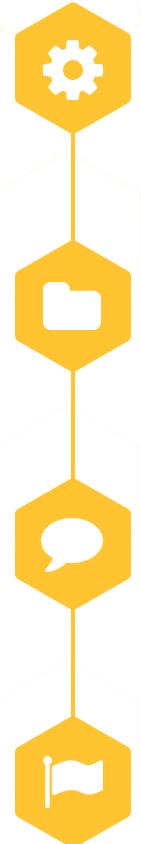




PrintVis

Partner Meeting 2022

Table Of Contents



Who are we?

Agenda

Get started

Hands on

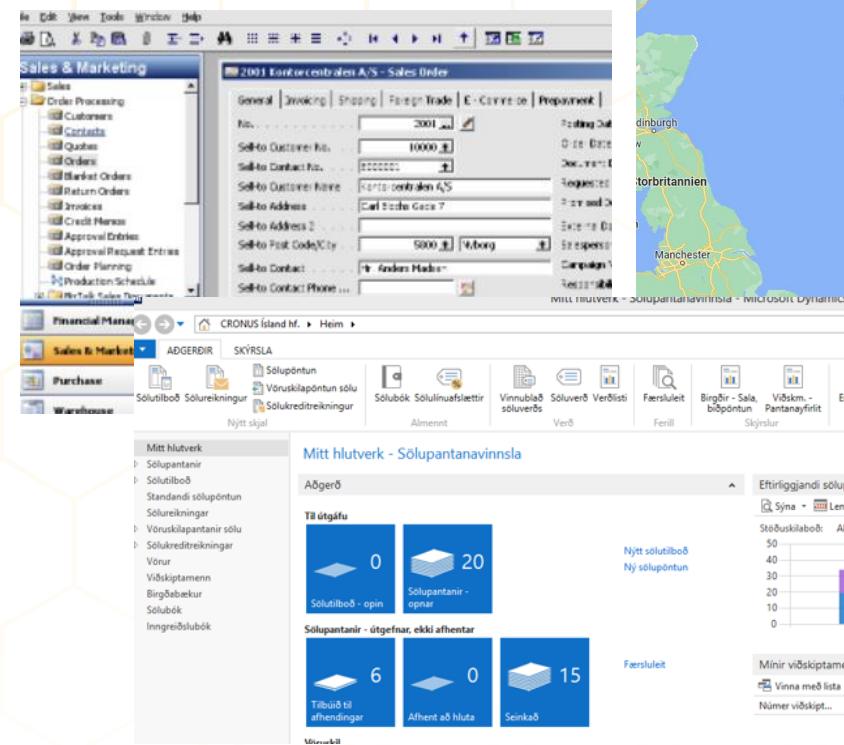
Who are we?

Rikard Hult

Employed at a printing company 2005-2012
 At a PrintVis Partner in Sweden 2012-2016
 PrintVis since 2016-



Interest beside PrintVis, Running and Orienteering



The screenshot shows the Microsoft Dynamics CRM interface. The main window displays a sales order for '2001 Konferenscentralen A/S'. The details include:

- General tab: Order No. 2001, Customer No. 10000, Contact Name Konsertcentralen A/S.
- Shipping tab: Shipping Date 01/01/2013, Document Type Requested.
- Prepayment tab: Payment Method Cash on Delivery.
- Customer tab: Address Carl Berné Gata 7, Post Code 5000, City Nyborg, Contact Person Mr. Anders Madsen, Contact Phone +45 26 12 34 56.

Below the main window, there are several tabs for other modules: Financial Management, Sales & Marketing, Purchase, and Warehouse. The Sales & Marketing tab is currently selected. On the left, a sidebar shows navigation links for Sales & Marketing, Financial Management, Purchase, and Warehouse. The bottom left corner shows a summary of current activity counts.



Livelox - Sverigepremiären, lång, H40

Who are we?

Catalin Dumitru

Graduated as Software developer 2018

Started at PrintVis in 2019 working mostly with Infrastructure and Development (DevOps)

As personal interests I like sports, traveling and learning about the outer space



Who are we?

Rainbow

Wye

Telos

Arqui Consult

Arqui Consult

Arqui Consult

Kodak

Sabre

Frans Selles

Mark Angeli

Erik Sørensen

André Botelho

Rui Pires

Pedro Coelho

René Mølsted

Srdan Grandic

Netherlands

US - Atlanta

Denmark

Portugal

Portugal

Portugal

Denmark/(US/CAN)

Serbia /(US/CAN)

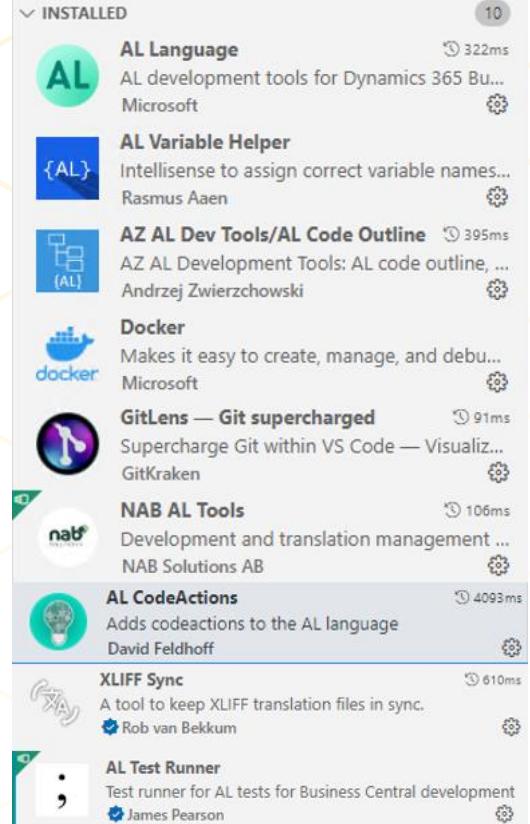
- Questions
 - Spending most time on what version of BC
 - Cloud or onprem
 - Experience with Navision/Nav/BC from what version?
 - Devops or Github? Or something else?
 - Experience from printing industry

Monday				Tuesday			
8.30-09:00		Welcome & round of presentations (30)				ShopFloor Metro	
09:00-10:30	sessions	Presentation of group		9.00-10.30	sessions	JobTicket	
		PrintVis Dev Environment				SharePoint Folders	
		VS Code extensions					
10.30-11.00	break			10.30-11.00	break		
11.00-12.30	sessions	Event Request				Invoice Template	
		Demo Data for Developers		11.00-12.30	sessions	Web2pv	
		Table Structure part 1					
12.30-13.15	Lunch			12.30-13.15	Lunch		
13.15-15.00	Sessions	Table Structure part 2		13:15-14:15		Create case with code	
		Status Code Events					
		Calculation Formulas					
		Unit Conversion					
		Userfields					
		Planning Board		14:15-15:00		Cloud Migration	
15.00-15.30	Break			15.00-15.30	Break		
15.30-16.30	Sessions	Upgrades (with all)		15.30-16.00	Sessions	Wrap-up with all	

Getting started

- [Download](https://github.com/printvis/PartnerMeeting2022) resources for this lab: <https://github.com/printvis/PartnerMeeting2022>
- Docker containers with traefik
 - Launch files
 - Script used
 - VS Code Extensions ->
- Create extension depending on PrintVis
 - Add something and publish
 - Add dependencies to PrintVis
- Use prefix on all, objects, fields,names
- If ID is used, put in 50000..99999 (even if its not mandatory)

VS code Extensions



Good looking names on variables

New objects

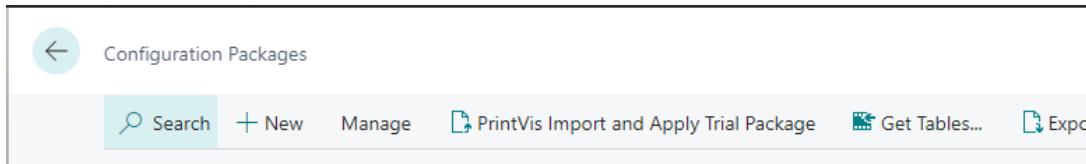
Makes working with Git easy

Translation tool

Run tests in VS Code

Demo Data for developers

- Check so Cronus data is loaded. (data that has customers)
- Apply PrintVis Configuration package 5 + 5 min
- While we waiting, PrintVis Demo



Goal:
PrintVis Demo data in one company

Event Request

- We are extending PrintVis all the time with new events
- New version is shipped every week
- Use the word document to request
- We approving or get back with an alternitive solution asap.
- If other changes needed create support ticket and explain why. (we will fix or guide you to solution)
 - Local procedurs => non local etc..
 - Adding procedurs to printvis objects
 - Access to global variables

Bug/issue Reporting

- If you find something that could be a potential bug
- Report it to support365.printvis.com
- Provide as much data as possible, debugger screenshots etc.. Solution proposals.
- If critical, new release ASAP
- If not critical next Tuesday/wendesday
- Support is monitored all business hours (europe) by developer

Demo flow – need to know level

- Initial setup
- Estimate
- Quote
- Order
- Material requirement
- Planning
- Planning board
- Shop floor
- Time registration
- Material registration
- Shipment
- Invoicing

Goal:
To be able to follow the flow

PrintVis User

- Change profile to PrintVis Admin
- PrintVis Licensed User
 - Add the needed user
 - You (admin)
 - Connect to BC user
- PrintVis Users
 - New users, select Admin
 - Enable: Coordinator, Estimator, Salesperson, Capacity Resource, Assign personal Journal, Allow Archiving, Allow User Field Setup
 - Planning Board: Full Access

User Setup

ADMIN - Admin

General

User ID	ADMIN	Coordinator	On
Business Central User Name	ADMIN	Production Employee	Off
Status	Enabled	Estimator	On
User Type	Full User	Salesperson	On
Contact No.		Salesperson	
Name	Admin	Capacity Resource	On
E-mail	admin@printvis.com	Create as Employee	Off

Permissions

Allow to edit Customer Accounting Fields	On	Invoicing
Default Salesperson	ADMIN	Department Location
Default Coordinator	ADMIN	Run Report on Login
Job Costing Journal	ADMIN	Item LookUp Show Inventory
Job Ticket Number Filter		Assign Personal Journal

Preferences

Allow User Field Setup	<input checked="" type="checkbox"/>	Deadline not Editable	Off
Planning Board	Full Access	Status Code new Request	
		Status Code new Quote	
		Status Code new Order	
		Default Order Type	
		Timeline	Nothing

PrintVis Licensed User List

PrintVis User ID ↑	User Type	Status	Business Central User Name
ALA	Full User	Enabled	-
ALA1	Full User	Enabled	-
BAS	Full User	Enabled	-
BOB	Full User	Enabled	-
CHRIS	Full User	Enabled	-

PrintVis General Setup

PrintVis General Setup

[User Fields](#) [Create PrintVis Standard Bitmaps](#)

