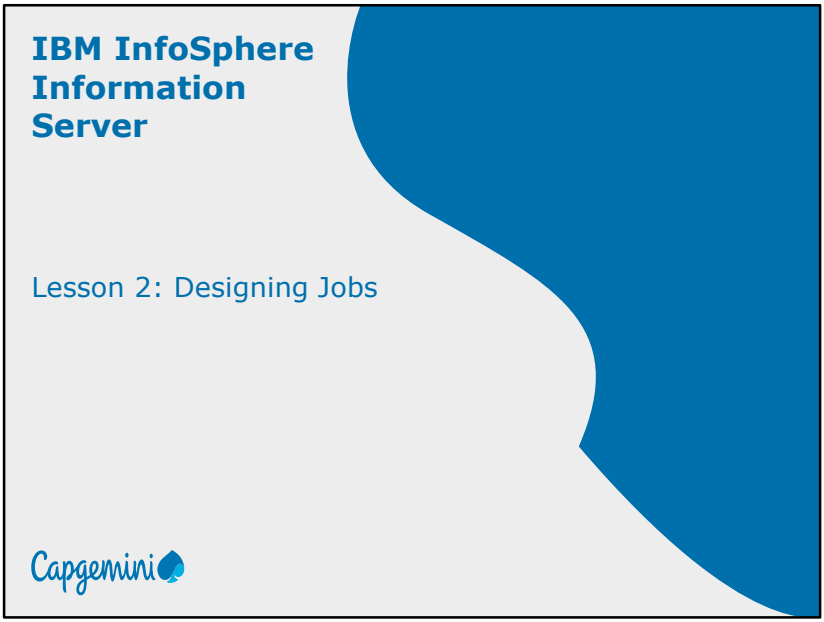

IBM InfoSphere
Information
Server
Lesson 2: Designing Jobs
Capgemini

Lesson Objectives

- On completion of this lesson, you will be able to:
- Understand DataStage designer
- Log onto DataStage
- Describe the DataStage workflow
- Create a Parameter Set
- Design a simple Parallel job in Designer
- Define a job parameter
- Use the Row Generator, Peek, and Annotation stages in a job
- Compile your job

IBM InfoSphere Information Server

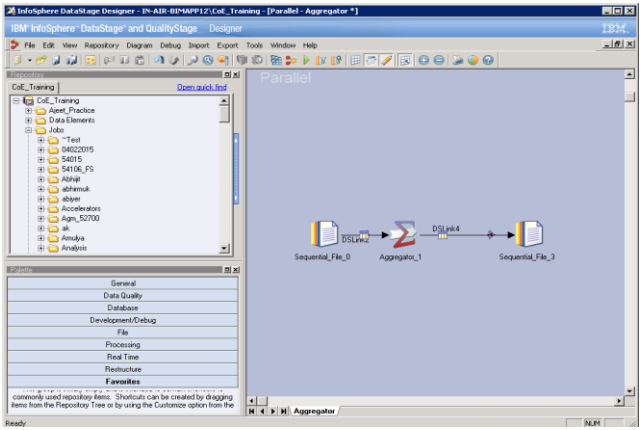
Lesson 2.1 : Introduction to Designer

Introduction to Designer and Job

- The DataStage Designer is the primary interface to the metadata repository and provides a graphical user interface that enables you to view, edit, and assemble DataStage objects from the repository needed to create an ETL job.
- An ETL job should include source and target stages. Additionally, your job can include transformation stages for data filtering, data validation, data aggregation, data calculations, data splitting for multiple outputs, and usage of user-defined variables or parameters. These stages allow the job design to be more flexible and reusable.

DataStage Designer enables you to:
Create, edit, and view objects in the repository.
Create, edit, and view data elements, table definitions, transforms, and routines.
Import and export DataStage components, such as projects, jobs, and job components.
Analyze the use of particular items in a project.
Edit and view user-defined object properties.
Create jobs, job sequences, containers, and job templates.
Create and use parameters within jobs.
Save, compile, and run jobs.

DataStage Designer Window

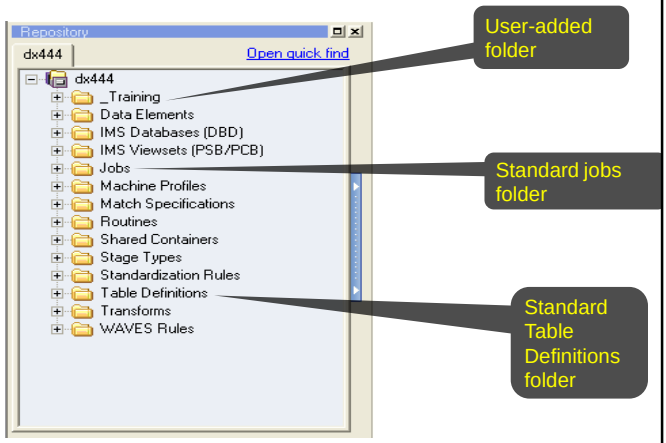


The DataStage Designer window, which is the graphical user interface used to view, configure, and assemble DataStage objects, contains the following components:

Repository Window: Displays project objects organized into categories. By default, the Repository window is located in the upper left corner of the Designer window. The project tree displays in this pane and contains the repository objects belonging to a project.

