ProHelp® EPM

Production & Process Monitoring System

Data Exchange Manual
For ProHelp® EPM, Release 6.0.0

MANUAL #810-0002

REV – C

February 12, 2003

TABLE OF CONTENTS

1. Introduction

MATTEC Corporation's ProHelp® EPM Production and Process Monitoring System is specifically designed for real-time monitoring of all types of production equipment. It is used extensively in the plastics injection molding, extrusion, blow molding, blown film, metal stamping, die casting, printing, painting, and assembly industries. The basis behind the benefits from the ProHelp® EPM system is the rationale that plant managers and operational people will take corrective actions to solve problems on production equipment when they are aware of such problems. ProHelp® EPM is the device to alert employees to problems immediately when the problems occur. Therefore, tremendous savings can occur in increased productivity and decreased scrap parts.

The ProHelp® EPM system combines computer hardware, computer software, and Machine Interface Units (MIUs) into an efficient system to provide real-time production monitoring, production reports, process alarms, plant scheduling aids, and SPC process and part capability analysis. Floor personnel can make use of the machine-mounted terminals to signal different departments for help, to view production results at the machine site, and to enter downtime reasons or scrap reasons. Production, downtime, and scrap reports may be generated on a shift and daily basis, or the user may generate these reports for extended time periods by specifying a start and end date for the desired report. Job history data is continuously summarized and available for management's review.

This manual describes the ProHelp® EPM facilities that support the sharing of data with other computer systems or applications. Each of these facilities accomplishes its task by reading or writing to a file system accessible to the user. ProHelp® EPM utilizes the Microsoft Windows NT operating system and the Microsoft SQL Server database. Users may connect to the system from most Microsoft Windows operating systems. As a result, ProHelp® EPM is quite powerful in its ability to share information with external systems.

Three facilities are described here:

Data Import

This optional feature is, for historical reasons, often referred to as "Download". It allows Tool IDs, Part IDs, Process Sheets, Job Descriptors, and Machine Schedules to be imported into ProHelp® EPM from an external database. The format for data import is a MATTEC-defined, delimited ASCII file. Reference Section 2 for additional information.

Data Export

This optional feature allows raw data from Production History, SPC History, Job Descriptors, Process Sheets, Part IDs, and Tool IDs to be exported from ProHelp® EPM. Reference Section 3 for additional information.

DTR Data Export

This optional feature allows the ProHelp® EPM system to be easily integrated with DTR Software International's *The Manufaturing Manager*™ (TMM) software. Reference Section 3.10 for additional information.

1.1 Conventions

Much of this document consists of record descriptions, which in turn consist of field descriptions. Fields are identified by their position in the record and the sequence of the field descriptions reflects that. Required fields for Data Import are indicated by “(required)” in the “FIELD” column.

In addition to its position or name, each field entry includes a type, format, and description. A field’s type may be one of the following:

FIELD TYPE	DESCRIPTION	EXAMPLES
Alphanumeric	Text that is a combination of letters, numbers, or both.	abc 123 abc123
Float	A "real number" that may include decimal places.	1.23 1.00 1
Integer	A "whole number" that may not include decimal places.	1 23 456
Date	The month, day, and 4-digit year, in the form MM/DD/YYYY, DD/MM/YYYY, or YYYY/MM/DD, depending upon System Configuration ("Units" tab). A 4-digit year is required. When data is output using Data Export, the date delimiter will either be a "/" or ".", depending upon System Configuration.	08/18/1999 12/31/2001 01/02/2003
Time	The time of day, using a 24-hour clock. Specified in the format hh:mm, where *hh* = hours using the 24-hour clock and *mm* = minutes. One or two digits are accepted for hours (hh), however, two digits are always required for minutes (mm). When data is output using Data Export, the time delimiter will either be a ":" or ".", depending upon System Configuration.	00:30 01:15 23:59 01:02

1.2 Copyright Notice

It is a violation of Mattec's copyright to use the DTR Data Export program, related database stored procedures, related database views, or other related utilities without purchasing the DTR Data Export Option from Mattec Corporation.