Logistics

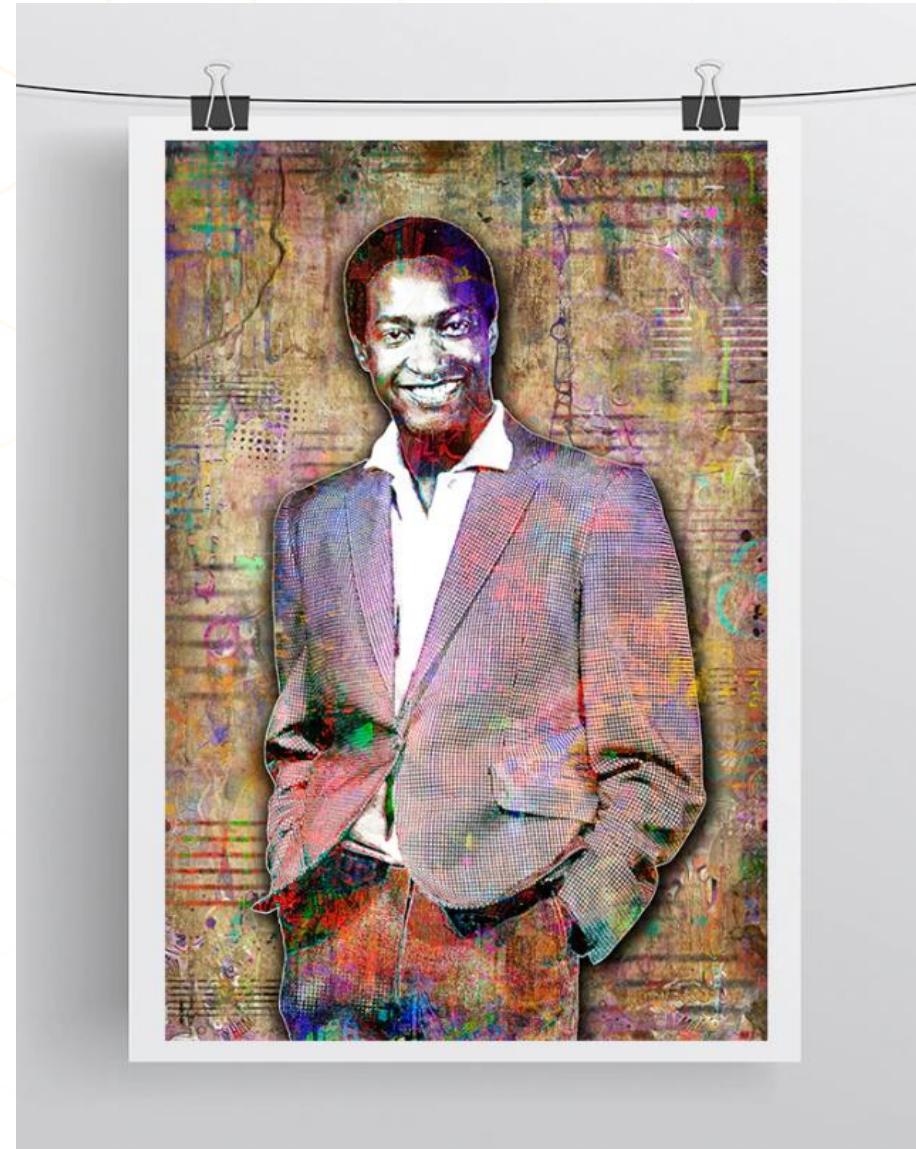
Shipment

Copy Shipments	Always
Default delivery address	Sell-To Address
Receipt Merge Code	
Way Bill Merge Code	
Package Labels Merge Code	
Package Label Text1	
Package Label Text2	
Package Label Text3	
Package Label Text4	
Package Label Text5	
Package Label Text6	

Item Journal Template Name	ITEM
Item Journal Name	PV
Warehouse Journal Template	ITEM
Warehouse Journal Batch	DEFAULT
Create Shipment Item	<input checked="" type="checkbox"/>
Shipment Item Template	ITEM01
Shipment Item No. Series	SHIP1
Tax Area Code Mandatory on Shipment	<input checked="" type="checkbox"/>
Case Shipment Method	No Sales Order
Shipment Sales Order Removal	On Ship

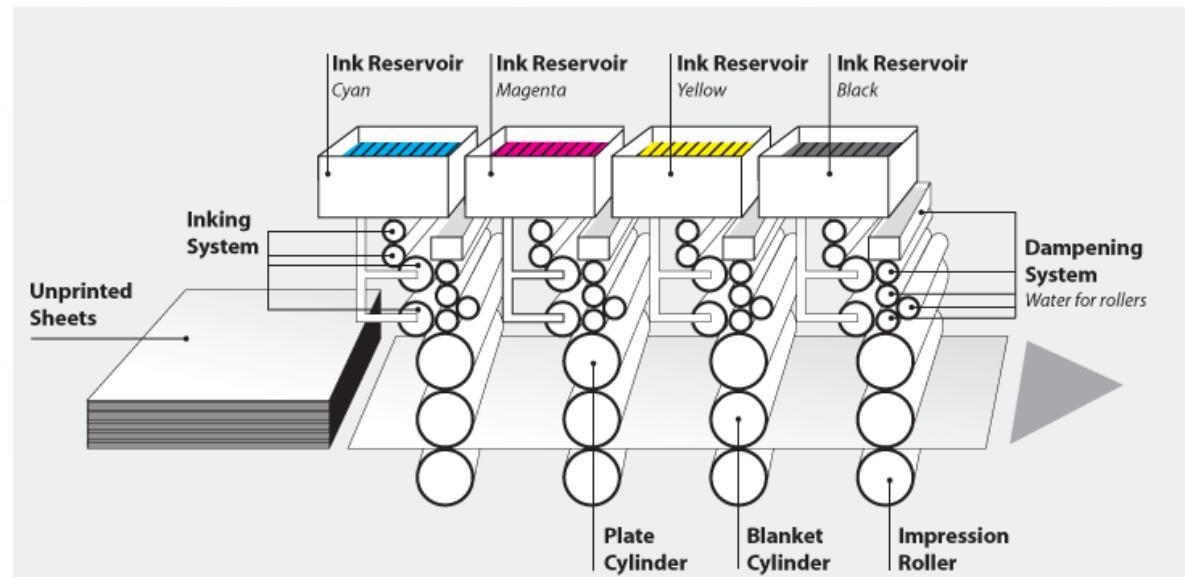
The Product

- Poster
- Size
 - 10 inches width
 - 15 inches height
- Material paper
 - White Gloss 35" x 23"
 - 68102104
- 4+0 colors



The Process

- Prepress 1 h
- 4 plates
- Ink
- Paper
- Trimming

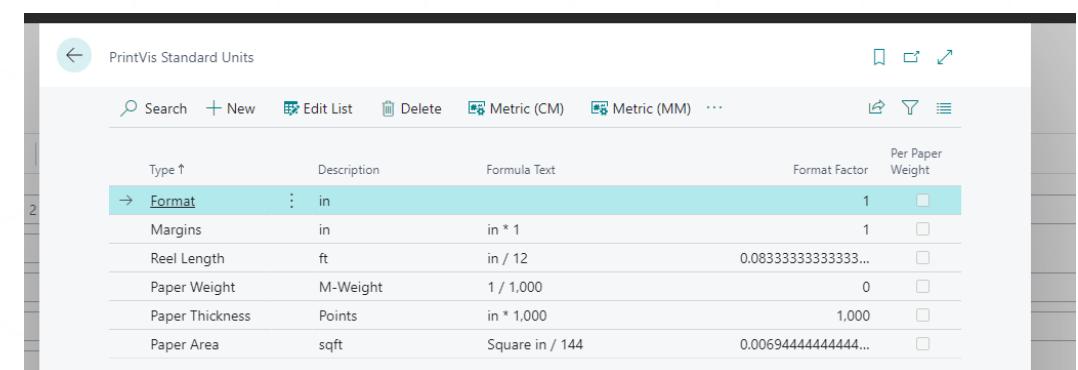


PrintVis Case

- Case management
- New Case
 - Select Customer
 - Order type PRINTED
 - Product Group POSTER
- Job line
 - Quantity 10000
 - Pages 1
 - Format Code 10x15
 - Colors front 4

This is like the sales order
 Describes the business

Format Code 10x15
 Width x Length PrintVis General Setup
 What unit? See PrintVis Standard Units



The screenshot shows a table titled "PrintVis Standard Units" with the following data:

Type ↑	Description	Formula Text	Format Factor	Per Paper Weight
→ Format	: in		1	<input type="checkbox"/>
Margins	in	in * 1	1	<input type="checkbox"/>
Reel Length	ft	in / 12	0.0833333333333333...	<input type="checkbox"/>
Paper Weight	M-Weight	1 / 1,000	0	<input type="checkbox"/>
Paper Thickness	Points	in * 1,000	1,000	<input type="checkbox"/>
Paper Area	sqft	Square in / 144	0.00694444444444...	<input type="checkbox"/>

PrintVis Job Card

- Job line
 - Click Job Card
 - Select Paper 68102104
 - Select Machine SHEETOFFSET
 - Check specification

Quantity

End Quantity Number of Sheet Sets Show less

Manual Final Quantity Off Total

Scrap

Gross Printed Sheets

Manual Printing Sheets Off

Imposition

A diagram illustrating the imposition layout. It consists of a 2x2 grid of four white rectangles, each with a black border. The grid is defined by thick yellow lines. A vertical purple line is positioned to the left of the first column, and a horizontal red line is positioned above the first row.

PrintVis Estimate

- DTP operation
 - Add hours on a line
 - Sheet Offset Printer
 - Notice all processes

Estimate contains all time and material for the job

Unit	Text		Info		Sheet No.	Process	Total Time
3000		DTP operations	+	-			4.00
4000		CTP Plate 23.62x31.5 M	+	1	1		0.50
→ 5060	:	Sheet Offset Printer	+	1	2		1.39
P_5060		Substrate Sheet Offset Printer	+	1			
7450		Trimming Pack	+	1	1		0.68
7450		Trimming Pack	+	1	1		0.68
9000		Shipping	+	-			0.17
			+				

PrintVis Quote

- Case card
 - Change status to Quote
 - Report -> Print/Send -> Quote

Job line is changed from
Request to Quote

Quote			
Station Road, 21 Adatum Corporation Cambridge, CB1 2FB Great Britain			
			
Quote No.	Quote Date	Your Reference	Responsible
Q000009	den 11 april 2022		Alex Accountant
we thank you for your inquiry and hereby take pleasure in offering you our best Price and Services for the following job specification: Job Name: Test			
Quantity	Description	Unit Price	Total price in GBP
10000	10.000 magazines, 32 pages in size 8.5x11 and 4+4 colors Pages: Text 32 Cover 2 Size: Text 8.5x11 Cover 8.5x11 Paper: Text Satin Coated Blauer Weight:70.00 Gloss EuroArt Weight:70.00 Offset: SHEETOFFSET SHEETOFFSET Colors: Text 4/4 Cover 4/0 Finishing: Trimming	0,05	500,00
we sincerely hope that our quotation will meet your expectation and look forward to serving you in this matter. Best Regards, Jim Olive			

```
Add-FontsToBCCContainer -containerName
PV20G -path c:\windows\fonts\*.ttf
```

PrintVis Order

- Case card
 - Set a requested delivery date on job
 - Change status to PLAN

New job line created Order

The screenshot shows the PrintVis software interface with a hexagonal background pattern. At the top, there's a navigation bar with tabs: Jobs, Manage (which is underlined), Job Processing, User Fields, Job, and Fewer options. Below the navigation bar are several icons: New Line, Delete Line, New Ver., Job Card, Estimating, Specifications, Shipments, and a grid icon. The main area features a table with the following data:

	Job ↓	Version ↓	Acti...	Type	Ordered Quantity	Pages	Format Code	Colors Front	Col...
	1		2	<input checked="" type="checkbox"/>	Order	10000	32	8.5X11	4
→	1	:	1	<input checked="" type="checkbox"/>	Quote	10000	32	8.5X11	4