Tool Palette: Contains objects that you add to your job design, such as stage types, file types, database types, and processor objects. You can drag these objects from the Palette into the Diagram window. By default, this window is displayed in the lower left corner, of the Designer window. This window appears to be empty until you open or create a job.

DataStage Project Repository



Repository Window: Displays project objects organized into categories. By default, the Repository window is located in the upper left corner of the Designer window. The project tree displays in this pane and contains the repository objects belonging to a project.

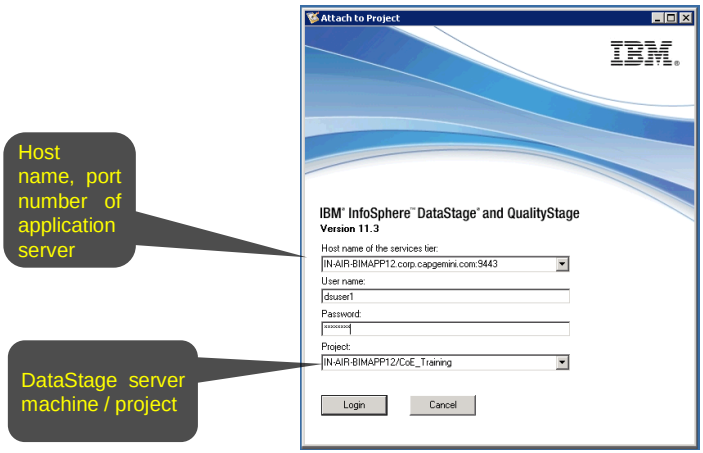
IBM InfoSphere Information Server

Lesson 2.2 : Development Workflow and logging into Designer

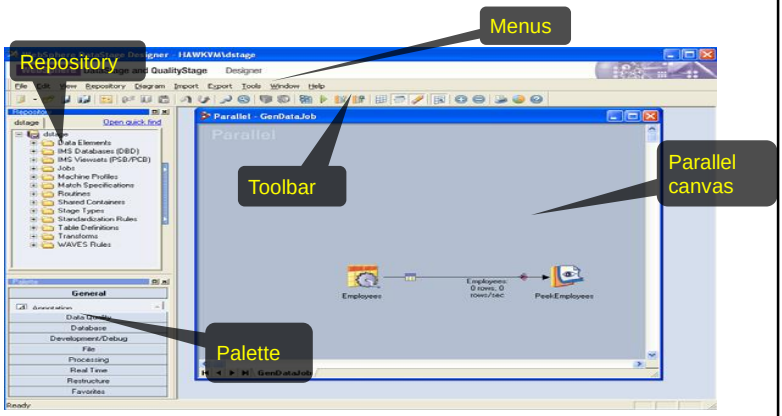
Development Workflow

- Define your project's properties: Administrator
- Open (attach to) your project
- Import metadata that defines the format of data stores your jobs will read from or write to
- Design the job: Designer
  - Define data extractions (reads)
  - Define data flows
  - Define data integration
  - Define data transformations
  - Define data constraints
  - Define data loads (writes)
  - Define data aggregations
- Compile and debug the job: Designer
- Run and monitor the job: Director as well as Designer.

Logging onto DataStage Designer



Designer Work Area



The appearance of the designer work space is configurable; the graphic shown here is only one example of how you might arrange components.

In the right center is the **Designer canvas**, where you create stages and links.

On the top left is the **Repository** window.

Items in the Repository, such as jobs and table definitions can be dragged to the canvas area.

Click **View>Repository** to display the **Repository** window

On the bottom left is the Tools Palette, which contains stages you can add to the canvas.

Click **View>Palette** to display the **Palette** window.

