**1. Create a New Crystal Report**

1. **Open Crystal Reports.**
2. **Go to**: *File* → *New* → *Blank Report* (or “Standard Report Wizard” if preferred).
3. When prompted, you will **choose or create a database connection**:
   * Expand *Create New Connection* → *OLE DB (ADO)* or *ODBC (RDO)*, depending on how you normally connect.
   * Enter your server name, database name, and credentials.

You should now have a new report with a data connection, but no data source defined yet.

**2. Create Parameters in Crystal Reports**

In your SSRS report, you had the following parameters:

1. **StartDate** (DateTime)
2. **EndDate** (DateTime)
3. **Division** (Text; effectively a string or number, e.g. 2 or 0)
4. **OrdStat** (Text; can be "invoiced", "complete")

We will reproduce these in Crystal:

1. **In the Field Explorer** (usually on the right side of the Crystal Reports window):
   * Right-click **Parameter Fields** → *New*.
   * Name it StartDate.
     + **Value Type** = *Date* (or *DateTime*, whichever suits your data).
     + Under **Options**, you can specify *Prompt Text*, e.g., “Enter Start Date”.
   * Repeat to create EndDate, Division, and OrdStat.
     + For **Division**, if it’s numeric in your database, set **Value Type** to *Number*. Otherwise, set to *String* (if your “division” in the DB is a string).
     + For **OrdStat**, set **Value Type** = *String*. You can also set a list of default values (like *invoiced*, *complete*).

You now have four parameters in the report. Next, you need to use them in your query/command.

**3. Use a Command Object (SQL Query) in Crystal**

Because your SSRS solution uses a conditional query (the IF…ELSE block depending on @OrdStat), the simplest equivalent is to embed the logic directly in a SQL “Command” in Crystal.

**3.1 Add a Command**

1. **Back in the Database Expert**:
   * Under your connection, right-click *Add Command* (or “New Command”).
2. A SQL editor window will appear. This is where you can write your SQL, referencing the parameters we just created in Crystal.

**3.2 Write the Conditional SQL**

Normally, Crystal does *not* support a T-SQL style IF…ELSE block *inside* the Command if you are just using a single piece of raw SQL. Instead, you have two main approaches:

**Approach A:** Use a Stored Procedure or a Table-Valued Function that implements the IF/ELSE logic in T-SQL, and simply call that procedure from Crystal.

**Approach B:** Write two separate queries and switch them with a CASE or parameter logic in the WHERE clause. For example, you can do something like:

sql

CopyEdit

SELECT

orders.order\_id as order\_id,

orders.status as orders\_status,

ar\_out\_header.transaction\_date as invoice\_date,

CONVERT(VARCHAR(20), pros.delivered\_on, 106) as delivery\_date,

CASE WHEN orders.billing\_group = 0 THEN 'Total Logistics Control'

ELSE 'Total Logistics Trucking' END as division,

ISNULL(companies.company\_name,'') as customer,

ISNULL(dbo.getRev('us', orders.order\_id), 0) as revenue\_us,

ISNULL(dbo.getRev('cdn', orders.order\_id), 0) as revenue\_cdn,

ISNULL(dbo.getCost('us', orders.order\_id), 0) as cost\_us,

ISNULL(dbo.getCost('cdn', orders.order\_id), 0) as cost\_cdn,

1 as counter

FROM orders WITH (NOLOCK)

LEFT JOIN ar\_out\_header WITH (NOLOCK)

ON orders.invoice\_id = ar\_out\_header.header\_number

JOIN companies WITH (NOLOCK)

ON companies.company\_id = orders.customer

JOIN (

SELECT order\_id, MAX(delivered\_on) as delivered\_on

FROM probills WITH (NOLOCK)

WHERE status = 'delivered'

AND ISNULL(last\_leg,'Y') = 'Y'

GROUP BY order\_id

) pros

ON orders.order\_id = pros.order\_id

WHERE

pros.delivered\_on >= {?StartDate}

AND pros.delivered\_on <= {?EndDate}

AND orders.billing\_group <> {?Division}

AND orders.service\_id <> 'No Charge'

AND (

-- if OrdStat = 'invoiced' then ar\_out\_header.transaction\_date > EndDate

({?OrdStat} = 'invoiced' AND ar\_out\_header.transaction\_date > {?EndDate})