PrintVis Planning

- Job Line
 - Planning
 - Plan from shipment date

Edit - Job Planning - M000010

Set Limits

Requested Shipment Date

Earliest Start Date

Method

Rough Planning

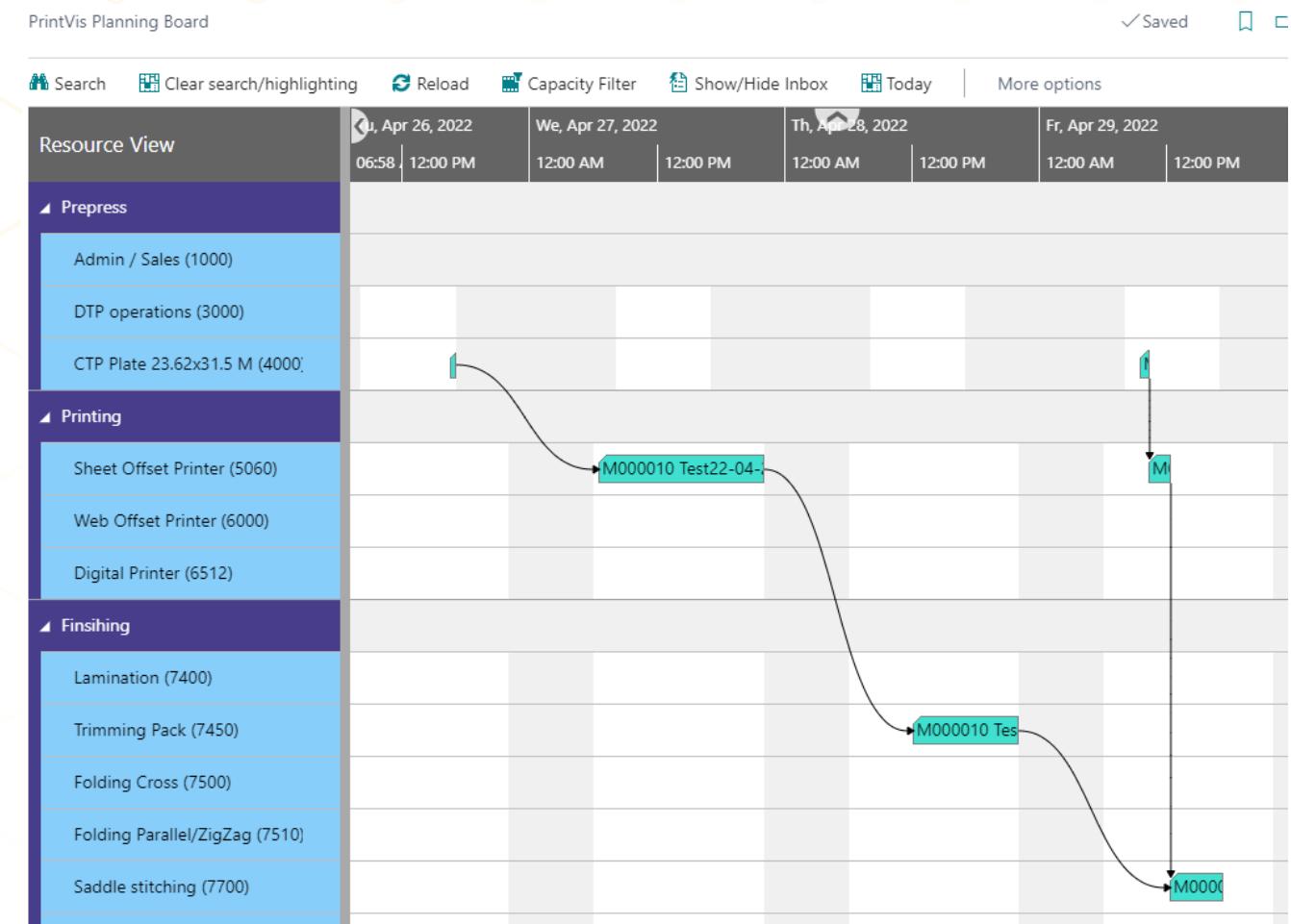
Planned Orientation

Manage More options

Mar...	Capacity Unit	Unit	Sorting Order ↑	Text	Job Item Description	Comment Text	Start	Ending	Loc...	Plan Status	Gang Job	Calc. Total Time	Planned Time	Ma.. Plan. Time
→	<input type="checkbox"/>	<input type="button" value="..."/>	PROOF-OUT	30	Proof sent to customer by	—	—		<input type="checkbox"/>	Not planned				C
	<input type="checkbox"/>		PROOF-OK	40	Proof approval from customer ...	—	—		<input type="checkbox"/>	Not planned				C
	<input type="checkbox"/>		PAPER	60	Paper in house by	—	—		<input type="checkbox"/>	Not planned				C
	<input type="checkbox"/>	3000	PREPRESS	3000	Prepress	—	—		<input type="checkbox"/>	Not planned				C
	<input type="checkbox"/>	4000	PLATES	4000	Plate production	Text block	—	2022-04-26 16:30	2022-04-26 17:00	<input type="checkbox"/>	Variable Pl...	0.50	0.50	C
	<input type="checkbox"/>	4000	PLATES	4000	Plate production	Cover	—	2022-04-29 09:35	2022-04-29 10:20	<input type="checkbox"/>	Variable Pl...	0.75	0.75	C
	<input type="checkbox"/>	5060	PRINTING	5060	Printing	Text block	—	2022-04-27 06:29	2022-04-27 22:00	<input type="checkbox"/>	Variable Pl...	13.96	15.51	C
	<input type="checkbox"/>	5060	PRINTING	5060	Printing	Cover	—	2022-04-29 10:20	2022-04-29 12:19	<input type="checkbox"/>	Variable Pl...	1.10	1.98	C
	<input type="checkbox"/>	7450	FINISHING	7450	Trimming Pack	Text block	—	2022-04-28 12:09	2022-04-28 22:00	<input type="checkbox"/>	Variable Pl...	1.76	9.85	C
	<input type="checkbox"/>	7700	FINISHING	7700	Saddle stitching	—	—	2022-04-29 12:19	2022-04-29 17:20	<input type="checkbox"/>	Variable Pl...	1.95	5.01	C
	<input type="checkbox"/>	9000	SHIPPING	9000	Shipping	—	—	2022-04-29 17:50	2022-04-29 18:00	<input type="checkbox"/>	Variable Pl...	0.02	0,17	C

PrintVis Planning Board

- Planning board
 - Planning Board Setup, add serial no
 - Search For PrintVis Planning Board



PrintVis Material Requirement

- Case Card
 - Material Requirement
 - Shows the requirement for the case

← PrintVis Material Requirement

	Order No.	Item Type	Item No.	Description	Quantity	Ma... Qua...	Quantity Needed	Reserved Quantity	PV Reserved Quantity	Open Qty. From Purchase Orders	Inventory available date	Inventory	Sub Con...
→	M000010	: Finished Good	STD_BOX	Standard size Box	20,00	<input type="checkbox"/>	20,00	-	-	-	2022-04-29	1 500	<input type="checkbox"/>
	M000010	Paper	69102343	Satin Coated Blazer	15 630,00	<input type="checkbox"/>	15 630,00	-	-	-	2022-04-27	10 000	<input type="checkbox"/>
	M000010	Plates	PL4000-100	Small Plates	4,00	<input type="checkbox"/>	4,00	-	-	-	2022-04-29	750	<input type="checkbox"/>
	M000010	Ink	C	Cyan	3,31	<input type="checkbox"/>	3,31	-	-	-	2022-04-29	1 000	<input type="checkbox"/>
	M000010	Ink	K	Black	3,53	<input type="checkbox"/>	3,53	-	-	-	2022-04-29	1 000	<input type="checkbox"/>
	M000010	Ink	M	Magenta	3,31	<input type="checkbox"/>	3,31	-	-	-	2022-04-29	1 000	<input type="checkbox"/>
	M000010	Ink	v	Vellum	3,31	<input type="checkbox"/>	3,31	-	-	-	2022-04-29	1 000	<input type="checkbox"/>

Reservation Entries

PrintVis Shop Floor

- Change Role to PrintVis Shop Floor Worker
 - Login with PPR
 - Select Cost Center Sheet Offset Printer
 - Find the job and start production

The screenshot shows a software interface for managing a printing job. At the top, there are navigation icons: a left arrow, a pencil, a magnifying glass, a plus sign, a trash can, and a refresh/crop icon. Below this is a large button labeled "All". Underneath are four tabs: "Show Completed", "Show Prod. Plan", "Manning Code Filter", and "Job Ticket". The main area displays a job card for "Peter Printer (5060 | Sheet Offset Print)".
Job Information:
Order No. **M000010**
Job Name **Test22-04-22 10:44**
Sheet ID **1**
Net Qty **11 760**
Scrap Qty **3 870**
Total Qty **15 630**
General:
Back (button)
Q&A Checklist (button)
Comments (button)
Signature **0**
Consume Mater... (button)
Return to Stock (button)
Produced Output (button)
Output Semi - F... (button)
Release Finishe... (button)
Ship Product (button)
Job Control:
Setup/Make rea... **100**
Production **200**
STOP JOB (button)
JOB COMPLETE (button with checkmark)
Waste Codes:
'No Waste' **--**
Customer Error **CUST**
Internal Error **INT**
Vendor Error **VEND**
ToolTip:
M000010
Test22-04-22 10:44
Status: Variable Planned
Part: Text block

Previous: 22-04-26 17:00
Starting: 22-04-27 06:29
- Buffer: 0 hours
Next: Trimming Pack
Starting: 22-04-28 08:29
Delivery: 22-04-30 00:00

Colors: 4+4: CMYK

PrintVis Shop Floor

- Consume material
 - Consume some paper
- Click stop
 - Time is registered
 - Complete the job
- Log out! <- possible demo ghost

PrintVis Shipment

- Change back Role to PrintVis Admin or PrintVis Coordinator
 - Open the case card again
 - Click on Shipment
 - Put in quantity to ship
 - Ship
 - Change status to INVOICE
- Methods
 - Create sales order per shipment
 - Just mark it as shipped

PrintVis Invoice

- Form the case card
 - Click on Invoice Guide
 - Click new
 - Click build
 - Post

Workshop run the demo flow

- Run through the demo flow
 - When you have a posted invoice, you are done.

Data Structure

The PrintVis Base App has around 250 tables which are a mixture of setup and transactional data.

PVS Case

PK: ID

Table:PVS Case	
Key field:ID	
Commonly used Fields	
Sell To No.	Who is ordering
Job Name	Text field for describing the case
Status Code	Current Status of the Case
Product Group and Order Type	Define type of case

The most central table in the system. Think of the case as a placeholder for all information about a particular customer task.

The primary key for the case is simply an entry no. called “ID”. During its lifecycle the case will change status, starting off as “request.” When/if the case changes status to “Quote” it will get a quote No. assigned (from a no. series) and when/if the case change status to “Order” it will get a Order No. assigned (from a no. series).

When there are no activities expected on a case it will be marked as “Archived” (a simple Boolean field) and assigned an “Archived Date.” This is typically done when an order is invoiced or a quote is rejected. This can be reverted and the case can again be removed from archive (simply by setting the Boolean field “Archived” to false)

PVS Job

PK: ID, Job, Version

The job table is the main placeholder for describing the product that we want to produce. In many companies you only have one job per case, whereas other companies are used to working with several jobs per case. This is a matter of company policies. The primary keys for the job are ID, Job, Version. Where the ID is the reference to the case, Job is the job no. and version gives you the opportunity to have several versions of the job. Only one of these versions will be the active version that is going to be produced.

ID	Job	Version	Status	Active	Quantity
123	1	1	Quote	Yes	5000
123	1	2	Quote	No	10000
123	1	3	Quote	No	15000

```
PVSJob.Setrange(ID,123);
PVSJob.Setrange(Active,true);
PVSJob.Setrange("Quote Calculation",true);
if PVSJob.FindSet() then...
```

ID	Job	Version	Status	Active	Quantity
123	1	1	Quote	No	5000
123	1	2	Quote	Yes	10000
123	1	3	Quote	No	15000
123	1	4	Order	Yes	7000

Table:PVS Job	
Key field:ID, Job, Version	
Commonly used Fields	
Quantity	Ordered/estimated quantity
Quoted Price	Quoted price for the line
Active	Check if the line is active
Requested Delivery Date	Requested Delivery Date

PVS Job Item

PK: ID, Job, Version, Job Item No., Entry No.

The job item tables is used to describe the different part of a product.
Book = Cover + text part.
Its here where the main machine is connected to the job.



1 · 2 · Order · 22-04-25 13:19

 Estimating  Automate all sheets  Userfields 1

Actions Fewer options

Job Items | Manage

 Create Sheet Delete Sheet Extra Job Item Delete Extra Job Item Job Signatures Specification Consumables Versions/Variants Combined Sheets Exit

No.↑	Sheet No.	Quantity	Component Type	Pages with Print	Main Pages	Colors Front	Colors Back	Job Item Format	Paper Weight	Paper Quality	Paper Item No.	List of Units	
→	1	1	50000	FLAT	1	<input type="checkbox"/>	4		10X10	80,0000	GLOSS	68102104	WEBOFFSET

PVS Sheet

PK: Sheet ID

The Sheet table stores information about the sheet, on the sheet the Job Item is produced/printed.

Information that is useful from here could be Size of the sheet. Quantity needed for producing the product.

Specifications

Web 1/1 | Flatwork - Sheet | Flatwork - Sheet

Scrap Process Flow More options

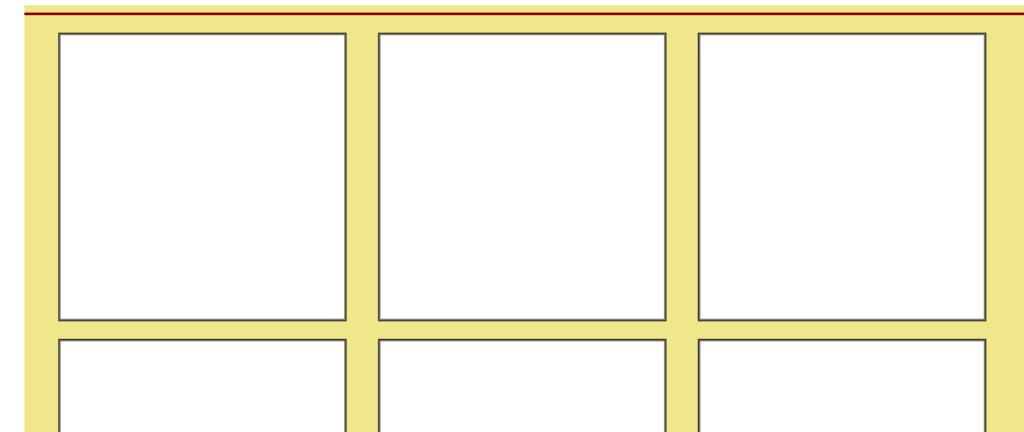
Web Information

Show more

Paper Item No.	68102104	...	Reel Width	35,000
Description	White Gloss		Reel Length	0,000
Paper Quality	GLOSS	...	Number of Reels	1,00
Weight	80,000		Print Reel Width	35,000
Weight Unit	BOOK		Cut Off	23,000
			Print Width/Reel Width	1

Quantity >

Imposition



PVS Job Process

PK: Process ID

The Job process tables stores information about the different processes, scrap is probably the most important thing to mention here.