1.3 Contacting Mattec

Before executing Data Import, Data Export, or the DTR Data Export you must purchase the Data Import / Export option and/or the DTR Data Export option from Mattec Corporation. For additional information, please contact Mattec's Sales Department at 1-513-683-1802.

It is easiest to obtain technical support by visiting Mattec's website at http://www.mattec.com or by emailing the help desk at helpdesk@mattec.com.

2. Data Import

The optional Data Import program provides a means for transferring Tool IDs, Part IDs, Process Sheets, Job Descriptors, and Machine Schedules from an external system, such as a mainframe, into ProHelp® EPM. This program reads an appropriately formatted ASCII text file and uses the information it contains to create or modify data. If errors are encountered during processing, ProHelp® EPM generates an error report and the user is notified.

2.1 File and Record Format

Each record in the ASCII file is essentially an action request and causes specific functions to occur during processing of that record by ProHelp® EPM. Several such actions may be requested:

SUGGESTED ORDER	ACTION	REQUEST CODE
1	Create or modify a Tool ID. Refer to Section 2.2.	TI
2	Create or modify a Part ID. Refer to Section 2.3.	PI
3	Create or modify a Process Sheet (basic information). Refer to Section 2.4.	PP
4	Create or modify a Process Sheet (process-specific information). Refer to Section 2.5.	PE
5	Create of modify a Job Descriptor. Refer to Section 2.6.	JB
6	Delete a Machine Schedule. Refer to Section 2.7.	SD
7	Create or append to a Machine Schedule. Refer to Section 2.8.	SA

The suggested order takes into account dependencies between the various actions. For example, a PE (Process Sheet, process-specific information) record can only include existing process sheets, including those created by PP records in the same file; before a PP or PE record can reference a Tool ID or Part ID, the Tool ID and Part ID must already exist, and so forth.

All types of records need not be present in the file. For example, if a record requests the creation of a Process Sheet (PP/PE) using Part ID "123", and ProHelp® EPM already has a record for Part ID "123", then there is no need to precede the request with a Part ID (PI) creation request.

All fields within a record are separated by commas (","). All records must be terminated with a carriage return and line feed. The first field of each line always consists of one of the two-character request codes listed above.

Character fields can optionally be included in double quotes (" "). However, single quotes (' ') may not be used. Because of this, it is preferred that you do NOT use quote marks at all.

2.2 Create or Modify a Tool ID

The Tool ID action request record consists of a two-character request code (TI), the name of the Tool Number (up to 18 characters), and additional fields as defined below:

FIELD	TYPE	LENGTH	DESCRIPTION
1 (required)	Alphanumeric	2	The request code (TI).
2 (required)	Alphanumeric	18	The name of the Tool Number.
3	Integer		Total cavities for the tool.
4	Integer		Active cavities for the tool. May not exceed the total cavities.
5	Float		Runner weight specified in grams.
6	Alphanumeric	20	Description.
7	Integer		The setup time for the tool specified in minutes.

Optional fields may be omitted though the field separators (commas) are required unless they are at the end of the record. If an optional field is omitted and the Tool ID record already exists, the field is not modified. If an optional field is omitted and the Tool ID record does not already exist, the field is set to blanks (Alphanumeric type) or 0 (Integer or Float types).

The "Last Update" field for the Tool ID will be automatically updated with the current date and time. Other fields in the Tool ID record will not be modified.

Examples:

```
TI,Tool1,3,2,4.5,My Tool,60
TI,Tool2,3,3,3,Another Tool
```

2.3 Create or Modify a Part ID

The Part ID action request record consists of the two-character request code (PI), the name of the Part Number, and additional fields as defined below:

FIELD	TYPE	LENGTH	DESCRIPTION
1 (required)	Alphanumeric	2	The request code (PI).
2 (required)	Alphanumeric	18	The name of the Part Number.
3	Alphanumeric	25	Material ID.
4	Alphanumeric	25	Color ID.
5	Float		Percent regrind.
6	Float		Part weight specified in grams.
7	Float		Part cost.
8	Float		Material cost.
9	Alphanumeric	20	Description.

