nested layout, it will be subtracted from the available quantity.

Attention:

• When Keep track of stock is selected and the stock reaches 0, the nesting server no longer create layouts that need that sheet size.

   This can cause graphics to be stuck in the substrate queue (when this was the only sheet size defined for that substrate queue, or when you chose to nest the graphics on that sheet size specifically).

   When you again get sheets available in that size, you should re-edit the available quantity.

   As introduced in Keeping Track of your Stock on page 15, the stock of sheet (or roll) sizes can also be controlled from an external system. Learn more in Import Substrate Sizes from XML on page 36.

• When Keep track of stock is not selected, no quantity is shown. The nesting server then also does not check this and will so always be willing to create a layout using this size.
6. Click OK.

The resulting list of sizes also show a column **Available in Stock**. When **Keep track of stock** was not selected for a size, no number will be shown.

![Edit Substrate Queue](image)

### Defining Nesting Settings for Sheets

1. Enter a **Minimum Fill** if you only want to output the sheet when it’s filled to a certain level. By default, the minimum fill is set to 75%.

   **Attention:** Do not put 100% here as only the graphics inside the cut path count as fill. The margins, gutters etc. all count as waste.

   **Note:** When nesting is based on an irregular shape, be aware that a minimum fill may need to be set quite low. Here is an example of a layout that looks rather full but still only has a fill rate of 38%:

![Layout Example](image)

2. **Overrule Minimum Fill:** Set the hours and minutes before the due date at which you no longer want to wait for the minimum fill, because the job-order becomes too urgent.

3. Select the **Nesting Type** to use for the layout:
   - **Minimal Waste** tries to reduce the waste in material to a minimum.
• **Minimum Layouts** aims to generate as few layouts as possible, potentially at the cost of having some more waste.

**Tip:** Having fewer layouts can reduce setup time, making this option interesting for offset printing.

• **Guillotine Cut** generates layouts that can be cut on a guillotine cutting table (which can only cut in straight edge-to-edge movements).

This usually reduces the number of cuts per layout but may increase the number of layouts and the waste.

In the example below, the layout on the left cannot be cut on a guillotine table. The one on the right can, but includes one graphics or copy less:

4. Select the **Search Method** to use for calculating the layout.

   Depending on if you are running a test or processing graphics for production, you can set the layout search from **Very Fast** to **Slow (Very Accurate)** (the faster the search, the less accurate the result can be).

5. If you want to nest the graphics in the order in which they are coming in, select **Respect the graphics order**.

6. Depending on the **Nesting Type** you selected, you can set extra options:

<table>
<thead>
<tr>
<th>If you have selected...</th>
<th>you can set:</th>
</tr>
</thead>
</table>
| Minimal Waste or Minimum Layouts | • the Fill Starting Point on Sheet (where the first graphic will be placed): at the Top Left, Top Right, Bottom Left or Bottom Right,  
| | • the Primary Fill Direction (in which direction the sheet will be filled): Horizontal or Vertical. |
| Guillotine Cut | the direction of the First Cut: Horizontal, Vertical or Don’t care. |

7. If you want, you can add a **Gutter** in between cut paths, if so, specify the gutter width.
8. If the graphics in your files have a bleed outside the cut path, you can clip it away by entering the bleed distance in **Bleed Clipping Mask**.

9. When working with **Guillotine Cut**, you can choose to allow the nesting server to print a few extra copies of some of the graphics, so that you have more printed sheets with the same layout. This reduces the amount of plates needed for offset printing, so reducing the amount of needed press setups.

   • If you want to do this, set a **Maximum Overrun**.
     
     For example set a **Maximum Overrun** of 10.0% if you don’t mind printing up to 10% more of some of the graphics to have less layouts.
     
     • If you only want to print the exact ordered quantity of each graphic, and don’t mind the number of layouts, keep the **Maximum Overrun** at 0.0%.

For example, you need to print 100 "BLUE" graphics and 10 "RED" graphics, on a sheet size that can fit 10 graphics.

   • If you don’t allow overrun, you will have 2 layouts:
     
     • A layout with 9 "BLUE" and 1 "RED" graphics, to be printed 10 times (giving 90 "BLUE" and 10 "RED" graphics).

     ![Diagram of layout with 9 BLUE and 1 RED graphics, printed 10 times.]

     • A layout with 10 "BLUE" graphics, to be printed 1 time (for the remaining "BLUE" graphics).

     ![Diagram of layout with 10 BLUE graphics, printed 1 time.]

   • If you allow a **Maximum Overrun** of 20.0%, you will have only 1 layout, with 9 "BLUE" and 1 "RED" graphics, to be printed 12 times.

     This gives 108 "BLUE" graphics (8% more than ordered) and 12 "RED" graphics (20% more than ordered).

     ![Diagram of layout with 9 BLUE and 1 RED graphics, printed 12 times.]

   • If you only want to print the exact ordered quantity of each graphic, and don’t mind the number of layouts, keep the **Maximum Overrun** at 0.0%.
10. In **Rotation**, specify if the nesting server is allowed to rotate the graphics in the layout to get a better fit. You can choose:

- **Allow Any Angle**,  
- **Allow 180 degrees** (clockwise),  
- **Allow 90, 180 or 270 degrees** (clockwise),  
- **Not Allowed**.

**Linking a Workflow to Your Substrate Queue**

Select the output **Workflow** to launch on the layouts that the nesting server created. This workflow minimally needs the **Export Nested Layouts** task.

This **output workflow** will be launched either:

- when the **minimum fill percentage** for that substrate queue is reached,
- in time for the submitted graphics's **due date** (if they have one),
- when you **force print** one or more graphics from the substrate queue.

**Attention:** You must link an output workflow to your substrate queue, otherwise you won’t get production files for the graphics you submitted to that substrate queue!

### 3.1.2 Adding a Roll Substrate Queue

Adding a substrate queue for a roll is very similar to adding one for a sheet, as explained in [*Adding a Sheet Substrate Queue*](#) on page 18.

When you add a **New Substrate Queue**, you there choose the **Roll** icon.

We here list the other differences:

**Adding Roll Sizes**

**Adding a Roll Size** is identical to [*Adding Sheet Sizes*](#) on page 19, with these differences:

1. The **Size** is the width of the web (across the roll).
2. You can define a **Min** and **Max** length (along the roll):

**Attention:** Nesting on roll will:

- generate a layout in case the layout exceeds the minimum length and reaches the minimum fill percentage. In case the due date has expired, a layout will be generated even if the minimum length is not reached.
- reject graphics that exceed the maximum length (these graphics can't fit on the substrate).
- always set the **Primary Fill Direction** or **First Cut** to the direction across the roll.

**Note:** In case the PDF file for printing needs a different orientation, use the rotation option in the **Export Nested Layouts** task.
3. **Stock**: With roll sizes, the stock is the amount of meters left on all your rolls of the same type of substrate. It is a 'global' stock for that roll. It is not the amount of rolls or the amount per roll.

**Defining Nesting Settings for Rolls**

Roll **Nesting Settings** are identical to **Sheet nesting settings**, with these differences:

- The **Primary Fill Direction** is always vertical, across the roll.

- **Fill Starting Point on Roll**: choose the icon for **Top Right** or **Top Left**. The filling will always be across the roll.

- When the nesting type is **Guillotine cut**, there is no **First Cut** setting. It will then always be across the roll.

**Linking a Workflow to Your Substrate Queue**

This is identical as we explained for sheet type substrates queues.

### 3.1.3 Editing a Substrate Queue

1. Double click the substrate queue that you want to edit.

   **Tip:** You can also go to **File > Open** or press **Ctrl + O** on Windows / **Cmd + O** on Mac, or click the action button 📜 and select **Open**.

2. In the **Edit Substrate Queue** dialog, you cannot change the substrate queue **Name**, but you can edit all the other settings.

   **Note:** If you want to change the substrate queue name, we recommend you duplicate the substrate under your chosen name and then delete the original substrate queue.

3. You can add, modify or delete sheet or roll **Sizes**.

   - To add a sheet size, click **Add** (on Windows) or + (on Mac) and define the sheet or roll size settings (for **sheets** or for **rolls**).

   - To modify a size, select it, click **Edit** (on Windows) or ⌘ (on Mac), and change the settings.

   - To delete a size, select it and click **Remove** (on Windows) or - (on Mac).

4. Click on **Nesting Settings...** to check or change **nesting settings** for this substrate queue.

5. Check or change the **Output Workflow** that this queue will launch on the nested layouts.

6. Click **OK** to save your changes or **Cancel** to revert.
Note: After changing the settings, the nesting server will automatically re-calculate nesting layouts for the graphics in that substrate queue, even if no new graphic was submitted.

3.1.4 Duplicating a Substrate Queue

1. Select the substrate queue that you want to duplicate, click the action button 🕵️‍♂️ ⬤ and select Duplicate....

   Tip: You can also go to File > Duplicate... or press Ctrl + D on Windows / Cmd + D on Mac.

2. In the Duplicate Substrate Queue dialog, enter a Name for the new substrate queue.

3. The queue settings are the same as the old substrate queue’s, but you can modify them if necessary.

   See:
   • Adding Sheet Sizes on page 19 or Adding Roll Sizes on page 23.
   • Defining Nesting Settings for Sheets on page 20 or Defining Nesting Settings for Rolls on page 24.
   • Linking a Workflow to Your Substrate Queue on page 23.

4. Click OK to save your new substrate queue.

3.1.5 Deleting a Substrate Queue

1. Select the substrate queue that you want to delete and press the delete key.

   Tip: You can also go to File > Delete or click the action button 🕵️‍♂️ ⬤ and select Delete.

2. In the dialog that pops up, click Delete to confirm that you want to delete that substrate queue.

   Note: When you confirm to delete, also a queue that still has graphic entries will be deleted.
3.2 Working with Graphics Submitted for Printing

3.2.1 Graphics Overview

List of Graphics per Substrate Queue
Each substrate queue indicates with a number how many graphics are in its queue.

Select the substrate queue to see the list of those graphics and printing information on the right.

Printing Information
For each graphic, you will see this information based on the settings when submitting the graphic to that queue:

- the File Name of the graphics and the page to be printed,
- the Substrate Size it should be printed on (if this was defined when submitting the graphic),
- the Quantity ordered,
- the Remaining Quantity to be printed (if some copies of the graphic have already been included in a nested layout this is what remains in the queue),
- the State of the graphic in this substrate queue,
- the date it was submitted (Submit Date),
- the Due Date,
- the Rotation,
- the File Path,
- (if used) Custom labels. These may have been used as grouping key or just serve a labels (here and in the report PDF). Learn more about their use in the page about the 'extra' tab of the 'Submit' task.

If you want to find a specific graphic in this overview, you can filter the list using the search field, or sort the graphics by clicking the column header. The sorting order has no influence on the output.

Tip:
You can also show / hide or reorganise columns by going to View > Select columns in 'Gang Run Printing'...
For example, you can also show columns with your labels and grouping keys and use that to sort graphics.
### Graphics States

<table>
<thead>
<tr>
<th>Icon</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>➤</td>
<td>The graphic is ready to be nested and output. When the nesting server calculates possible layouts, it will try to include all of the graphics in this state in the most efficient layout(s) possible.</td>
</tr>
<tr>
<td>❌</td>
<td>The graphic is on hold. Graphics in this state will not be included in the layout calculations.</td>
</tr>
</tbody>
</table>
| ✗    | The graphic can not be placed on a layout because they are too big. It could be that:  
  - The graphic is too big for any available size of that substrate. In this case, consider either defining a bigger sheet size (if you have some available), or choosing a different substrate when submitting the graphic to the queue.  
  - The graphic is too big for the size you chose to nest it on (when submitting it to the queue). In this case, choose a bigger size, or set the graphic to be nested on Any size of your substrate when submitting it to the queue.  
  - The graphic is too big to fit within the rotation possibilities defined for that substrate queue. In this case, change the rotation parameter to Allow any angle or change it in the Submit to Gang Run task (which can overrule the substrate queue setting). |

If none of these solutions help, you need to tile your graphic before submitting it to the substrate queue.

### 3.2.2 Controlling the Printing Flow

By default, graphics will be sent to the output workflow automatically either when the desired minimum fill is reached or at the time you set (a certain amount of time before the due date).

However, you can:

- Put graphics on hold: select the graphic(s) and click the Hold button 🔄. Graphics on hold have a 'hold' icon 🌵 next to them in the table.

  **Note:** You can put one or more graphics (or even all graphics in the queue) on hold, but you cannot put the queue itself on hold. Even if all of the queue's graphics are on hold, new graphics submitted to the queue will still trigger a layout calculation.

- Release graphics that are on hold: select the graphic(s) and click the Release button 🔄. Graphics that are not on hold have a 'ready' icon 🔄.

- Force-print graphics: select the graphic(s) and click the Print Now button 🔄.

  **Note:** To get a better substrate usage, some other graphics may be printed along the graphic(s) you force-print. In this case, you will see that either these graphics disappear from the list, or their Remaining Quantity is reduced.

Graphics that are on hold will not be included. If you want to force-print a graphic on hold, you need to release it first.
Tip: When, even after a while, still nothing happens with the graphics you asked to ‘print now’, then check if the substrate size that was requested for that graphics
• (still) exists
• still has stock left (learn more in Keeping Track of your Stock on page 15).

Remove a graphic from the queue: select the graphic and click the Delete button.

Note: This will not delete the graphic file itself, only remove it from the substrate queue.
4. Automating your Workflow Using XML

The use of XML enables many ways to integrate Automation Engine’s Gang Run Printing mode with an external system, typically the system that manages orders, production details and stock.

This bidirectional communication via XML results in a high form of automation.

We can distinguish a basic and an advanced level of such integration:

- **Basic use of XML in Gang Run Printing**

  ![Diagram: Using XML in GRP (basic use)]

  - An XML file describing a job is sent to Automation Engine (typically a Folder Access Point). The XML informs the workflow of the name of the graphic file (only one per XML) and also what to do with it (substrate, quantity, etc.). This is illustrated in detail in the example [Basic XML driven workflow on page 110](#).

  **Note:** These XML files can be very simple and serve a workflow where you mostly get orders for copies of one graphic at a time. See the below ‘advanced’ section to read about more complex XML files that ask the nesting server to calculate layouts for a list of files.

  - A dedicated task uses an XML file to update the list of the available substrate sizes and also of their current quantity in stock. Learn more in [Import Substrate Sizes from XML on page 36](#).