OR

-- else scenario (i.e. if OrdStat != 'invoiced') => ar\_out\_header.transaction\_date = '2001-01-01'

({?OrdStat} <> 'invoiced' AND ISNULL(ar\_out\_header.transaction\_date, '2001-01-01') = '2001-01-01')

)

ORDER BY ISNULL(companies.company\_name, '')

Note:

* This uses a single query with a WHERE clause that checks what the parameter OrdStat is. If OrdStat is 'invoiced', it enforces one condition. Otherwise, it enforces the other condition.
* You no longer need a T-SQL IF…ELSE; you handle it in the WHERE logic with OR conditions.

**3.3 Referencing Crystal Parameters in the Command**

To reference a Crystal parameter inside a Command query:

1. Create the parameter (which you did in Step 2).
2. In the Command text, insert {?ParameterName} wherever you’d normally have a T-SQL variable reference.
3. Once you click *OK*, Crystal will try to link these references to your actual parameter fields.

**4. Verify and Finish Command Setup**

1. **Click OK** in the Command editor, and you will be prompted to enter initial values for your parameters so Crystal can run the query.
2. Once the query is accepted, the fields returned by the query should appear under **Database Fields** → *Command* in the **Field Explorer**.

**5. Place Fields on the Report**

1. Open the **Field Explorer** → expand *Database Fields* → expand *Command* (the name of your new command).
2. Drag and drop the fields you want onto the report “Details” section (or into Group sections, etc.).
3. For example, drag order\_id, invoice\_date, delivery\_date, etc. onto the Details band.

**6. Parameter Prompting and Testing**

1. When you **refresh** the report (or preview it), Crystal will prompt for the four parameters.
2. Enter some test values— for instance:
   * StartDate = 2023-01-01
   * EndDate = 2023-12-31
   * Division = 2
   * OrdStat = “invoiced”
3. Confirm that your data (the rows that come through) matches expectations.

**7. Optional: Grouping, Summaries, Totals**

You can replicate some grouping or summarizing from your SSRS version:

1. **Create a Group** by going to: *Insert* → *Group…*
   * For example, group on division or customer.
2. **Add summary totals** (like sum of revenue\_us, etc.) by:
   * Right-clicking the field → *Insert* → *Summary*.
   * Choose “Sum” as the operation, and place it in the Group Footer or Report Footer.

**8. Format and Finalize**

1. Adjust fonts, spacing, and alignment as desired.
2. Insert page headers, titles, or date/time fields from *Insert* → *Text Object* or *Special Fields* (e.g., Print Date, Page N of M, etc.).
3. Save your report.

**9. Deployment / Distribution**

Once the report is created, you can:

* Run it directly in Crystal Reports.
* Export it to PDF, Excel, etc.
* If you have a Crystal Reports runtime viewer or a .NET application that loads .rpt files, you can use it to distribute the report.

**Key Notes**

* In SSRS, you wrote T-SQL with IF @OrdStat = 'INVOICED' … ELSE …; in Crystal, the typical approach is either to handle that logic in the WHERE clause as shown or to put it into a stored procedure.
* Make sure your parameter data types in Crystal match the database field types you compare against (DateTime vs. Date, String vs. Number).
* If you run into issues with the ISNULL(ar\_out\_header.transaction\_date, '2001-01-01') part, confirm that your date field can safely compare to '2001-01-01' as a default. Or you might do something like ISNULL(ar\_out\_header.transaction\_date, '1900-01-01')—the essential idea is that you’re matching the “ELSE” path from SSRS.

**Summary**

1. **Define the same parameters** in Crystal as you had in SSRS.
2. **Write a single “Command”** that references those parameters in the SQL’s WHERE clause to replicate the T-SQL IF/ELSE logic.
3. **Preview your results** in Crystal, place fields on the report, and add any grouping/summary fields you want.
4. **Test, format, and save**.

That should get you from the SSRS logic to a functioning Crystal Report. If you ever need more complex logic that is hard to replicate with a single query, put it in a stored procedure and just call that from Crystal—using the same parameters.

Hopefully, this step-by-step explanation helps you feel confident building your first Crystal Report that mimics your existing SSRS report!