Optional fields may be omitted though the field separators (commas) are required unless they are at the end of the record. If an optional field is omitted and the Part ID record already exists, the field is not modified. If an optional field is omitted and the Part ID record does not already exist, the field is set to blanks (Alphanumeric type) or 0 (Integer or Float types).

The "Last Update" field for the Part ID will be automatically updated with the current date and time. Other fields in the Part ID record will not be modified.

Examples:

```
PI,Part1,Material1,Color1,10.1,2,3,4
PI,Part2,Material2,,6,1.1,2.2,3.3,4.4
```

2.4 Create or Modify a Process Sheet (Basic Information)

The Process Sheet (Basic Information) action request record consists of the two-character request code (PP), the name of the Machine Number, the name of the Tool Number, the name of the Part Number, and additional fields as defined below:

FIELD	TYPE	LENGTH	DESCRIPTION
1 (required)	Alphanumeric	2	The request code (PP).
2 (required)	Alphanumeric	6	The name of the Machine Number.
3 (required)	Alphanumeric	18	The name of the Tool Number.
4 (required)	Alphanumeric	18	The name of the Part Number.
5 (*)	Float		Expected percent up.
6 (*)	Float		Expected cycle time specified in seconds.
7 (*)	Float		Expected percent good.
8	Float		Labor hour factor.
9	Float		Labor cost.
10	Alphanumeric	20	Description 1.
11	Alphanumeric	20	Description 2.
12 (*)	Float		Non-production limit specified in seconds.

* - This field is required if this is a new record, otherwise, the field is optional.

Optional fields may be omitted though the field separators (commas) are required unless they are at the end of the record. If an optional field is omitted and the Process Sheet record already exists, the field is not modified. If an optional field is omitted and the Process Sheet record does not already exist, the field is set to blanks (Alphanumeric type) or 0 (Integer or Float types).

The "Last Update" field for the Process Sheet will be automatically updated with the current date and time. Other fields in the Process Sheet record will not be modified.

When creating a Process Sheet, you must have one PP action request and multiple PE action requests, one for every parameter that is defined in the machine's configuration. The PP action request must be executed before the PE action request.

Note: If the PP and PE action requests are not done properly, the user will not be able to edit the Process Sheet via Edit Facilities. The Process Sheet record must be corrected using valid PP and PE action requests before Edit Facilities will work properly. In particular, the PE requests must correspond precisely with the parameter numbers that are defined in the machine's configuration program. Reference Section 2.5 for additional information.

Examples:

```
PP,Mach1,Tool1,Part1,99,10.5,94,1.5,4.53,Description1,Description2,30.0
PP,01,123501,124F-13,98.5,10,94.5
```

2.5 Create or Modify a Process Sheet (Process-specific Information)

The Process Sheet (Process-specific Information) action request record consists of the two-character request code (PE), the name of the Machine Number, the name of the Tool Number, the name of the Part Number, and additional fields as defined below:

FIELD	TYPE	LENGTH	DESCRIPTION
1 (required)	Alphanumeric	2	The request code (PE).
2 (required)	Alphanumeric	6	The name of the Machine Number.
3 (required)	Alphanumeric	18	The name of the Tool Number.
4 (required)	Alphanumeric	18	The name of the Part Number.
5 (required)	Integer		The parameter number. This number may have the values 0-63 and must **correspond precisely** with the parameters that are defined in the machine's configuration (and stored in the dbo.MachParm table in the field ParmNo).
6	Float		The parameter's upper limit.
7	Float		The parameter's nominal value. This value may not exceed the parameter's upper limit or be below the parameter's lower limit.
8	Float		The parameter's lower limit.

Optional fields may be omitted though the field separators (commas) are required unless they are at the end of the record. Any fields not present in the record are set to 0 (Integer or Float types).