  - The task that exports the nested layouts into print and cut files also writes report files. You can choose to have them in XML form, so that an external system can use them for production status (how many were already nested?) or for post-production cost-calculation (material usage/waste). Learn more in [Export Nested Layouts on page 70](#).

- **Using XML for Advanced Communication with the Nesting Server**

  The dedicated [Submit XML to Gang Run](#) task enables these advanced forms of XML based communication:

  - The business system sends XML files that represent ‘nesting requests’, possibly of complex orders. They can be about different graphics, each with their own quantity, substrate, etc. By doing this, the business system can ask the nesting server possible


layouts for estimation purpose only. It then uses the return data for its own cost calculation and maybe production preparation.

- After the layouts have been exported, the feedback XML to the business system describes the whole order and includes references to the actual production files.

Here’s a schema about this workflow:

The XML files that the business system sends must match a *detailed specification*. Learn more in *Submit XML to Gang Run* on page 60.
5. The Gang Run Printing Tasks

Note: In a standalone setup of Device Manager, several of these tasks are not available.

5.1 Prepare Graphics for Nesting

Use this task to prepare individual graphics for Gang Run Printing. You can create a cut path, extend the graphics and add SmartMarks.

This task supports these input file types:

- PDF (single page or multipage),
- Normalized PDF (single page or multipage).
- TIFF (CMYK or grayscale),
- JPEG (CMYK or grayscale. Progressive JPEG compression is not supported.),
- PSD (CMYK or grayscale),

Typically, you use this task before a Submit to Gang Run task in an input workflow. The input workflow prepares the graphics and submits them to a substrate queue on the nesting server (see Working with Gang Run Printing on page 9 for an overview of the whole Gang Run Printing process).

5.1.1 Create Cut Path From

In Create Cut Path from, define what to use as a cut path, and how to generate one.

<table>
<thead>
<tr>
<th>If you want to...</th>
<th>do the following...</th>
</tr>
</thead>
</table>
| use an Ink as cut path | 1. Select Ink.  
2. Enter the name of the ink to use in the field below. |

Tip:

- If the ink to be used for the cut path has different names in different input files (for example if you have customers using names in different languages like cut, découpe, taglio...), you can add several ink names, separated by a comma (, ). Their order is important because the task will use the first ink it finds in the list as cut path!

- You can use wildcards and/or SmartNames to define ink names. For example use Pantone* if you want to use any Pantone ink used in the file as a cut path.
### If you want to... | do the following...
--- | ---
**Tip:** When building a workflow using an example file, click ![ink](ink.png) to select one or more of the inks from the example file. If you always receive CMYK graphics with one spot color used for the cut path, select **ANY SPOTCOLOR**.

3. Define what to do if the ink(s) you defined are **not found** in the file:
   - If you want to use the trim box as cut path instead, select **Create cut path from trim box**.
   - If you want the task to end in error, select **Generate an error**.

| use the **Trim Box** as cut path | Select **Trim Box**. |
| use the cut path defined in an external ARD, PDF or Normalized PDF File | Select **File** and **Browse** to that file (or use SmartNames to define the path to the file). |
| create a cut path from the contour of the graphic on its **White Background** | Select **White Background**. **Note:** This will include both the contour of the design on the white background and the contours of any "white holes" in the design in the cut path. You can then remove the "white holes" contours using the **Remove holes** option. |

### Optimize Cut Path

This **Optimize**... dialog offers ways to optimize the shape of your cut path.

**Important:** Especially when you created the cut path with the **From White Background** method, you typically need to optimize the automatically generated cut path contours to facilitate and speed up the cutting process.

You can:

- **Remove holes** in your cut path.
  - Select this if the cut path was extracted from a white background and your graphic contains white. This option makes sure that the white parts in your graphic are not considered part of the background (which would create holes in your graphic).
- Smooth small jagged edges in your cut path.
  - Select **Smooth cut path** and enter the desired minimum distance. Any point that deviates from the cut path by less than that distance will be smoothed.
- Remove small objects from your cut path.
Select **Remove small contours** and enter the minimum contour size. Any objects smaller than this will be removed from the cut path.

- Move the cut path inwards or outwards.

Select **Offset cut path** and enter a positive offset value (to move it outwards) or a negative one (to move it inwards).

### 5.1.2 Cut Path's Ink Name

In **Cut Path's Ink Name**, enter the ink name to use for the cut path you just defined.

- If you defined the cut path based on an ink in your file, this will rename that ink to the name you enter here.

**Tip:** Use this to standardize cut path ink names across files (especially when your files come from different customers).

- If you defined the cut path from the file's trim box, white background, or an external file, this will add the cut path in that ink in your graphic file.

Later in the **input workflow**, the cut path ink name you enter here will be used to identify the cut path in the **Submit to Gang Run** task.

### 5.1.3 Extend Graphics

If you want to extend the graphics, choose one of these modes:

**Create Bleed**

If the graphic contains no bleed, you can add some here. Select **Create Bleed** and choose from these **Options**:

- the **Bleed Distance** (the distance outside the cut path).
- the **Bleed Type**. You can choose between:
  - Choose **Clone** to take the part of the graphic just inside the cut path and clone it to the outside of the cut path.
  - **Clone From**: choose the distance on the inside of the cut path from which data will be cloned. This is useful in cases where graphic elements are very close to the cut but are not required to bleed out. This can avoid unwanted effects. An example:
This option works on cut paths with any shape (free form or rectangular).

**Note:** When an RGB image is cloned, the cloned bleed will contain CMYK information. The image data within the cut path remains unchanged.

- Choose **Mirror** to take part of the graphic (as much as the bleed distance) and mirror it to the outside of the cut path. This option is only useful on rectangular cut paths.

  The visual effect is less distracting and is advised with large bleeds or when the bleed remains visible (as in below example of a photo canvas).

**Extend Content**

This mode mirrors the graphic content over the 4 sides and corners of the original cut path, and extends the cut path. This can be used to create a hem for a flag, banner or canvas.
In the Options, 
- Set the **Hem Size**, defining how much the cut path needs to be extended.
- Define the shape for the **Corner**: square, round, bevel or cut-out. Use **Edit...** to control the shape with a SmartName. Learn more about using SmartNames for buttons in *SmartNames of Parameter Values*.
- For round and beveled corner shapes you can supply a **Bevel Distance**.
- In **Limit Bleed**, you can add bleed on the extended cut path. As the hems are created by mirroring the graphics, the bleed will also make use of those mirrored graphics.

### 5.1.4 SmartMarks

You can add **SmartMarks** to the resulting graphics (a SmartMarks set that you created in an Esko editor *i-cut Layout* or *PackEdge*).

For example, you can add a **Text on Cut Path** mark to automatically add the name of the graphics file and other info on the cut path, for example in the bleed zone. This information is then visible on the cutting table and so helps organize your printed graphics after cutting. An example:
5.2 Import Substrate Sizes from XML

**Concept**

This task allows to use an XML file to

- import (and create) substrate sizes into the Gang Run Printing database
- update substrate size information (for example the available quantity).

This XML file typically comes from a business system, the MIS or ERP that also manages the stock of printing material. Such an XML file is very useful for example to regularly import a list of left-over stock entries (aka "offcuts").

**Usage**

The task panel serves to map the language in the XML to the required terms in Automation Engine for these substrate parameters. The result will be visible in the list of substrate queues or in the details of their substrate sizes.
XML Element Name: You first need to specify the name of the XML element that contains the substrate size information.

Then, for each Size Parameter, select and click Edit to specify the corresponding XML Element or Attribute that is used in the XML file.

The Size Parameter 'Name' is the name of the substrate size and the Size Parameter 'Substrate' is the name of the substrate queue that that size will be added to (or be updated when already there).

Important: Substrate queues mentioned in the XML file that do not exist yet in Automation Engine are automatically created using the default nesting settings. Note that you will still need to complete their settings by adding an output workflow.

Important: All sizes are assuming the units as defined in the Pilot's Preferences.

These examples illustrate the XML specifications:

**Example XML to Import Sheet Substrate Sizes**

In this example, as shown in above screen shot, the XML Element Name is 'Size' and there are 2 such elements. Their XML attributes are also those that the default task ticket proposes. The 2 substrates described in this XML file are sheets. That is defined by their attribute Type P (for 'Plano').
Example XML to Import Roll Substrate Sizes

To define a roll size, the XML file needs XML attributes that the ticket then maps to

- **Type**, which for a roll is 'R'
- **Width**
- **Roll Minimum Length** and **Roll Maximum Length**.

An example (the default ticket will map these to the right **Size Parameters**):

```xml
<?xml version="1.0"?>
<ForEsko>
  <Size>
    <Substrate>MyStandardRoll</Substrate>
    <Name>RollNarrow</Name>
    <Type>R</Type>
    <Width>390</Width>
    <RollLengthMin>720</RollLengthMin>
    <RollLengthMax>900</RollLengthMax>
    <TopMargin>5</TopMargin>
    <BottomMargin>5</BottomMargin>
    <LeftMargin>5</LeftMargin>
    <RightMargin>5</RightMargin>
  </Size>
</ForEsko>
```

Method and Example XML to Update the Available Quantities of Substrate Sizes
**Note:** For sheets, the quantity is the number of sheets. For rolls the quantity is the amount of units (as defined in *Preferences*).

In the task ticket,

- the *Size Parameter* "Available quantity" allows to define a stock quantity of a substrate size.

- the *Size Parameter* "Use available quantity" allows to define whether the quantity of that size should be tracked (counted) when nesting layouts use that substrate size.

See an example (same syntax for sheet or roll):

```xml
<?xml version="1.0"?>
<ForEsko>
  <Size>
    <Substrate>MyStandardRoll</Substrate>
    <Name>RollNarrow</Name>
    <Type>R</Type>
    <Width>390</Width>
    <RollLengthMin>720</RollLengthMin>
    <RollLengthMax>900</RollLengthMax>
    <AvailableQuantity>5550</AvailableQuantity>
    <UseAvailableQuantity>1</UseAvailableQuantity>
    <TopMargin>5</TopMargin>
    <BottomMargin>5</BottomMargin>
    <LeftMargin>5</LeftMargin>
    <RightMargin>5</RightMargin>
  </Size>
</ForEsko>
```

The above example created or updated this *Roll Size* in this *Substrate Queue*:
For the **Size Parameter “Use available quantity”** (in this XML described as `<UseAvailableQuantity>`), you can set these values:

- 0: this disables **Keep track of stock**.
- 1: this enables **Keep track of stock** (used in above XML example).
- 2: this deletes the substrate size (for example because all left-over pieces have been used in production).
- 3: same as value 2.
- 4: this disables **Keep track of stock** and removes the substrate size if the quantity defined in the XML $<=$ 0.
- 5: this enables **Keep track of stock** and removes the substrate size if the quantity defined in the XML $<=$ 0.

**Note:** When using the values 4 or 5, the MIS (business system) only needs to pass 1 parameter to manage the stock: the one that maps to **Available quantity**. And when that quantity is $<=$ 0, Automation Engine will also remove that substrate size. That way, there is no need for the MIS to explicitly mention the value 2.

So depending on the logic the MIS applies,

- It can either use the value 0 or 1 and at a certain time also use 2 if it wants to remove a substrate size. Automation Engine then does not remove the substrate size unless it is explicitly told so (by that value 2).
- or the MIS can focus on the value of the available quantity and in its XML consistently use the value 4 or 5. When that available quantity reaches $<=$ 0, Automation Engine then automatically also removes the substrate size.
Important Note when you used this task before v 14.1

In Automation Engine v 14.0 this task was named "Import Sheet Sizes from XML". In v 14.1, dedicated support for roll sizes was added and this task was renamed to "Import Substrate Sizes from XML".

Also, since v 14.1, the default ticket for the task Import Substrate Sizes from XML was adapted: SheetSize was renamed to Size, Sheet Name was renamed to Name, SheetHeight was renamed to Height and SheetWidth was renamed to Width.

This means that if you used the default ticket in v 14.0, that the default ticket in v 14.1 will generate an error.

To correct this situation, choose one of the following approaches:

• adapt the XML files to make sure they use the new names (advised).
• copy the default ticket to a new ticket, adapt the names to the old names and use this new ticket.

5.3 Create Tiles

Use this task to tile oversize graphics, used for example for billboards, into smaller graphics to be printed. You can select the size and number of the tiles, and allow for an overlap if necessary.

You can send these graphic file formats to this task:

• PDF (single or multipage),
• Normalized PDF (single or multipage),
• TIFF (CMYK or grayscale), 8-bit TIFF also supported.
• JPEG (CMYK or grayscale. RGB JPEGs result in CMYK PDF. Progressive JPEG compression is not supported),
• PSD (CMYK or grayscale).

Note:

• For each image file or single page PDF, the task outputs a multipage PDF in which each page contains a tile.
• For multipage PDFs, the task outputs a multipage PDF (in which each page contains a tile) for each page of the input multipage PDF.

1. In Method, choose how to determine the number and size of the tiles:

• If the tile size is the most important (if for example you can only print a certain size), choose Use Fixed Tile Size and enter the horizontal and vertical Tile Size.

  Note: The last tiles might be smaller (depending on the tile size and the size of the original file).

  Select Allow Rotation if you don't mind having the graphic rotated by 90 degrees in case this fits better and generates less tiles.

• If the number of tiles is more important to you than their size, choose Use Grid and enter the Number of Tiles you want horizontally and vertically.
• If you don’t know the exact number or size of the tiles you want, but you can only print up to a certain size, choose **Automatic Tiling** and enter the **Maximum Tile Size** horizontally ↔ and vertically ↑.

The number and size of the tiles are calculated to have the least waste and the least tiles possible, within that maximum size.

All tiles will have the same size, except when **adding an overlap**.

Select **Allow Rotation** if you don’t mind having the graphic rotated by 90 degrees in case this fits better and generates less tiles.

• If you have a tiling template that you have created in **i-cut Layout** and want to use here, choose **Use Template** and select your template in the **Template** list.

---

**Note:**

To do this, you need to:

1. Make sure your i-cut Layout application is connected to your Automation Engine server.
2. Create and save the template in i-cut Layout (this automatically saves it to the correct location on your Automation Engine server: \AEservername\bg_data_custom_v01\dat\tiling).

See the i-cut Layout manual for more information about connecting to an Automation Engine server and creating a tiling template.

