




GS524 – Embedding Business Rules and Exceptions in Globalization Studio

Compliance rules vary. Some customers want PDFs, others require XML. Some invoices must be signed, others not. Dynamics 365 Globalization Studio enables you to embed those differences as **rules**, not code.

In this article, you'll learn how to:

- Embed business-specific logic and exceptions into your features
- Dynamically control steps in the **processing pipeline** using **context-aware applicability rules**
- Add validations directly in **Format Designer**
- Build localization logic that adapts to different business cases

 Builds on: [GS504 – Applicability Rules](#), [GS514 – Output Routing](#), [GS525 – Validation and Storage](#)

Why Add Business Logic?

Real-world compliance is **not one-size-fits-all**. You need to reflect:

Type	Example
Country rules	Only Spain uses FACe
Customer exceptions	B2G needs digital signature, B2B does not
Invoice thresholds	Sign invoices only if > €500
Delivery logic	Exports vs. domestic

Hardcoding these rules in X++ creates rigidity. Using **Globalization Studio + ER**, they become **configurable and transparent**.



← Save Validate Options 🔍

V3_Spanish electronic invoice (ES) 19 : Sales invoice gen... | Standard view ▾

Feature version setup

Processing pipeline

Applicability rules

Variables

Parameters

Set up applicability rule
+ New Delete Group clause Ungroup clause

<input type="radio"/>	<input type="radio"/>	And/or	Field	Operator type	Value	Data type
<input type="radio"/>	<input type="radio"/>	And	Country/ISOCode	Equals	ES	string
			DocumentType	Equals	Customer invoice	string
			InvoiceAmount	Less than	500	number

Step-by-Step: Add Pipeline Applicability Rules using Context

☒ Step 1: Find Context Configuration

1. Go to **Electronic Feature > Click Application Setup**

Note - The **Application setup** is essential for:

- Automating electronic invoice creation and submission
- Ensuring regulatory compliance
- Supporting multiple document types and legal entities
- Driving dynamic behavior using context and validation rules

Feature

Feature version

ISO Country/region codes

Base feature

Base version

V3_Spanish electronic invoice...

19

ES

Spanish elec...

Microsoft

3

Application setup is active

☒ No

Versions Configurations Feature parameters **Setups** Tags

+ Add ▾ Delete Edit Application setup

Feature setup	Description	Setup type
Sales invoice generation (ES) derived (1)	Sales invoice generation (ES) derived (1)	Processing pipeline
Project invoice generation (ES) derived (1)	Inherited feature setup from 'Project invoice ge...	Processing pipeline

2. Check **Context** name. Remember – this is by document type



Table name	Context	Electronic document mapping	Override el...
Customer invoice journal	Customer invoice context...	Invoice model mapping	Customer Invoice
Project invoice	Customer invoice context...	Invoice model mapping	Project Invoice

3. Context variables

👉 This defines dynamic runtime values available for rules.

As seen in your screenshot, these values are later used in applicability rules to filter steps by country, channel, or entity.

Context

A Context defines runtime values that are used by: Applicability rules, Pipelines, Processing logic

Contexts

Name	Configuration version	Shared
Customer invoice context	Customer invoice conte... 55	Shared

Items

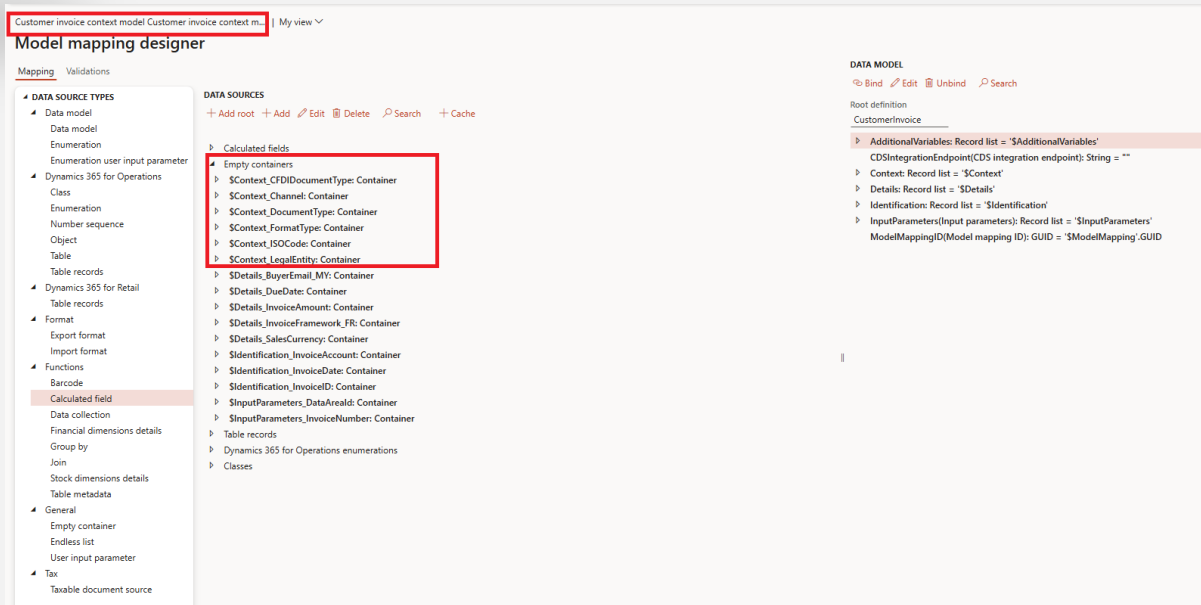
Name	Type	Value
"SubmissionType"	"string"	CASE(FIRSTORNULL(CustInvoiceJour.'<Relations'.EInv...
"Channel"	"string"	"IndEmail"
"DocumentType"	"string"	IF('\$Context_ISOCode'.Value="SA", IF(CustInvoiceJour....
"CountryISOCode"	"string"	SysCountryRegionCode.countryInfo()
"LegalEntityID"	"string"	CustInvoiceJour.DataAreald

