

Batch Pilot Glossary

This *Glossary* contains brief definitions of terms associated with *Batch Pilot*.

Action

A concise procedure that is part of a rule. Together, a rule's actions determine how a particular *rules-based* task is to deal with a specific object of the Document Hierarchy...a particular page or field, for example.

An action belongs to a RuleSet Type such as Locate, Recognition, Validate, or Export. (These are just a few of the RuleSet Types.)

The setup phase of the Task Definition process uses *Batch Pilot's Setup* dialog to assign RuleSet Types to a task. *See also: Action, Rules, RuleSet Type, Task Definitions.*

Actions Library

The combination of action categories that is available to rules of a specific RuleSet Type. Actions that belong to a particular category – Locate actions, for example – are part of a Rule Runner Actions file (.rra). A RuleSet Actions Library contains one or more files. *See also: Action, Rules, RuleSet Type,*

Admin Database

The *Taskmaster* database that contains definitions of workflows, jobs, tasks, users and workstations. *See also: Engine Database, LookUp Database, Rules Database, Taskmaster.*

Administrative Privileges

Attributes of a User Definition or User Group Definition that designate the ability of an individual or members of a User Group to access an application's administrative dialogs and data, and carry out administrative functions. *See also: User Definition, Job/Task Permissions.*

Application

The principal organizing entity of a *Taskmaster* configuration. An application's Document Hierarchy identifies *structural* objects at four levels: **Batch**, **Document**, **Page** and **Field**. A Workflow Hierarchy organizes the application's *processing* entities at three levels: Workflow, Job and Task. *See also: Document Hierarchy, Workflow Hierarchy.*

The application includes four principal databases (Admin, Engine, Rules and LookUp); at least one *Data Entry* panel for Verification tasks; a *Rule Manager*; and a *Report Viewer*.

Application Configuration Interface

A *Batch Pilot* product used to configure elements which are essential to an application's *setup* or *runtime* procedures. *Rule Manager* is an example of an Application Configuration Interface. *See also: Rule Manager.*

Autoform

The *Batch Pilot* utility that uses the **Field** objects of a Document Hierarchy to automatically assemble the *runtime* form of a Verification Task Project. This form serves as the Verification task's *Data Entry* panel. *See also: Runtime Form.*

Batch

A *Taskmaster* application's processing container **and** the organizing object at the top of its Document Hierarchy. *See also: Application, Document Hierarchy.*

Batch Pilot

The environment in which *rules-based* tasks are designed, developed and run. This environment constructs a task's *setup* and *runtime* forms, binds them to a Task Project, and adds settings that link the Task Project to a Task Definition prepared by *Taskmaster*. *Batch Pilot* technology also constructs and maintains supporting tools such as *Rule Manager* and the *Enhanced Notification Utility*. *See also: Datacap Form, Runtime Form, Setup Form, Task Project, Task Settings, Task Definition, Task Project.*

Batch Pilot Window

Batch Pilot's fully-equipped window - used for form design, Task Project definition, scripting, and task and utility operations.

Binding

Batch Pilot's design mechanism that links a Task Project's *Setup* and *Runtime* forms to specific objects of an application's Document Hierarchy. *See also: Document Hierarchy, Runtime Form, Setup Form.*

Binding also refers to the *Rule Manager* procedure that assigns rules of a specific RuleSet Type to an object of the application's Document Hierarchy. *See also: Document Hierarchy, Rule Manager.*

Character

A distinct, separate object of a word detected on the processing file of a *source* page. *See also: Line, Word, Processing file, Source Page.*

Confidence Level

A value between “0” and “9” stipulating a minimum degree of certainty on the part of a “recognition” task that it has accurately interpreted data in a particular field of a *source* page, or has accurately recognized the page itself. The Required Confidence Levels for a page and its fields are defined in the task’s Task Definition. “0” is the least stringent rating, while “9” means that Rule Runner will accept a field’s value only if it has interpreted the value with complete confidence. *See also: Field, RuleRunner Task, Source Page, Task Definition.*

Control

A User Interface object such as a field, button frame or image that is placed on a Datacap form – and becomes part of the form. The *Batch Pilot Toolbox* carries a complete set of controls.

Datacap Forms

See Forms.

Data Source Name (DSN)

Specifications defining the link between a specific data **source**—usually a database such as an application’s Engine, Admin, Rules or LookUp database—and an **Open Database Connectivity (ODBC)** “driver.”