With a tiling template:

• you can see exactly how your graphic will be tiled when creating the template in i-cut Layout (if you create the template from the same/a similar graphic that you will use in Automation Engine),
• you make sure you have consistent tiling in i-cut Layout and Automation Engine (for example if you use i-cut Layout for some graphics orders and Automation Engine for others).

**Attention:** If you want to add overlaps to your tiles when using a template, you need to already add them in the template in i-cut Layout. You cannot add overlaps to tiles created from a template in this **Create Tiles** task.

If you enable **Fit graphics to grid**, your graphic will be scaled to match the grid size, both vertically and horizontally. This can result in non-proportional scaling!

2. Open **Tiling Sequence...** to define the sequence of the tiles (not available when you use a template).

   a) In **Tile Start Position**, select where you want the first overlap to be: on the top left, top right, bottom left or bottom right tile.

**Note:**

You can use SmartNames for most options of this dialog. For button options like Tile **Start Position**, **Direction** and **Pattern**, you can set a SmartName by clicking \[\], and edit it by clicking **Edit...**
Tip:
To know what the SmartName should resolve to to get a certain value for a parameter, do the following:

1. Right-click the parameter value and select **Copy Parameter Value**.
   - For a parameter with a field to edit (like **Overlap**), enter the desired value in the field then right-click in that field.
   - For a parameter with buttons (like **Tile Start Position**), right-click the button of your choice.

2. Paste this in a text editor.

   For example, the SmartName to get a top left **Tile Start Position** should resolve to \( tl \).

b) In **Direction**, select whether the overlaps should run horizontally \( \rightarrow \) or vertically \( \downarrow \).

   Depending on the **Tile Start Position** you chose, horizontal overlaps can run from left to right \( \rightarrow \) or from right to left \( \leftarrow \), and vertical overlaps can run from top to bottom \( \downarrow \) or from bottom to top \( \uparrow \).

c) In **Pattern**, choose whether you want the overlaps to run snake-wise \( \rightarrow \) or in zig-zag \( Z \).

   The exact snake-wise or zig-zag pattern depends on what **Tile Start Position** and **Direction** you selected.

   For example, when starting on the top left tile \( \uparrow \) and running horizontally \( \rightarrow \) with a snake-wise pattern \( \rightarrow \), you will get something like this:
We advise that you run a test to determine exactly what pattern works best for your graphic.

**Note:** When using overlaps, the direction is often decided by physical laws like the direction of rain and wind.

**Note:** When using a vertical snake direction, the tiles in the uneven columns will be rotated 180 degrees. This is done to optimize color consistency between rows when printing on large printers. You can see this rotation in their page in the output multipage PDF and also in the tiling report. See this illustration where 2 columns of tiles will be created with a 180 degree rotation:

3. Define what to use as Cut Path.

**Note:** Whereas the *Prepare Graphics for Nesting* task allows you to generate a cut path if the graphic doesn't contain one, the *Create Tiles* task needs the cut path to be already available, either as an ink or by using the trim box as cut path.

**Note:** The task actually checks this information first because it is the cut path that decides the size of the input graphic that should be tiled. This overall size then influences the required number of tiles (unless you set a fix number).

<table>
<thead>
<tr>
<th>If you want to...</th>
<th>do the following...</th>
</tr>
</thead>
<tbody>
<tr>
<td>use an ink as cut path</td>
<td>1. Enter it in the Cut Path field.</td>
</tr>
</tbody>
</table>
If you want to... | do the following...
---|---

**Tip:**
- If the ink used for the cut path has different names in different input files (for example if you have customers from different countries using names like *cut*, *découpe*, *taglio*...), you can add several ink names, separated by a comma (, ).

  The order you use is important as the task will use the first ink it finds in the list as cut path!

- You can use wildcards and/or use SmartNames to define ink names. For example use *Pantone* if you want to use any Pantone ink used in the file as a cut path.

**Tip:**
When building a workflow using an example file, click to select one or more of the inks from the example file.

- If you always receive CMYK graphics with one spot color used for the cut path, select *Any SpotColor*.

2. Define what to do if the ink(s) you defined are not present in the file:
- If you want to use the trim box as cut path instead, select *Create cut path from trim box if no ink is found*.
- If you don’t want to use the trim box but want the task to end in error, leave *Create cut path from trim box if no ink is found* deselected.

| use the trim box as cut path | Leave the **Cut Path** field empty.

If you always want to use the trim box as cut path (usually for rectangular graphics like business cards), you don’t need to select *Create cut path from trim box if no ink is found* (as this option is only applicable when you want an ink as cut path as your first choice).

The task will tile the graphic that is inside the cut path. The tiles created will have a cut path in the same ink as the input graphic’s cut path (if the cut path is created from the trim box, this will be the *Cut* ink).

4. Graphics for large banners or billboards are often not supplied in the size at which they will be printed. In that case, you can *Scale* the tiles here.

  For example, enter 1000% if your input graphic is only 10% of the size you will need it to be.
5. If the input file (the non-tiled full graphic) has a bleed, the resulting tiles will show that bleed on their outside sides (except those surrounded by other tiles). You can limit that bleed on the tiles by setting a maximum bleed in **Bleed Clipping Mask**.

See this example of a full graphic with a bleed. The trim box is shown in red. This graphic will be tiled in 4 tiles:

![Example graphic showing bleed and trim box](image)

Here is the below left tile:

![Example tile with bleed and overlap](image)

In this example the bleed of the full graphic is similar to tile's overlap. But sometimes this bleed can be too large for a tile. That is why you can reduce that original bleed by setting a **Bleed Clipping Mask** on tile level.

**Note:** In i-cut Layout, in the **Tiling** dialog, this same function is named **Tile Bleed**.

6. If you want, you can add **SmartMarks** to the tiles. SmartMarks sets can be created in i-cut Layout or PackEdge. We advise to create them in i-cut Layout because it offers extra SmartMarks fields that are specific for tiling.

You can for example add SmartMarks containing glueing instructions in the overlap zones. **Below page** illustrates a typical example.

7. If you want to include overlaps (we mentioned a few types when introducing **Tiling** on page 6), select **Add Overlap** and click **Options**...

   a) Enter the **Overlap Distance**.

   b) You can choose to add an **Overlap on both sides** of each tile (using the overlap distance on both sides).

   In this case, you can not add a glue area.
c) If you want, also add a **Glue Area**.
This is an extra area that is not printed:

![Glue Area Diagram]

(d) Click **OK**.

**Attention:** Adding overlaps and glue areas increases the size of a tile (except if it is the last one on the grid). If you selected the tiling method **Use Fixed Tile Size** or **Automatic Tiling**, you should take these extra distances into account.

8. If you want a report detailing how the graphic was tiled, select **Create tiling report** and click the **Options...** button to define some options like the report **Size** and the used **Units**.

See (part of) an example of a report in below described example.

### 5.3.1 Example of SmartMarks for Tiles

**Concept**

This page shows how to add specific tiling related information into the glue or overlap areas of a tile. A typical example is to put the number of the neighbouring tile in this area. This helps those who apply the tiles.

See this (first page of a) tiling report where we added red squares to help you recognize the SmartMarks we added:
Notice that tile nr. 3 only has these marks in its glue area on the right. This is because the SmartMarks know that this tile has no glue area on any other side.

The next image zooms in on this mark (a view from I-cut Layout or PackEdge), both in contour and preview mode. Its shows an overlap of 10 mm and a glue area of 20 mm. Notice that the text appears mainly in the glue area, but also partially in the overlap. And to make sure it can always be read, a white rectangle is placed behind the text area:
How to Create this SmartMark Set

**Note:** In below instructions, we assume you have basic knowledge of both i-cut Layout and SmartMarks.

Follow below steps to create a set of SmartMarks as shown above.

1. You can use any of your files that may need tiling. Choose one and write down its size (V-H).
2. In i-cut Layout, create a **New** layout. Set the height and width to that of the file you choose to use. In our example we will create a layout of 4000 mm by 6000 mm. The name or place is not important. Click **OK**.
3. In **Graphics** mode, click on + to **Add a Graphic**. Select the file you chose in step1 and click **OK**.
4. In the list of **Graphics**, double click the thumbnail of the graphic.
5. While in **Graphic View** ( ), choose **Tools > Tiling**.
6. In the **Create Tiling Grid** dialog, specify a number of tiles: set 3 vertical and 3 horizontal and click **OK**.
7. In the **Tiling** dialog, select **Include Overlaps** and click on **Set Up Overlaps**. Set an **Overlap** of 10 mm, deselect **Overlap on both sides** and set a glue area of 20 mm. Set a **Tile Bleed** of 3 mm (though we will not use this in the SmartMark). Keep the dialog open.
8. Select a tile that is surrounded by other tiles on all 4 sides. The **Tiling** dialog shows the settings for this tile:
The SmartMark set that we will make needs to add marks on every overlap that the tile has, no matter what tile it is, so no matter of that tile has overlaps on one or two or more sides. To do this, we need to simulate a tile that has an overlap on all 4 sides.

9. Select the central tile (tile 5). In the Tiling dialog, make sure all 4 sides have an overlap. Based on above example, you therefore need to manually add a 1.0 mm overlap to also the left and bottom side:

We can now use this tile 5 to create our SmartMark set.

10. In the Tiling dialog, click on Create Tiles. The 9 tiles are created and added to the list of Graphics.

11. In the list of Graphics, scroll down to tile 5 and double click its thumbnail to open it. Then go to Production > SmartMarks. See how the SmartMarks dialog already shows many named objects, including 4 Overlap and 4 Glue objects.

12. We now start adding the marks. We start with the one on the right side, though the order is not important. In the SmartMarks dialog, select the named object Right Overlap, then click on + and choose to add a Text Mark.

Note: In this example, we will position the text inside the overlap area. You can also put it in the glue area, but often users avoid any printing the glue area.

13. In the Text Mark dialog,
   - As Text: keep Automatic Size selected and add the text as shown below. To insert the smart text fields, click on Smart Text, scroll down to the category i-cut layout, choose
tiling and in Options, choose to insert (in two times) [Right Tile] and [Current Tile number].

Click on Style, set the font size to 36 and align the text centered (H & V).

**Note:** This font size may be too large when you create tiles with smaller glue areas. This could cause the text to enter in the tile area itself. To avoid this, choose Smart Text and in the category Typographic choose Automatic Point Size and set a minimum and maximum size.

- In Inks, choose Darkest to avoid too much ink in this glue area. Also select Ignore Technical Inks.
- In Position, attach this text mark to the Right Overlap. Position it as shown below. Add a small offset versus the (left) end of the overlap area. Make sure you rotate the text 90 degrees.

Click Apply.

14. As good practice, rename the text mark to a logical name like "Text Mark - Name of Right Tile". To do this, select the text mark, then right-click it to enable renaming it.

15. Now add a white rectangle below this text box. Add the mark of category rectangle and manually drag and drop it onto the text mark. It should appear as shown here:

In the SmartMark Options - Rectangle dialog:
- in the Rectangle tab: select Fit (to object) to create a rectangle that adapts to the text box (both V and H). Set an Extra Height and Width of 1 mm.
• in the **Inks** tab, select **Knockout**.

• in the **Position** tab, position the rectangle central versus the text mark **object**.

![Image](image.png)

Click **Apply**.

16. Save your SmartMarks set with a logical name.
17. Now repeat the above steps 11 to 14 for the 3 other sides.
18. Save your SmartMarks set again (and close i-cut Layout).
19. In Automation Engine Pilot, test this SmartMarks set by using it in the **Create Tiles** task.

---

### 5.4 Submit to Gang Run

Use this task to **submit** a graphic file to a **substrate queue**, where it will be nested with other graphics to optimize substrate usage.

You can also submit ARD files (1-up CAD). Those will only be nested with other ARD files. This is useful for example when you want to fill a sheet with samples of shapes and send that merged resulted to a Kongsberg cutting table to cut these samples.

In this task, you can specify parameters (quantity, scale factor...) for nesting graphics based on their cut path.

You can send these file formats to this task:

- TIFF (CMYK or grayscale),
- JPEG (CMYK or grayscale, progressive JPEG compression is not supported),
- PSD (CMYK or grayscale),
- PDF (single page or multipage),
- Normalized PDF (single page or multipage),
- ARD files (1-up CAD).

Learn more in:

*General Tab* on page 53

*Back Tab* on page 56

*Substrate Tab* on page 56
5.4.1 General Tab

1. Define how many of each input graphic you want to print.
   You can either:
   - Enter the **Quantity** to print.
   - Select **Maximum Fill** if you want to print as many copies as will fit in one sheet.

2. Typically, graphics for large banners or billboards are not supplied in the size at which they
   will be printed out. In this case, you can **Scale** them up here.
   For example, enter 1000% if your graphic is only 10% of the size you will need it to be.

3. Define the **Rotation**. This is helpful when all or some of the graphics are only allowed a
   specific rotation.
   You can either:
   - Use the rotation that was **Set by the Substrate Queue**.
   - **Force a 0, 90, 180 or 270** degrees clockwise rotation. This choice overrules the substrate
     queue **nesting settings**.
   - **Force Allow Any Angle**: You don't care, you let the software decide. Also this choice
     overrules the substrate queue **nesting settings**.
   - Select a SmartName.

   **Note:** The **Gang Run Printing** view also shows the rotation for each entry in that queue.

4. Define what to use as **Cut Path**.
   **Note:** Whereas the **Prepare Graphics for Nesting** task allows you to generate a cut path if
   the graphic doesn't contain one, in the **Submit to Gang Run** task the cut path needs to be
   already available, either as an ink or by using the trim box as cut path.

<table>
<thead>
<tr>
<th>If you want to...</th>
<th>do the following...</th>
</tr>
</thead>
<tbody>
<tr>
<td>use an ink as cut path</td>
<td>1. Enter it in the <strong>Cut Path</strong> field.</td>
</tr>
</tbody>
</table>
### If you want to…

<table>
<thead>
<tr>
<th>do the following…</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Tip:</strong></td>
</tr>
<tr>
<td>• If the ink used for the cut path has different names in different input files (for example if you have customers from different countries using names like cut, découpe, taglio...), you can add several ink names, separated by a comma (, ). The order you use is important as the task will use the first ink it finds in the list as cut path!</td>
</tr>
<tr>
<td>• You can use wildcards and/or use SmartNames to define ink names. For example use Pantone* if you want to use any Pantone ink used in the file as a cut path.</td>
</tr>
</tbody>
</table>

**Tip:**

When building a workflow using an example file, click ![link](image) to select one or more of the inks from the example file.