Identification

Name	Type	Value
"InvoiceAccount"	"string"	CustInvoiceJour.InvoiceAc...
"InvoiceDate"	"dateTime"	DATEFORMAT(CustInvoic...
"InvoiceID"	"string"	CustInvoiceJour.InvoiceId

This section defines unique identifiers for each transaction instance. These are critical for: Tracking execution, Logging submission results, Linking with Submission History, Supporting retry and audit logs

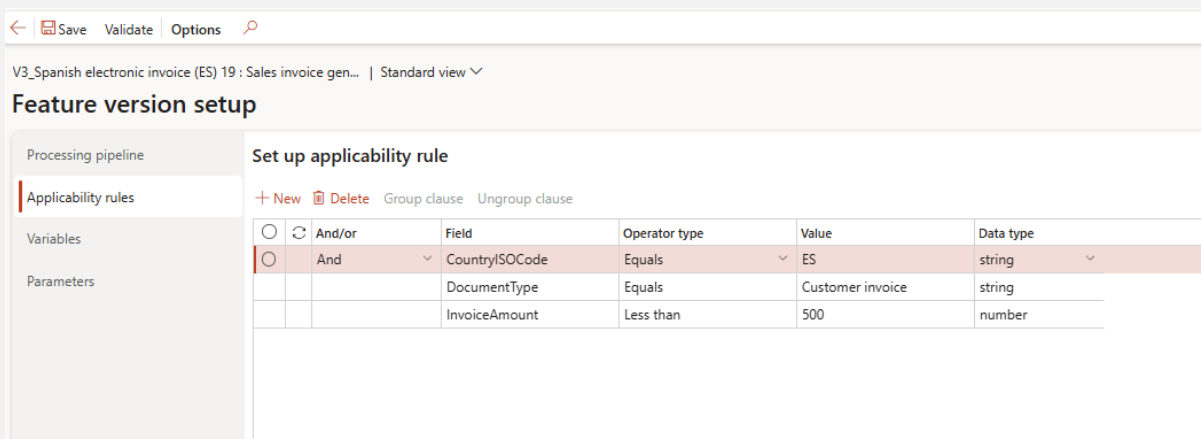
4. Context variables are located under empty container in ER configuration



5. You can extend these variables as per requirement or add new once.

☑ Step 2: Use the Context in Applicability Rules

1. Go to **Globalization Studio > Feature > Applicability Rules**
2. Add a rule like:



3. This restricts the **entire feature** to apply only for invoices in Spain

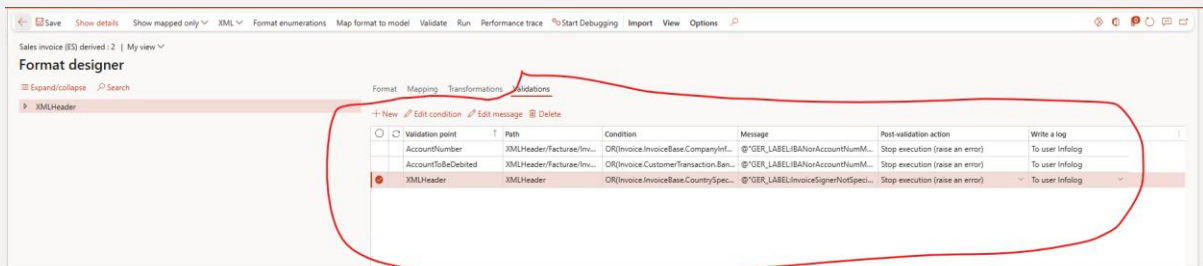
🔧 Step-by-Step: Add Validation in Format Designer

In **Format Designer**, validations are used to enforce business logic on the XML structure. For example, validating IBAN numbers or signer parameters.