IBM InfoSphere Information Server

Lesson 2.3 : Import metadata into the Repository

Importing Table Definitions

- Table Definitions describe the format and columns of files and tables
- You can import Table Definitions for:
  - Sequential files
  - Relational tables
  - COBOL files
  - Many other things
- Table Definitions can be loaded into job stages

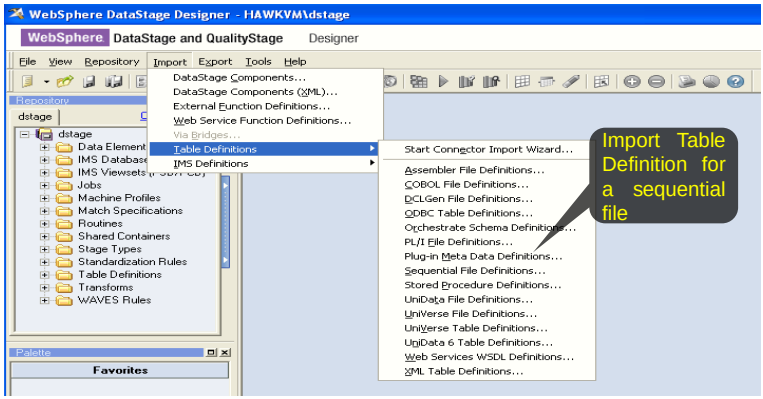
Importing Table Definitions

- Table Definitions define the formats of a variety of data files and tables. These definitions can then be used and reused in your jobs to specify the formats of data stores.
  - For example, you can import the format and column definitions of the **Customers.txt** file.
- You can then load this into the sequential source stage of a job that extracts data from the Customers.txt file. You can load this same metadata into other stages that access data with the same format. In this sense the metadata is *reusable*. It can be used with any file or data store with the same format.
- If the column definitions are similar to what you need you can modify the definitions and save the Table Definition under a new name.
- You can import and define several different kinds of Table Definitions including: Sequential files and ODBC data sources.

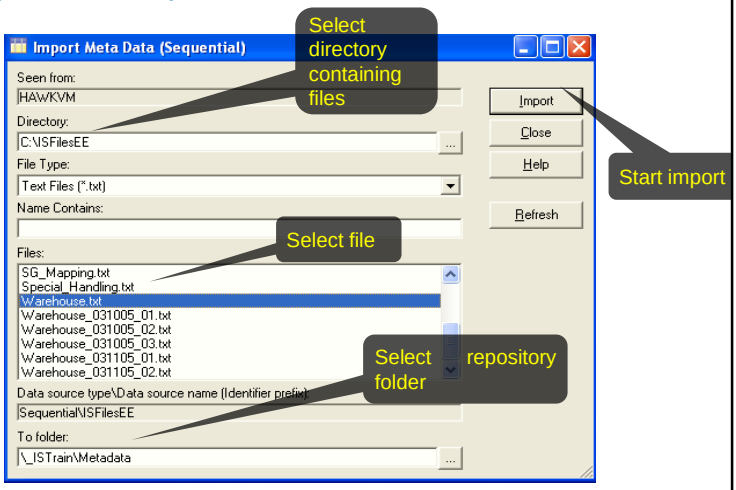
Sequential File Import Procedure

- To start the import, click Import>Table Definitions>Sequential File Definitions.
  - The **Import Meta Data (Sequential)** window is displayed.
- Select directory containing sequential file and then the file
  - The **Files** box is then populated with the files you can import. Select the file to import.
- Select or specify a folder to import into.
- Select a repository folder to store the Table Definition in
- Examined format and column definitions and edit as necessary.