If you always receive CMYK graphics with one spot color used for the cut path, select **Any SpotColor**.

2. Define what to do if the ink(s) you defined are not present in the file:

• If you want to use the trim box as cut path instead, select **Create cut path from CAD object or trim box if no ink is found**.

• If you don’t want to use the trim box but want the task to end in error, leave **Create cut path from CAD object or trim box if no ink is found** deselected.

| use the trim box as cut path | Leave the **Cut Path** field empty. If you always want to use the trim box as cut path (usually for rectangular graphics like business cards), you don’t need to select **Create cut path from CAD object or trim box if no ink is found** (as this option is only applicable when you want an ink as cut path as your first choice). |

| gang run ARD files | ARD files can only be submitted when the field **Cut Path from Ink** is left empty and when the option **Create cut path from CAD object or trim box** is selected. When this is not the case, the submit task ends in error with the message "Only default cutting path is allowed". |

5. If you want to set a due date (when the sheet with the nested graphics needs to be printed), select **Due Date** and choose when you want the due date (and time) to be.
• If you set a due date, the nesting server will always generate a nested layout that includes this graphic before your due date, even when the minimum fill has not been reached.

**Tip:** In your substrate queue settings, you can choose to generate the nested layout and send it to the output workflow a specific time before the due date. This allows time for printing, finishing, packing...

• If you don’t set a due date, the nesting server will only generate a nested layout when the minimum fill is reached.

| Select... | to... |  
|-----------|-------|---
| Immediate | generate a nested layout including your graphic immediately (this is the same as when you use *Print Now* in the *Gang Run Printing* view). |  
| Today     | set the due time to a certain time today (this is the default option).  
The nesting server will generate a nested layout including your graphic the specified amount of time before the due time. |  
| Tomorrow  | set the due date to a certain time tomorrow.  
The nesting server will generate a nested layout including your graphic the specified amount of time before the due time. |  
| In ... hours | set the due time a certain number of hours after the task is launched (you can set this with a SmartName). |  
| In ... days | set the due date a certain number of days after the task is launched (you can set this with a SmartName).  
You also need to set the due time. |  
| On        | set the due date to a specified date (and time) in the future. |  
| Job due date | set the due date to the due date specified in the *Job* setup (only valid when the graphics are stored in a *Job folder*). |  
| Select SmartName... | set the due date using a SmartName.  
This opens the *SmartNames* dialog where you can select your SmartName. |
5.4.2 Back Tab

- You can here specify what to print on the back of each graphic:
  - **No graphics**: choose this if you don't want any back side for the graphic.
  - **Even pages of multipage file**: choose this to print the odd pages of the file on the front, and the even pages on the back (this only works with multipage PDFs).
  - **Graphics from front side**: choose this to print the same graphic on the front and on the back.

  If your graphic has an irregular shape, you can select **Flip horizontally** so both graphics face the same direction and have the exact same contour for cutting.

  **Note**: If your graphic contains text, you shouldn't flip it as this will make the text unreadable.

  - **Other graphics**: choose this to print different graphics on the back, and **Browse** to these graphics.

  **Tip**: Give the graphics to print on the back a consistent name, and use SmartNames to retrieve them (for example, `[File]_back.[Extension of File]`).

5.4.3 Substrate Tab

1. Choose the **Substrate Queue** to send your graphics to.
Attention:
You should choose (one of) the substrate(s) you have defined as a substrate queue in the Gang Run Printing view.

If you submit your graphics to a non-existent substrate, a substrate queue will be created automatically for this substrate, but it won't be ready to process your graphics (so your graphics will be stuck in the queue until you configure it properly).

2. If this substrate has several sizes defined, you can choose the name of the Substrate Size to use.
   Leave this on Any to let Automation Engine find the best fit.

Attention: You here define the Name of the substrate size as defined in the setup of the substrate queue. Do not enter actual size information here.

5.4.4 Extra Tab

In this tab you can enter:

1. A Customer ID identifying the customer who ordered the graphics. Whether it is also used as a grouping key is one of the settings the Configure tool.

2. An Order ID: Every graphic file that you here submit with the same Order ID will show up as part of one same order in the Orders View. When working with the job concept, a logic value would be the SmartName [Order ID]. When you don’t enter a value, a system generated ID will be used.

3. Your desired values for the custom fields that you defined in the Configure tool.

You can use both customer ID and custom fields either as:

- labels in the reports from the nested layouts or the graphics,
- grouping keys to define which graphics can be nested together.

Example 1: Using Custom Fields as Labels

You get graphics orders from national and international customers. You are using different shipping companies for national and international orders.

When generating layouts, you want to indicate which graphics are for national and for international shipping, so the cutting operator can store the finished graphics in the right places.

1. In the Configure tool, define a Custom Field as Shipping.
   The corresponding Custom Field parameter in this task tab will now be called Shipping instead.

2. When launching the task on graphics from international orders, enter International in that parameter, otherwise enter National.

3. In your output workflow, choose to generate a layout PDF report in the Export Nested Layouts task.

4. In the layout report, you will see for each graphic name:
   - the label National if the copies of this graphic need to be shipped nationally,
• the label **International** if the copies of this graphic need to be shipped internationally.

5. If you open the report in an application that can show PDF layers (for example Adobe Reader), you will see that the report has a layer called **Shipping**.

If you hide the layers above this layer (so it only shows the layers **Base** and **Shipping**), you will see that:

• graphics that need to be shipped nationally are shown in the color of the **National** label (purple in the example below).

• graphics that need to be shipped internationally are shown in the color of the **International** label (green in the example below, cut paths are shown in red).
**Note:** The way the software chooses the 'random' color for the color patches is identical to how it is done when SmartMarks can use a "random color based on". A specific value (text?) will always return a same color based on that value.

**Example 2: Using Custom Fields as Grouping Keys**

You need to print different graphics on the same substrate, but some will need to be laminated and some not.

1. In the **Configure tool**, define a **Custom Field** as Lamination, and select **Use as grouping key**.

   The corresponding **Custom Field** parameter in this task tab will now be called **Lamination** instead.

2. When launching the task on graphics to be laminated, enter **Yes** in that parameter, otherwise enter **No**.

3. The nesting server will generate layouts with either:
   - only graphics to be laminated,
   - only graphics that shouldn't be laminated.

**Note:** If you generate a layout report, it will have the label **Yes** if the graphics on that layout need to be laminated, **No** if they don't.
See an elaborate example on using grouping keys and how they show up on reports in step 10 of the below example on Working with Grouping Keys on page 97.

5.5 Submit XML to Gang Run

5.5.1 Concept

The Submit XML to Gang Run task was designed to enable a business system (MIS, ERP...) full interaction with Automation Engine's Gang Run Printing mode.

The business system writes an XML file that serves as the input file of the Submit XML to Gang Run task. This XML is minimally a request for information about a nesting layout, possibly of a combination of different graphics (known as ganging). The MIS can then interpret the detailed response XML that it gets back (for example if the waste factor is acceptable). The MIS so decides if and when to re-send that XML file, but now using an extra option to also request links to the actual production files of the nested layout(s).

Let's have a closer look at the two main steps in this process:

- **Request for Nesting information**

  To assist with estimating the cost of producing specific jobs, the MIS asks Automation Engine's "Nesting Server" a detailed calculation of a layout of (a batch of) graphics on a specified Substrate Queue. The business system wants answers to these questions: How much material will be consumed with this substrate size and this quantity? What will be the waste factor?

The nesting server calculates and creates the requested layout files and also launches its assigned output workflow. The XML attribute production="FALSE" indicates that the response XML can happen immediately, because it does not have to wait for the production files to be created. The MIS interprets the Response XML.
• **Request to Nest and Produce**

When the XML contains the attribute `production="TRUE"`, the response XML will contain links to the production files. To enable this, the response XML is only sent when the workflow that created them has finished.

**Schema: Request for Production (production="TRUE")**

5.5.2 Using the Submit XML to Gang Run Task

The task ticket has no interface options. Everything needs to be defined in the input XML file. As usual, this task can also be triggered by the external system, for example by using a Folder Access Point.

Here is some background on what happens when the task is launched:

- The graphics ("<jobs>") listed in the XML enter into the specified Gang Run Substrate Queue(s).

- The "Nesting Server" starts calculating possible layouts. Depending on any specified grouping keys (learn more in Working with Substrate Queues) it is possible that other graphics that were already present in those substrate queues join the nested layout.

- The nesting server decides and creates an optimized layout.

- When all graphics ("<jobs>") for a certain NestingRequest have been placed, a NestingResponse output XML file is written (on the requested location defined in the input XML file).

- The MIS monitors the URL that it specified as ReturnURL and reads the NestingResponse XML as soon as it becomes available.
Note: As with the Submit to Gang Run task, the "<jobs>" listed in the XML can also refer to ARD files.

5.5.3 Examples of a Nesting Request XML

Example Nesting Request for 3 Jobs and 2 Substrates

In our example, your customer with ID 123456 asks you what it will cost to produce the following items, before noon of 5th of June of 2014:

- 10 items of AmericanCars_7 on Dibond-4mm, 1100x2400 mm
- 12 items of AmericanCars_2 on Dibond-4mm, 600x1100 mm
- 12 items of AmericanCars_1 on Forex-3mm, any size

Your company’s sales price will depend on how much it will cost yourself to produce these items. Surely, you want to produce them the most cost-effective way. To get this information, your MIS creates this XML file:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<NestingRequest Id="2014-84103">
  <ReturnURL>file://aeserver01/Data1/SnDResponses</ReturnURL>
  <CustomerId>123456</CustomerId>
  <Substrate>Dibond-4mm</Substrate>
  <SubstrateSize>1100x2400</SubstrateSize>
  <Quantity>10</Quantity>
  <DieShape>Cut</DieShape>
  <GroupingKey1>Finishing Methods</GroupingKey1>
  <GroupingKey4>2014-84103</GroupingKey4>
  <DueDate>2014-06-05T12:00:00+01:00</DueDate>
  <JOBS>
    - <JOB>
      <FileName>file://aeserver01/Data1/Jobs/SnDJob1/sndprepared/AmericanCars_7.pdf</FileName>
      <Quantity>10</Quantity>
    </JOB>
    - <JOB>
      <FileName>file://aeserver01/Data1/Jobs/SnDJob1/sndprepared/AmericanCars_2.pdf</FileName>
      <SubstrateSize>600x1100</SubstrateSize>
    </JOB>
    - <JOB>
      <FileName>file://aeserver01/Data1/Jobs/SnDJob1/sndprepared/AmericanCars_1.pdf</FileName>
      <Substrate>Forex-3mm</Substrate>
    </JOB>
  </JOBS>
</NestingRequest>
```

Note: This example shows that a NestingRequest can contain multiple jobs, possibly on multiple substrates, possibly on a specified, a non-specified or on multiple substrate sizes. This allows the XML to describe a production order containing multiple components as 1 NestingRequest.

This is a conceptual difference with the Submit to Gang Run task, which can also report back an XML file but always on 1 single job. The nature of the XML version of this task is the ability to calculate and feedback on batches of jobs.
Examples with front and back graphics:

**Attention:** In the 2 below examples, only one `<JOB>` element of the XML is shown:

- **Using the same graphic on the back but flipped:**

```
<JOB>
  <FileName>file://AEserver01/Data1/labels/111_kriek_belle.vue.pdf</FileName>
  <Quantity>3</Quantity>
  <Scale>100</Scale>
  <DieShape>BORDERS</DieShape>
  <Bleed>2.0</Bleed>
  <SameGraphic>1</SameGraphic>
  <FlipGraphic>1</FlipGraphic>
</JOB>
```

The same graphic is used a back-graphic, but will be flipped.

- **Using a specific graphic for the back and filling the sheet:**