Edit - Technical Specifications: Advanced - 230

Manage Page

Process

ProcessID 2 Cost Center... 6000 Web ID 1

Process Name Web Offset Printer Cost Center... Web Offset Printer

Process Details >

Processes in Job Item / Job >

Scrap Process

Start Scrap	<input type="text"/>	Scrap for Following Pl...	<input type="text"/> 50
Scrap % Running	<input type="text"/> 2	Man. Scrap for Followi...	<input checked="" type="checkbox"/>
Manual Scrap %	<input checked="" type="checkbox"/>	No. of Shifts / Stops	<input type="text"/>
No Of Webs Sets	<input type="text"/> 1	Man. No. of Stops	<input checked="" type="checkbox"/>
Per Section Scrap	<input type="text"/>	Per Stop Scrap	<input type="text"/> 150
Man. Section Scrap	<input checked="" type="checkbox"/>	Man. per Stop Scrap	<input checked="" type="checkbox"/>
Section Following Scrap	<input type="text"/>	No. of Plates to change	<input type="text"/> 0,00
Man. Section Following	<input checked="" type="checkbox"/>	Man. No. of Plates to ...	<input checked="" type="checkbox"/>
Processing Per Web	<input type="text"/> 1	Per Plate in Shift Scrap	<input type="text"/> 50
Manual Processing	<input checked="" type="checkbox"/>	Man. per Plate in Shift...	<input checked="" type="checkbox"/>
Per Processing Scrap	<input type="text"/> 150		
Man. Per Processing S...	<input checked="" type="checkbox"/>		

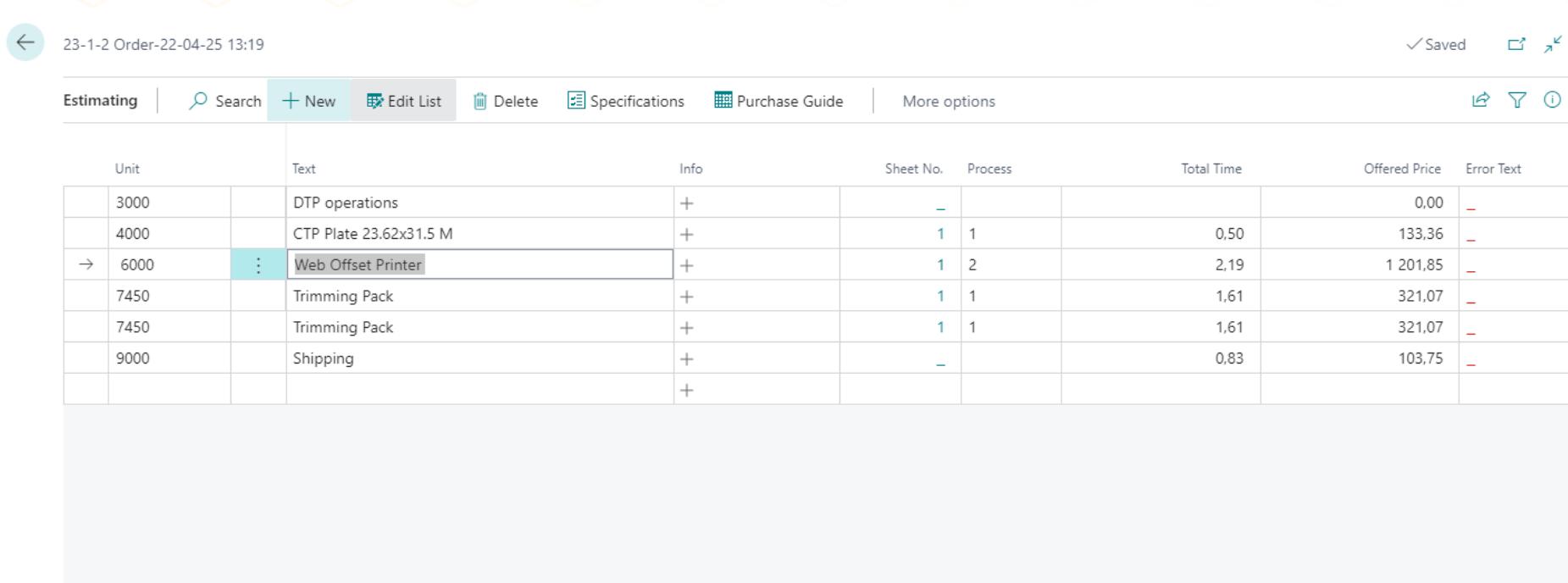
Close

PVS Job Calculation Unit

PK: ID, Job, Version, Entry No.

A puzzle piece of the estimation. The Calculation unit could be connected to a Sheet and a Process.

The Calculation Unit is also connected to a Planning unit, with the field Plan ID



The screenshot shows a software interface for managing calculation units. At the top, there's a header bar with a back arrow, the identifier '23-1-2 Order-22-04-25 13:19', a 'Saved' status indicator, and several icons for search, new entry, edit list, delete, specifications, purchase guide, and more options. Below the header is a toolbar with buttons for Estimating, Search, New (highlighted in blue), Edit List, Delete, Specifications, Purchase Guide, and More options. The main area is a table listing calculation units:

Unit	Text		Info		Sheet No.	Process	Total Time	Offered Price	Error Text
3000		DTP operations	+	-				0,00	-
4000		CTP Plate 23.62x31.5 M	+	1	1		0,50	133,36	-
→ 6000	⋮	Web Offset Printer	+	1	2		2,19	1 201,85	-
7450		Trimming Pack	+	1	1		1,61	321,07	-
7450		Trimming Pack	+	1	1		1,61	321,07	-
9000		Shipping	+	-			0,83	103,75	-
			+						

PVS Job Calculation Detail

PK: ID, Job, Version, Entry No., Line Type, Unit Entry No.

The smallest puzzle piece of the estimate.

It could either be a material line or a hour line.

Material line:

Item

Quantity

Cost and sales prices

Hour line:

Operation (what to do)

Hours

Cost and sales prices

Both lines has also:

Cost Center – what machine/resource

Unit of measure – how to consume

Line Type – Normal or additional

PVS Job Planning Unit

PK: Simulation Version, ID, Job, Plan ID, Entry No.

Things that needs to be planned or are already planned. Planning against capacity units.

To get the Calculation details connected to the planning unit, the Plan ID is the field that exists in both tables.

Case Card ▼ Saved

Edit - Job Planning - M000013

Set Limits

Requested Shipment Date	<input type="text" value="2022-04-30 00:00"/>	Method	Rough Planning	Time Date
Earliest Start Date	<input type="text"/>	Planned Orientation		

Manage Drop an action here | Actions Related Reports Fewer options

Mar...	Capacity Unit	Unit	Sorting Order ↑	Text	Start	Ending	Loc...	Plan Status	Calc. Total Time	Planned Time	Ma... Plan... Time	Roughly Planned Start no earlier than	Ma... Earli... Start	Roughly Planned Latest Start
<input type="checkbox"/>		PROOF-OUT	30	Proof sent to customer by			<input type="checkbox"/>	Not planned			<input type="checkbox"/>		<input type="checkbox"/>	22-04-25 09:00
<input type="checkbox"/>		PROOF-OK	40	Proof approval from customer ...			<input type="checkbox"/>	Not planned			<input type="checkbox"/>		<input type="checkbox"/>	22-04-25 10:00
<input type="checkbox"/>		PAPER	60	Paper in house by			<input type="checkbox"/>	Not planned			<input type="checkbox"/>		<input type="checkbox"/>	22-04-25 17:00
<input type="checkbox"/>	3000	PREPRESS	3000	Prepress			<input type="checkbox"/>	Not planned			<input type="checkbox"/>		<input type="checkbox"/>	22-04-25 17:00
<input type="checkbox"/>	4000	PLATES	4000	Plate production	2022-04-26 00:00	2022-04-26 00:30	<input checked="" type="checkbox"/>	Locked	0,50	0,50	<input type="checkbox"/>		<input type="checkbox"/>	22-04-29 16:05
→ <input type="checkbox"/>	6000	PRINTING	6000	Printing			<input type="checkbox"/>	Not planned	2,19	2,19	<input type="checkbox"/>	2022-04-26 00:30	<input type="checkbox"/>	22-04-29 16:35
<input type="checkbox"/>	7450	FINISHING	7450	Trimming Pack			<input type="checkbox"/>	Not planned	3,22	3,22	<input type="checkbox"/>	2022-04-26 02:41	<input type="checkbox"/>	22-04-29 18:46
<input type="checkbox"/>	9000	SHIPPING	9000	Shipping			<input type="checkbox"/>	Not planned	0,83	0,83	<input type="checkbox"/>	2022-04-26 06:24	<input type="checkbox"/>	22-04-29 23:09

PVS Job Costing Journal/Line

PK: Simulation Version, ID, Job, Plan ID, Entry No.

When users reporting time or material everything ends up in this PrintVis Job Costing Journal. Then when the Journal is posted different thing will happen with lines of type time and material.

PVS Job Costing Labor Entry
Contains all posted time

PVS Job Costing Entry
Contains all posted time and material, has connection to Item Ledger Entry or PVS Job Costing Labor Entry. Also contains information of posted invoices.

PVS Shop Floor Journal Entry

PK: Simulation Version, ID, Job, Plan ID, Entry No.

Used on shopfloor to track ongoing time reporting.

Also used to collect the reporting from shop floor if realtime posting is deactivated.

PVS Template

Template is just a normal case.

On the Job line its possible to mark it as a template.

Template could be connected to Product groups or Items.

Workshop Status Code

- Status Codes are used on Cases
- Triggers functionality / checking data
- Used in all installations
- What to extend?
- Check data before status change
- Modify data after status change

1. Before status change to Quote, Check quoted amount, if > 1000 then error out
2. Create a table + List Page
 1. PTE Status Change Log
 1. ID PK
 2. Entry No. PK
 3. Old status Code[50]
 4. New status Code[50]
 5. On the page add system field modified at
 2. Add record after status code is changed

```
[EventSubscriber(ObjectType::Codeunit, Codeunit::"PVS Main", 'OnCaseBeforeStatusChange', '')  
[EventSubscriber(ObjectType::Codeunit, Codeunit::"PVS Main", 'OnCaseAfterStatusChange', '')
```

Workshop User Field 1/2

- Can be set for different tables
 - PVS Case, PVS Job, Customer, etc.
- Can be grouped and assigned to a PrintVis Department
- Can be used in
 - Calculation formulas
 - Merge fields
 - Printed reports

Use the **PVS Userfield Management** codeunit

```
UserfieldManagement.Get_UserField_Value(...);
```

```
UserfieldManagement.Set_UserField_Value(...);
```

Workshop User Field 2/2

- Task 1
 - Create user field on case level
 - Extend Report PVS Word Dataset Case Quote (6010405) with the created user field
 - Create custom layout and add the field
 - Run the report with the custom layout
- Task 2
 - Create user field on job level
 - Modify the user field with the current date time when Case status changes to Plan
 - Subscribe to **OnCaseBeforeStatusChange** event in PVS Main codeunit
 - Add the userfield value to the job list part on the case card.

