

Field Name	Field Type	Formula / Entry Option
Active EPP Proof #	Calculation (Text)	Unstored calculation = Active EPP Proof::EPP Proof #
Active EPP Proof Number of sigs	Calculation (Number)	Unstored calculation = Active EPP Proof::Number of Sigs
Active Oce File #	Calculation (Text)	Unstored calculation = Active OceFiles::Oce File #
Active Oce Proof #	Calculation (Text)	Unstored calculation = Active OceProof::Oce Proof #
Active Plate Set #	Calculation (Text)	Unstored calculation = Active PlateSet::Plate Set #
Active Plates QCed?	Calculation (Text)	Unstored calculation = If(IsEmpty(Active PlateSet::Initials Plate Checked), "No", "Yes")
Allow Orders?	Text	Auto-enter: "No" Value List (Custom Values): Yes No Required value Indexed
Assembly Instructions Exist	Calculation (Text)	Evaluate even if all referenced fields are empty Unstored calculation = If(AssemblyInst::Serial # > 0, "Show", "Create")
Available Sample	Calculation (Text)	Unstored calculation = Case(Master Samples::Part # = Part #, "Master Sample", Signed Proofs::Part # = Part #, "Signed Proof", Other Samples::Part # = Part #, Other Samples::Sample Type, "")
Average Min Max Run	Calculation (Number)	Indexed = Max - Min + (Min / 2)
Average Pull	Calculation (Number)	Unstored calculation = Average(Inventory Pulls::Quantity) * (-1)
Binding Type	Text	Indexed
BRC?	Text	Auto-enter: "No" Value List (Custom Values): Yes No Do not allow user to override validation Indexed
Buyout Turn days	Number	
Calc Novellus Price	Calculation (Number)	Unstored calculation = If(Part Category = "Book" and Novellus Pricing Category = "Clean Room", 5, "") + If(Part Category = "Book", 4, "") + Sum(SubParts::Novellus price)
Check for duplicate PN	Calculation (Text)	Unstored calculation = If(Concat Ext PN::Part # > 1, "Warning! Duplicate Customer PN.", "")
Colors side one	Number	Indexed
Colors side two	Number	Indexed
Combined Current Inventory	Calculation (Number)	Unstored calculation = Current Inventory + Current Remote Inventory
Combined Net Inventory	Calculation (Number)	Unstored calculation = Net Inventory + Net Remote Inventory
Concat Extended Part	Calculation (Text)	Indexed = Customer Abbreviation & " " & Concat Part Number
Concat Extended Part if Buyout	Calculation (Text)	Indexed = If(Production Path = "Buyout", Customer Abbreviation & " " & Concat Part Number, "")
Concat Part Number	Calculation (Text)	Indexed = If(IsEmpty(Customer Part Number), Part Description, Customer Part Number) & " " & If(not IsEmpty(Revision), "rev " & Revision & " ", "") & Part Category

Field Name	Field Type	Formula / Entry Option
Concat Specs	Calculation (Text)	<p>Unstored calculation  = Case(Production Path = "POD",  NumToText(Number of Pages) &amp; "pp. " &amp; NumToText(Final width) &amp; "x" &amp;  NumToText(Final length) &amp; " " &amp; Binding Type &amp; " " &amp; Part Category &amp; " on " &amp;  Paper standard name &amp; ".",</p> <p>Production Path = "Offset",  If(Number of Pages &gt;0, NumToText(Number of Pages) &amp; "pp. ", "") &amp;  NumToText(Final width) &amp; "x" &amp; NumToText(Final length) &amp; " " &amp; Binding Type  &amp; " " &amp; Part Category &amp; If(Shell Part # &gt; 0, " imprinted on " &amp; Shell concat part  number &amp; " shells", " printed on "&amp; Paper standard name &amp; " stock") &amp; ". Inks: "  &amp; Colors side one &amp; "/" &amp; Colors side two &amp;  " Flat Size: " &amp; Flat Width &amp; "x" &amp; Flat Length,</p> <p>Production Path = "Bindery",  Binding Type &amp; " " &amp; NumToText(Final width) &amp; "x" &amp; NumToText(Final length)  &amp; " Spine: " &amp; Top Thickness &amp; " inches."  )</p>