```
<JOB>
  <FileName>file://AEserver01/Data1/labels/111_kriek_belle.vue.pdf</FileName>
  <Quantity>Maximum</Quantity>
  <Scale>100</Scale>
  <DieShape>BORDERS</DieShape>
  <Bleed>2.0</Bleed>
  <BackGraphic>file://AEserver01/Data1/labels/111_kriek_belle.vue_back.pdf</BackGraphic>
</JOB>
```

**Nesting Request for a Roll substrate**

There is no difference in the nesting request XML when the substrate is a roll.

### 5.5.4 XML documentation

#### Technical summary

The **Submit XML to Gang Run** task always outputs a return XML. Its path is specified by the `ReturnURL= element in the input NestingRequest XML.

Once the nesting request has been processed fully, the nesting server writes a `NestingResponse_[date].xml` file in the specified `ReturnURL` folder. This response can always be linked to the original request via the `id=` attribute. The (MIS) system that submitted the NestingRequest should monitor that `ReturnURL` folder and so pick up the response to its request.

When adding the attribute `production="TRUE"` to the input XML, the return XML will also include references to the actual production files.

#### Documentation of the XML for the Submit XML to Gang Run task

<table>
<thead>
<tr>
<th>XML Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NestingRequest</td>
<td>Mandatory root element (max 1). Many sub elements can be specified directly under this element and/or under a sub-element JOB. In the first case they serve as 'global' parameter, in the second as a job-specific one. A job parameter overrules a same type already specified 'globally'.</td>
</tr>
<tr>
<td>XML Element</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Id (mandatory)</td>
<td>The Id attribute of the element NestingRequest comes back in the NestingResponse output XML file. Based on this Id, the MIS knows which response matches which request.</td>
</tr>
<tr>
<td>ReturnURL</td>
<td>A file::-based URL that specifies the folder where the NestingResponse output XML file will be written once all jobs have been nested.</td>
</tr>
<tr>
<td>production=&quot;FALSE&quot;</td>
<td>Attribute of the element ReturnURL. Default setting.</td>
</tr>
<tr>
<td>production=&quot;TRUE&quot;</td>
<td>Attribute of the element ReturnURL. As described in Concept, this attribute means that the response XML will wait for the workflow to finish to also add the references to the files produced by the Export Nested Layouts task.</td>
</tr>
<tr>
<td>DueDate</td>
<td>To be defined in ISO 8601 format.</td>
</tr>
<tr>
<td>GroupingKey (n)</td>
<td>Add up to 4 grouping keys to help control groups of nested jobs. Tip: Using the NestingRequest Id as grouping key prevents these jobs from being nested with any job that is not part of this request.</td>
</tr>
<tr>
<td>JOBS (mandatory when more than 1 JOB element)</td>
<td>Contains one or more JOB element(s).</td>
</tr>
<tr>
<td>JOB (mandatory)</td>
<td>Contains the definition of a graphic. At least 1 JOB element must be present.</td>
</tr>
<tr>
<td>FileName (mandatory)</td>
<td>A file::-based URL indicating the one up graphics to be nested. A sub element of a JOB element.</td>
</tr>
<tr>
<td>Substrate</td>
<td>The name of the Substrate Queue that will calculate the layout(s). <strong>Attention:</strong> This element can be specified as global setting under the NestingRequest element. When it occurs under a JOB element, it is valid for that job only, and overrules any global values appearing under the NestingRequest element.</td>
</tr>
<tr>
<td>SubstrateSize</td>
<td>The name of the sheet or roll Size of the specified substrate queue. Same remark as for Substrate: Can be used as global or job element.</td>
</tr>
<tr>
<td>Quantity</td>
<td>Desired quantity for the specified job(s). Same remark as for Substrate: Can be used as global or job element.</td>
</tr>
</tbody>
</table>
| Rotation         | Rotation of the graphic. When not specified, the rotation setting specified in the substrate queue ("nesting settings") will be used. Possible values are:  
<p>|                  | • sub : use the rotation settings as defined by the substrate queue's nesting settings.                                                     |
|                  | • 0 or 90 or 180 or 270 : fixed rotation angle (clockwise). This will overrule the rotation setting specified in the substrate queue's nesting settings. |
|                  | • any : this allows any rotation for the graphic. This will overrule the rotation setting specified in the substrate queue's nesting settings.   |
| Scale            | To scale the graphic. Value unit is percentage. No scale = 100. Same remark as for Substrate: Can be used as global or job element.             |</p>
<table>
<thead>
<tr>
<th>XML Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DieShape</td>
<td>The die shape for the graphic. Allowed values: the name of a spot color, the keywords TRIMBOX, BORDERS or DEFAULT. TRIMBOX (or BORDERS) will use the trim box of the PDF graphic. DEFAULT will search for a CAD object inside the normalized PDF(STA) file; if no CAD object is found, the trim box will be used.</td>
</tr>
<tr>
<td>Bleed</td>
<td>The amount of bleed (in mm) allowed outside the die shape. If no bleed is specified, no bleed mask will be applied, unless the input is a PDFSTA file, in which case the bleed setting from the PDFSTA file will be used.</td>
</tr>
<tr>
<td>Page Number</td>
<td>Page selection of the multi-page PDF. Can only be used inside a JOB element.</td>
</tr>
<tr>
<td>SameGraphic</td>
<td>Value on 1 means that the (front) graphic defined in FileName will be used as back graphic. Can only be used inside a JOB element.</td>
</tr>
<tr>
<td>FlipGraphic</td>
<td>Only valid when value of SameGraphic is on 1. FlipGraphic on 1 defines that the back graphic will be flipped around the vertical axis (mirrored). Mind not to flip graphics that contain text. Can only be used inside a JOB element.</td>
</tr>
<tr>
<td>RectoVerso</td>
<td>When the input file is a multipage PDF, this will use the uneven pages as front pages and the even pages as back pages. This element overrules any use of the elements SameGraphic and FlipGraphic. Can only be used inside a JOB element.</td>
</tr>
<tr>
<td>BackGraphic</td>
<td>A file:·based URL indicating the graphic to be used as back graphic. Can only be used inside a JOB element.</td>
</tr>
</tbody>
</table>

5.5.5 Example of a Nesting Response XML

A NestingResponse XML file contains 2 main elements: Sheets and Jobs.

- The Sheets element describes which (part of) sheets were used to nest the given files. It refers to the files by a <Job id=""/> element.
- The Jobs element lists all jobs that were nested.

The below example shows these 3 sheet elements:

- The first sheet element states, in UsedSheets, that it will take 6 sheets of Forex-3mm 900x900 to deliver the wanted Quantity of 12 AmericanCars_1. The MIS can calculate that such a sheet places 2 of these graphics. The waste is 0,54 m2. The MIS can calculate that this is 11%.

- The second sheet element states, in UsedSheets, that it will take 12 sheets of Dibond-4mm 600x1100 to deliver the wanted Quantity of 12 AmericanCars2. The MIS can calculate that only 1 fits on such a sheet. The waste is 2,56 m2. The MIS can calculate that this is 32%.

- The third sheet element states, in UsedSheets, that it will take 2 sheets of Dibond-4mm 1100x2400 to deliver the wanted Quantity of 10 AmericanCars_7. The MIS can calculate that such a sheet places 5 of these graphics. The waste is 0,82 m2. The MIS can calculate that this is 16%.
<xml version="1.0" encoding="UTF-8">
  <NestingResponse id="2014-08-103" Created="2014-05-20T10:01:06.206+02:00">
    <Jobs>
      <Job id="419801a2-a743-4628-8119-3aba1f1418154" Created="2014-05-20T10:01:00.360+02:00">
        <Substrate id="Foreax-3mm" SubstrateSize="900x900">
          <Job id="ad01056b-ce2d-4fbf-a9de-aee1efa5655e">
            <Quantity>12</Quantity>
            <UsedSheets>60</UsedSheets>
            <UsedSurface m2="4.319400209051265"/>
            <TotalSurface m2="4.8599999999999999"/>
            <Waste m2="0.5450197901457349"/>
          </Job>
        </Substrate>
      </Job>
    </Jobs>
  </NestingResponse>
</xml>

Response XML for a Roll substrate
A response XML for roll substrates is the same as for sheet substrates, except for an extra `RollLength` element in each `Sheet` Element.

An example (only one `Sheet` element is unfolded):

```xml
<?xml version="1.0" encoding="UTF-8"?>
<Sheets>
  <Sheet id="0741c48f-9b23-4b0f-a967-30dd294dc802" Created="2015-06-09T12:05:31.795+02:00">
    <Workfloor id="9c1ede38-6a93-4d7a-99c6-3e94ce4f0998">  
      <Substrate id="Roll-1mm"/>
      <SubstrateSize id="Large"/>
      <SubstrateSize/>
    </Workfloor>
    <RollLength units="mm">3829.079</RollLength>
  </Sheet>
</Sheets>
```

### 5.5.6 Example of a Response XML after the Production Workflow

As mentioned earlier, the `NestingResponse` output XML can be extended to contain references to the actual production files.

**Note:** Production files are the files generated by the Export Nested Layouts task, as part of the Output Workflow of the Substrate Queue.

To do this, the MIS must add the attribute `production="TRUE"` to the `ReturnURL` element of the `NestingRequest` XML.

An example:

```xml
<?xml version="1.0"?>
<NestingRequest Id="21185">
  <ReturnURL production="TRUE">file://A1server01/Data1/SnDResponses</ReturnURL>
</NestingRequest>
```

As a result, each `/NestingResponse/Sheets/Sheet` element will be extended with a `FileList` element.

For each production file, there will be a `File` element attached to the `FileList` element.

**Example of a NestingResponse XML with references to production files:**
5.5.7 Using Workflow Parameters in this Process

Concept

Having these production files created on specific folders and with specific names will be important for the MIS to recognize and read them.

You can organize the production files that were created by a NestingRequest by using some workflow parameters that are available in the workflow that was launched on the generated Nested Layout files. See an example below where we picked some from the list of SmartNames.
Also the **Grouping Key**'s and custom fields used in the **Submit to gang Run** task can be used to help determine names of folders and files. Note that these only appear in the list of SmartNames after you first used that workflow ticket.

**Example**

- In *Pilot- Configure - Gang Run Printing*, we renamed the customfield 4 to **OrderId** and confirmed that we want to use this field as a grouping key.
- In the **Export Nested Layouts** ticket of the workflow that the substrate queue launches on the generated nested layouts, set up the **Output Folder** for the **Production Files** as in this example:

  ![Output Folder Configuration](image)

  **Note:** Learn more about these SmartNames in [this page about the ‘Export Nested Layouts’ task](#).

- As a result, the production files are organized in a folder structure that makes it easy for the MIS to find the files related to its initial **NestingRequest**:

  ![Folder Structure](image)

### 5.5.8 Workflow Extensions

**Concept**

Production files that are mentioned in the **NestingResponse output XML** can then be picked up again in an Automation Engine workflow for further processing. You can use the workflow control **Select Referenced File** to do this.

**Example workflow**

This enables a lot of extensions to the workflow. In this example we select the production files and ZIP them before sending them to an external system or partner. You can also choose to delete those files locally.
5.6 Export Nested Layouts

Use this task to create production files (a PDF print file and a cutting file) from your nested Gang Run Printing layouts.

Typically, the Export Nested Layouts task is the main step of your Gang Run Printing output workflow.

When working with Gang Run Printing:

1. You submit the graphics to be nested and printed through the input workflow.
2. The nesting server automatically nests the graphics on the substrate, and sends the resulting nested layouts to the output workflow that is linked to that substrate queue.
3. The output workflow generates production files from the nested layouts.

Note:

In a workflow, this task has five output pins: Print Files, Cutting Files, Inputs and Error. This makes it easy to redirect each type of file to the right place.

If you ask to Create Reports, the task will show an extra output pin for these report files.

Learn more in:
- Production Files Tab on page 71
- Sheet Tab on page 73
- Reports Tab on page 76

Managing the Nested Layout PDFPLA files

The nested layouts that the nesting server creates are written as PDFPLA files on a central folder. By default, this folder is ExampleJobContainer/SnDtemp. To optimize data traffic, we advise to have this folder on a disk away from your system drive. You can change this folder in the configure panel.
These PDFPLA files are not cleaned up automatically. We advise to build this cleanup into your output workflow. This is easily done as in below example, where you delete this PDFPLA input file that arrives on the "Inputs" output pin:

5.6.1 Production Files Tab

The task will output:

- **PDF print files** that are to be printed. When the input PDFPLA file only contains CAD data (you only submitted ARD files to the Gang Run), the option **Save Print Files** is skipped.

- **Cutting files**. These are used by cutting tables to print the printed sheet or roll. There are many formats available.

<table>
<thead>
<tr>
<th>Note:</th>
<th>Every resulting layout is a single page PDF print file, unless you are printing front and back (in which case the back graphic(s) will be on a second page).</th>
</tr>
</thead>
</table>

1. In **Save Cutting Files**, choose the cutting file format you want to generate (i-cut, HPGL, PDF, MFG, ACM or JDF for Kongsberg cutting tables or JDF for guillotine cutters).

**Tip:**

- If you have several cutting tables and you don’t know yet which one you will use, you can generate several file formats: choose **Select File Types...**, select all the file types you want to generate and click **OK**.

- If you want to check your cutting files for quality control, you can also output a PDF file and inspect it in Acrobat.

When generating an **ACM** file (for an Esko Kongsberg table), you can select **Print Side Up** to cut with the print side visible.

For all cutting file types except **JDF**, you can influence the knife or cutting head’s cutting sequence:

- Leave **Sequence lines** deselected if you want to use the original cut paths for the cutting lines (this way, common lines between graphics are only cut once, but every graphic is still cut individually).
• Select **Sequence lines** if you want the vertical and horizontal cutting lines to be concatenated so that the knife can cut longer lines at once (the cutting process is faster when the knife doesn't need to be lifted and brought down, and accelerate and decelerate many times).

If you selected **Sequence lines**, you can choose to **Cut Across Gaps** to cut longer lines at once even if there is a *gutter* between your graphics.

---

**Attention:**

When you export an *i-cut* file, make sure you have *i-cut marks* in your file as they are needed to define where to start cutting.

Other SmartMarks in your file will not be used for that purpose when working with an *i-cut* file.

Learn more about *i-cut* marks in the *i-cut Layout manual*.

---

**Note:** When the PDFPLA only contains CAD data (you only submitted ARD files to the Gang Run), then all structural lines get exported when choosing MFG or ACM (using the default ArtiosCAD sample making output). With all the other cutting formats, only the cut and crease lines get exported.

2. By default, all inks with the ink type "technical", including the ink used for the cut path, are considered structural inks, and are included in the cutting file but not in the print file.
If you want to include other inks from your graphic in the cutting file but not in the print file, enter their names in **Add Extra Structural Inks**.

When adding several inks, you should separate their names with a comma, for example Crease,Kiss-cut. You can also use wildcards (*) and SmartNames.

### 3. Output Name

If you want to give your output files a specific name, enter it in **Output Name**. By default, the output files have an automatically generated name containing unique identifiers. The default ticket proposes you to use smart workflow parameters. It also proposes to end the name with a unique ID like [**Short Task ID**]. This will prevent that output files overwrite each other (which can happen for example when several layouts are generated):

![Output Name: [wfp.substrate]_[wfp.substrate.size]_[wfp.substrate.size.quantity]_[Short Task ID] □](image)

Table: Meaning of SmartNames useful in this workflow:

<table>
<thead>
<tr>
<th>SmartName (wfp)</th>
<th>meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>substrate</td>
<td>This is the name of the substrate queue.</td>
</tr>
<tr>
<td>substrate.size</td>
<td>This is the name of the sheet (or roll) size.</td>
</tr>
<tr>
<td>substrate.size.quantity</td>
<td>This is the number of times the nested layout needs to be printed in order to get to the requested quantity of the submitted graphics.</td>
</tr>
<tr>
<td>substrate.size.type</td>
<td>This is the substrate type: Sheet (value P for plano) or Roll (value R).</td>
</tr>
<tr>
<td>customer.id</td>
<td>Mind that this one will only appear when you selected the option Use the Customer field as grouping key in the Configure tool. This is the one you used as ‘CustomerId’ in the Submit to Gang Run task (tab ‘Extra’).</td>
</tr>
</tbody>
</table>

**Note:** When multiple copies of the same layout need to be printed, the number of copies will automatically be added to the PDF file name.

### 4. Define where to save the production files and the report(s):

- If you want to save all output files in the same folder, select **Save all files in the same folder**, and browse to that **Output Folder**.
- Otherwise, deselect **Save all files in the same folder** and browse to your specific folder for the **Print Files**, **Cutting Files** and **Report Files**.

### 5.6.2 Sheet Tab

1. If you want, you can add **SmartMarks** to the nested layout (a SmartMarks set that you created in i-cut Layout or PackEdge).

You can for example use a SmartMarks set containing registration and/or cut marks. You can use SmartNames to enter your SmartMarks set’s name.
2. When the *nesting server* generates a nested layout, it places the layout on the sheet based on the settings of the substrate queue (sheet size and margins and nesting settings).

However, for printing and finishing purposes, you may want to reposition the nested layout on the sheet. In **Center Layout on Sheet** you can choose to center the layout either **Vertically**, **Horizontally**, or **Vertically and Horizontally**.

In the example below, the nested layout (left) has been centered both vertically and horizontally (right).

3. If you are working with substrate rolls or large sheets, you can choose to fit the final PDF that will be printed (the "sheet" file containing your nested layout) to one or both of the layout's dimensions.

This way, you can cut closer to the layout and have less substrate waste (for rolls) or save the remaining blank area as an offcut so you can reuse it (for large sheets).

For **Fit Sheet**, choose either:

- **To Layout Height** to fit the sheet height to the height of your nested layout.

- **To Layout Width** to fit the sheet width to the width of your nested layout.
• **To Layout Width & Height** to fit the sheet height and width to the height and width of your nested layout.

4. If you are printing front and back, define how the sheet will be turned in **Backing Up**: either **Work-and-Turn** or **Work-and-Tumble**.

This influences how the back graphics will be placed in your final PDF.

Front (at left) and back (at right) with **Work-and-Turn**:

Front and back with **Work-and-Tumble**:
5.6.3 Reports Tab

Reports can help production and finishing and can also serve as feedback to your business system.

XML Reports

- Select **For every layout** if you want an XML report describing the layout: its size and substrate, the graphics used, their quantity, cut files, URLs, etc.

- Select **For every graphic** if you want an XML report of each graphic used in that layout. This file describes the layout that this graphic was used in, how many times it was used, remaining quantity, etc.

**Example of XML Report of a Layout:**
Table: Clarification of above XML Report of a Layout

<table>
<thead>
<tr>
<th>Part of Element</th>
<th>Element</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>LAYOUT</td>
<td>TotalLength</td>
<td>In mm, the total length of the cut path.</td>
</tr>
<tr>
<td>LAYOUT</td>
<td>COPIES</td>
<td>The amount of times that his layout will need to be printed (as calculated by the nesting server).</td>
</tr>
<tr>
<td>LAYOUT</td>
<td>YIELD</td>
<td>The surface of used substrate material by this layout ((WIDTH * HEIGHT) * %)). The % of used material is shown in the PDF report (PDF).</td>
</tr>
<tr>
<td>JOB</td>
<td>NUMBERUP</td>
<td>Amount of times this &lt;JOB&gt; (graphic) is present on this layout.</td>
</tr>
</tbody>
</table>
## Automation Engine

<table>
<thead>
<tr>
<th>Part of Element</th>
<th>Element</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>JOB</td>
<td>REMAINING</td>
<td>Amount of times this &lt;JOB&gt; (graphic) still needs to be printed to produce the requested amount.</td>
</tr>
<tr>
<td>JOB</td>
<td>VSIZE and HSIZE</td>
<td>In mm, V and H size of surface that this &lt;JOB&gt; (graphic) occupies on this layout (bounding box), before any rotation. See also below illustration.</td>
</tr>
<tr>
<td>POSITION</td>
<td>V and H</td>
<td>In mm, V and H position of this &lt;JOB&gt; (graphic), calculated from the top left of he bounding box of the graphic to the top left of the layout. See this example (to keep the overview, all cut paths were based on the graphic’s trim box and not on ‘white background’):</td>
</tr>
</tbody>
</table>

![Diagram](image)

**Example of XML Report of a Single Graphic:**