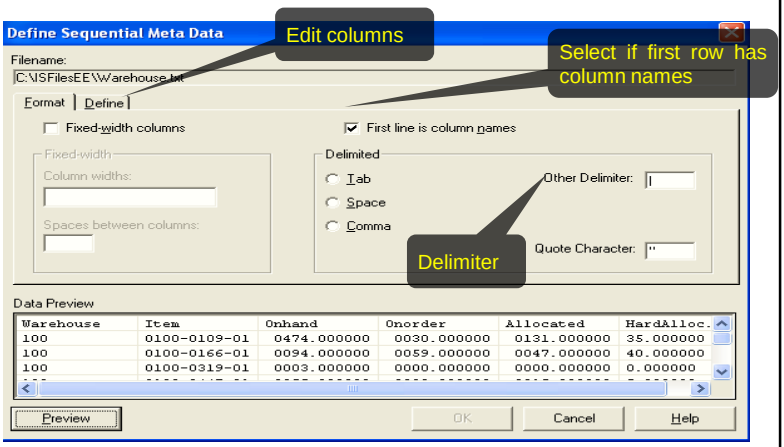
Importing Sequential Metadata



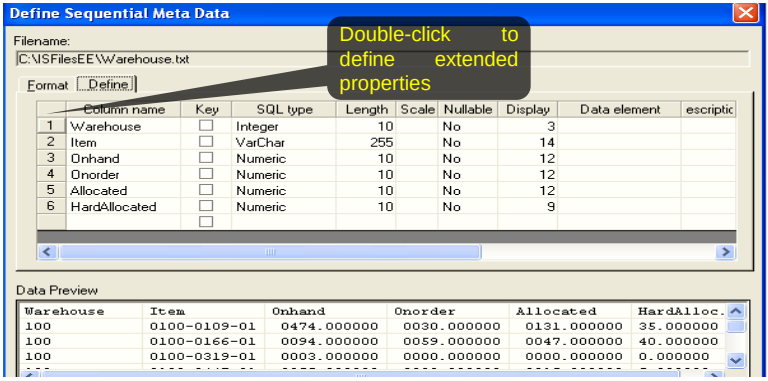
Sequential Import Window



Specify Format

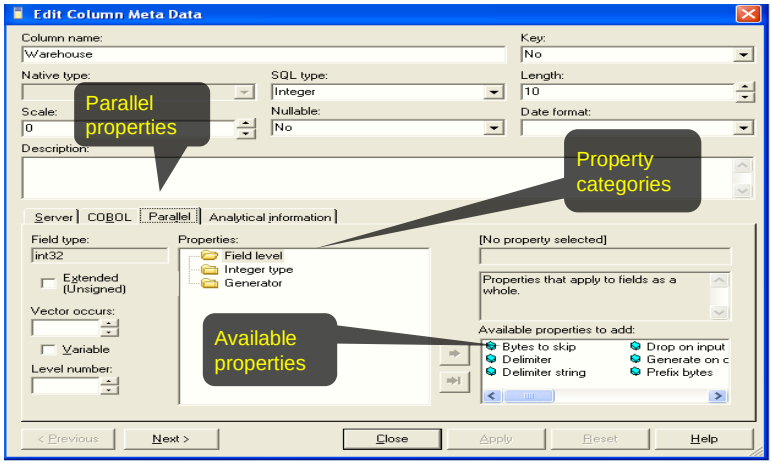


Edit Column Names and Types



You can also add "Extended Properties". Double-click on the number to the left of the column name to open up a window in which you specify these extended properties. Extended properties are discussed later in this course.

Extended Properties window



Capgemini Internal

Table Definition General Tab

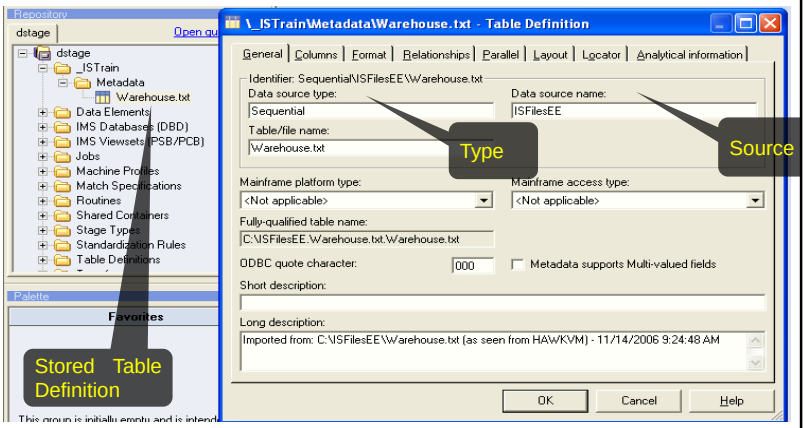


Table Definition General Tab
In the Repository window, select the folder that contains the Table Definition.
Double-click the Table Definition to open the **Table Definition** window.
Click the **Columns** tab to view and modify any column definitions.
Select the **Format** tab to edit the file format specification.
Select the Parallel tab to specify parallel format properties.

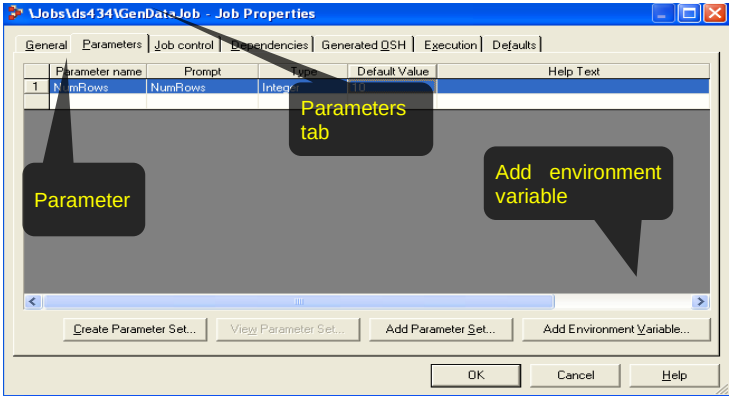
IBM InfoSphere Information Server

Lesson 2.4 : Job Parameter and parameter Sets

Job Parameter

- Defined in Job Properties window
- Makes the job more flexible
- Parameters can be:
  - Used in directory and file names
  - Used to specify property values
  - Used in constraints and derivations
- Parameter values are determined at run time
- When used for directory and files names and property values, they are surrounded with pound signs (#)
  - –E.g., #NumRows#
- Job parameters can reference DataStage environment variables
  - –Prefaced by $, e.g., $APT_CONFIG_FILE