Concat Specs Export	Calculation (Text)	<p>Unstored calculation  = "¶" &amp; Part Category &amp; ":" &amp; "¶" &amp;</p> <p>Case(Production Path = "POD",  "Pages: " &amp; NumToText(Number of Pages) &amp; "¶" &amp;  "Paper: " &amp; Paper standard name,</p> <p>Production Path = "Offset",  If(Number of Pages &gt;0, "Pages: " &amp; NumToText(Number of Pages) &amp; "¶", "") &amp;  If(Shell Part # &gt; 0, "Shell: " &amp; Shell concat part number, "Paper: "&amp; Paper  standard name) &amp; "¶" &amp;  "Colors: " &amp; Colors side one &amp; "/" &amp; Colors side two &amp; "¶" &amp;  If(not IsEmpty(Inks Side One), "Inks: " &amp;  GetRepetition(Inks Side One, 1) &amp;  If(not IsEmpty(GetRepetition(Inks Side One, 2)), " " &amp; GetRepetition(Inks  Side One, 2), "") &amp;  If(not IsEmpty(GetRepetition(Inks Side One, 3)), " " &amp; GetRepetition(Inks  Side One, 3), "") &amp;  If(not IsEmpty(GetRepetition(Inks Side One, 4)), " " &amp; GetRepetition(Inks  Side One, 4), "") &amp;  If(not IsEmpty(GetRepetition(Inks Side One, 5)), " " &amp; GetRepetition(Inks  Side One, 5), "") &amp;  If(not IsEmpty(GetRepetition(Inks Side One, 6)), " " &amp; GetRepetition(Inks  Side One, 6), "") &amp;  If(not IsEmpty(GetRepetition(Inks Side One, 7)), " " &amp; GetRepetition(Inks  Side One, 7), "") &amp;  If(not IsEmpty(GetRepetition(Inks Side One, 8)), " " &amp; GetRepetition(Inks  Side One, 8), "") &amp;  If(not IsEmpty(GetRepetition(Inks Side Two, 1)), " / " &amp; GetRepetition(Inks  Side Two, 1), "") &amp;  If(not IsEmpty(GetRepetition(Inks Side Two, 2)), " " &amp; GetRepetition(Inks  Side Two, 2), "") &amp;  If(not IsEmpty(GetRepetition(Inks Side Two, 3)), " " &amp; GetRepetition(Inks  Side Two, 3), "") &amp;  If(not IsEmpty(GetRepetition(Inks Side Two, 4)), " " &amp; GetRepetition(Inks  Side Two, 4), "") &amp;  If(not IsEmpty(GetRepetition(Inks Side Two, 5)), " " &amp; GetRepetition(Inks  Side Two, 5), "") &amp;  If(not IsEmpty(GetRepetition(Inks Side Two, 6)), " " &amp; GetRepetition(Inks  Side Two, 6), "") &amp;  If(not IsEmpty(GetRepetition(Inks Side Two, 7)), " " &amp; GetRepetition(Inks  Side Two, 7), "") &amp;  If(not IsEmpty(GetRepetition(Inks Side Two, 8)), " " &amp; GetRepetition(Inks  Side Two, 8), "") &amp; "¶", "") &amp;  If(Spine &gt; 0, "Spine: " &amp; Spine, "") &amp;</p>

Field Name	Field Type	Formula / Entry Option
		"Flat Size: " & Flat Width & "x" & Flat Length,  Production Path = "Bindery", "Binding: " & If(Binding Type="Wire-O" or Binding Type="Comb", Wire Color & " ", "") & Binding Type & "¶" & "Final Size: " & NumToText(Final width) & "x" & NumToText(Final length) & If(Drill?="yes", "¶Drill: Yes" , "") & If(Shrinkwrap?="Yes", "¶Shrinkwrap: " & If(Shrink type = "Anti-Static", "Anti-Static", "Yes"), "") ) & "¶****"
Constant One	Calculation (Number)	= 1
Count Items	Summary (Number)	= Count of Part #
Count Pulls	Calculation (Number)	Unstored calculation = Count(Inventory Pulls::Quantity)
Create First Article?	Global (Text)	
Create Proofs?	Global (Text)	
Creation Date	Date	Auto-enter the: "Creation Date" Prevent data that is automatically entered from being changed. Indexed
Creation Time	Time	Auto-enter the: "Creation Time" Prevent data that is automatically entered from being changed.