Workshop Calculation formula 1/3

- Could be used in several places
 - Estimate
 - Machine setup
 - Speed / scrap tables
 - Pricelists
- Triggers functionality / checking data
- Used in all installations

```
codeunit 50151 "PTE Calculation Formula"
{
    trigger OnRun()
    begin
        SingleInstance.Get_Current_CalcUnitDetailRec(JobCalculationDetail);
        JobCalculationDetail."Qty. Calculated" := 0; // Clear the field to be calculated
        JobCalculationDetail."Qty. Calculated" := Random(10000); // Assign value
        SingleInstance.Set_Current_CalcUnitDetailRec(JobCalculationDetail); // Push back the result
    end;

    var
        JobCalculationDetail: Record "PVS Job Calculation Detail";
        SingleInstance: Codeunit "PVS SingleInstance";
}
}
```

Workshop Calculation formula 2/3

Calculation Formulas

- Some data is calculated on temp tables

- Calculation Details

```
SingleInstance.Get_CalcUnitDetailsRecTmp(JobCalculationDetailTemp); //read calculation lines
```

- Sheet

```
SingleInstance.Get_SheetRecTmp(JobSheetTemp); // Use this if you need info from the sheet
```

- Process

```
SingleInstance.Get_ProcessRecTmp(JobProcessTemp); // Need info from the process
```

- The rest is possible to read as normal

```
codeunit 50152 "PTE Calculation Formula 2"
{
    trigger OnRun()
    begin
        SingleInstance.Get_Current_CalcUnitDetailRec(JobCalculationDetail);
        SingleInstance.Get_SheetRecTmp(JobSheetTemp);
        JobCalculationDetail."Qty. Calculated" := 0; // Clear the field to be calculated
        JobSheetTemp.SetRange("Sheet ID", JobCalculationDetail."Sheet ID");
        if JobSheetTemp.FindFirst() then
            JobCalculationDetail."Qty. Calculated" := JobSheetTemp.Weight;
        SingleInstance.Set_Current_CalcUnitDetailRec(JobCalculationDetail); // Push back the result
    end;

    var
        JobSheetTemp: Record "PVS Job Sheet" temporary;
        JobCalculationDetail: Record "PVS Job Calculation Detail";
        SingleInstance: Codeunit "PVS SingleInstance";
```

Workshop Calculation formula 3/3

Calculation Formulas

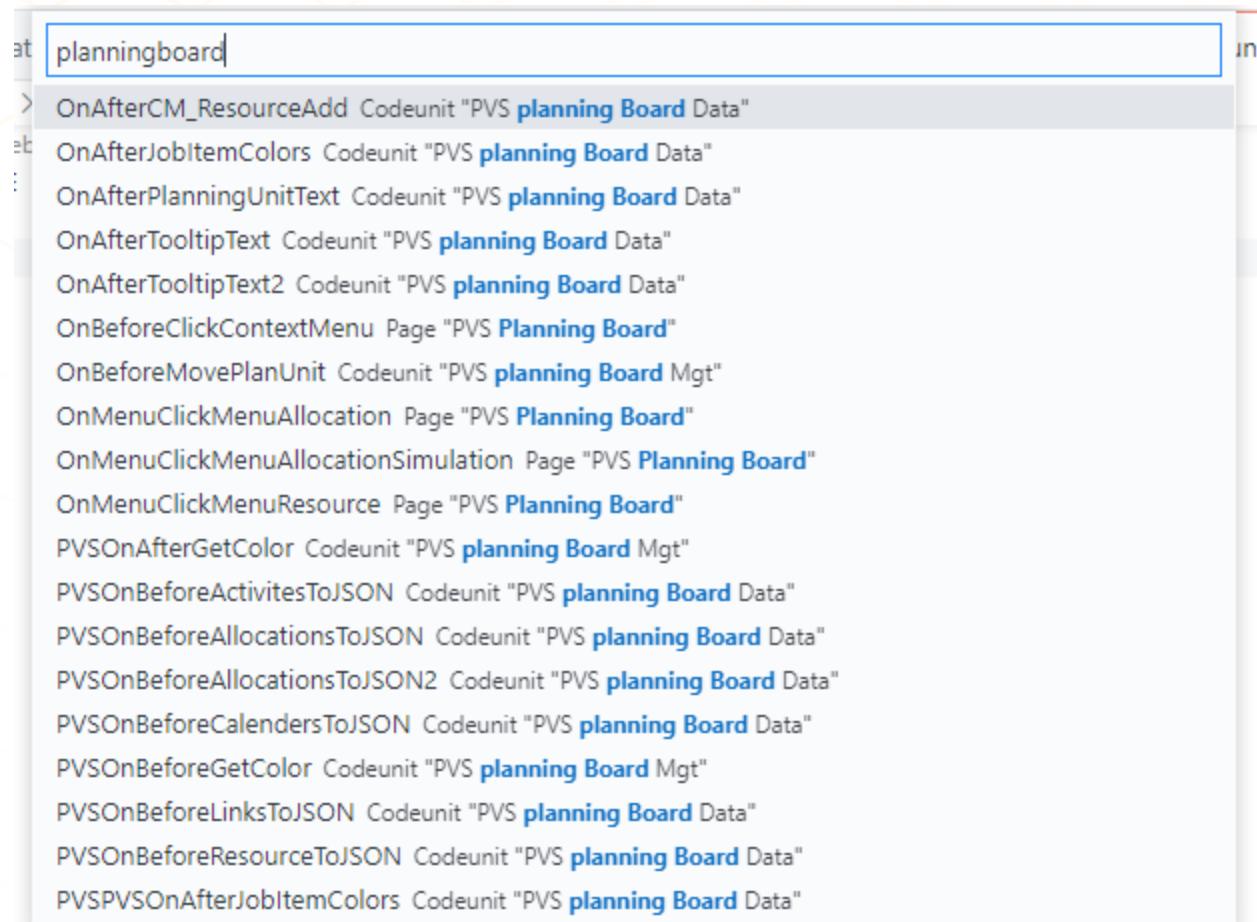
- Task 1
 - Create formula 1010 that's returns the weekday (DATE2DWY(TODAY(), 1);)
- Task 2
 - Create formula 1011 that returns the area of the printed sheet
 - Sheet.width x Sheet.length
- Create both formulas in same codeunit

```
case JobCalculationDetail."Formula Code" of  
    1010:Formula_1010_Weekday();  
    1011:Formula_1011_Area();  
end;
```

Workshop Planning Board 1/2

Planning Board

- Based on Netronics product
- Possible to manipulate data
- Tooltips
 - Possible to do setup against most common tables
 - Extend with event
 - HTML based
- Colors
 - Possible to setup
 - Extend with event
 - Css-colors
- Menu
 - Extend enums



The screenshot shows a software interface with a search bar at the top containing the text "planningboard". Below the search bar is a list of events, all of which are associated with the codeunit "PVS planning Board Data". The events listed are:

- OnAfterCM_ResourceAdd
- OnAfterJobItemColors
- OnAfterPlanningUnitText
- OnAfterTooltipText
- OnAfterTooltipText2
- OnBeforeClickContextMenu
- OnBeforeMovePlanUnit
- OnMenuClickMenuAllocation
- OnMenuClickMenuAllocationSimulation
- OnMenuClickMenuResource
- PVSOnAfterGetColor
- PVSOnBeforeActivitesToJSON
- PVSOnBeforeAllocationsToJSON
- PVSOnBeforeAllocationsToJSON2
- PVSOnBeforeCalendersToJSON
- PVSOnBeforeGetColor
- PVSOnBeforeLinksToJSON
- PVSOnBeforeResourceToJSON
- PVSPVSONAfterJobItemColors

Workshop Planning Board 2/2

Planning Board

- Task 1
 - Add Customers name from the case to the tooltip with an event
- Task 2
 - Set the color to Purple with an event
- Task 3
 - Extend the menu with two options
 - Change status to JOBCOSTING, reload planning for the case
 - Open the first shipment of the Case

Workshop Shop Floor 1/3

Shop Floor

- Most common is to add an extra tile
- Two events are used
 - One for adding the tile

```
[EventSubscriber(ObjectType::Codeunit, Codeunit::"PVS Shop Floor JS Add-in Mgt", 'PVSAAddMoreTilesJobCard')
```

- One for processing the click

```
[EventSubscriber(ObjectType::Codeunit, Codeunit::"PVS Shop Floor JS Add-in Mgt", 'OnAfterClickTileJobTicket')
```

```
[EventSubscriber(ObjectType::Codeunit, Codeunit::"PVS Shop Floor JS Add-in Mgt", 'PVSAAddMoreTilesJobCard', '', false, false)]
local procedure PVSAAddMoreTilesJobCard(var TempPVSSortingBuffertTiles: Record "PVS Sorting Buffer"; var isHandled: Boolean);
begin
    TempPVSSortingBuffertTiles.Init();
    TempPVSSortingBuffertTiles.PK1_Integer1 := 1;
    TempPVSSortingBuffertTiles.Text1 := '#CUSTOMSHIP#';
    TempPVSSortingBuffertTiles.Text2 := 'Shipments';
    TempPVSSortingBuffertTiles.Insert();
    isHandled := true;
end;
```

Workshop Shop Floor 2/3

Shop Floor

- Another change could be that the customer would like to have more info on right side
- Since PV20 we could do more setup.... Should be enough.
- If not...

```
[EventSubscriber(ObjectType::Codeunit, Codeunit::"PVS Shop Floor JS Add-in Mgt",
'OnAfterBuildPanelBuffer', '', false, false)]
```

```
if pagename <> 'JobCard' then
    exit;

PlanRec.SetPosition(JSAddinMgt.DecodeBase64ElementName(in_ControlID));
PlanRec.Find('=');

CaseRec.Get(PlanRec.ID);
```

```
in_PanelBuffer.Reset();
in_PanelBuffer.SetRange(Code1, SF_FIELDINFO);
if in_PanelBuffer.FindFirst() then begin
    TempSortingBuffer := in_PanelBuffer;
    TempSortingBuffer.Text1 := 'Customer';
    TempSortingBuffer.PK1_Integer2 := 1;
    TempSortingBuffer.Text2 := CaseRec."Sell-To Name";
    in_PanelBuffer.Init();
    in_PanelBuffer := TempSortingBuffer;
    if not in_PanelBuffer.Insert() then
        in_PanelBuffer.Modify();
end;
```

```
in_PanelBuffer.reset;
Page.Run(Page::"PTE Sorting Buffer", in_PanelBuffer);
```

Login	Alex Accountant (4000 CTP Plate 23.6 ^
Job Information	
Customer	PrintVis Inc.
Order No.	M000004
Job Name	Spring Catalog 2020
Sheet ID	1
Net Qty	3 000
Scrap Qty	1 345
Total Qty	4 345

ToolTip
M000004
Spring Catalog 2020
Status: Not planned
Part: Text block
Previous:
Starting:
- Buffer: 0 hours
Next:
Starting:
Delivery:
Colors: 4+4: CMYK
Paper Name: White Gloss
Number of Sheet Sets: 3
Width: 35
Length: 23
Finishing
Random: \$TEST
Comments

Workshop Shop Floor 3/3

Shop Floor

- Task 1
 - Add a tiles to open the shipment list filter on the Case ID
- Task 2
 - Add customer name in panel on right side
- Task 3
 - When shopfloor start and stop, send a GET to `webhook.site`

```
[EventSubscriber(ObjectType::Codeunit, Codeunit::"PVS Shop Floor Management", 'OnBeforeJournal_Special_Start_Stop', '', false, false)]
```

```
local procedure sendToSomewhere(TextToSend: Text)
var
    myHTTPClient: HttpClient;
    myResponseMessage: HttpResponseMessage;
begin
    myHTTPClient.Get('https://webhook.site/9113df2f-c59d-4823-9408-610cf80774e6?string=' + TextToSend,
myResponseMessage);
end;
```

Workshop Shop Floor Extra

- Start and stop time registration. Scenario integration?
Customization?
 - InData
 - Order No. = use one of the created cases that are in status PLAN
 - CostCenterCode = 6512
 - UnitOfMeasure = 200
 - PVSUser = PPR
1. Procedure Start
 1. Find if " PVS Shop Floor Journal Entry" exists, if not create a new one
`ShopFloorManagement.Journal_Add_Entry`
 2. Set uom on the created line
 3. Start the line
`ShopFloorManagement.Journal_Start_Entry()`
 1. Procedure Start
 1. Stop the line
`ShopFloorManagement.Journal_Stop_Entry()`

How to test? Postman... or just create a codeunit/page action calling your code

Workshop Invoice Template 1/6

Invoice Template

- Invoice templates are used to generate invoices
 - Possible to add "custom" lines when invoice is built up
 - Possible to manipulate the invoice line before insert
 - Possible to replace with custom build