Defining a Job Parameter

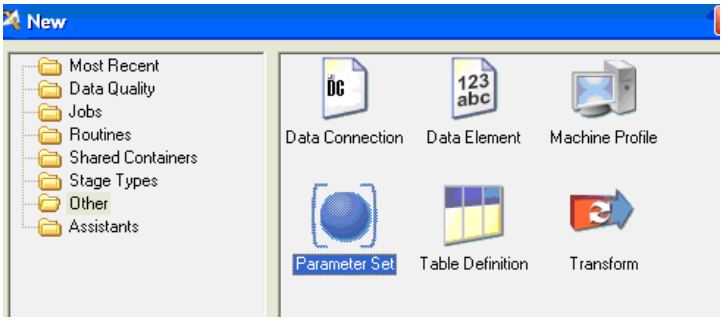


- Click the Job Properties icon to open this window.

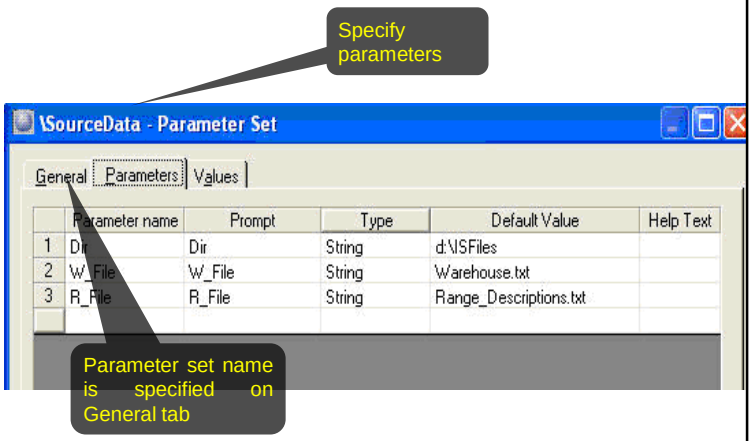
Parameter Sets

- Store a collection of parameters in a named object
- One or more values files can be named and specified
  - A values file stores values for specified parameters
  - Values are picked up at runtime
- Parameter Sets can be added to the job parameters specified on the Parameters tab in the job properties

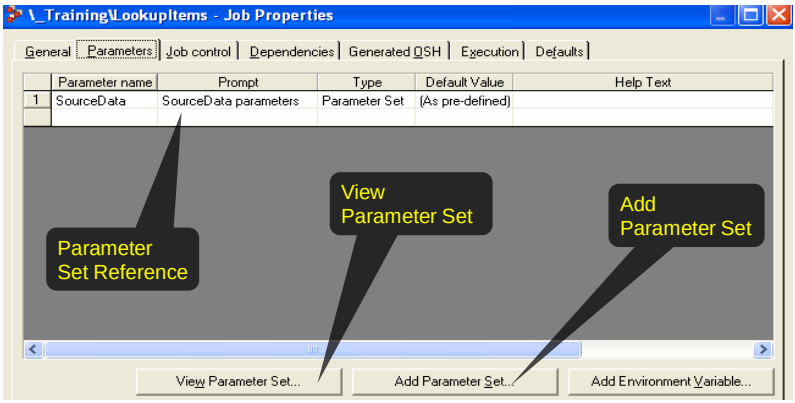
Creating a New Parameter Set



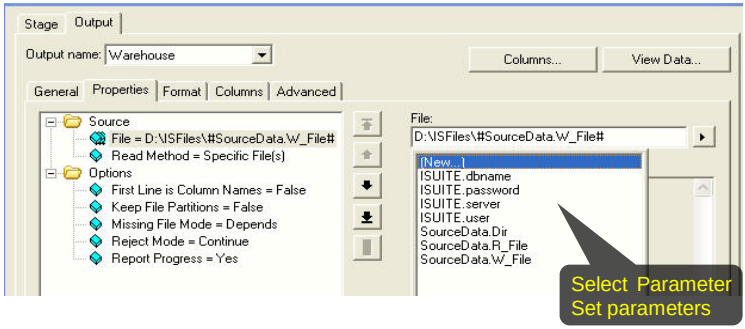
Parameters Tab



Adding a Parameter Set to Job Properties



Using Parameter Set Parameters



Notice that Parameter Set parameters are qualified by the name of the Parameter Set.