☑ Step 1: Open Format Designer

1. Go to **Electronic Reporting > Configurations**
2. Select your format (e.g., Sales Invoice (ES))
3. Navigate to the **Validations** tab



☑ Step 2: Add Validation Conditions

Your screenshot includes the following conditions:

Validation Point

Condition Sample

Function

AccountNum
ber

OR(Invoice.InvoiceBase.CompanyInfo.BankAccount.IBAN <>
"",
Invoice.InvoiceBase.CompanyInfo.BankAccount.AccountNum
<> "")

This validation ensures that the **company's bank account details** are present in the invoice, either:

- An **IBAN**, or
- A **local bank account number**

◆ Why It's Needed:

For Spanish e-invoicing (e.g., submission to FACe), it is often mandatory to include **payment account**



Validation
Point

Condition Sample

Function

details of the
issuer
(supplier/company) in the XML.
Missing this
could result in:

- Submission rejection by government platforms
- Invalid invoice formats

This validation ensures that the **company's bank account details** are present in the invoice , either:

- An **IBAN**, or
- A **local bank account number**

AccountToBe
Debited

OR(Invoice.CustomerTransaction.BankAccount.IBAN <> "",
Invoice.CustomerTransaction.BankAccount.AccountNum <> "")

◆ **Why It's
Needed:**

For Spanish e-invoicing (e.g., submission to FAcE), it is often mandatory to



Validation Point	Condition Sample	Function
		<p>include payment account details of the issuer (supplier/company) in the XML. Missing this could result in:</p> <ul style="list-style-type: none">• Submission rejection by government platforms• Invalid invoice formats
XMLHeader	<p>OR(Invoice.InvoiceBase.CountrySpecificData.ElInvoiceParameters_IT.ElInvoiceSignerType_ES=</p> <p>Enums.ModelElInvoiceSignerType_ES.Issuer,</p> <p>Invoice.InvoiceBase.CountrySpecificData.ElInvoiceParameters_IT.ElInvoiceSignerType_ES=</p> <p>Enums.ModelElInvoiceSignerType_ES.Receiver,</p> <p>Invoice.InvoiceBase.CountrySpecificData.ElInvoiceParameters_IT.ElInvoiceSignerType_ES=</p> <p>Enums.ModelElInvoiceSignerType_ES.ThirdParty)</p>	<p>Validates that the invoice signer type (ElInvoiceSignerType_ES) is one of the allowed values:</p> <ul style="list-style-type: none">• Issuer• Receiver• ThirdParty <p>◆ Why It's Needed:</p> <p>The Spanish e-invoice model expects the</p>



Validation
Point

Condition Sample

Function

invoice XML to
include the
correct **signer
classification**,
depending on:

- Whether
the
invoice
is self-
billed
- Who
generat
ed or
signed
the
docume
nt
- Whether
a third
party is
involved
in the
process

This validation
ensures that a
valid **enum
value** is
selected ,
otherwise the
XML may:

- Be
structur
ally
invalid
- Get
rejected
by the
Spanish
e-



Validation
Point

Condition Sample

Function

invoicin
g web
service

- If these conditions fail, the export will be blocked and a validation error will appear.
- 🔧 Tip: Use **Edit Message** to provide a clear business-friendly error message like:
“ ⚠ IBAN or Account Number is required for customer payment details.”

Path : Specifies the node or section in the XML format where the validation applies.

This is the error message shown when the validation fails. Alerts the user to why the document was blocked or rejected during generation or submission.

Defines what the system should do when the condition is not met.

Determines where to record the validation result if the condition fails.

📁 Advanced Use Case: Mixed B2B/B2G Customers in Spain

Contoso Spain sells to:

- **Public sector (B2G)** → requires Facturae XML + signature + FAcE submission
- **Private sector (B2B)** → needs only PDF via email

How to handle both in one pipeline:

Step	Applicability Condition
Sign	Context.CustomerGroup = "B2G"
Submit	Context.CustomerGroup = "B2G"
Store to Azure	Context.CustomerGroup = "B2G"
BDM PDF	Context.CustomerGroup = "B2B"



Step	Applicability Condition
Email	Context.CustomerGroup = "B2B"

Best Practices for Managing Rules

Tip	Why It Helps
Name context variables clearly	IsExport, NeedsSignature, CustomerGroup
Keep validations user-readable	Auditors can understand failure reason
Separate logic cleanly	Don't mix country + customer logic
Use comments in rules	Explain why the rule exists
Test edge cases	Ensure robustness (nulls, missing fields)

Where to Monitor Rule Outcomes

Tool	View
Pipeline Execution Log	See if step was executed or skipped
ER Execution Log	See calculated values, e.g., IsB2G
Submission History	Result per document, incl. validation failures
Power BI Audit Report	End-to-end compliance view

Related Articles

- [GS504 – Applicability Rules](#)
- [GS514 – Output Storage](#)
- [GS517 – Logs and Errors](#)

Coming Up Next

In [GS525 – Document Archival and Compliance Audits in Globalization Studio](#), we'll explain how to securely store signed documents and submission logs for long-term audit compliance. You'll learn how to:



- Use Azure Key Vault and Blob Storage for secure retention
- Configure SharePoint for business-friendly document access
- Apply metadata and retention policies based on country rules
- Enable audit-ready traceability across all compliance features

 Continue reading: [GS525 – Document Archival & Compliance Audits](#) →