Data Edit Field

An interactive field in a Verification task’s *Data Entry* panel. This field displays the *recognized* value for a **Field** object of a Document Hierarchy; the accompanying **Snippet** shows the *image* of the field and its value; and a **Label** identifies the field.

After comparing the two values, a Data Entry operator can make appropriate changes to the value in the Data Edit field.

The *Autoform* utility automatically generates a *Data Entry* panel with a field trio – Snippet, Data Edit field and Label – for each **Field** object of the Document Hierarchy. *See also: Autoform, Data Entry Panel, Label, Snippet.*

Data Entry Panel

A *runtime* form which presents the Verification task’s operator with information taken from certain fields of a *recognized* page. For most fields, the information appears in two formats: a Snippet shows the image of the source field and any value it contains; an interactive Data Edit field displays the field’s recognized value *See also: Autoform.*

Data File (.xml)

A file containing the *recognized* values of a **source** page. After a “recognition” task fully processes a **source** page, the task adds these values to a Data file it assembles just for that page. If the succeeding Verification task alters a value, the task updates the Data file – and Export tasks access the files to create an Export file or update an Export database. *See also: Source Page.*

DCEdit Control

See Data Edit Field.

DCO

An acronym for the **Datacap Object**.

Properties and methods of this object are responsible for the definition of an application’s Document Hierarchy; the application of rules to specific objects of the Document Hierarchy; and for the ways in which the rules are carried out after they have been assigned to tasks of a *Taskmaster* workflow. *See also: Objects, Document Hierarchy, Rules.*

DCImage Control

The control that displays the image of a **source** page, or of a field within the page.

Development Toolbox

A container with a set of *Batch Pilot* drag-and-drop controls you can place anywhere on a form you’re constructing. Right-clicking on a control reveals a list of formatting options and accesses the control’s **Properties** dialog. You can add and remove controls to or from the *Development Toolbox* without difficulty. *See also: Datacap Form.*

Dictionary

A property of an object of the Document Hierarchy (usually a **Field** object.) The dictionary contains a set of acceptable values for the objects, and can be defined using the *Dictionaries* dialog of the *Document Hierarchy Setup* window.

Document

An object on the second level of an application’s Document Hierarchy. A **Document** object consists of one or more **Page** objects, and is the primary organizing element of the hierarchy’s top-level **Batch** object. Page files (.xml) generated by each task identify the documents in the current batch, and the pages in each document. *See also: Batch, Page File, Document Hierarchy, Page, Page Type,*

Document Integrity

Settings of the Document Hierarchy that determine the makeup of the documents in a batch, in terms of the *number* and *type* of pages a document must contain. *See also: Document, Document Hierarchy, Page.*

Engine Database

The *Taskmaster* database containing processing results according to task, batch, document and page. *See also: Admin Database, Rules Database, LookUp Database.*

Field

An object at the lowest level of an application's Document Hierarchy...and a child of a **Page** object one level above.

Fields on a *source* page contain values that are recognized by a RuleRunner task, *see also: Document Hierarchy, Page.*

File Server

The server on which the files of a **Datacap** configuration reside. Phase 1 of the Installation process copies these files from the Installation CD-ROM to the File Server. The files include *Batch Pilot* software and pre-defined forms. *See also: Forms.*

Filter

Optional settings in the *Filter* tab of a Task Project's **Task Settings** dialog. You can use these settings to limit the scope of a task's processing activity to batches, documents and pages which meet the filter's selection criteria.

Form

The electronic interface between a user and an administrative or operating facet of a *Batch Pilot* product. A Task Project, for example, consists of a *setup* form and a *runtime* form – or a master form with *setup* and *runtime* tabs. In contrast, the **Rule Manager Window** is a special purpose *Batch Pilot* form. These forms are stored as Datacap Form files (.dcf). *See also: Task Project.*

Form Design

The development mode of the **Batch Pilot Window** that permits you to add and remove controls to a form. To access this mode, you click on the **Design** icon in the window's toolbar. *See also: Form Scripting.*

Form Scripting

The development mode of the *Batch Pilot Window* that you use to prepare and test scripts for the overall form – and for individual controls within the form. To access this mode, you click on the **Edit Script** icon in the window's toolbar. See also: *Form Design*.

Form Testing

The development mode of the *Batch Pilot Window* that tests scripts you've prepared – or the form itself. The **Run Script** icon tests scripts; the **Test Form** icon tests the current form.