The "Last Update" field for the Process Sheet will be automatically updated with the current date and time. Other fields in the Process Sheet record will not be modified.

When creating a Process Sheet, you must have one PP action request and multiple PE action requests, one for every parameter that is defined in the machine's configuration. The PP action request must be executed before the PE action request.

Note: If the PP and PE action requests are not done properly, the user will not be able to edit the Process Sheet via Edit Facilities. The Process Sheet record must be corrected using valid PP and PE action requests before Edit Facilities will work properly. In particular, the PE requests must correspond precisely with the parameter numbers that are defined in the machine's configuration program (and stored in the MachParm database table).

Examples:

```
PE,01,Tool1,Part1,0,10,9.5,9
PE,01,Tool1,Part2,5,21,20.5,20
PE,01,Tool1,Part3,60,21.3,21.2,21.1
```

2.6 Create or Modify a Job Descriptor

The Job Descriptor action request record consists of the two-character request code (JB), the name of the Job Number, the name of the Machine Number, the name of the Tool Number, the name of the Part Number, and additional fields as defined below:

FIELD	TYPE	LENGTH	DESCRIPTION
1 (required)	Alphanumeric	2	The request code (JB).
2 (required)	Alphanumeric	18	The name of the production Job Number.
2 (required)	Alphanumeric	6	The name of the Machine Number.
3 (required)	Alphanumeric	18	The name of the Tool Number.
4 (required)	Alphanumeric	18	The name of the Part Number.
5 (required)	Date	10	The job's desired start date.
6	Time	5	The job's desired start time.
7 (required)	Date	10	The job's desired stop date.
8	Time	5	The job's desired stop time.
9 (required)	Integer		The lot size.
10	Alphanumeric	18	The name of the Customer.
11	Alphanumeric	20	Comment 1.
12	Alphanumeric	20	Comment 2.

Optional fields may be omitted though the field separators (commas) are required unless they are at the end of the record.

The Job Number that is specified must be a unique identifier for this job. Although ProHelp® EPM supports non-unique Job Numbers (that is, multiple Job Descriptors with the same Job Number), the Data Import function requires unique Job Numbers.

The following describe the action that is taken for the "JB" action request:

- If the specified Job Descriptor doesn't exist, the job will be created. It may then be scheduled using an Append to Machine Schedule (“SA”) request.
- If the specified Job Descriptor exists, all fields except the Machine Number, Tool Number, and Part Number will be updated. Any fields not present in the record are set to blanks (Alphanumeric) or 0 (Integer or Float types).

Examples:

```
JB,Job1,Mach1,Tool1,Part1,01/31/2001,23:01,12/31/2003,03:59,1000,cust1,comment 1,c2
JB,2-CHR,Mach1,Tool1,Part2,12/15/2001,12:00,12/25/2001,12:00,1500
```

2.7 Delete a Machine Schedule

The Delete Machine Schedule action request record consists of the two-character request code (SD) and the name of the Machine Number. Job Descriptors that are scheduled on the current machine are deleted from the machine's schedule (except for the currently running job).

FIELD	TYPE	LENGTH	DESCRIPTION
1 (required)	Alphanumeric	2	The request code (SD).
2 (required)	Alphanumeric	6	The name of the Machine Number.

The Delete Machine Schedule action request can be used in combination with one or more Machine Schedule action requests (SA) to rewrite a machine's schedule.

Examples:

```
SD,Mach1
SA,Mach1,Job1
SA,Mach1,Job2
```

2.8 Create or Append to a Machine Schedule

The Machine Schedule action request record consists of the two-character request code (SA), the name of the Machine Number, and the name of the Job Number to be scheduled.

The Job Number in the action request record is appended to the end of the current schedule. If the Job Number is already scheduled, or if the job is currently running, the action request is aborted and an error is reported.

FIELD	TYPE	LENGTH	DESCRIPTION
1 (required)	Alphanumeric	2	The request code (SA).
2 (required)	Alphanumeric	6	The name of the Machine Number.
3 (required)	Alphanumeric	18	The name of the production Job Number.