```xml
<JOB>
<JOBNAME>
<JOB ExtId>
<JOB PageCount>
<JOB PageNr>
<JOB Rotation>
<JOB Substrate>
<JOB SubstrateSize>
```
Automation Engine

In addition to above table for the XML report for layout, this table shows specifics for the XML report of a single graphic:

**Table: Clarification of above XML Report of a Single Graphic**

<table>
<thead>
<tr>
<th>Part of Element</th>
<th>Element</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>JOB</td>
<td>Quantity</td>
<td>Remaining quantity for the nesting server to be put on other layouts to reach the requested amount for this graphic.</td>
</tr>
<tr>
<td>JOB</td>
<td>DieShape</td>
<td>Name of the cut path as defined in the <em>Submit to Gang Run</em> task.</td>
</tr>
<tr>
<td>JOB</td>
<td>Bleed</td>
<td>Maximum bleed for this graphic. This value is the one set as Bleed Clipping Mask in the nesting settings of the substrate queue.</td>
</tr>
<tr>
<td>Layouts</td>
<td>COPIES</td>
<td>The amount of times that his layout will need to be printed (as calculated by the nesting server).</td>
</tr>
<tr>
<td>Layouts</td>
<td>NUMBERUP</td>
<td>The amount of times this graphic is placed on this layout.</td>
</tr>
<tr>
<td>JOB</td>
<td>ExtId</td>
<td>Esko internal ID, not useful to non-Esko systems.</td>
</tr>
<tr>
<td>JOB</td>
<td>ForceOut</td>
<td>Y indicates that this layout was made after someone used the Print now button or after a deadline was reached. This element is not present when the layout was made because the nesting server reached a minimum fill.</td>
</tr>
<tr>
<td>JOB</td>
<td>Id</td>
<td>Esko internal ID, not useful to non-Esko systems.</td>
</tr>
<tr>
<td>JOB</td>
<td>PageCount</td>
<td>Total number of pages of the file in FileName.</td>
</tr>
<tr>
<td>JOB</td>
<td>PageNr</td>
<td>Number of page of file in FileName that was placed on this layout.</td>
</tr>
</tbody>
</table>
| JOB             | Rotation| The value indicates the rotation of this graphic as it was set in the *Submit to Gang Run* task. Sub means "Set by Substrate Queue", Any
PDF Report

A PDF report can be made **For every layout**. Such a report is typically used as instruction for the cutting table operator.

The template of this PDF is fixed, but you can define:

- the **Report Size** (A4 or Letter),
- the **Report Resolution** used for the graphics preview in the report (in **ppi**),
- the **Units** used in the report (**mm** or **inch**).

An example:
A PDF report shows following information:

- (on top) name of the layout and, when not 1, the amount of copies this layout needs to be printed.
- preview and size of the layout
- an area with text:
  - the first line mentions this information: name_of_substrate, name_of_substrate_size, # = amount_of_copies_this_layout_needs_to_be_printed, used_material %, wasted_material %
- URLs to the print and cut file(s).
- below right, a thumbnail showing the back graphics (蕊). When there are no back graphics, a icon is shown instead.
- a list of the graphics on this layout. Per graphic:
  - a patch with an arbitrary chosen color that is also used in the PDF layer "Dummy Colors" (learn more below),
  - name and size of the graphic file on the layout,
  - amount of times the graphic is placed on this layout / total requested amount. When maximum fill was chosen, only the amount in this layout is shown.
  - (when used) custom labels: their name and a patch with an arbitrary chosen color. See further below to learn how hiding PDF layers visualises which graphics have these labels.

Layers in Layout Reports

When graphics look much alike, it can be helpful to not see the preview but see dummy colors in stead. This makes it much easier to see which one is placed where. An example:

In these PDF reports, the preview of the graphics is a separate layer. You can see or hide layers by switching them on or off in Acrobat. You can also have your workflow create extra versions of this report by removing the preview layer for example. See such a workflow in this step of one of the examples at the end of this chapter.
Also the custom labels are shown in a separate layer (learn about custom labels in *Graphics Overview* on page 26). See this report example in Adobe Acrobat of a layout containing graphics with two custom labels, one for finishing (need for lamination) and one for shipping (type of transport). Only the **Base** layer and the **Transport** layer are shown:

By hiding some layers, it quickly becomes visible which graphics need their transport by air and which ones by road. This quick visual helps your colleagues in the packing and shipping.

See more examples of PDF reports in **Step 10** of the example on *Working with Grouping Keys* on page 97.
6. The Orders View

**Concept**

This view is targeted to customer service people or production floor managers who want to track orders that used the Gang Run mode and that then (typically after printing) will be produced on a Kongsberg cutting table.

**Note:** Learn more about using Automation Engine to manage your production on a Kongsberg table in the chapter *Kongsberg Production Workflow*.

All information in this view is updated live. It shows the status of all parts of a job Order that was processed by a Gang Run queue. It helps to monitor orders, especially those where multiple graphics need to be printed and cut.

This workflow and View is also valid when you submit ARD or MFG CAD files to a Gang Run queue. For sake of convenience, this view describes both graphics or ARD / MFG input files as 'One-Ups'.

- An Order is created when you submit file(s) to a Gang Run queue and you there specified an Order ID (in the tab 'Extra').
**Note:** An example: When you have 7 one-ups residing in a same Automation Engine Job and you first submit 5 of them and later the 2 remaining, then:

- Order ID = [Job Name]_[Short task ID] will result in 2 different orders (the task IDs are different).

- Order ID = [Job Name] will assemble all 7 one-ups in one same order.

To get feedback from the Kongsberg table in this Orders view, the substrate queue needs to, in its Export Nested Layouts task, create a cutting file of the type "JDF file for Kongsberg". Only this JDF file format can provide that feedback.

- **An Order is Done** when all ordered quantities are produced by one or multiple Kongsberg cutting tables, on one or multiple sheets. When later, extra one-ups enter the gang run queue that use that same Order ID, the state changes from (Done) back to (Partially Done).

- **An Order is removed from this list** when it was removed by the automatic clean up tool as configured for Device databases or when its total number of ordered copies changed to 0 (see below).

**Columns**

**Note:** As in many Views in the Pilot, you can

- re-arrange columns by dragging their title to a new position,
- sort them by clicking on a column title,
- right-click a column title and choose to Hide this one or any other. In the list that appears, select Select Columns... if you want to change or re-arrange multiple items.

- use filters and optionally save them as a Smart View.

- **Order ID**: ID defined in the Submit to Gang Run task that was launched on the one-ups in this order.

- **Customer Name**: Name of the customer of which the Customer ID was defined in the Submit to Gang Run task that was launched on the one-ups in this order.

- **Quantity**: Total of all ordered copies, as requested in the General tab of the Submit to Gang Run task, for all one-ups belonging to this order. For example: The Quantity of the order is 44 when it expects 44 fruit displays in total, of which 22 apple and 22 banana variants.

When a one-up is submitted to the Gang Run with quantity "Maximum Fill", its (part in the) ordered Quantity is counted or shown as "0" until the nesting has taken place. Only then, the amount of this one-up that was placed on the layout is added to the total amount of one-ups for the order.
**Attention:** When one or more one-ups are deleted from a Gang Run substrate queue, the order’s details in this view automatically adapt:

- The **Remaining Quantity** of those one-ups is subtracted from the total amount of ordered copies (**Quantity**).
- When the total ordered amount of copies for an order reaches zero, the order is removed.
- The one-ups that, by removing them from the Gang Run, are no longer ordered are also removed from the list of shown one-ups in the right pane.

- **In Gang Run**: Amount of ordered copies (of which the one-ups are still) present in a Gang Run queue.
- **In Output Workflow**: Amount of ordered copies (of which the one-ups are still) in the output workflow of a Gang Run queue.
- **Queued on Table**: Amount of ordered copies present in an iPC’s **My Jobs** list (still to be produced).
- **Done**: Amount of ordered copies produced on a Kongsberg table.

**Status**. General status of the order:

- **In Gang Run**: When all ordered copies are still in the Gang Run queue.
- **Done**: When all ordered copies are produced by one or multiple Kongsberg tables.
- **Partially Done**: This is the status shown in all other cases.
- **Created Date**: Date when order was created, i.e.: when the (first) one-up was submitted for this order.
- **Due Date**: Due date of the first submitted one-up for this order.
- **Finished Date**: Date when the order reached the status **Done**.

**The Right Pane: Pie Chart and Details**

- A colored pie chart shows the percentage of ordered copies in one of these states:
  - **In Gang Run**: Part that is still in the substrate queue, waiting to be merged.
  - **In Output Workflow**: Part that are in the workflow where print and cutting files are being made of the nested layout.
  - **Queued on Table**: Part that has its cutting file present as a job in a Kongsberg queue (iPC).
  - **Done**: Part that has been produced on the Kongsberg table. These numbers are updated each time you close or suspend a job in iPC.
- **Click** to see the amount and the names of all the **One-Ups** that were submitted and the amount and names of all the **Layouts** that substrate queues created for this order.
• "**In Gang Run**" shows the amount and the names of one-ups that are still in their substrate queue. Click ➔ to switch to the **Gang Run Printing view** and see the substrate queue where the related one-ups still are. When you here click on the name of the one-up, the same view opens but that one-up is also selected.

![In Gang Run](image1.png)

**Tip:** Use the arrow buttons ⬅️ ➔ ➙ ➙ in the Pilot tool bar to come back to this previous view.

• "**In Output Workflow**" shows the amount and the names of layouts that the gang run created but that have not yet arrived in a Kongsberg queue (iPC). Click ➔ or on a layout name to open the Tasks view and see only the corresponding tasks.

![In Output Workflow](image2.png)

• "**Queued on Table**" shows the amount and the names of layouts that are present in a Kongsberg queue (iPC). Click ➔ or on a layout name to open the **Table Queues view** and see that item selected.

![Queued on Table](image3.png)

**Seeing Overruns**

When the percentage of copies of one-ups in one part of the chart is larger than 100%, the part above 100% is shown in a slightly darker color. This visualises the size of the overrun.

See this example where 25 were ordered but 31 were produced ("Done" = 124%):

![Overruns](image4.png)

The amount of copies can become larger than 100% in these cases:

• When the gang run queue places more copies on a sheet than initially ordered (see the option **Maximum Overrun** in the Substrate Queue's Nesting Settings).

• When the Kongsberg table operator decides to cut more sheets than initially ordered.

• When more sheets are submitted to the Kongsberg table than initially ordered. This can happen when you overruled the number of copies in the **Submit to Kongsberg table** task.
For orders with a large number of copies, instead of leaving the option **Copies** to its default value **From Input File**, you could edit that field and for example submit half of the copies to table A and then launch the task again and submit the other half to table B.
7. Examples

These examples will help you build and understand some typical workflows.

- **This example** shows a basic workflow. It uses the Shuttle and shows how to use Public and Workflow parameters.
- **This example** shows how grouping keys help control the nesting. It uses a Folder Access Point and SmartNames to extract parameters from the names of the incoming files.
- **This example** shows how a workflow that is based on delivery of XML files. All required parameters are defined in the XMLs, including references to graphics.

7.1 Basic workflow using the Shuttle

**Summary**

In this example, a user will use the Shuttle to drop PDF files onto a "Submit to Gang Run" workflow. He will be asked to specify some options for that workflow. All the rest, including the output of nested layouts, will happen fully automatic.

This example specifically guides you in using workflow parameters that also become public parameters.

We assume you have been introduced to the tools we use in this example:

- The **Pilot**. To create the workflows and the substrate queue.
- The **Workflow Editor**.
- **Public parameters**.
- **Workflow parameters**.
- The **Shuttle**. This basic client tool enables to drag and drop files onto Automation Engine workflows.

7.1.1 Step 1 - Create the Substrate Queues

In the Pilot, create these 3 substrate queues:

- **PVC** (sheet)
- **AluComp 8 mm** (sheet)
- **Vinyl Roll 2 mm** (roll)

For each of these substrate queues,

- add minimally 1 substrate size
• in their **Nesting Settings**, set the minimum fill to 50%
• we do not have an output workflow yet, so leave that setting open. The icon of the substrate queue will show a warning sign, but we will ignore and solve this in step 5 of this example.

### 7.1.2 Step 2 - Create the Submit Workflow

The "submit" workflow is the workflow that will be launched on the files dropped on the Shuttle. Typically it prepares your PDFs and ends with the **Submit to Gang Run** task. The workflow that we here create is one of many possible variations. More important is the way we create the workflow parameters and public parameters.

Here is our example:

Follow these steps to create this workflow:

1. In the Pilot **Tickets** view, click ![open icon] to open the workflow editor and create a workflow. First simply add and connect the above steps. We will add details later, including the Router's 3 different output pins. Learn more about creating workflows [here](#).
2. Save your workflow as "**GRP_EXA1_Submit**".
3. Go to the menu **Advanced > Manage Workflow Parameters**. Click **Add** to add these 3 parameters, their value is at this point less important:

   ![Manage Workflow Parameters]

When finished, close this dialog. We can now use these new workflow parameters in the tickets.

4. The **Router** uses the new (workflow) SmartName `[wfp.Substrate]` that you just created. We route based on the name of the substrate queue, a decision taken by the Shuttle user.

   **Note:** You can use "**is equal to**" and enter their name in detail or use "**contains**" if this is easier.
5. Next we have 2 different **Prepare Graphics for Nesting** steps. This is only a suggestion; their only difference could for example be the way they create the cut path or the SmartMarks they add. In our example, the bottom step is used for roll-output.

6. Connect the **Router** output pins to their matching **Prepare Graphics for Nesting** step.

7. In the **Submit to Gang Run** step, add the other 2 workflow parameters for **Quantity** and **Scale** as shown here:

![Workflow parameter settings](image)

8. Workflow parameters automatically become public parameters. Follow the next steps to fine-tune the dialog in which they will be shown to the Shuttle user. Still in the workflow editor, go to the menu **Advanced > Manage Public Parameters**. The dialog will already list the workflow parameters. We only need to add some details.
9. Add the specifics as shown in above screen shot. Click **OK** to confirm.
10. Save your workflow and close the workflow editor.

### 7.1.3 Step 3 - Create the Output Workflow

The output workflow is the one launched by the substrate queue on the nested layouts that it created.

We will use below example. Just like with the 'Submit' workflow in Step 2, this is also just one of many possible variations. Again, the most important here is how we use workflow parameters.

Follow these steps to create such a workflow:
1. As you did in the previous step, just simply add the workflow's steps. We will add details and more transitions later. The "Remove Preview Layer" step at the end is actually an Optimize PDF document ticket that was renamed. Learn more about creating workflows here.

2. Save your workflow as "GRP_EXA1_Output".

3. Go to the menu Advanced > Manage Workflow Parameters. You will see 4 substrate related parameters that were added by default. This is because we used the Export Nested Layouts step in this workflow. In this example, we do not need any others. Close the dialog.

We will now use one of these workflow parameters in the tickets:

4. Much like in Step 2, the Router should use the (workflow) SmartName [wfp.substrate].

5. The 2 Export Nested Layouts only differ in the set of SmartMarks that they add. The Vinyl route uses the bottom one. Here is an example of how we use the workflow parameter [wfp.substrate] in output file names and folders:
6. The default output pins of both Export Nested Layouts steps (print, cut, report and input files) connect to an own end OK step that was renamed.

7. The Report files of the Export Nested Layouts steps continue to an Optimize PDF document step that creates an extra report in which the preview layer was removed:

Learn more about layout reports and their layers in here.

8. Save your workflow and close the workflow editor.

7.1.4 Step 4 - Add the Output Workflow to the Substrate Queue

Now complete the setup of the 3 substrate queues.

For each substrate queue, follow these steps:

1. double click the substrate queue to open and edit it.

2. in Output Workflow, select the workflow we created in Step 3 ("GRP_EXA1_Output").

3. click OK.

Here is an example of the PVC substrate queue:
7.1.5 Step 5 - Add the Submit Workflow to the Shuttle

Our Gang Run workflow is now ready for use from the Pilot.

When you set up a **Folder Access Point** and have it launch the submit workflow on the PDFs that you drop on that folder, you will not have an opportunity to specify important parameters like **Quantity** and **Substrate Queue**...

That is why we will here use the **Shuttle**. This tool enables us to have the user that drops the files answer some quick questions about the wanted parameters.

Follow these steps to set this up:

1. On your client, start the **Shuttle** and login with an Automation Engine user name and password.
