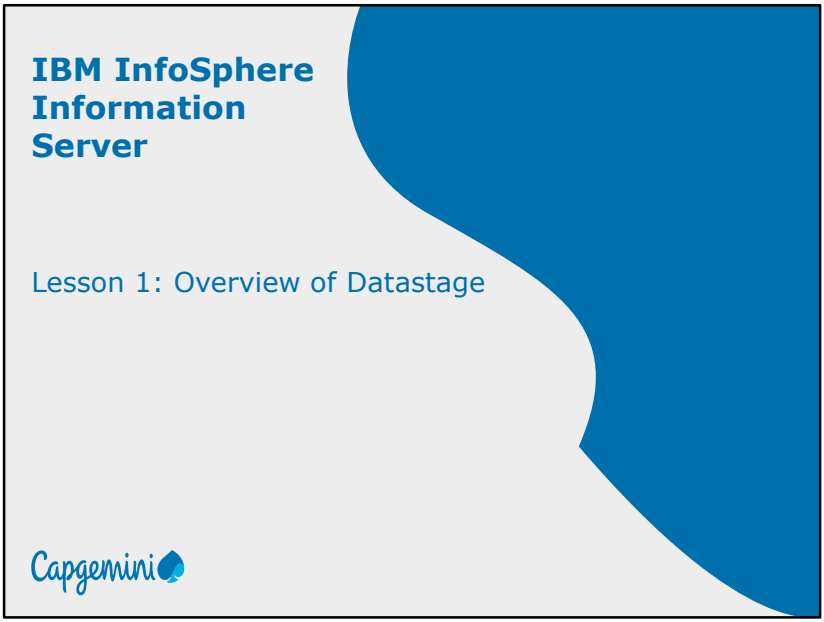

IBM InfoSphere
Information
Server
Lesson 1: Overview of Datastage
Capgemini

## Lesson Objectives

- On completion of this lesson, you will be able to:
- Understand DataStage History
- Overview on DataStage Architecture
- Type of Datastage products and jobs
- Setting Up Your DataStage Environment

## Introduction to IBM Infosphere and Datastage

- It is an ETL tool and part of the IBM Information Platforms Solutions suite and IBM InfoSphere.
- It uses a graphical notation to construct data integration solutions and is available in various versions such as the Server Edition, the Enterprise Edition, and the MVS Edition.
- InfoSphere DataStage is a powerful data integration tool.
- It uses a client/server design where jobs are created and administered via a Windows client against central repository on a server.
- The IBM InfoSphere DataStage is capable of integrating data on demand across multiple and high volumes of data sources and target applications using a high performance parallel framework.
- InfoSphere DataStage also facilitates extended metadata management and enterprise connectivity.

## Purpose of using datastage

- Design jobs for Extraction, Transformation, and Loading (ETL)
- Ideal tool for data integration projects – such as, data warehouses, data marts, and system migrations
- Import, export, create, and manage metadata for use within jobs
- Schedule, run, and monitor jobs, all within DataStage
- Administer your DataStage development and execution environments
- Create batch controlling(sequencer) jobs

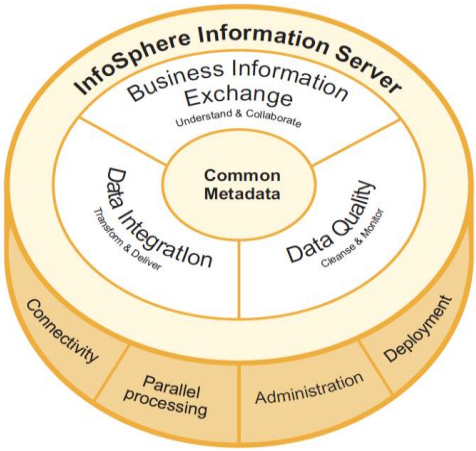
## History of Datastage

- Data stage and its version ownership moved from one to another during period.
- Vmark:1997(Lee Cheffler):Data Integrator
- Informix-Ascential :2000 :Datastag Server Jobs
- Ascential-Orchestrate :2002:
- In 2005, IBM acquired Ascential Software and moved the products into the WebSphere Information Integration suite.