Creator	Text	Auto-enter the: "Creator Name" Prevent data that is automatically entered from being changed. Indexed
Current Inventory	Calculation (Number)	Unstored calculation = Sum(InventoryLots::Current Quantity)
Current Remote Inventory	Calculation (Number)	Unstored calculation = Remote Adds + Remote Adjustments - Remote Top Pulls - Remote Sub Pulls
Customer #	Text	Value List (By Field): "Customer #" from file "Customers.fp5" Do not allow user to override validation Required value Indexed
Customer # if Buyout	Calculation (Text)	Indexed = If(Production Path = "Buyout", Customer #, "")
Customer # if Offset	Calculation (Text)	Indexed = If(Production Path = "Offset", Customer #, "")
Customer # if POD	Calculation (Text)	Indexed = If(Production Path = "POD", Customer #, "")
Customer # Mirror	Calculation (Text)	Indexed = Customer #
Customer # Orderable	Calculation (Text)	Indexed = If(Allow Orders? = "Yes", Customer #, "")
Customer Abbrev Mirror	Calculation (Text)	Indexed = Customer Abbreviation
Customer Abbreviation	Text	Lookup: Use relationship "Customers" "Customer #" = "Customers::Customer #" If exact match, copy "Customers::Customer Abbreviation" don't copy "Customers::Customer Abbreviation" if empty If no match: "Do not Copy" Indexed
Customer Name	Text	Lookup: Use relationship "Customers" "Customer #" = "Customers::Customer #" If exact match, copy "Customers::Customer Name" don't copy "Customers::Customer Name" if empty If no match: "Do not Copy" Indexed
Customer Name Mirror	Calculation (Text)	Indexed = Customer Name

Field Name	Field Type	Formula / Entry Option
Customer Owned?	Text	Auto-enter: "No" Value List (Custom Values): Yes No Indexed
Customer Part # if Orderable	Calculation (Text)	Indexed = If(Allow Orders? = "Yes", Concat Part Number, "")
Customer Part Number	Text	Indexed
Customer Part Number Mirror	Calculation (Text)	Indexed = Customer Part Number
Default Carton Material #	Text	
Default Carton Material Code	Calculation (Number)	Unstored calculation = Materials::Material Code
Default CSR Employee #	Calculation (Text)	Unstored calculation = Customers::CSR Employee #
Default Supplier #	Text	Indexed
Default Supplier Code	Calculation (Text)	Unstored calculation = Suppliers::Supplier name
Drill for	Text	Indexed
Drill Size	Text	
Drill?	Text	Lookup: Use relationship "Specs" "Spec Template #" = "Specs::Spec Template #" If exact match, copy "Specs::Drill?" don't copy "Specs::Drill?" if empty If no match: "Do not Copy"
Final length	Number	Indexed
Final width	Number	Indexed
First Article Due Date	Global (Date)	
First Article Due Time	Global (Time)	
First Article Qty	Global (Number)	
Flat Length	Calculation (Number)	Unstored calculation = If(not IsEmpty(Override Flat Length),Override Flat Length,  Case(Part Category = "Cover",Final length, ""))
Flat Width	Calculation (Number)	Unstored calculation = If(not IsEmpty(Override Flat Width),Override Flat Width, Case(Part Category = "Cover" and Binding Type = "Perfect Bind", Spine + 2 * Final width, Part Category = "Cover" and Binding Type = "Saddle Stitch", 2 * Final width, Part Category = "Cover" , Final width , ""))
Free Date 1	Date	
Free Number 1	Number	
Free Number 2	Number	
Free Number 3	Number	
Free Text 1	Text	
Free Text 2	Text	
Free Text 3	Text	
Free Time 1	Time	
Global Date	Global (Date)	
Global Number 1	Global (Number)	
Global Number 2	Global (Number)	
Global Number 3	Global (Number)	
Global Text 1	Global (Text)	

Field Name	Field Type	Formula / Entry Option
Global Text 2	Global (Text)	
Global Text 3	Global (Text)	
Inks Side One	Text	Repeating field with 8 repetitions Indexed
Inks Side Two	Text	Repeating field with 8 repetitions Indexed
Inventory?	Text	Indexed
iUniverse Link	Calculation (Text)	= If(iUniverse Orderable? = "yes", Customer Part Number,"")
iUniverse Orderable?	Text	Auto-enter: "No" Indexed
iUniverse Pages	Number	
iUniverse Price	Number	