2. Use the Shuttle's **View** menu to make sure you see the **Launch Panel**.
3. Click left-below on the + icon to create a launch pad for a workflow. Files that you drag and drop onto this pad will be launched on an Automation Engine workflow.
4. Select the workflow from the list. We here only need the submit workflow, the one we named "GRP_EXA1_Submit".
7.1.6 Step 6 - Test and Monitor

We test our workflow by dropping files on the Shuttle and defining the public parameters. All the rest happens fully automatic. Many of below steps only serve to illustrate how you can monitor this workflow in more detail.

1. On your local PC or any place you can access in your network, select files that will share the same production parameters. In our example we select 3 PDFs of which we need 10 prints on PVC sheet material.
2. Drag and drop these files together on the launch pad of our submit workflow.
3. A New Task dialog will appear asking you only to define the values for the public parameters:

   ![New Task dialog](image)

   
   Note: The way they appear in this dialog (name, order, default value) is as we set up in Step 2 in the Public parameters dialog.

4. Set the Quantity to 10, change the Substrate to PVC and click on Launch. You will see the tasks start in the Shuttle's Tasks view.
5. Just for information, we have a look at the Substrate Queue in the Pilot, which is waiting for more graphics to reach its minimum fill.

6. We go back to the Shuttle and now drag and drop the same 2 files + 2 other files on that workflow launch pad. This time we will want a different quantity and scale.
7. Because Shuttle recognizes files, it now first asks if you want to be shown the parameters again that you used the last time for that same file:

   ![Screenshot of Shuttle interface]

   We here click on Reuse.

8. A New Task dialog opens and shows the parameters that we used previously for the mentioned file. For the 4 files that we are dropping now, set the Quantity to 20, keep (reuse) the Substrate to PVC, and change the Scale to 200. Then click on Launch.

9. In the Pilot, we can monitor the PVC substrate queue. In our example, it reaches a minimum fill and so starts creating layouts on which it then launches the output workflow.

   **Tip:** You can force this layout-creation by using the Print Now button or by lowering the Minimum Fill percentage in the substrate queue's nesting settings.

10. When files are disappearing from the substrate queue, you can go to the Tasks view to monitor their output. Double click (one of) the launched output workflow(s) to open the workflow editor. This is how one of them looks in our example:
7.2 Working with Grouping Keys

As explained in *Working with Substrate Queues*, **Grouping Keys** can help control which graphics end up on the same layout.
In this example we will submit a group of PDFs to be nested but we will make sure there are no layouts with graphics from different customers nor layouts that combine graphics who need lamination and others that don't.

**Where can we get that information from a PDF (it's customer and whether it needs to be laminated)?**

There are multiple ways to get such information:

- When using the Shuttle, **a user could type in** the customer name (as a public parameter). This concept was illustrated in the previous example *Basic workflow using the Shuttle* on page 88.

- When the PDF resides in a Job Folder of an Automation Engine **Job**, then the customer and production parameters will be available as Job Parameter **SmartNames**. However, many customers using the Gang Run Printing mode choose to not use this **job concept**.

- These parameters can also be part of an **XML** that your MIS (business system) sends to Automation Engine. This is a powerful automated workflow that is illustrated in the *example of an XML driven workflow*.

- In this example, we will assume that those parameters are **part of the name of the PDF** input files that are dropped on a **Folder Access Point**. SmartNames will extract those fields from the file name and workflow tickets will then use those SmartNames as values.

**Tools used in this example**

Next to the Gang Run Printing view, we also make use of

- **String Extract SmartNames**
- A **Folder Access Point**
- **Building Workflows**.

### 7.2.1 Step 1 - Set up the Grouping Keys

Grouping keys can be created in the **Configure** panel. Learn details about this panel in *Gang Run Printing*.

For this example, follow these steps:

1. In the Pilot, go to **Tools > Configure**.
2. In the **Gang Run Printing** item of this Configure panel,
• name the **Custom Field 1** "Lamination Yes No" and select the option **Use as grouping key**

• select the option **Use the Customer field as grouping key**.

3. Save these settings (File > Save) and close the panel.

We will use these grouping keys in **Step 4**, when we pick them up in the Submit to Gang Run ticket.

### 7.2.2 Step 2 - Create the SmartNames

We will use these 4 files:

<table>
<thead>
<tr>
<th>PDF file name</th>
<th>Preview</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stevens_Dog_LamNo_36.pdf</td>
<td><img src="image" alt="Dog" /></td>
</tr>
<tr>
<td>Stevens_Entrance_LamYes_36.pdf</td>
<td><img src="image" alt="Entrance" /></td>
</tr>
<tr>
<td>ZooWorld_Flamingo_LamYes_90.pdf</td>
<td><img src="image" alt="Flamingo" /></td>
</tr>
<tr>
<td>ZooWorld_elephant_LamYes_20.pdf</td>
<td><img src="image" alt="Elephant" /></td>
</tr>
</tbody>
</table>
Notice how the PDF file names are constructed as CustomerName_DesignName_NeedForLamination_Quantity.pdf. We will use SmartNames to extract the needed information from the file name.

**Note:** To give you an idea when selecting your own graphics for this example, know that above samples are all approximately 200 mm by 200 mm.

### String Extract SmartNames

Follow these steps to create the required SmartNames (find more details in Use Separation Character):

1. In the Pilot's SmartNames view, click ![new smartname](new_smartname.png) to create a new SmartName.
2. Keep the default type String Extract.
3. Click Insert SmartName and in the category File Information, select the File and Insert it onto the canvas.
4. In Extract part, select use separation character.
5. In Separation Character, select the underscore character.
6. In String Part,
   a. select 1, click Save As... and name it Underscore_Sep_Part1.
   b. select 2, click Save As... and name it Underscore_Sep_Part2.
   c. select 3, click Save As... and name it Underscore_Sep_Part3.
   d. select 4, click Save As... and name it Underscore_Sep_Part4.
7. Now test the SmartName by, in the Extract part of... canvas, replacing [File] by typing an example of such a complete file name. Then change the String Part settings below to immediately see the result below in the dialog. There is no need to save at this point.
8. Click Cancel to close the dialog.
9. Back in the SmartNames view, see the resulting 4 SmartNames:

![SmartNames screenshot](smartnames_screenshot.png)

The screen shot also shows the description (added in the save-dialog). We also used the function Resolve all using and selected a valid example file. This allows to preview the Results in the extra column. Learn more on this in this page in the chapter on SmartNames.

### 7.2.3 Step 3 - Create the Substrate Queue

1. In the Pilot's Gang Run Printing view, go the menu File > New Substrate Queue...
2. Choose Sheet or Roll (not relevant for this example) and click OK.
3. Set the name to "PVC" and add a Size of 2500 x 1250 mm. In Nesting Settings..., set the Minimum fill to 45%. We will add the output workflow later in Step 7.
4. Click OK to confirm and close.

7.2.4 Step 4 - Create the Submit Workflow

The "submit" workflow is the workflow that will be launched on the files that are dropped onto the Folder Access Point. Typically it prepares your PDFs and ends with the Submit to Gang Run task.

The workflow that we create in this step is one of many possible variations. More important is the way we use the SmartNames.

Here is our example:

Follow these steps to create this workflow:

1. In the Pilot's Tickets view, click to open the workflow editor and create a workflow. First simply add and connect the above steps. We will add details later. Learn more about creating workflows here.
2. Save your workflow as "GRP_EXA2_Submit".
3. In the Prepare Graphics for Nesting step, use White Background to base our cut path on (with our example PDFs). Choose bleed and SmartMarks as you wish. We propose to write the resulting prepared files in a specific folder and with a specific name like [File]_Prep.

Note: Notice that this will add another suffix with an underscore to the input file name. This will not break the extraction logic that the SmartName uses.

4. In the Submit to Gang Run step, use the SmartNames:
   - in the tab General, add the SmartName that uses the extracted value for Quantity:
     ![General Tab](image)
   - in the tab Substrate, do not use a SmartName for the substrate but we pick PVC from the list.
Attention: This makes our submit workflow one that is only fit for this substrate queue. We will remember this in Step 5 when we define the name of the Folder Access Point that will start this workflow.

- in the tab Extra, add the SmartName that uses the extracted value for Customer ID and for the custom field Lamination Yes No (that we created in Step 1):

5. Save your workflow and close the workflow editor.

7.2.5 Step 5 - Create the Folder Access Point

Creating Folder Access Points is documented in detail in Creating or modifying a Folder Access Point.

Follow these steps to create what we need for this example:

1. In the Pilot, go to Tools > Access Points.
2. Click on , choose the type “Folder” and click OK.
3. Choose a logic name. Remember that the workflow it will launch is specific for the substrate queue "PVC”. We therefore advise to put that important fact in the name. For example: "GRP_EXA2_PVC”.
4. In Folder settings, choose the folder that will serve as Access Point for the incoming PDFs.
5. In Automatic Actions, define a processing folder and choose to Launch a Workflow and select "GRP_EXA2_Submit" (the one we created in Step 4).
6. Click **OK** to close the dialog.
7. Check the updated list of Access Points. Notice whether it is **Active** or not.

### 7.2.6 Step 6 - Create the Output workflow

We still need to define what needs to happen to the layouts that our Substrate Queue "PVC" will create.

We will create this basic example (a simplified version of the one we already used in *Example 1*):

Follow these steps to create this output workflow:

1. In the Pilot **Tickets** view, click ![button](image) to open the workflow editor and create a workflow. First simply add and connect the above steps. Learn more about creating workflows [here](#).
2. Save your workflow as "**GRP_EXA2_Output**".
3. In the **Export Nested Layouts** step, the only thing that is specific in this example is the smart definition of the **Output Name** of (all) our **Production File(s)**:
• the SmartName \[wfp\textunderscore substrate\] is a workflow parameter that is created automatically when you use this task type in a workflow. Click on \[\] and select it from the list of SmartNames, from its category "Workflow Parameters".

• the SmartName \[wfp\textunderscore Lamination Yes No\] is also one that we want to add here but it will not yet appear in the list of SmartNames. Remember that this one started as a custom field (and grouping key) that we created in \textit{Step 1}.

This custom field will only appear automatically as a SmartName Workflow Parameter after the nesting server passed this information a first time to this output workflow. This means that we first have to run the whole workflow once and can then come back to this part of the workflow to pick up that new SmartName.

At this time, we will continue without adding that extra SmartName \[wfp\textunderscore Lamination Yes No\], but we will come back to this ticket in \textit{Step 9}.

4. Much like in the previous example, the \textbf{Report} files continue to an \textbf{Optimize PDF document} step that creates an extra report in which the preview layer was removed (\textit{same settings as in previous example}).

5. Save the workflow.

7.2.7 \textbf{Step 7 - Add the Output Workflow to the Substrate Queue}

Although the output workflow is not final yet, we can already complete the setup of the substrate queue by adding its name to the queue's setup.

Follow these steps:

1. Double click the substrate queue "PVC" to open and edit it.
2. In \textit{Output Workflow}, select the workflow we created ("GRP\_EXA2\_Output").
3. Click \textbf{OK}.

7.2.8 \textbf{Step 8 - Test a first time}

We now do a test-run of our whole example workflow, starting with the Folder Access Point.

Follow these steps:

1. Drop the 4 files described in \textit{Step 2} on the folder that the \textbf{Folder Access Point} is monitoring.
2. Unless a minimum fill is not reached, all the rest should happen automatically.
   • The Pilot's \textbf{Tasks} view should show the 4 submit workflows, one for each input file.
• Shortly after, the **Gang Run Printing** view should show graphics entering our PVC substrate queue.

• If, after a few minutes, these jobs are not output automatically, you can force the layout creation by using the **Print Now** button or by lowering the **Minimum Fill** percentage in the substrate queue’s **Nesting Settings**.

• Once output workflows have started (we only need one), we can open one of them and add the extra workflow parameter SmartName (see **next step**).

### 7.2.9 Step 9 - Add the Custom Workflow Parameter to the Output Workflow

After this first execution of this workflow, we can pick up the extra workflow parameter and start using it inside the workflow tickets.

Follow these steps:

1. In the Pilot **Tasks** view, double-click (one of the ) finished output workflows to open them in the workflow editor.
2. Go to **Advanced > manage Workflow Parameters** and notice that we now have 2 extra workflow parameters:
3. Double click the step **Export Nested Layouts**. In the tab **Production Files**, in the field for **Output Name**, click on [ ] and see the new workflow parameter SmartNames. Select `[wfp.Lamination Yes No]` and insert it to the Output Name field. This should be your result: `Output Name: [Time]_{wfp.substrate}_{wfp.Lamination Yes No}`.

4. Close the **Export Nested Layouts** step and choose **Save Ticket as...** to save it with the same name it had before: "GRP_EXA2_Output". Confirm to replace the ticket with the same name.

The workflow is now complete.

### 7.2.10 Step 10 - Test Again and See Result of Grouping Keys

We now test the completed workflow again. We will also indicate the effect of the grouping keys that we used.

Follow these steps to test the whole workflow:

1. Remove any remaining graphics in the PVC substrate queue.
2. As we did in **Step 8**, again drop the 4 input PDF files onto the **Folder Access Point**.
3. Now monitor what is happening in the Pilot's Gang Run Printing view: we see all 4 graphics and also see that the custom fields created **Step 1** are appearing as extra columns:
4. In this example, the combination of the grouping keys and the queue's nesting settings makes that no layouts are decided (and no output workflows are launched).

To force this output, we select all 4 graphics and click the **Print Now** button 🖥️. The nesting server calculates and creates layouts.

5. In the Tasks view, we see the output workflows start, one for each created layout. Below we show the output PDF reports of each of these workflows. Open it by double clicking the Report PDF on the Report Files output pin:

**Layout 1: the "Dog" file**

This file used the grouping key values "Stevens" and "LamNo". No other file shared both of those, so the output depended solely on the queue's minimum fill percentage.

The resulting report shows us
• only "dog" signs, no other graphics, not even the "Entrance" file that was also for customer Stevens.

• a fill rate of 31.1 %, lower than the queue's minimum fill rate of 45%.

• below right, the report mentions the custom fields as 2 extra labels.

**Layout 2: the "Entrance" file**

This file used the grouping key values "Stevens" and "LamYes". There were 2 other graphics with "LamYes", but they had a different customer grouping key. So also here, the output depended solely on the queue's minimum fill percentage.

The resulting report shows us
• only "entrance" signs, no other graphics, not even the "dog" file that was also for customer Stevens.

• a fill rate of 39.42%, lower than the queue's minimum fill rate of 45%.

• below right, the report mentions the custom fields as 2 extra labels.

**Layout 3: the "Flamingo & Elephant" files**

These graphics used the grouping key values "ZooWorld and "LamYes". No other file shared both of those, so the output depended solely on the queue's minimum fill percentage.

The resulting report shows us
only the 2 graphics from customer ZooWorld, no other files. Although there was some extra space available on the sheet, we here do not see the "Entrance" job that also had the grouping key LamYes, because it did not have the same grouping key for customer ID.

• a fill rate of 38.57%, also lower than the queue's minimum fill rate of 45%.

• below right, the report mentions the custom fields as 2 extra labels.

### 7.3 Basic XML driven workflow

**Summary**

In this example, we simulate a business system that writes XML files onto a Folder Access Point. These simple XML files contain both a reference to a graphic file and parameters that describe what to do with it.

As explained in *Automating your Workflow Using XML* on page 29, the difference with using the *Submit XML to Gang Run* task is that these XML files are much simpler and also that we here stick to descriptions of 1 graphic file at the time.

In this example we assume you have read the previous examples and are thus familiar with these tools in the *Pilot*:

• creating a *Folder Access Point*

• working with *Substrate Queues*

• working with the *Workflow Editor* and *Workflow parameters*. 
7.3.1 Step 1 - Content of XML files

Incoming XML Files
The business system sends XML files that have a simple structure. They only list the name of the graphic file and some production parameters. They are all for the same substrate queue and size. In our example, we use 10 different XML files:

- 6 XML files represent large graphics of "American Cars" for the customer "ToyToy". Some graphics need scaling, some need different quantities. Here is one example:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<JOB>
  <Customer>ToyToy</Customer>
  <FileName>AmericanCars_1.pdf</FileName>
  <Quantity>10</Quantity>
  <Scale>50</Scale>
  <Substrate>AluComp 8 mm</Substrate>
  <Size>2500x1250</Size>
  <LaminationYesNo>LamNo</LaminationYesNo>
</JOB>
```