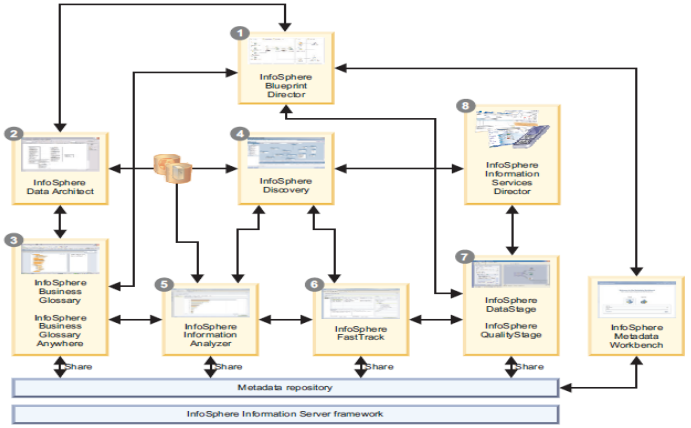
## History of Datastage

- DataStage Standard Edition was the original DataStage product and is also known as DataStage Server Edition.
- DataStage Enterprise Edition was originally called Orchestrate, then renamed to Parallel Extender when purchased by Ascential.
- DataStage TX was originally known as Mercator and renamed when purchased by Ascential.
- DataStage SOA was originally known as the Real Time Integration pack.

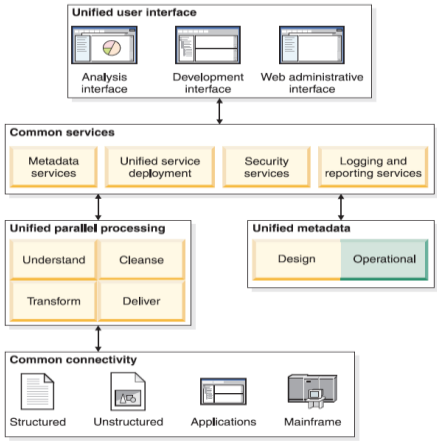
## Information Server Product Components