Label at SW?	Text	Auto-enter: "No" Indexed
Laminate?	Text	Auto-enter: "No" Value List (Custom Values): Yes No Required value Indexed
Mark for conversion	Text	Indexed
Max	Number	Indexed
Min	Number	Indexed
Mirror part description	Calculation (Text)	Indexed = Part Description
Modification Date	Date	Auto-enter the: "Modification Date" Prevent data that is automatically entered from being changed. Indexed
Modification Time	Time	Auto-enter the: "Modification Time" Prevent data that is automatically entered from being changed. Indexed
Modifier	Text	Auto-enter the: "Modifier Name" Indexed
Net Inventory	Calculation (Number)	Unstored calculation = Current Inventory + Pending Adds - Pending Pulls
Net Remote Inventory	Calculation (Number)	Unstored calculation = Current Remote Inventory + Remote Pending Adds - Remote Pending Sub Pulls - Remote Pending Top Pulls
Notes	Text	Indexed
Novellus Pricing Category	Text	Value List (Custom Values): Clean Room Training Tech Pubs Problem  Indexed
Number of active EPP proofs	Calculation (Number)	Unstored calculation = Sum(EPPProofs::One if NotRejected)
Number of active Oce Files	Calculation (Number)	Unstored calculation = Sum(Active OceFiles::One If Active)
Number of active Oce proofs	Calculation (Number)	Unstored calculation = Sum(OceProofs::One If Active)
Number of Pages	Number	Indexed
Number of Pages of texts	Calculation (Number)	Unstored calculation = Sum(SubParts Text only::Number of Pages)
Number of Sigs of Text	Calculation (Text)	Unstored calculation = SubParts Text only::Number of Sigs from EPP proof

Field Name	Field Type	Formula / Entry Option
NVT Category	Text	Lookup: Use relationship "Courses" "NVT Course ID" = "Courses::Course ID" If exact match, copy "Courses::Category" don't copy "Courses::Category" if empty If no match: "Do not Copy"
NVT Category ID	Text	Lookup: Use relationship "Courses" "NVT Course ID" = "Courses::Course ID" If exact match, copy "Courses::Category ID" don't copy "Courses::Category ID" if empty If no match: "Do not Copy"
NVT Course Description	Text	Lookup: Use relationship "Courses" "NVT Course ID" = "Courses::Course ID" If exact match, copy "Courses::Course Description" don't copy "Courses::Course Description" if empty If no match: "Do not Copy"
NVT Course ID	Text	
NVT Item Description	Text	Indexed
NVT Item ID	Text	
NVT Notes	Text	
NVT Website?	Text	Indexed
Override Flat Length	Number	Indexed
Override Flat Width	Number	Indexed
Override top shipping weight	Number	Indexed
Override Unit Cost	Number	
Override WireO Guage	Text	Indexed
Pages for drilling	Calculation (Number)	Unstored calculation = Sum(SubParts::Number of Pages)
Paper standard #	Text	Indexed
Paper standard name	Calculation (Text)	Unstored calculation = PaperStandards::Standard Name
Paper Standard Name of Text	Calculation (Text)	Unstored calculation = SubParts Text only::Paper standard name
Part #	Text	Serial Number with Current Value: "P27239" Increment: "1" Prevent data that is automatically entered from being changed. Indexed
Part # if buyout	Calculation (Text)	Indexed = If(Production Path = "Buyout", Part #, "")
Part # unless buyout	Calculation (Text)	Indexed = If(Production Path ≠ "Buyout", Part #, "")
Part as number	Calculation (Number)	Indexed = TextToNum(Part #)
Part Category	Text	Indexed
Part Category Mirror	Calculation (Text)	Indexed = Part Category
Part Description	Text	Indexed
Part Description Mirror	Calculation (Text)	Indexed = Part Description
Parts List	Global (Text)	
Pending Adds	Calculation (Number)	Unstored calculation = Sum(WorkOrders::Pending Inventory Add) + Sum(POLine Adds::Pending Inventory Add)
Pending Pulls	Calculation (Number)	Unstored calculation = Sum(PendingPulls::Qty Pending)
Plate set # Display	Calculation (Text)	Unstored calculation = Active PlateSet::Plate set # Display
Product Family	Text	Indexed

Field Name	Field Type	Formula / Entry Option