- 4 XML files represent smaller graphics for displays of movie characters for the customer "MovieWorld". Here is one example:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<JOB>
  <Customer>MovieWorld</Customer>
  <FileName>shrek.pdf</FileName>
  <Quantity>20</Quantity>
  <Scale>100</Scale>
  <Substrate>AluComp 8 mm</Substrate>
  <Size>2500x1250</Size>
  <LaminationYesNo>LamNo</LaminationYesNo>
</JOB>
```

They share a same value for the element "LaminationYesNo" that we will again use as a grouping key. So in our final output, we do allow these graphics to appear on a same layout. Learn more about using grouping keys in the previous example.

Graphic Files Data Structure
In this example, the business system does not know the exact place of the graphics data (full URL).

In our example, all graphics data are available as PDF in a folder named after the customer. This means that the file name and the customer name will suffice because we will use those as SmartNames to find the data.

7.3.2 Step 2 - Set up the Grouping Keys
This step is identical to the one in the previous example, except that in this example we will not use the Customer field as grouping key. So that option needs to be switched off.
7.3.3 Step 3 - Create the Substrate Queue

In this example, we only need one substrate queue. Learn how to create Substrate Queues here.

Very similar to how we did it in the previous examples, follow these steps:

1. Create a substrate queue of the type Sheet and name it "AluComp 8 mm".
2. Add a substrate size with the name "2500x1250" and give it those dimensions (in mm).
3. In Nesting Settings, set the minimum fill to 50%.
4. We do not have an output workflow yet, so leave that setting open. The icon of the substrate queue will show a warning sign, but we will ignore this for now and solve this in step 7. Click OK to close.

7.3.4 Step 4 - Create the Submit Workflow

This submit workflow is very similar to the one we created in step 4 of the previous example. We here only list the differences.

Follow these steps:

1. Create a workflow with these steps (you can already rename them as we did here):

   ![Diagram of workflow steps]

2. Save this workflow as "GRP_EXA3_Submit".
3. Still in the workflow editor, go to Advanced > Manage Workflow Parameters... In this menu,
   a. click Load Workflow Parameters from XML file... and select one of the XML files that the business system sends. In the dialog that asks to select an XML element, leave the field blank and click OK. This creates the [wfp... ] SmartNames that we will use later.
   b. select Use values from XML file and keep the default path [URL of Folder of Input]/[File].xml. This tells the workflow that the values for those workflow parameter SmartNames are those as specified in the input file of the workflow.

This should be the result:
Learn more on using XML when working with workflow parameters in *Using Workflow Parameters*.

Close the dialog and save your workflow again.

**4. In the Select File step,**

- select *Select other Files*

- in *name matches*, enter the SmartName `[wfp.FileName]`

- in *Look in Folder*, enter a path that uses the SmartName `[wfp.Customer]`.

**5. The Router serves to find if the graphic file already has a Separation Name that can serve as cut path.**
6. The two **Prepare Graphics for Nesting** steps differ in their setting **Create Cut Path from**:
   - one uses an **Ink Name**:
     
     ![Create Cut Path from: Ink](image)
     
     Ink Name: `cut*, *coupe*, taglio`
     
   - the other one creates the path from the **White Background**:
     
     ![Create Cut Path from: White Background](image)

Choose bleed and SmartMarks as you wish. We propose to write the resulting prepared files in a specific folder and with a specific name like `[File]_Prep`.

7. In the **Submit to Gang Run** step, use the SmartNames:
   - in the tab **General**, use the SmartName `[wfp.Quantity]` to define the **Quantity**.
   - in the tab **Substrate**, use the SmartNames `[wfp.Substrate]` and `[wfp.Size]` to define the **Substrate** and **Substrate Size**.
   - in the tab **Extra**, use the SmartName `[wfp.LaminationYesNo]` to define the value for the custom field **Lamination Yes No**.

8. Click **OK** to close the step. Save your workflow and close the workflow editor.

### 7.3.5 Step 5 - Create Folder Access Point

Setting up this Access Point is very similar to the one we set up *in the previous example*. Follow those same steps and mind these 2 differences:

- The behavior of (the workflow started by) this Access Point is decided by the values in the XML files, so its name can be generic, for example "GRP_HOT_SINGLE_XML".
• In **Automatic Actions**, the workflow to launch is "GRP_EXA3_Submit" (the one we created in the previous step).

7.3.6 Step 6 - Create the Output Workflow

The output workflow can be identical to *the one we used in the previous example*. It was named "GRP_EXA2_Output".

The only SmartName we use in this output workflow is to help name the output **production files**.

7.3.7 Step 7 - Add the Output Workflow to the Substrate Queue

Follow these steps:

1. Double click the substrate queue **AluComp 3.4 mm** to open and edit it.
2. In **Output Workflow**, select the workflow "GRP_EXA2_Output" (yes the same one as in our previous example).
3. Click **OK**.

7.3.8 Step 8 - Test

All we need to do to test this workflow is dropping the XML files onto the Folder Access Point. For every XML, one graphic file should appear in the substrate queue.

Depending on the size of the graphics and the nesting settings, some or all graphics might automatically end up on a layout and continue to the output workflow.

Here is an example result of one of the output workflows:

![Diagram of workflow]

The **Reports** output pin shows us a link to the report PDF shown below (and also lists the other report data in XML form).
Automation Engine

154935_AluComp 8 mm

[Image of a patterned design with red cars and characters]

AluComp 8 mm - 2500x1250 # = 1 66.42% 34.58%
\ewe11d1b6WDE-AE-DATA/GRP_output154935_AluComp 8 mm.pdf
\ewe11d1b6WDE-AE-DATA/GRP_output154935_AluComp 8 mm_CUT.pdf
\ewe11d1b6WDE-AE-DATA/GRP_output154935_AluComp 8 mm_cut.pdf

<table>
<thead>
<tr>
<th>File Name</th>
<th>Width</th>
<th>Height</th>
<th>Pages</th>
<th>Laminate</th>
</tr>
</thead>
<tbody>
<tr>
<td>shrek_Prepp.pdf</td>
<td>336.33</td>
<td>207.71</td>
<td>14/20</td>
<td>LamNo</td>
</tr>
<tr>
<td>hiccup_Prepp.pdf</td>
<td>334.78</td>
<td>98.6</td>
<td>6/6</td>
<td>LamNo</td>
</tr>
<tr>
<td>AmericanCars_2_Prepp.pdf</td>
<td>489.21</td>
<td>883.2</td>
<td>5/7</td>
<td>LamNo</td>
</tr>
</tbody>
</table>