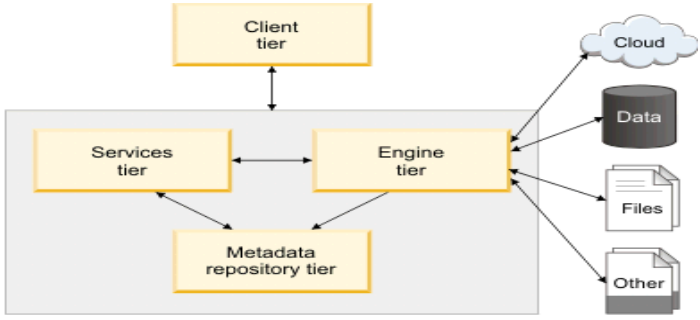
## Information Infosphere Family



## Datastage Architecture



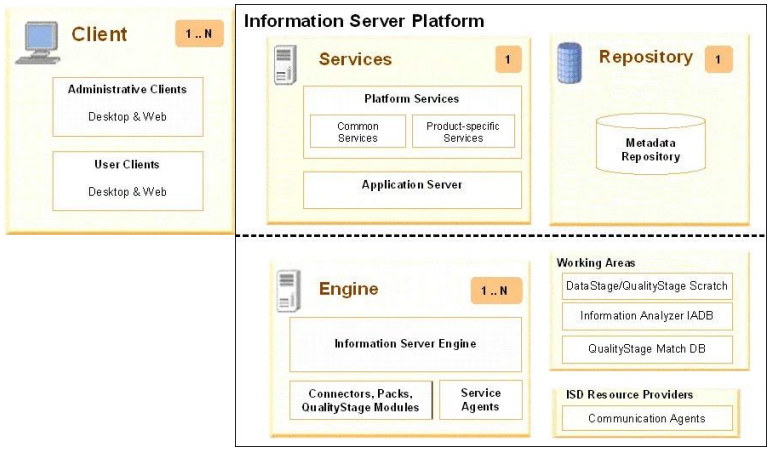
## Datastage Architecture Simplified



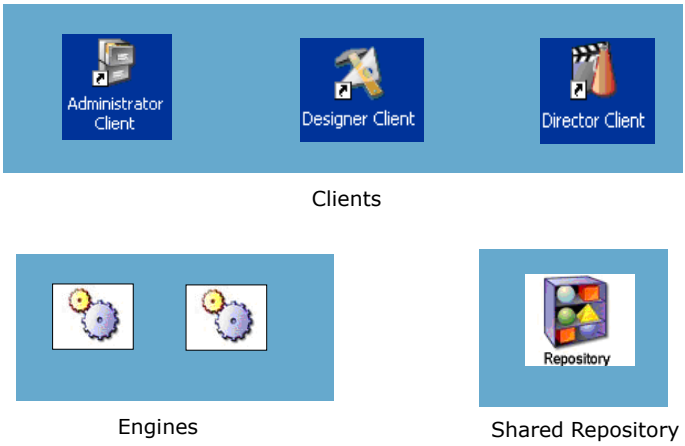
## Datastage Architecture – Tiers Explained

| Tier | Description |
|---|---|
| Client tier | Includes the client programs and consoles that are used for development and administration, and the computers where they are installed. you |
| Engine tier | The engine tier includes the logical group of components (the InfoSphere Information Server engine components, service agents, and so on) and the computer where those components are installed. The engine runs jobs and other tasks for product modules. |
| Services tier | The services tier includes the application server, common services, and product services for the suite and product modules, and the computer where those components are installed. The services tier provides common services (such as metadata and logging) and services that are specific to certain product modules. On the services tier, WebSphere® Application Server hosts the services. The services tier also hosts InfoSphere Information Server applications that are web-based. |
| Metadata repository tier | The metadata repository tier includes the metadata repository, the InfoSphere Information Analyzer analysis database (if installed), and the computer where these components are installed. The metadata repository contains the shared metadata, data, and configuration information for InfoSphere Information Server product modules. The analysis database stores extended analysis data for InfoSphere Information Analyzer. |

## Datastage Architecture in Detail



## DataStage Components



## DataStage Components

- Core client applications of DataStage are common in case of all these versions.
- These components are, Administrator, Designer, Director and Manager.
- Administrator: Administrator is the user interface of DataStage which, as the name suggests, is responsible for administering DataStage projects. It does so by managing the global settings and interacting with the system. This components performs the administration task of setting up users and project properties, and adding, moving and deleting projects. It sets up the purging criteria. It specifies general server defaults. The DataStage repository is provided with a command interface by this component. It also sets up parallel jobs default, job scheduling options, user privileges, job monitoring limits etc.

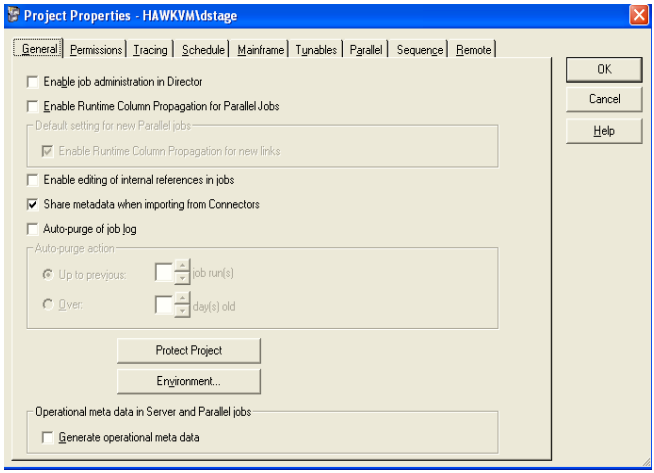
## DataStage Components

- Manager: The DataStage Manager is considered the main interface the Repository of DataStage as it is used for the purpose of viewing and editing the contents of the DataStage Repository. It allows the browsing of the Repository as well. Its main use is the storage and management of reusable meta data. Tables and files layouts, jobs, transforms, routines etc. that are defined in the project are also displayed by it.