High Confidence Value

A value in a field of a scanned page which a RuleRunner task interprets with a degree of certainty that equals or exceeds the application's requirements.

Hot Key

A *Batch Pilot* feature that establishes unique keyboard combinations and assigns them to a Task Project or to a form. A Hot Key at the project level is available to *any* form bound to the project. A Hot Key you assign to a form is only available to that form. The script you prepare for the **Pilot** object's **OnHotKey** event determines the response of a Hot Key. See also: *Form, Pilot Object, Task Project*.

Image File

An electronic representation a scanned page. The Image file is a digital picture of the actual page, usually stored as a .tif file.

Job

A set of interrelated *Taskmaster* tasks that operate together to accomplish a specific goal. Usually, the purpose of a job is to convert paper documents to captured data. Some special purpose jobs – a FixUp job, for example – handle problems with unsatisfactory images or document content. A job is an component of a workflow. See also: *Workflow Hierarchy, Job, Task*.

Job/Task Combination

The workflow's *fundamental processing unit*, forged by the assignment of a specific task to a specific job when workflows, jobs and tasks are defined. See also: *Workflow, Job, Task*.

Job-Task Shortcut

An icon in *Taskmaster's Operations* window that initiates one or more Job/Task combinations. A shortcut is an element of Application Security: it is available *only* to

authorized operators, Supervisors and Administrators working from a station with comparable Job-Task permissions. *See also: Job-Task Permissions.*

Job-Task Permissions

Authorizations permitting individuals to carry out certain Job/Task combinations. An application's Security Program includes these authorizations in its User, User Group, Station and Shortcut Definitions. *See also: Job/Task Combination, Administrative Privileges.*

Label Control

A control that is part of a Datacap form (.dcf).

A **Label** control is a member of the field trio on a **Data Entry** panel that includes a Snippet and a Data Edit field. *See also: Data Edit Field, Data Entry Panel, Snippet.*

Log File

A file that tracks a task's progress as it processes a batch. The *Log* tab of a Task Project's *Task Settings* dialog contains Log criteria.

LookUp Database

An application's validation database. An application's Verification procedures compare the *recognized* values for certain fields of a *source* page to standard values in tables of the LookUp database – and report any discrepancies. *See also: Admin Database, Engine Database, Rules Database.*

Main Job

A workflow's primary job, usually consisting of tasks in at least four broad categories: Scan, Recognition, Verification and Export. *See also: Task Definitions.*

Modes, Batch Pilot

Batch Pilot's four environments. *Design* constructs a form; *Coding* adds or modifies the code for a form and its controls; *Project Definition* sets up a Task Project and incorporates its forms, and tests the project; *Processing* operates the form or Task Project.

Multi-tab Form

A form with more than one tab. The settings in each tab are responsible for discrete setup or processing activities. *See also: Form.*

Objects

Elements at the four levels of an application's Document Hierarchy: **Batch**, **Workflow**, **Document**, **Page** and **Field**. *Batch Pilot's* Task Development process assigns properties, binds a Task Project's forms, and applies rules to these objects. *See also: Document Hierarchy.*

Operator

An individual authorized by an application's Administrator to run one more tasks: Scan and Verification tasks usually require direct operator participation. Application Security procedures require a unique User ID and Password for each operator, and formal permission to carry out a specific Job/Task combination. *See also: Job/Task Combinations, Job-Task Permissions.*

Page

An object on the third level of an application's Document Hierarchy. A **Page** object is an element of a **Document** object one level above, and contains **Field** objects one level below. *See also: Document Hierarchy.*

Page File (.xml)

A file generated by a task as it processes a batch. The Page file file (.xml) contains identifying codes and processing statistics about the batch, and about the documents and pages the batch contains. After a task has finished with a batch and assembled the file, the next task in the current job refers to it for processing benchmarks. *See also: Data File, Job, Task.*

Page Type

The unique identifying property of a **Page** object of the Document Hierarchy. A **Document** object must include one or more **Page** objects – each with its own Page Type value.

Panel

See Data Entry Panel.

Privileges

See Administrative Privileges.

Processing File

A file that contains processing information about the current *source* page. This information identifies the page according to its type; the identity of its Fingerprint Class and fingerprint; the locations of its fields; and preliminary recognized values of these fields. *See also: Source Page.*

Project

See Task Project.