Workshop Invoice Template 2/6

Invoice Template

- Possible to add "custom" lines when invoice is built up

```
[EventSubscriber(ObjectType::Codeunit, Codeunit::"PVS Invoice Template Mgmt", 'BuildInvoiceTemplateLine', '', false, false)]
procedure BuildInvoiceTemplateLine(var SalesHeader: Record "Sales Header"; var InvoiceTemplateHeader: Record "PVS Invoice Template Header";
var InvoiceTemplateLine: Record "PVS Invoice Template Line"; var Job: Record "PVS Job");
var
    InvoiceTemplateManagement: Codeunit "PVS Invoice Template Mgmt";
    InvoiceTemplateLineTemp: Record "PVS Invoice Template Line" temporary;
begin
    IF InvoiceTemplateLine."No." <> 'TEST' then
        exit;
    InvoiceTemplateLineTemp.init;
    InvoiceTemplateLineTemp.Type := InvoiceTemplateLineTemp.Type::"G/L Account";
    InvoiceTemplateLineTemp.Description := 'This line is created by special function';
    InvoiceTemplateLineTemp."Manual Description" := true;
    InvoiceTemplateLineTemp.Quantity := 123;
    InvoiceTemplateLineTemp."Unit Price" := 100;
    InvoiceTemplateLineTemp.Insert();
    //Insert Invoice line
    InvoiceTemplateManagement.Insert_Salesline_New(SalesHeader, InvoiceTemplateLineTemp, InvoiceTemplateHeader, Job, '', '', '', '');
end;
```

Workshop Invoice Template 3/6

Invoice Template

- Possible to manipulate the invoice line before insert

```
[EventSubscriber(ObjectType::Codeunit, Codeunit::"PVS Invoice Template Mgmt", 'Insert_Salesline_New_OnBeforeModify', '', false, false)]
local procedure Insert_Salesline_New_OnBeforeModify(var SalesHeader: Record "Sales Header"; var SalesLine: Record "Sales Line"; var
InvoiceTemplateHeader: Record "PVS Invoice Template Header"; var InvoiceTemplateLine: Record "PVS Invoice Template Line"; var Job: Record "PVS
Job");
begin
  if InvoiceTemplateHeader.Code <> 'STD' then
    exit;
  if InvoiceTemplateLine.Type <> InvoiceTemplateLine.Type::"Main Invoice" then
    exit;
  SalesLine.Description := SalesLine.Description + 'X';
end;
```

Workshop Invoice Template 4/6

Invoice Template

- Possible to replace with custom build
- The Case ID could be picked up from the Sales Header, apply it on all lines that are created

```
[EventSubscriber(ObjectType::Codeunit, Codeunit::"PVS Invoice Template Mgmt", 'BuildInvoiceTemplateHeader', '', false, false)]
local procedure BuildInvoiceTemplateHeader(var SalesHeader: Record "Sales Header"; var InvoiceTemplateHeader: Record "PVS Invoice Template Header");
var
    SalesLine: Record "Sales Line";
begin
    if InvoiceTemplateHeader.Code <> 'TEST' then
        exit;
    SalesLine.Init();
    SalesLine."Document Type" := SalesHeader."Document Type";
    SalesLine."Document No." := SalesHeader."No.";
    SalesLine."Line No." := 10000;
    SalesLine.Insert();
    SalesLine.Validate(Type, SalesLine.Type::"G/L Account");
    SalesLine.Validate("No.", '6110');
    SalesLine.Validate(Quantity, 123);
    SalesLine.Validate("Unit Price", 1);
    SalesLine."PVS ID" := SalesHeader."PVS Order ID";
    SalesLine.Modify(true);
end;
```

Workshop Invoice Template 5/6

Invoice Template

- Task 1

- Build one invoice line per Calculation Unit

20-1-2 Order-22-04-25 08:5722-04-25 09:44

✓ Saved

Estimating | Search + New Edit List Delete Specifications ...

Unit	Text		Info		Sheet No.	Pro...	Total Time	Offered Price	Error Text
3000	DTP operations		+		-			0.00	-
→ 4000	CTP Plate 23.62x31.5 M		+		1	1	0.50	133,36	-
5060	Sheet Offset Printer		+		1	2	1,51	556,64	-
P_5060	Substrate Sheet Offset Printer		+		1			302,58	-
7450	Trimming Pack		+		1	1	10,91	2 177,26	-
7450	Trimming Pack		+		1	1	0,83	166,46	-
9000	Shipping		+		-		0,33	41,50	-
			+						

Invoice Template

✓ Saved

Invoice Template Assign Type

General

Code CALCUNITS Build from Order

Description Invoice Method Invoice based on Case level

Invoice Template Line Manage

New Line Delete Line Card

Level	Type	No.	Description	Ma... Des...	Quantity	Quantity Type
Case	Merge Fields	INV		<input type="checkbox"/>		
Case	Special Fun...	CALCUNIT		<input type="checkbox"/>		

> Page Inspection

See information about the page, its different elements, and the source behind the data it displays.

Page

PVS Job Calculation Unit 1 (6010376, List)

Table

PVS Job Calculation Unit (6010321)

View table

Table Fields Extensions Page Filters

ID (1, Integer, PK)
20
PrintVis
Job (2, Integer, PK)

Workshop Invoice Template 6/6

Invoice Template

- Task 2
 - Round the unit price on the main invoice line.

The screenshot shows the PRINTVIS software interface for managing invoice templates. The top navigation bar includes 'Invoice Template' (highlighted in blue), 'STD', 'Card', and 'List'. Below the navigation is a toolbar with icons for back, forward, edit, new, and delete. A reminder banner at the top states: 'Reminder: your work date is 2022-04-11 Use today | Change to... | Turn off reminder'. The main area is divided into sections: 'General' (Code: STD, Description: Standard, Build from: Order, Invoice Method: Invoice based on Case level), 'Invoice Template Line' (Manage tab selected), and a table for defining invoice lines. The table has columns: Level, Type, No., Description, Manual Descript..., Quantity, Quantity Type, Unit Price, Use Item No., and Comment. The first row is a header with checkboxes for 'Manual Descript...' and 'Unit Price'. The second row is for 'Merge Fields' (Type: Merge Fields, No: INV, Description: One invoice line per case, checked for both 'Manual Descript...' and 'Unit Price'). The third row is for 'Main Invoice' (Type: Main Invoice, No: , Description: Text line in bottom of invoice, checked for both 'Manual Descript...' and 'Unit Price'). The fourth row is for 'Text line in bottom of invoice' (Type: Text line in bottom of invoice, No: , Description: , checked for both 'Manual Descript...' and 'Unit Price').

Level	Type	No.	Description	Manual Descript...	Quantity	Quantity Type	Unit Price	Use Item No.	Comment
→ Case	Merge Fields	INV		<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	
Case	Main Invoice		One invoice line per case	<input checked="" type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	
Case			Text line in bottom of invoice	<input checked="" type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	

Workshop Folders SharePoint 1/2

Folders / SharePoint

- Folder setup
 - SharePoint
 - Folder group
 - Test
- PrintVis General Setup
 - Skip Background Processing
- Task
 - On status change to Order
 - Save report PVS Job Ticket as PDF and upload to the first SharePoint folder for the case

```
Cloudmgt.GetFolder(0, Folder."Cloud ID", Folder."Cloud ID", false, TempCloudStorage);
if TempCloudStorage.Count > 0 then
    ServerFileName := ServerFileName + Format(TempCloudStorage.Count);
ServerFileName += '.pdf';

if Cloudmgt.UploadFile(TempBlob, ServerFileName, Folder."Cloud ID", outNewFileID) then;
```

Workshop Folders SharePoint 2/2

Folders / Sharepoint

```
Folder.SetRange(ID, inID);
if not Folder.FindFirst() then begin
  Folder.Folder_Create_Folders();
  Folder.SetRange(ID, inID);
  if not Folder.FindFirst() then
    exit;
end;

recRef.Open(Database::"PVS Case");
fldRef := recRef.Field(1);
fldRef.SetRange(inID);

TempBlob.CreateOutStream(oStream);
if recRef.FindFirst() then
  Report.SaveAs(Report::"PVS Job Ticket", ReportParameters, ReportFormat::Pdf, oStream, recRef)
else
  exit;

Cloudmgt.GetFolder(0, Folder."Cloud ID", Folder."Cloud ID", false, TempCloudStorage);
if TempCloudStorage.Count > 0 then
  ServerFileName := ServerFileName + Format(TempCloudStorage.Count);
ServerFileName += '.pdf';

if Cloudmgt.UploadFile(TempBlob, ServerFileName, Folder."Cloud ID", outNewFileID) then;
```

- Create Folders
- Run report -> PDF -> TempBlob
- Upload the file

Workshop Job Ticket 1/3

Job Ticket

- Most complex report in PrintVis
- Very often customized
- Dataset based on PVS Buffer Table

My Company		Order No. M000001-1		7. February 2021 kt 12:08:35 Page 1/1	
customer: PrintVis Inc. 315 West Ponce de Leon Avenue Suite 440 30030 Decatur	cust. no.: PVS-C0001 Phone no.: FAX no.: e-mail: info@printvis.com	salesperson: SAS coordinator: COC quote no.: 0000005 copied from:			
contact: reference:	customer summary: PDF ready for CTP	Eco-label			
quantity: 1.000 format: 6X9 shipment:	Greatest Airplane Stories 1,000 books, 64 pages in size 6X9 and 1+1 colors				
Prepress					
We supply Proofs, CTP Plates, Print Finishing					
Proofs Yes					
Proof type Electronic					
Number of 1					
Imposition Standard					
Line 2 – section 3					
Line 4 – section 2					
Line 8 – section 4					
Line 12 – section 20	3000 DTP operations 3000 Aligning pages	0,50 Hours 0,50 Hours	0,00 0,00		
Line 14 – section 2	Plate production				
	4000 CTP Plate 23.62x31.5 M	0,50 Hours	0,00	SheetID 1	
Line 18 – section 50	Printing				
Line 19 – section 51	Printing Machine	Sheet	Print-Method	Printed Net	Paper
				Gross	
					Weight Format
					No. of Sheets
Line 22 – section 52	Text block, Colors: 1+1 Black				
	Job Offset Printer	8 Perf	25,0 x 9,70	8000 9135	58102415 White Offset Roll Husky
					60,00 25,0000 9135
					x 9,7000
	Cover, Colors: 4+0 Cyan toner Black toner Magenta toner Yellow toner				
	Digital Printer	2 Ink/Tn	14,0 x 10,00	2000 2010	68102167 C1S Cover Tango
		0			10,00 14,0000 2010
					x 10,0000
					10,0000
	Sheeter	2 Ink/Tn	14,0 x 10,00	2010 2010	68102167 C1S Cover Tango
		0			10,00 14,0000 2010
					x 10,0000

- Each record in the buffer table corresponds to a layout section

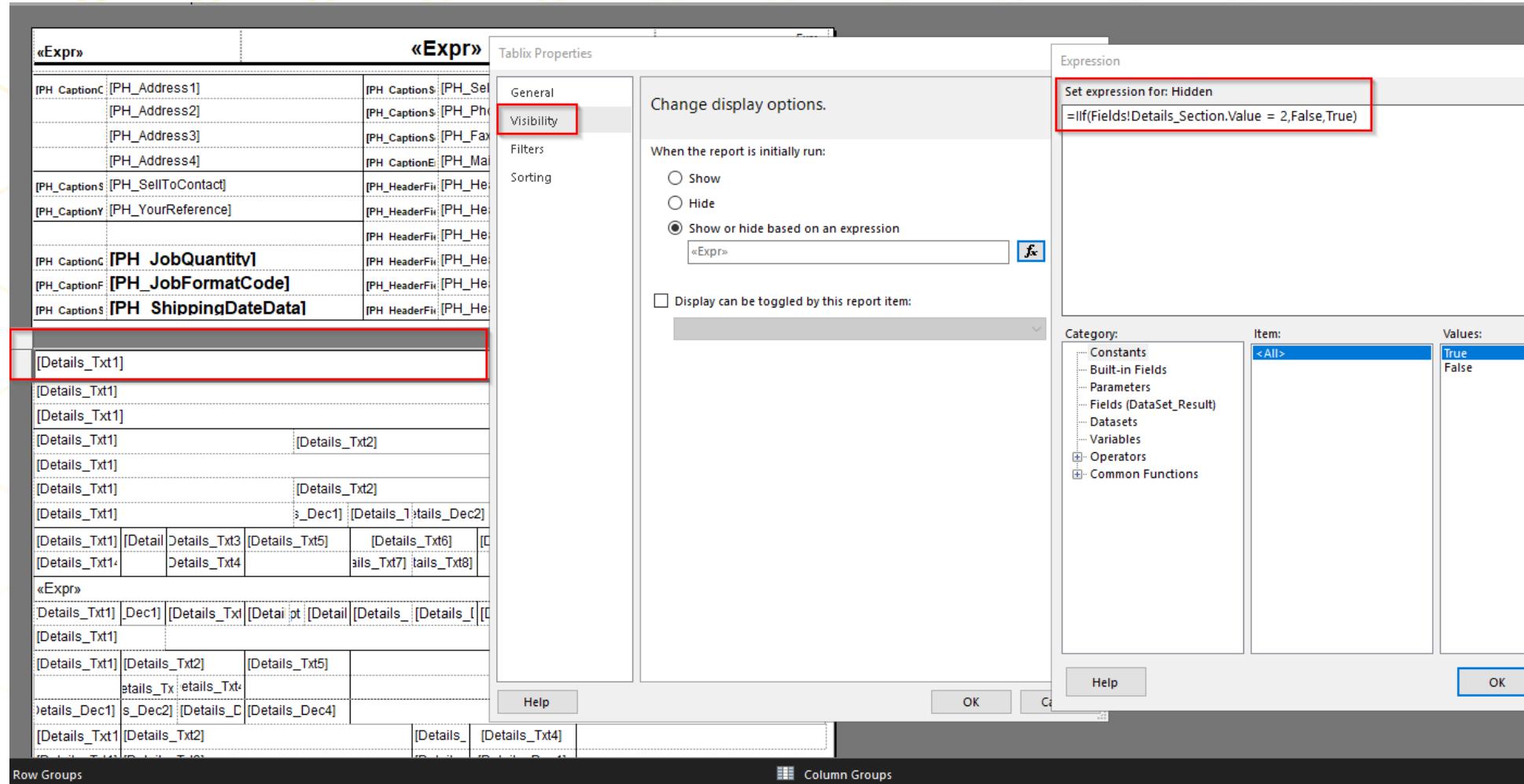
LineType:

- 1 : Delimiter Line
- 2 : Header (Department Name etc.)
- 3 : Comment Text (Full Width)
- 4 : Side by Side: Comment Text (Half Width) + UserFields
- 5 : UserFields
- 6 : Side by Side: Comment Text (Half Width) + UserFields (Same as 4)
- 7 : UserFields (Same as 5)
- 20 : Calc. Unit Details
- 50 : Press - Process Head
- 51 : Press - Process Subject Body
- 52 : Press - Process Body
- 55 : Press - Plate Change Head
- 57 : Press - Plate Change Body
- 60 : Postpress - Process Head
- 61 : Postpress - Process Subject Body
- 62 : Postpress - Process Body
- 70 : Press Continuous - Process Head
- 71 : Press Continuous - Process Subject Body
- 72 : Press Continuous - Process Body
- 80 : Shipments

Workshop JobTicket 2/3

Job Ticket

Layout visibility expression for each section based on section value



The screenshot shows a report design in SSMS. A section labeled 'Details_Txt1' is highlighted with a red border. The 'Tablix Properties' dialog is open, specifically the 'Visibility' tab. An expression is defined under 'When the report is initially run':

```
=IIf(Fields!Details_Section.Value = 2, False, True)
```

The 'Category' pane lists various items including 'Common Functions'. The 'Item' pane shows '<All>' and the 'Values' pane shows 'True' and 'False'.

Workshop JobTicket 3/3

Job Ticket

```
[EventSubscriber(ObjectType::Codeunit, Codeunit::"PVS JobTicket Management", 'OnAfterGetReportBuffer', '',
true, true)]
local procedure OnAfterGetReportBuffer(var in_JobRec: Record "PVS Job"; var in_VERSION_COUNTER: Integer;
var Out_BufferRecTmp: Record "PVS Sorting Buffer" temporary; var Result: Boolean)
begin
    // the generated dataset is in Out_BufferRecTmp
    // this can now be manipulated
end;
```

- Task 1
 - Change Job Item Description
- Task 2
 - Add extra line under each printing process
 - Create new line under Report Section 52 with a new key – use PK1_Integer2
 - Set a new report section and details text to be printed
 - Add new section in the layout

Workshop KPI 1/2

KPI Calculations

Possible to calculate/update 10 KPI fields on PVS Job
KPI's can be set on different product groups

Example:

```
[EventSubscriber(ObjectType::Table, Database::"PVS Job", 'OnAfterValidate_Price_Method', '', true,  
false)]  
procedure OnAfterValidate_Price_Method(var in_Rec: Record "PVS Job");  
var  
KPI_Value1, KPI_Value2, KPI_Value3: Decimal;  
begin  
// If you only want to calculate for a certain prod group you can add this check  
if in_Rec."Product Group" <> 'POSTER' then  
exit;  
// Calculate up to 16 KPI values somehow  
KPI_Value1 := 13;  
KPI_Value2 := 23;  
KPI_Value3 := KPI_Value1 + KPI_Value2;  
// Assign the KPI values to the Job  
in_Rec."Misc. 1" := KPI_Value1;  
in_Rec."Misc. 2" := KPI_Value2;  
in_Rec."Misc. 3" := KPI_Value3;  
end;
```

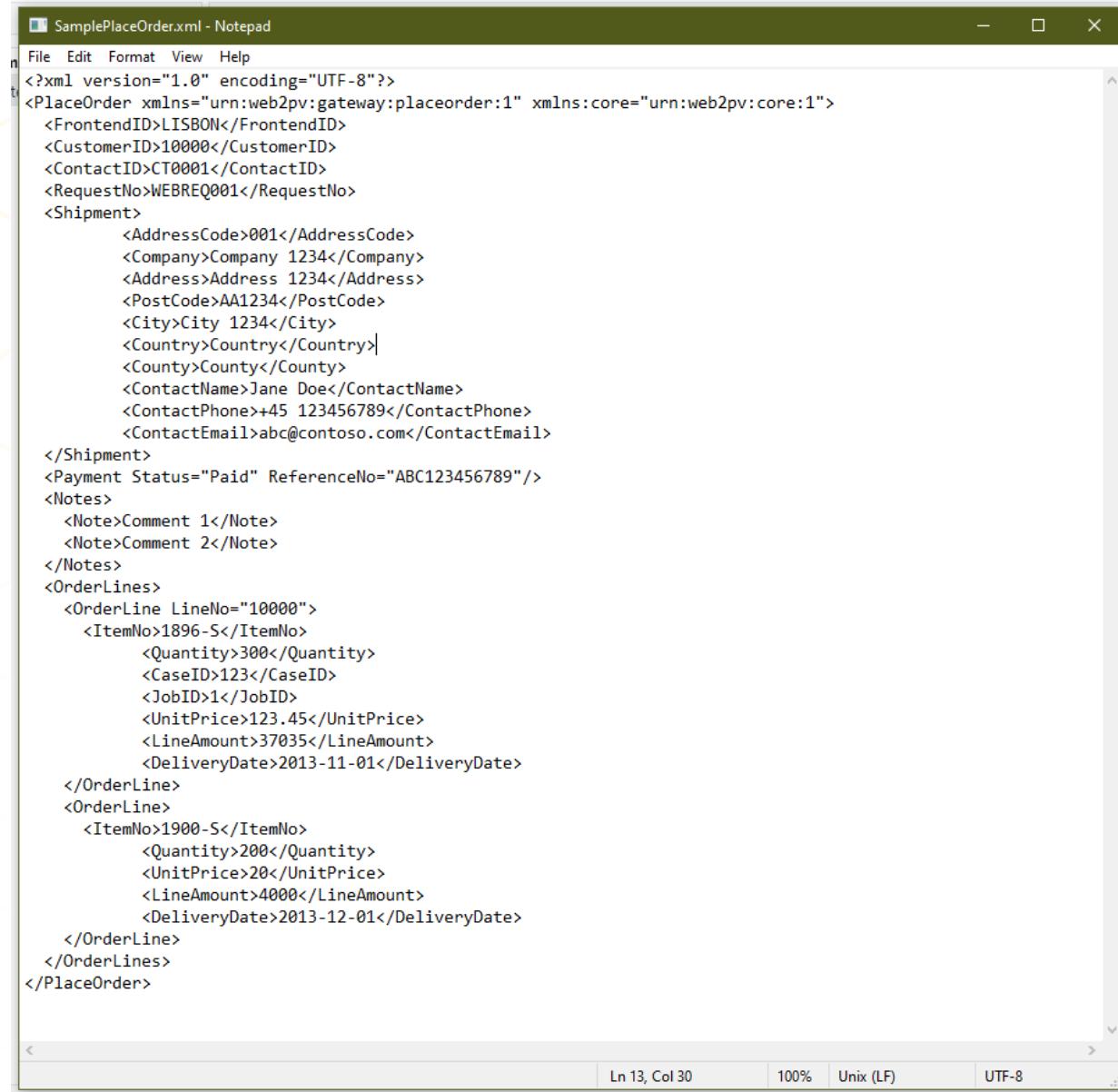
Workshop KPI 2/2

KPI Calculations

- Task 1
 - Create KPI field that displays cost of all plates in the estimate connected to the job
- Task 2
 - Create KPI field that displays the price per 10pcs
 - Add the KPI field next to the quoted price on the job list part on the case card.

Workshop Web2pv

- Setup
 - Web2pv FrontEnd
 - Code = LISBON
 - Always Sales order
 - Webshop Order Nos.
 - Sales Order Nos.
 - Accounts
 - Login ID = 1
 - Frontend Login ID = 1
 - Customer 10000
 - Web2pv catalogs
 - Code = X
 - New line. Index = 1
 - Catalog index.
 - Item No. 1896-S
 - Item No. 1900-S - production
 - Web2pv FrontEnd Catalog



The screenshot shows a Windows Notepad window titled "SamplePlaceOrder.xml - Notepad". The XML code represents a place order with the following structure:

```
<?xml version="1.0" encoding="UTF-8"?>
<PlaceOrder xmlns="urn:web2pv:gateway:placeorder:1" xmlns:core="urn:web2pv:core:1">
  <FrontendID>LISBON</FrontendID>
  <CustomerID>10000</CustomerID>
  <ContactID>CT0001</ContactID>
  <RequestNo>WEBREQ001</RequestNo>
  <Shipment>
    <AddressCode>001</AddressCode>
    <Company>Company 1234</Company>
    <Address>Address 1234</Address>
    <PostCode>AA1234</PostCode>
    <City>City 1234</City>
    <Country>Country</Country>
    <County>County</County>
    <ContactName>Jane Doe</ContactName>
    <ContactPhone>+45 123456789</ContactPhone>
    <ContactEmail>abc@contoso.com</ContactEmail>
  </Shipment>
  <Payment Status="Paid" ReferenceNo="ABC123456789"/>
  <Notes>
    <Note>Comment 1</Note>
    <Note>Comment 2</Note>
  </Notes>
  <OrderLines>
    <OrderLine LineNo="10000">
      <ItemNo>1896-S</ItemNo>
      <Quantity>300</Quantity>
      <CaseID>123</CaseID>
      <JobID>1</JobID>
      <UnitPrice>123.45</UnitPrice>
      <LineAmount>37035</LineAmount>
      <DeliveryDate>2013-11-01</DeliveryDate>
    </OrderLine>
    <OrderLine>
      <ItemNo>1900-S</ItemNo>
      <Quantity>200</Quantity>
      <UnitPrice>20</UnitPrice>
      <LineAmount>4000</LineAmount>
      <DeliveryDate>2013-12-01</DeliveryDate>
    </OrderLine>
  </OrderLines>
</PlaceOrder>
```

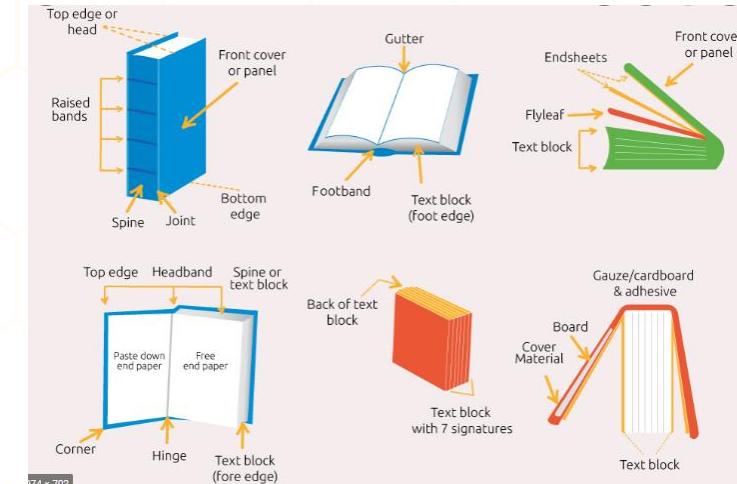
The XML code defines a place order with two items. The first item has an item number of 1896-S, a quantity of 300, and a unit price of 123.45. The second item has an item number of 1900-S, a quantity of 200, and a unit price of 20. Both items have a delivery date of November 1, 2013, and a line amount of 37035 and 4000 respectively. The order is marked as paid with reference number ABC123456789. The shipment address is set to company 1234 at address 1234, post code AA1234, city City 1234, and country Country. The contact information for the shipment includes Jane Doe, phone +45 123456789, and email abc@contoso.com. There are two comments in the notes section: "Comment 1" and "Comment 2".