IBM InfoSphere Information Server

Lesson 2.5 : Creating Parallel Jobs

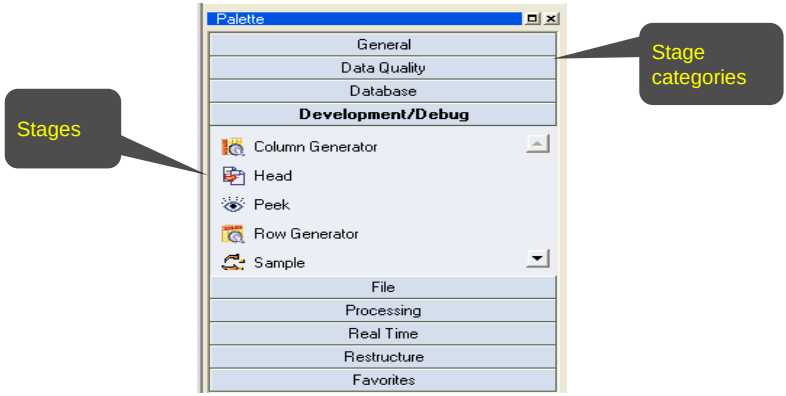
What Is a Parallel Job?

- Executable DataStage program
- Created in DataStage Designer
  - Can use components from Repository
- Built using a graphical user interface
- Compiles into Orchestrate script language (OSH) and object code (from generated C++)

Job

- A *job* is an executable DataStage program.
- In DataStage, you can design and run jobs that perform many useful data integration tasks, including data extraction, data conversion, data aggregation, data loading, etc.
- DataStage jobs are:
  - Designed and built in Designer.
  - Scheduled, invoked, and monitored in Director.
  - Executed under the control of DataStage.

Tools Palette



- The tool palette contains icons that represent the components you can add to your job design

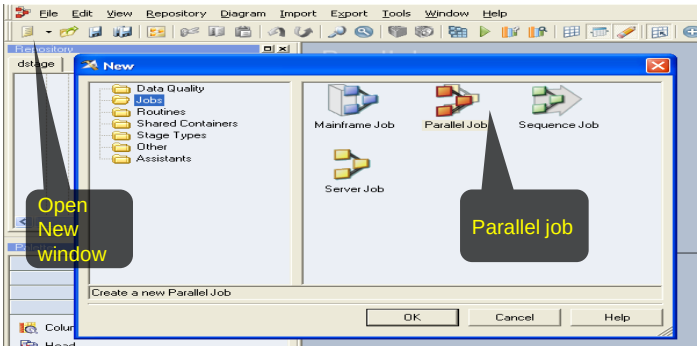
Adding Stages and Links

- Drag stages from the Tools Palette to the diagram
  - Can also be dragged from Stage Type branch to the diagram
- Draw links from source to target stage
  - Right mouse over source stage
  - Release mouse button over target stage

Job Creation Example Sequence

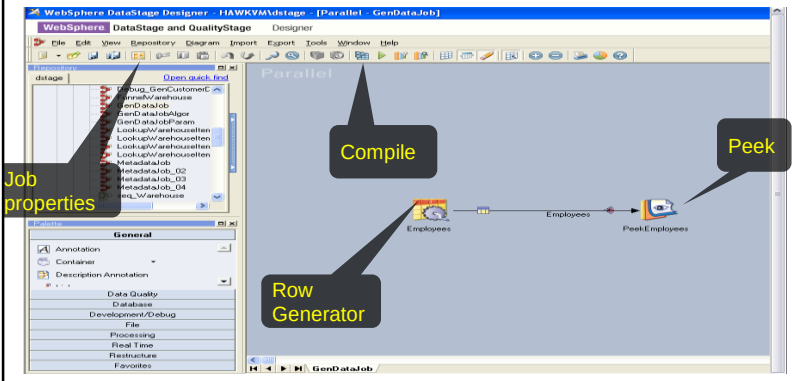
- Brief walkthrough of procedure
- Assumes Table Definition of source already exists in the repository

Create New Parallel Job



- Click the New button in the toolbar to open the New window. Click on the Parallel Job icon to create a new parallel job (the focus of this course).

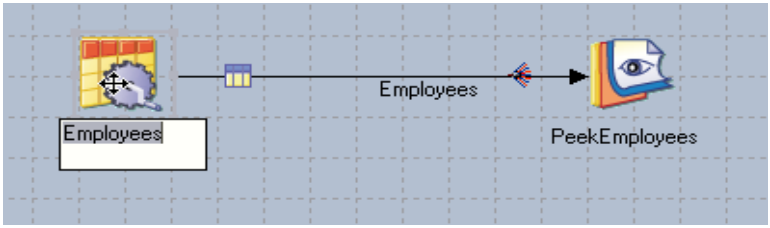
Drag Stages and Links From Palette



The tools palette may be shown as a floating dock or placed along a border. Alternatively, it may be hidden and the developer may choose to pull needed stages from the repository onto the design work area.