Recognition Engine

Software that a “recognition” task uses to read and interpret the values in the fields of a *source* page. Usually, an application uses different engines to read machine-printed and hand-printed text, and to “vote” on values in certain fields.

Recognition Task

The task of a workflow’s Main job that is responsible for identifying pages according to their Page Types; for locating, reading, interpreting and assessing data in the fields of a *source* page; and for adding the data and the assessments of the data’s clarity to a Data file for that page. To carry out its work, the task applies rules of various kinds to the *source* page and its fields. Often, the task is termed a *RuleRunner* task.

Recognized Value

The *interpreted* value of a word on a *source* page.

Rule

A set of instructions that determine how a task is to carry out a narrowly defined activity involving a single object of the Document Hierarchy.

A rule consists of one or more actions.

Rules are categorized according to their RuleSet Type: **Locate** rules, for example, find fields and their values; **Export** rules extract values from a Data file and add them to an Export file or database. Multiple rules of the same type that present alternative ways to deal with the same object form a RuleSet. *See also: Action, RuleSet, RuleSet Type.*

Rule Manager

Development technology responsible for definitions of an application’s Document Hierarchy, fingerprints, rules, RuleSets and RuleSet Types. *Rule Manager* activities take place within the *Rule Manager Window* - a product of the *Batch Pilot* workshop.

Rules-based Tasks

These are tasks that operate in response to rules and their actions. The rules are applied directly to objects of a Document Hierarchy that has also been constructed exclusively for a particular workflow.

The Document Hierarchy has objects at four levels: **Batch, Document, Page** and **Field**. Rules are categorized according to a RuleSet Type such as Recognition, Locate, or Export. Tasks developed in the *Batch Pilot* workshop are *rules-based* tasks.

Rules Database

A database with complete information about an application's rules, actions and RuleSets – and their association with individual objects of the Document Hierarchy. *See also: Actions, Rules, RuleSets, Document Hierarchy.*

RuleSet

A group of rules of the same RuleSet Type that apply to same object of the Document Hierarchy. As an example, the **Locate** rules that look for the **Date** field on a *source* page belong to the same RuleSet. *See also: RuleSet Type, Source Page.*

RuleSet Type

A category that defines the purpose of a particular RuleSet.

Runtime Form

A form associated with a particular task's operations. Background tasks such as *RuleRunner* and *Export* use similar *runtime* forms that display information but do not require operator involvement. In contrast, a Verify task's **Data Entry** panel is an example of a *runtime* form that differs from application to application, and demands an operator's attention. *See also: Setup Form*

Script-based Tasks

Tasks that process documents in response to the specifications of a FormSpec. Although the setup of individual tasks permits limited customization, an application's versatility results primarily from scripts that expand (or limit) a task's role as it processes a particular page, or a field within a page. *See also: Rules-based Tasks.*

Scripts

Code that powers a control or form, or the actions that are part of a rule. *See also: Action, Control, Form, Rule.*

Security

Procedures that define and govern the access of individuals and workstations to an application's databases, jobs and tasks; and limit an individual's involvement in administrative activities. *See also: Administrative Privileges, Job-Task Permissions.*

Setup Form

A form associated with a particular task's definition. Setup of an application's *rules-based* tasks rely on a similar form that assigns RuleSet Types to the task. Scan tasks and FixUp tasks use far more detailed forms that relate a task to a physical scanner and scanning procedures, and to a scanner driver. *See also: Runtime Form*

Source Page

A **Page** object of the Document Hierarchy with **Field** objects. Rules associated with the **Field** objects of a *source* page determine how the workflow recognizes, extracts, verifies and validates the fields' values – and exports the values to a file or database. *See also: Page, Document Hierarchy.*

Stock Form

A *setup* form or *runtime* form - or a Multi-tab Form – that has been formally constructed and tested, and is in a folder of the **Datacap** directory's **BPilot** sub-directory. You can bind a stock form (.dcf) without difficulty to a new Task Project you're assembling...or to a stock Task Project (.bpp) if it's also in the folder. *See also: Binding, Task Project.*

Task

The processing object of a Workflow Hierarchy. A task must be part of a job; a job must be part of a workflow. The task's identity is defined in *Taskmaster* when it is added to the Workflow Hierarchy. However, a *rules-based* task that is a product of the *Batch Pilot* workshop requires setup specifications and detailed settings from the underlying Task Project. *See also: Task Project, Workflow Hierarchy.*