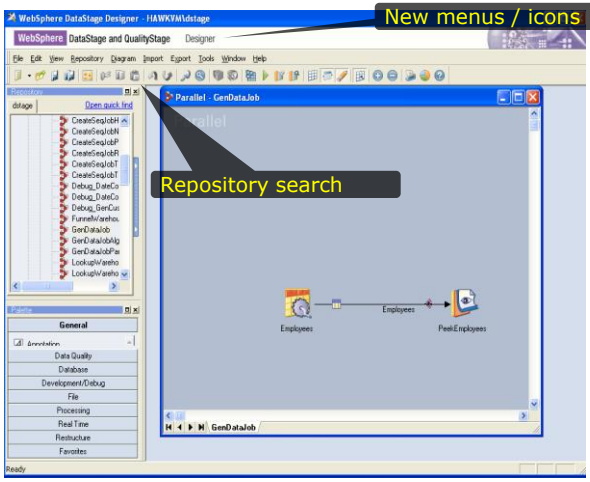
## DataStage Components

- Designer: The DataStage jobs or applications are created using the design interface. These jobs are compiled to form executable programs. The jobs individually specify the sources of data, required transforms, as well as the destination of data. The executables which are created from compiling these jobs are scheduled by the DatStage Director. The Server runs these executables. Designer is a graphical application which is user-friendly. Visual data flow method is applied by this user interface for the extraction, cleansing, transformation, integration and loading of data. DataStage developers prefer to use this module.
- Director: The main task of this DataStage interface is scheduling the executable programs formed by the compilation of jobs. It runs, validates, schedules and monitors the server jobs and parallel jobs. The main users of this component are testers and operators.

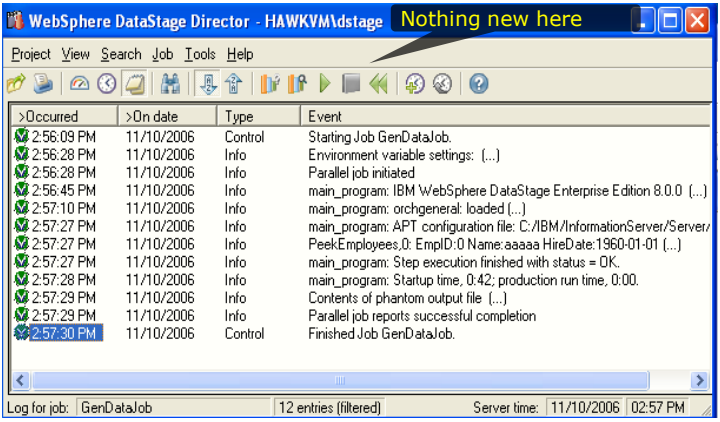
## DataStage Administrator



## DataStage Designer



## DataStage Director



## Developing in DataStage

- Define global and project properties in Administrator
- Import metadata into the Repository
- Build job in Designer
- Compile job in Designer
- Run and monitor job log messages in Director as well as Designer.

## Types of DataStage Jobs

- Parallel jobs
  - Executed by the DataStage parallel engine
  - Built-in functionality for pipeline and partition parallelism
  - Compiled into OSH (Orchestrate Scripting Language)
  - OSH executes Operators
  - Executable C++ class instances
  - Runtime monitoring in DataStage Director and Designer
- Server jobs
  - Executed by the DataStage server engine
  - Compiled into Basic
  - Runtime monitoring in DataStage Director and Designer

## Types of DataStage Jobs

- Job sequences (batch jobs, controlling jobs)
  - Master Server jobs that kick-off jobs and other activities
  - Can kick-off Server or Parallel jobs
  - Runtime monitoring in DataStage Director and Designer
  - Executed by the Server engine