Create Case with Code 1/2

- Scope
 - Create Case, Job that is planned
1. Create the Case
 1. Add the sell to
 2. Set Status Code ORDER
 2. Create the Job with Skip Calc = true
 1. Validate Product Group POSTER, ordered Qty. 20000, format code 10x10, requested delivery dateTime
 3. Get the Job Item against the Job
 4. Get the Sheet against the Job Item
 1. Validate Paper '68102104' and Controlling unit 'DIGITAL'
 5. Add new Job Calculation Unit, Validate the Unit field. (do not insert)
 6. Find the Calc Unit Detail line that was added, line type normal, operation 100
 7. Get the job again since its updated. Set skip calc = false, and modify.
 8. Get the case and change status to PLAN
 9. `PlanningManagement.Auto_Plan_All(0, false, JobRec.ID, JobRec.Job, false, false); //auto plan from delivery date`

Create Case with Code 2/2

- Task
 - Create Case
 - Customer 1000
 - Order Type PRINTED
 - Prod Group BOOKS
 - Status code ORDER
 - Job line
 - 10000 order qty
 - Format A4
 - Requested delivery datetime +14d 140000T
 - Job Card - Job Item
 - TEXT line
 - Pages with print = 64
 - Paper 68102104
 - List of units SHEETOFFSET
 - COVER line
 - Pages with print = 2
 - Color front 4 + color back 0
 - Paper 68102141
 - Lists Of Units DIGITAL
 - Add calculation unit 1000
 - Find Calcunit detail operation 100 on the inserted unit. Validate Hours = 1. Manual Hours = true;
 - Change status to PLAN
 - Autoplan on delivery date



Change status to PLAN
Autoplan

Workshop Unit Conversion

In the printing industry, many different units are used.

For example, paper could be handled in weight *and* pieces.

On the item the consultant/user has entered known information about the item's dimension, and with help of that we have standard functions to transform/convert from different units.

PrintVis Standard Units

```
PVSIItemManagement.Item_InventoryUnit_Factor(Item,FromUnit,ToUnit) = faktor
```

- Workshop
 - Convert 2660 pcs of item '100GLS BC' into Weight.
 - Convert 2660 pcs of item '100GLS BC' into Area

Upgrade from BC16 – BC20

Always better to upgrade

- Increased performance, new features
- Less time spent manually porting fixes

Consider new server configuration settings

Plan and test the upgrade process to measure expected downtime

Check all 3rd party apps are available for the target version

Create a template script that you can reuse

PrintVis Upgrade example script

<https://github.com/printvis/PrintVis.OnPrem.UpgradeExample>

Cloud Migration

Upgrade from PrintVis 14 to Cloud - <https://github.com/printvis/PrintVis.CloudUpgrade>

Creating DEMO data in cloud

CDX tenant - <https://aka.ms/cdx>

Migrate from PrintVis Demo database

<https://printvis.com/blogpost/create-printvis-demo-environments-for-prospects-and-consultants/>

- Task
 - Create tenant through CDX
 - Perform cloud migration of a company

DevOps and Containers

Our setup

- Azure DevOps Project with multiple apps
- Pull-Requests
 - Code review
 - Build validation + test automation
 - Associated work items
- Deployments
 - Scheduled or on demand
 - Test demo and support environments
- Releases
 - Based on hotfix branches
 - Cloud app with validation
 - Onprem apps with runtime packages
 - Uploaded to Azure Blob Storage

Containers

- Using BcContainerHelper PowerShell module
- Test, demo and support environments
- Builds, releases, upgrades
- Test automation
- Development environments
- Using Traefik to securely expose all BC endpoints

Free monthly Azure credits with Visual Studio subscription

build

Consider

- Pipelines for build your extensions against next minor + major
- Test automation in some way
- Devops? Github with Al-Go?
- Catch errors before customer catch it.
- Less things to worry about
- Make someone (customer) pay for the time/cost (not our job... but it could be a cool project during lunch hours)

Bonus Extra

PVS Setup

- Cost Center
 - Cost Center Configuration
 - Operation
 - Speed table
 - Calculation Unit
 - With operation
 - Calculation Unit
 - With items
 - Calculation Unit List
 - Capacity unit
 - Opening hours
 - Test the Cost center until planning

Cost Center

[X](#) Reminder: your work date is 2022-04-11 [Use today](#) | [Change to...](#) | [Turn off reminder](#)

 Configurations  Fixed Asset  Item Group filter  Merge

General

Code 6300

Name

Type Machine 

Department

Department PRINTING 

Department Name Printing

Machine Type

Machine Type Printing machine 

Imaging

Printing Machine Digital 

Finishing

Shipment Miscellaneous



Configuration



✓ Saved

6300 · 6300C

X Reminder: your work date is 2022-04-11 Use today | Change to... | Turn off reminder

[Rates](#)[All Rates](#)[Copy from Configuration](#)[Create Capacity Unit](#)[Create Calculation Unit](#)[Calculation Units](#)[Scrap Table](#)[Tool Usage](#)

General

Cost Center Code	6300	Standard	<input checked="" type="checkbox"/>
Configuration	6300C	Type	Machine
Name		Machine Type	Printing Machine
Speed Table		Printing Machine	Digital
Scrap Table		Configuration Type	Basic
Blocked	<input checked="" type="checkbox"/>		

Operations

Manage

[New Line](#)[Delete Line](#)[Rates](#)[Speed Table](#)[Price List](#)

Operation ↑	Text	Sub Contracting	Unit of Measure	Formula Code	Speed Table	Price List Code	Price List Look-up	Include in Re-run	Include in Add. Q. Price
100	Make Ready	<input type="checkbox"/>		1111	6300C_100		Cost and Sales Prices	<input type="checkbox"/>	<input type="checkbox"/>
→ 200	: Production	<input type="checkbox"/>		14	6300C_200		Cost and Sales Prices	<input type="checkbox"/>	<input type="checkbox"/>

Machine

Configuration Sheet SUB

Max. Printing Format Length	20.0000
Max. Printing Format Width	20.0000
Max. Number of Colors	4
Colors Front	4
Colors Back	4
Varnish Work	<input checked="" type="checkbox"/>



Create Calculation unit from the configuration, make 2



Calculation Units



✓ Saved



Calc. Unit · 6300

Merge

General

Code	6300	List of Units	<input checked="" type="checkbox"/>
Text	Printer	Hide	<input checked="" type="checkbox"/>
External Text		Canceled	<input checked="" type="checkbox"/>
Department	PRINTING	No Surcharge Configurations	<input checked="" type="checkbox"/>
Department Name	Printing	Call Colors/Glue	<input checked="" type="checkbox"/>
Cost Center Code	6300	Global Unit	<input checked="" type="checkbox"/>
Cost Center Name	-	Surcharge to	Cost Center
Configuration	6300C	Surcharge to unit	
Process Type	Digital	Default Planning Unit	
Process Start	<input checked="" type="checkbox"/>	Distribution	No
Mandatory Process	<input checked="" type="checkbox"/>		

Detail

Manage

New Line Delete Line

Type	Cost Center Code	Configuration	Operation	Req...	Sub Con...	Item Type	Item No.	Text	Formula Code	Quantity	Speed	Hours	Standard	Incl... in Re-run	Incl... in Add. Q. Price	Price List Code
Operation	6300	6300C	100	<input type="checkbox"/>	<input type="checkbox"/>			Make Ready	1111		0,00		Yes	<input type="checkbox"/>	<input type="checkbox"/>	
→ Operation	6300	6300C	200	<input type="checkbox"/>	<input type="checkbox"/>			Production	14		0,00		Yes	<input type="checkbox"/>	<input type="checkbox"/>	

[Calculation Units](#)

✓ Saved

Calc. Unit · 6301

[X Reminder: your work date is 2022-04-11](#) [Use today](#) | [Change to...](#) | [Turn off reminder](#)

General

Code	6301	List of Units	<input checked="" type="checkbox"/>
Text	Printer Substrate	Hide	<input checked="" type="checkbox"/>
External Text		Canceled	<input checked="" type="checkbox"/>
Department	PRINTING	No Surcharge Configurations	<input checked="" type="checkbox"/>
Department Name	Printing	Call Colors/Glue	<input checked="" type="checkbox"/>
Cost Center Code	6300	Global Unit	<input checked="" type="checkbox"/>
Cost Center Name	-	Surcharge to	Cost Center
Configuration	6300C	Surcharge to unit	
Process Type	Digital	Default Planning Unit	
Process Start	<input checked="" type="checkbox"/>	Distribution	No
Mandatory Process	<input checked="" type="checkbox"/>		

[Detail](#) | [Manage](#)[New Line](#) [Delete Line](#)

Type	Cost Center Code	Configuration	Operation	Req...	Sub Con...	Item Type	Item No.	Text	Formula Code	Quantity	Speed	Hours	Standard	Incl... in Add. Q. Price	Price Li
→ Item	: 6300			<input type="checkbox"/>	<input type="checkbox"/>	Paper			14	0,00			Yes	<input type="checkbox"/>	<input type="checkbox"/>

[Calculation Units](#)[✓ Saved](#)

Calc. Unit

[X](#) Reminder: your work date is 2022-04-11 | [Use today](#) | [Change to...](#) | [Turn off reminder](#)

[Merge](#)

General

Code	P_6300	List of Units	<input checked="" type="checkbox"/>
Text	Printer	Hide	<input type="checkbox"/>
External Text	<input type="text"/>	Canceled	<input type="checkbox"/>
Department	<input type="text"/>	No Surcharge Configurations	<input type="checkbox"/>
Department Name	<input type="text"/>	Call Colors/Glue	<input type="checkbox"/>
Cost Center Code	<input type="text"/>	Global Unit	<input type="checkbox"/>
Cost Center Name	<input type="text"/>	Surcharge to	<input type="text"/> Cost Center
Configuration	<input type="text"/>	Surcharge to unit	<input type="text"/>
Process Type	<input type="text"/>	Default Planning Unit	<input type="text"/>
Process Start	<input checked="" type="checkbox"/>	Distribution	<input type="text"/> No
Mandatory Process	<input checked="" type="checkbox"/>		

[Units](#) | [Manage](#)

Unit Code	Unit Text	Sorting Order
→ 6300	Printer	0
6301	Printer Substrate	0

Show

Show in Case Management List

Hide during Calculation

Include in machine choice

Always

Show Capacity

Cost Center

6300

Configurations Fixed Asset Item Group filter Merge

General

Code 6300 Capacity Unit 6300

Name Printer Capacity Unit Name Printer

Capacity Unit Card

6300

Blockings Exceptions Capacity Calendar Statistics Merge

General

Code 6300 Calendar Group

Name Printer Standard Efficiency

Department PRINTING Team Manning 100,00

Department Group Planning Units 100,00

Group Code Simultaneous Jobs

Type Normal Planning Principles

Shared Capacity Unit

No Capacity Check (switch)

Time Rounding Up 0,00

Pool Planned (switch)

Time Rounding Down 0,00

Ongoing Shift Change (switch)

Extra Working Hours 0,00

Respect Open Hours (switch)

PrintVis Opening Hours Setup

Saved

Functions Navigate View

Apply Changes Copy To Period Copy From Period Undo (Unassigned) Apply All

Week Number	Week Day	Date ↑	Description	Default Calendar	Administration
13	tredag	2022-04-01			
13	lördag	2022-04-02			

Document Mgt

- Create document against Case (sales order, purchase order)
- Word template
- Merge fields
 - Predefined tables by setup
 - Possible to extend
- Sharepoint

Task 1 – Setup of sharepoint. Create document setup, add template, add std mergefields. Test

Task 2 – Use one of the examples from github

Document Mgt

```
[EventSubscriber(ObjectType::Table, Database::"PVS Merge Fields", 'OnMergeCustomEvent', '', false, false)]
local procedure OnMergeCustomEvent(var Global_DocRec: Record "PVS Document";
var outResultTxt: Text; var IsHandled: Boolean; Bookmark: Text)
var
    CaseRec: Record "PVS Case";
begin
    if Bookmark = 'SAMPLE' then begin //Bookmark from the mergefield setup
        ContentTxt := '';
        CaseRec.Get(Global_DocRec.ID1);
        outResultTxt := CaseRec."Quote No.";
    end;
end;
```

Thank You

See You Next Time!