Production Path	Text	Indexed
Proof Due Date	Global (Date)	
Proof Due Time	Global (Time)	
Proof Qty	Global (Number)	
Pulls in Min	Calculation (Number)	Unstored calculation = Round(Min / Average Pull,1)
Pulls per Run	Calculation (Number)	Unstored calculation = Round(Average Min Max Run / Average Pull,1)
Qty Over Max	Calculation (Number)	Unstored calculation = Case(Max > 0 and Current Inventory > Max, Current Inventory - Max, Max > 0, 0,"")
Qty Under Min	Calculation (Number)	Unstored calculation = Case(Min > 0 and Current Inventory < Min, Min - Current Inventory, Min > 0,0,"")
Receive overs percentage	Number	Auto-enter calculation = If(Production Path = "Buyout", .05,"")
Remote Adds	Calculation (Number)	Unstored calculation = Sum(RemoteAdds::Actual Quantity) + Sum(RemotePendingAdds::Actual Quantity)
Remote Adjustments	Calculation (Number)	Unstored calculation = Sum(RemoteAdjusts::Actual Quantity)
Remote Pending Adds	Calculation (Number)	Unstored calculation = Sum(RemotePendingAdds::Balance)
Remote Pending Sub Pulls	Calculation (Number)	Unstored calculation = Sum(SubParts reversed::Remote Pending Pulls)
Remote Pending Top Pulls	Calculation (Number)	Unstored calculation = Sum(RemotePendingPulls::Balance)
Remote Sub Pulls	Calculation (Number)	Unstored calculation = Sum(SubParts reversed::Remote Pulls)
Remote Top Pulls	Calculation (Number)	Unstored calculation = Sum(RemotePulls::Actual Quantity) + Sum(RemotePendingPulls::Actual Quantity)
Remote Total Pending Pulls	Calculation (Number)	Unstored calculation = Remote Pending Sub Pulls + Remote Pending Top Pulls
Revision	Text	Indexed
Revision Mirror	Calculation (Text)	Indexed = Revision
Run a quantity?	Global (Text)	
Run Due Date	Global (Date)	
Run Due Time	Global (Time)	
Run Qty	Global (Number)	
Sched Block Notes	Text	Indexed
Self Cover	Text	Value List (Custom Values): Yes No Do not allow user to override validation Indexed
Shell concat part number	Calculation (Text)	Unstored calculation = Self Shell::Concat Part Number
Shell Part #	Text	Indexed
Shell popup	Calculation (Text)	Indexed = If(Part Category = "Shell", Concat Part Number,"")
ShellOrPaper	Calculation (Text)	= If(not IsEmpty(Shell Part #), "Shell", "Paper")
Shipping Weight of Paper	Calculation (Number)	Unstored calculation = Case(Part Category = "Text", Round(Number of Pages /2, 0) * Final width * Final length *PaperStandards::Std lbs per square inch, Part Category = "Cover", Flat Width * Flat Length * If(Binding Type = "Wire-O", 2,1) * PaperStandards::Std lbs per square inch)

Field Name	Field Type	Formula / Entry Option
Show on Website?	Text	Auto-enter: "No" Value List (Custom Values): Yes No Required value Indexed
Shrink type	Text	Auto-enter calculation = If(Shrinkwrap? = "Yes", "Standard", "") Indexed
Shrinkwrap Layers	Number	Lookup: Use relationship "Specs" "Spec Template #" = "Specs::Spec Template #" If exact match, copy "Specs::Shrinkwrap Layers" don't copy "Specs::Shrinkwrap Layers" if empty If no match: "Do not Copy"
Shrinkwrap?	Text	Lookup: Use relationship "Specs" "Spec Template #" = "Specs::Spec Template #" If exact match, copy "Specs::Shrinkwrap?" don't copy "Specs::Shrinkwrap?" if empty If no match: "Do not Copy" Indexed
Side Glue Override	Text	Value List (Custom Values): Yes No Do not allow user to override validation Indexed
Side Glue?	Calculation (Text)	= Case(Binding Type ≠ "Perfect Bind", "", Side Glue Override = "No", Side Glue Override, "Yes")
Sig Count active proof	Calculation (Number)	Unstored calculation = Active EPP Proof::Number of Sigs
Sort Sequence	Number	
Spec Template #	Text	
Specs Export Description	Calculation (Text)	= Concat Extended Part & "¶" & Part Description & "¶****"
Spine	Calculation (Number)	Unstored calculation = Round(If(Part Category = "Cover" and Binding Type = "Perfect Bind", SubParts reversed::Spine, ""), 2)
Standard buy price	Number	Indexed