## Design Elements of Parallel Jobs

- Stages
  - Implemented as OSH operators (pre-built components)
  - Passive stages (E and L of ETL)
    - Read data
    - Write data
    - •E.g., Sequential File, DB2, Oracle, Peek stages
  - Processor (active) stages (T of ETL)
    - Transform data
    - Filter data
    - Aggregate data
    - Generate data
    - Split / Merge data
    - E.g., Transformer, Aggregator, Join, Sort stages
- Links
  - "Pipes" through which the data moves from stage to stage

## Unit summary

- Having completed this unit, you should be able to:
- List and describe the uses of DataStage
- Describe Information Server
- List and describe the DataStage clients
- Describe the DataStage workflow
- List and compare the different types of DataStage jobs

## Summary

- In this module, you learned about the following:
  - Datawarehousing strategies
  - Datawarehousing architecture
  - Need for ETL
  - Meaning of ETL


Add the notes here.

# IBM WebSphere DataStage

Lesson 1.1: Setting up Your DataStage Environment

## Module Objectives

- Setting project properties in Administrator
- Defining Environment Variables
- Importing / Exporting DataStage objects in Manager
- Importing Table Definitions defining sources and targets in Manager

## Setting Project Properties

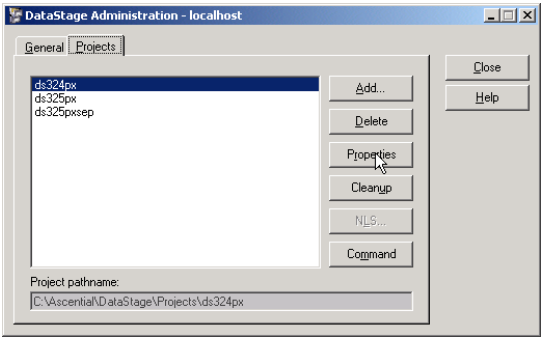
## Project Properties

- Projects can be created and deleted in Administrator
  - Each project is associated with a directory on the DataStage Server
- Project properties, defaults, and environmental variables are specified in Administrator
  - Can be overridden at the job level

In DataStage all development work is done within a project. Projects are created during installation and after installation using Administrator. Each project is associated with a directory. The directory stores the objects (jobs, metadata, custom routines, etc.) created in the project. Before you can work in a project you must attach to it (open it).
You can set the default properties of a project using DataStage Administrator. Many of these properties can be overridden at the job level.

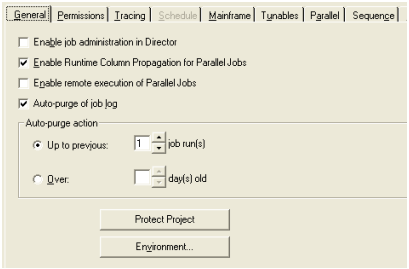
## Setting Project Properties

- To set project properties, log onto Administrator, select your project, and then click "Properties"



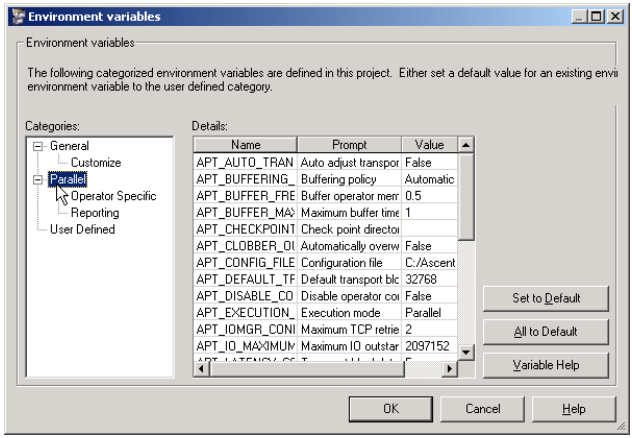

The logon screen for Administrator does not provide the option to select a specific project (unlike the other DataStage clients).