Task Definition

A *Taskmaster* process that defines a processing task as an object of a Workflow Hierarchy made up of a workflow, its jobs and the jobs' tasks. When a Task Definition incorporates the forms and settings of a *Batch Pilot* Task Project, a Task Module links the task to the project...and *Taskmaster* to *Batch Pilot*. *See also: Task Module, Task Project.*

Taskmaster

The administrative engine of an application. *Taskmaster* defines and manages the workflow—beginning with the physical scanning of a page, and ending when the values in the fields of a *source* page have been verified and exported. *See also: Source Page*

Taskmaster Administrator

A secondary window of the *Taskmaster Window*, used to define an application's Workflow Hierarchy and Security Program.

Taskmaster Client

The object of the *Taskmaster* configuration responsible for initiating and monitoring a job's tasks. Taskmaster Clients access the application's databases and supporting files through a Taskmaster Server. The application's Security Program defines the scope of a client's activity: Administrative clients can carry out a full range of tasks and supplementary procedures. *See also: Taskmaster Server, Databases.*

Taskmaster Web

A processing environment that uses a Taskmaster Web Site to link remote Taskmaster Web Clients to an application's administrative and operating functions.

Task Module

The **required** attribute of a Task Definition that links *Taskmaster* to *Batch Pilot* - and the Task Definition to a Task Project (.bpp). The Task Module also determines if a task can create new batches, or route batches to "child" jobs. *See also: Task Definition, Task Project.*

Task Project

The *Batch Pilot* structure that is the foundation for Task Development. A typical Task Project consists of two Datacap forms (.dcf) – a **Setup** form and a **Runtime** form – and specifications of the project's **Task Settings** dialog. *Taskmaster's* Task Definition procedures use the **Setup** form to assign administrative and operating parameters to a task; when the task runs, the project's **Runtime** form serves as the user interface.

Task Projects are retained as **Batch Pilot Project** files (.bpp). The construction of a new Task Project **requires** an affiliation with an existing Document Hierarchy (.xml). *See also: Document Hierarchy, Runtime Form, Setup Form, Task Settings.*

Task Settings

Attributes of a Task Project specified in the tabs of the project's **Task Settings** dialog. Settings in the *General* tab identify the project's Document Hierarchy file (.xml) – and designate the *Task Module* settings to determine where batches will be stored if the project is the basis for "batch creation" tasks, and the conditions a task might encounter in the case of "job router" tasks.

Filter settings intentionally limit the scope of a task's operations. *Log* settings determine the content (if any) of a task's Log files, while *Statistic* settings govern whether or not the task will contribute to **Statistics** tables of the Engine database. *See also: Task Modules, Task Project.*

Task Simulator

The feature of *Batch Pilot* development that tests a new form or Task Project.

.tif File

See Image File.

Toolbox

See Development Toolbox.

User

1. An individual who fills out the fields of a paper copy of a form.
2. An individual who has access to one or more objects of an application and its databases. The application's Security Program provides each user with a User ID and Password, and specify the limits of a user's processing and database access "privileges." *See also: Form-Filler, Security.*

Validation

The process of making sure that a field's value meets specified criteria. For example, the criteria for a field might be that the field can only contain a number within a range of numbers. The process is governed by **Validate** rules that apply to specific **Field** objects of the Document Hierarchy, and are called by a job's RuleRunner and Verification tasks. *See also: LookUp Database.*

Verification Task

A core task of a Main job. This operator-intensive task retrieves information from the Image and Data files of a "problem" page and displays it in the fields of the application's **Data Entry** panel. After review by the operator, the task moves to the next problem page until the batch is verified. *See also: Data Entry Panel.*

Workflow

The top level of an application's Workflow Hierarchy. A job, the tasks assigned to the job, any child jobs that branch from the job, and the tasks of these child jobs. In the *Batch Pilot* environment, a Document Hierarchy is an essential property of a workflow. *See also: Document Hierarchy, Job, Task, Workflow Hierarchy.*

Workflow Hierarchy

The formal structure an application employs to retrieve, interpret, verify and store the data on a scanned page. A Workflow Hierarchy has three tiers. At the top is the workflow itself, which focuses on a particular form or operates within a particular environment. Jobs are objects of a workflow and occupy the second tier: a workflow usually has a Main job, and may have child jobs to handle special functions. Tasks are objects of a job, and are on the hierarchy's third tier. *See also: Job, Task, Workflow.*