Standard qty per carton	Number	
Status	Text	Auto-enter: "Active" Value List (Custom Values): Active Obsolete Indexed
Status Date	Date	Auto-enter the: "Creation Date"
Status Mirror	Calculation (Text)	Indexed = Status
Sum Unit Cost Extended	Summary (Number)	= Total of Unit Cost Extended
Sum Unit Price Extended	Summary (Number)	= Total of Unit Price Extended
Sum Unit Valuation Extended	Summary (Number)	= Total of Unit Valuation Extended
Temp Job #	Global (Text)	
Temp PN	Calculation (Number)	Indexed = TextToNum(Part #)
Temp Status	Global (Text)	
Temp Work Order #	Global (Text)	
Template Type	Text	
Text Type	Text	Lookup: Use relationship "Specs" "Spec Template #" = "Specs::Spec Template #" If exact match, copy "Specs::Text Type" don't copy "Specs::Text Type" if empty If no match: "Do not Copy"



Field Name	Field Type	Formula / Entry Option
Thickness	Calculation (Number)	Unstored calculation = Case(Part Category = "Text" or Part Category = "Foldout Set", PaperStandards::Std Caliper * Round(Number of Pages / 2, 0) , Part Category = "Cover" and Shell Part # > 0, Thickness of Shell, Part Category = "Cover", Thickness as paper, Part Category = "Shell", Thickness as paper )
Thickness as paper	Calculation (Number)	Unstored calculation = PaperStandards::Std Caliper *  Case(Part Category = "Text", Round(Number of Pages / 2, 0) , Part Category = "Cover", 2, Part Category = "Shell", 2 )
Thickness of Shell	Calculation (Number)	Unstored calculation = Self Shell::Thickness
Top Shipping Weight	Calculation (Number)	Unstored calculation = If(Override top shipping weight > 0, Override top shipping weight, Sum(SubParts::Shipping Weight))
Top Thickness	Calculation (Number)	Unstored calculation = Sum(SubParts::Thickness)
Unit Cost	Number	Indexed
Unit Cost Extended	Calculation (Number)	Unstored calculation = If(Override Unit Cost > 0, Override Unit Cost, Unit Cost) * Current Inventory
Unit Price Extended	Calculation (Number)	Unstored calculation = Unit Sell Price * Current Inventory
Unit Sell Price	Number	Indexed
Unit Valuation	Calculation (Number)	Indexed = Case(Unit Cost > 0, Unit Cost, Unit Sell Price > 0, Unit Sell Price * .6, 0)
Unit Valuation Extended	Calculation (Number)	Unstored calculation = Unit Valuation * Current Inventory
Use Inventory Concat Part Number	Calculation (Text)	Unstored calculation = Use Inventory Part::Concat Extended Part
Use Inventory Net Inventory	Calculation (Number)	Unstored calculation = Use Inventory Part::Net Inventory
Use Inventory Part #	Text	Indexed
Wire Color	Text	Value List (Custom Values): Black White Blue
WireO Gauge	Calculation (Text)	Unstored calculation = If(not IsEmpty(Override WireO Gauge), Override WireO Gauge,  Case(WireO Sixteenths = "", "", WireO Sixteenths ≤ 4, "1/4", WireO Sixteenths = 5, "5/16", WireO Sixteenths = 6, "3/8", WireO Sixteenths = 7, "7/16", WireO Sixteenths = 8, "1/2", WireO Sixteenths = 9, "9/16",  WireO Sixteenths = 10, "5/8", WireO Sixteenths ≤ 12, "3/4", WireO Sixteenths ≤ 14, "7/8", WireO Sixteenths ≤ 16, "1", WireO Sixteenths ≤ 20, "1 1/4", "error"))

Field Name	Field Type	Formula / Entry Option
WireO Loops per Inch	Calculation (Text)	Unstored calculation = Case(Binding Type = "Comb", "4 per inch", WireO Sixteenths = "", "", WireO Sixteenths ≤ 9, "3 per inch", WireO Sixteenths ≤ 20, "2 per inch", "error")
WireO Sixteenths	Calculation (Number)	Unstored calculation = If((Binding Type = "Wire-O" or Binding Type = "Comb") and Part Category = "Book", Truncate(Sum(SubParts::Thickness) *16,0) + 3,"")
Work Order Instructions	Text	Indexed
Work Order Instructions Mirror	Calculation (Text)	= Work Order Instructions
g_Status	Global (Text)	
g_Work Order Type	Global (Text)	
g_Customer # From Job	Global (Text)	
History Concat	Calculation (Text)	Unstored calculation = Part # & "   " & g_Status & "   " & g_Work Order Type & "   " & g_Customer # From Job
Epac Owned Part #	Text	Indexed