## Project Properties General Tab




The General tab is used to specify Runtime Column Propogation as a default, job log purging requirements, and Environment variables.
Check Enable job administration in Director to enable some administrative functions to be available in Director.
Use the Permissions tab to specify who has permission to develop and run DataStage jobs.
Use the Tracing tab to turn on DataStage Client/Server tracing information.
Use the Schedule tab to specify the user ID under which to run scheduled jobs.
The Mainframe and Tunables tabs don't apply to Parallel jobs and are not discussed in this course.
Use the Parallel tab to specify defaults for Parallel jobs.
The "Enable remote execution of Parallel Jobs" option is available only for USS development.

## Environment Variables

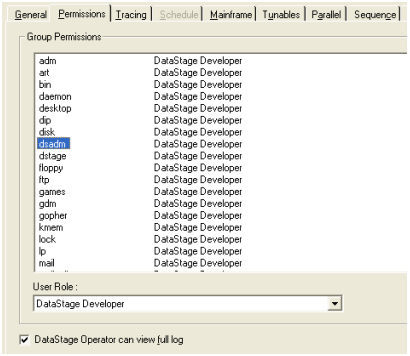



Click the Environment button on the General tab to specify environment variables. The variables listed under the Parallel branch apply to Parallel jobs.

You can also specify your own environment variables under the User Defined branch. These variables can be passed to jobs through their job parameters to provide project level job defaults.

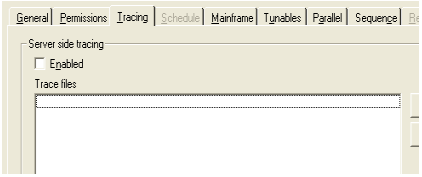
There are also other environment variables that are hidden from the GUI. See the Parallel Job Advanced Developers Guide documentation for a list of the environment variables.

## Permissions Tab



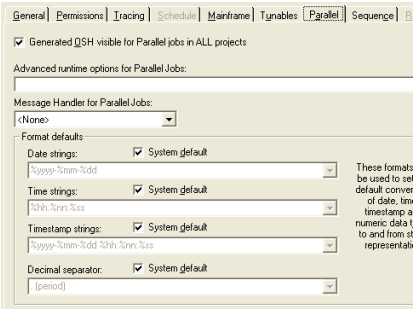

Use this page to set user group permissions for accessing and using DataStage. All DataStage users must belong to a recognized user role before they can log on to DataStage. This helps to prevent unauthorized access to DataStage projects.
There are three roles of DataStage user:
DataStage Developer, who has full access to all areas of a DataStage project.
DataStage Operator, who can run and manage released DataStage jobs.
DataStage Manager, who has full access to project, plus ability to create and manipulate protected projects.
<None>, who does not have permission to log on to DataStage.
UNIX note: In UNIX, the groups displayed are defined in /etc/group.

## Tracing Tab




This tab is used to enable and disable server-side tracing.
The default is for server-side tracing to be disabled. When you enable it, information about server activity is recorded for any clients that subsequently attach to the project. This information is written to trace files. Users with in-depth knowledge of the system software can use it to help identify the cause of a client problem. If tracing is enabled, users receive a warning message whenever they invoke a DataStage client.
Warning: Tracing causes a lot of server system overhead. This should only be used to diagnose serious problems.

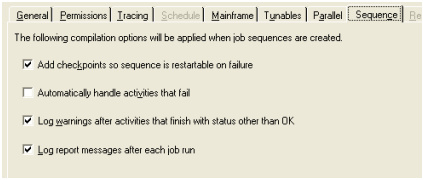
## Parallel Tab




Use this tab to specify Parallel job defaults. In addition to displaying the OSH generated by DataStage from Parallel jobs, you specify default formats for dates and times.

In general, you should choose to display the OSH. This provides useful information about how your job works.

## Sequence Tab



## Q&A

- Overall Components in datastage ?

## Q&A

- Server, Client and Shared repository

True or False? The directory you export to is on the DataStage client machine, not on the DataStage server machine.
True: Correct! The directory you select for export must be addressable by your client machine.
False: Incorrect. The directory you select for export must be addressable by your client machine.