Renaming Links and Stages

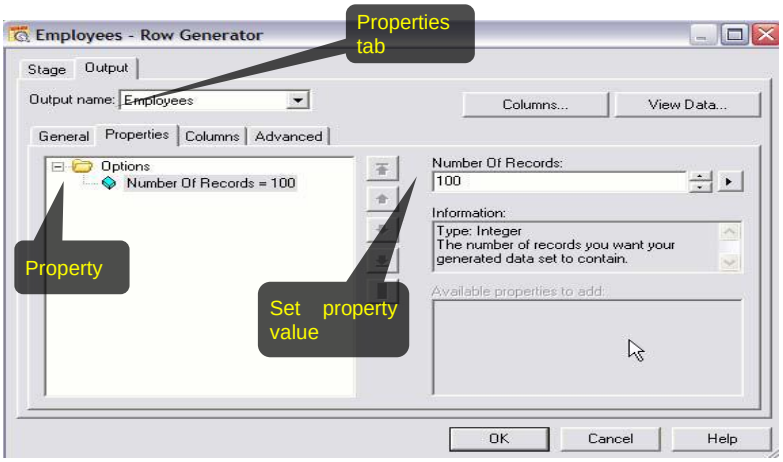
- Click on a stage or link to rename it
- Meaningful names have many benefits
  - Documentation
  - Clarity
  - Fewer development errors



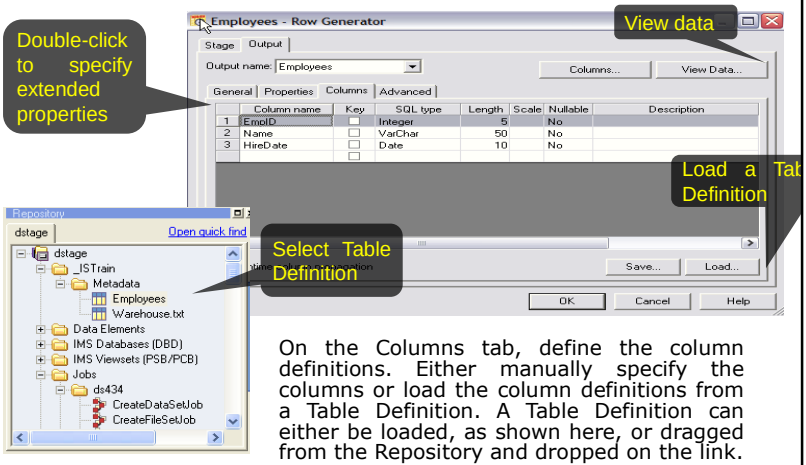
Row Generator Stage

- Produces mock data for specified columns
- No inputs link; single output link
- On Properties tab, specify number of rows
- On Columns tab, load or specify column definitions
  - Open Extended Properties window to specify the values to be generated for that column
  - A number of algorithms for generating values are available depending on the data type
- Algorithms for Integer type
  - Random
  - Cycle: Initial value, increment
- Algorithms for string type: Cycle , alphabet
- Algorithms for date type: Random, cycle

Inside the Row Generator Stage

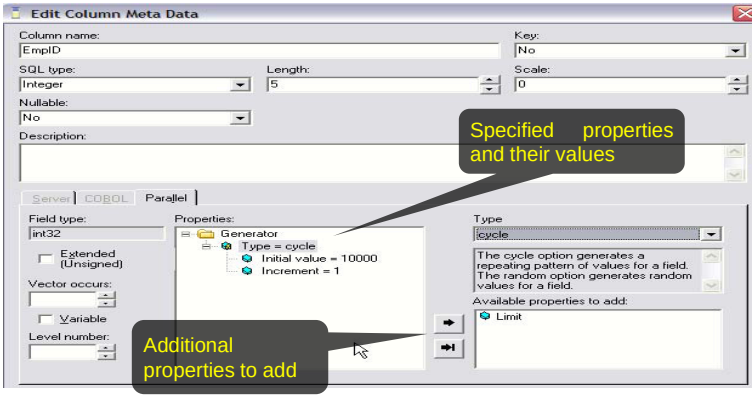


Columns Tab



On the Columns tab, define the column definitions. Either manually specify the columns or load the column definitions from a Table Definition. A Table Definition can either be loaded, as shown here, or dragged from the Repository and dropped on the link.

Extended Properties

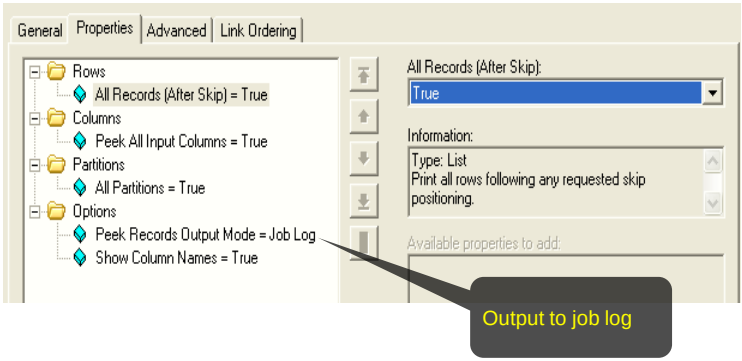


Double-click on the column number to define the extended properties for the column

Peek Stage

- Displays field values
  - Displayed in job log or sent to a file
  - Skip records option
  - Can control number of records to be displayed
  - Shows data in each partition, labeled 0, 1, 2, …
- Useful stub stage for iterative job development
  - Develop job to a stopping point and check the data
- The peek stage will display column values in a job's output messages log.

Peek Stage Properties



You can also output from the Peek stage to a file.

Peek Stage

- Displays field values
  - Displayed in job log or sent to a file
  - Skip records option
  - Can control number of records to be displayed
  - Shows data in each partition, labeled 0, 1, 2, …
- Useful stub stage for iterative job development
  - Develop job to a stopping point and check the data
- The peek stage will display column values in a job's output messages log.

IBM InfoSphere Information Server

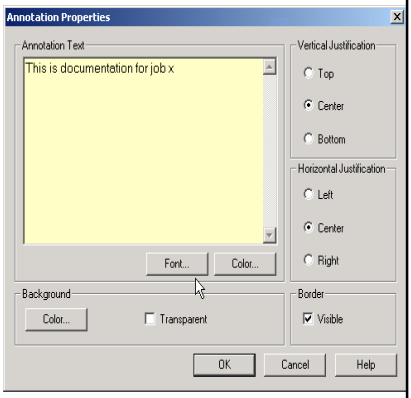
Lesson 2.6 : Adding Job Documentation

Adding Job Documentation

- Job Properties
  - Short and long descriptions
- Annotation stage
  - Added from the Tools Palette
  - Displays formatted text descriptions on diagram
  - This documentation is displayed in Manager and Director in addition to the job diagram.

Annotation Stage Properties

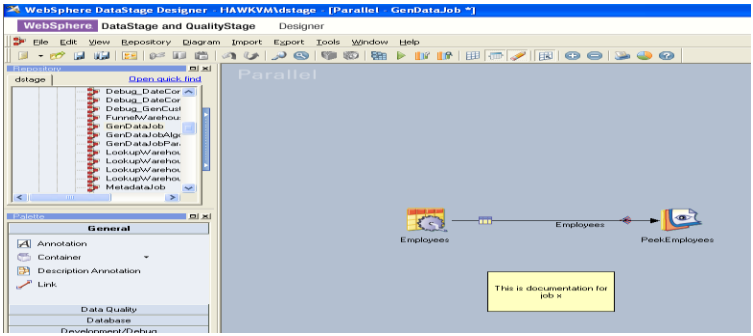
- You can type in whatever you want; the default text comes from the short description of the jobs properties you entered, if any.
- Add one or more Annotation stages to the canvas to document your job.
- An Annotation stage works like a text box with various formatting options.
- You can optionally show or hide the Annotation stages by pressing a button on the toolbar.
- There are two Annotation stages. The Description Annotation stage correlates its text with the Descriptions specified as part of the job properties.



IBM InfoSphere Information Server

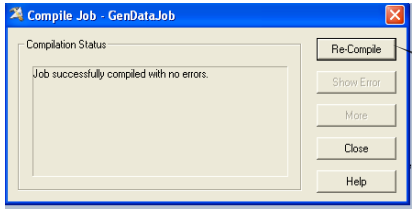
Lesson 2.7 : Compiling a Job

Compiling a Job



- Before you can run your job, you must compile it.
- To compile it, click File>Compile or click the Compile button on the toolbar.
- The Compile Job window displays the status of the compile.
- A compile will generate OSH.

Errors or Successful Message



- If an error occurs:
- Click Show Error to identify the stage where the error occurred.
  - This will highlight the stage in error.
- Click More to retrieve more information about the error. This can be lengthy for parallel jobs.
- Many errors also show up on the diagram if "Show Stage Validation Errors" is turned on.

Unit summary

- Having completed this unit, you should be able to:
- Design a simple Parallel job in Designer
- Define a job parameter
- Use the Row Generator, Peek, and Annotation stages in a job
- Compile your job
- Run your job in Director
- View the job log

Q&A

1. True or False ? Can you import Table Definitions for files with fixed-length record formats?
2. Which stage can be used to display output data in the job log?
3. Which stage is used for documenting your job on the job canvas?

Q&A

1. Yes. Record lengths are determined by the lengths of the individual columns.
2. Peek stage
3. Annotation stage