Machine Schedule action requests may only contain jobs for which Job Descriptors already exist. The Job Descriptor may have just been created using a previous Job Descriptor action request (JB).

The Job Number that is specified must be a unique identifier for this job. Although ProHelp® EPM supports non-unique Job Number (that is, multiple Job Numbers with the same name), the Data Import function requires unique Job Numbers.

Examples:

```
SA,Mach1,Job1
SA,Mach1,Job2
```

3. Data Export

The Data Export program is actually comprised of many separate programs that allow the user to export the following information:

DATA EXPORT TYPE
Production History. Refer to Section 3.3.
SPC History. Refer to Section 3.4.
Job Descriptor information. Refer to Section 3.5.
Process Sheet information. Refer to Section 3.6.
Part ID Information. Refer to Section 3.7.
Tool ID information. Refer to Section 3.8.
DTR data export. Refer to Section 3.10.

The following sections describe each of these Data Export programs.

3.1 Limiting The Amount Of Exported Data

When you select to export Production History data and specify the "Specific Shift" filter, the "Shift" that is displayed always includes the numbers 1-10. These correspond to the internal "ShiftSeq" field that is used in the database (ShiftProd.ShiftSeq), where "1" represents the first shift of the specified day, "2" represents the second shift of the specified day, etc.

These numbers are not the "Shift Name" that is specified in the Shift Boundaries Edit Facilities program.

3.2 Weighted Values

Weighted values apply primarily to the Production History data export (Section 3.3) and the Job Descriptor data export (Section 3.5).

Weighted values are used for family molding. Weighted values are stored for bachelor jobs, but the weighted value (e.g., weighted down time) will contain the same value as the non-weighted value (e.g., down time).

When considering family molding, however, weighted values are very important. These values allow for the display of overall family-specific information.

Consider this example: Component jobs “son1” and “son2” ran for 8 hours at the same time. These are related family jobs. A production report would display 8 hours of run time for “son1” and 8 hours of run time for “son2”. However, the total and grand total for the report would show only 8 hours of run time, not 16 hours of run time. That’s because the family only ran for 8 hours.

If you looked at the raw data, the Production History database table would contain 8 hours of total time for “son1” and 4 hours of weighted total time for “son1”. The same would be true for “son2”. Thus, the detail records in the report would display total time, whereas the total lines would display the sum of the weighted total times.

(Some people want to see the sum of the non-weighted values in the totals and grand totals displays. Thus, in the example above, the report would display 8 hours of run time for “son1”, 8 hours of run time for “son2”, and 16 hours of total run time. Obviously this can be accomplished by displaying the sum of the non-weighted total times.)

Weighted values are accumulated in real-time and vary depending on the number of son jobs that are currently running for the family. Thus, if two son jobs are running, each job gets ½ of the run time (actually total time), down time, etc. stored in its weighted values. If three son jobs are running, each job gets a 1/3rd weighting, etc.

3.3 Production History Data Export

The Production History data export selection allows the user to export production-related information by shift. This program can be useful for updating external systems with production-related information for jobs by shift, including the number of parts produced for a shift and the number of packed parts for a shift.

The Production History data export is very similar to the Job History data export. The Production History selection exports data by shift, whereas the Job History selection exports data for the entire job. Data may be exported for the current day, previous day, or a date/shift range for one or all departments.

The following table describes the fields that are exported in the Production History data export program:

#	FIELD	TYPE (LENGTH)	DB SOURCE	DESCRIPTION
1	JobID	Alphanumeric (18)	JobQueue	The name of the Job Number.
2	MachID	Alphanumeric (6)	MachCon	The name of the Machine Number.
3	PartID	Alphanumeric (18)	PartIDs	The name of the Part Number.
4	MoldID	Alphanumeric (18)	MoldIDs	The name of the Tool Number.
5	MatlID	Alphanumeric (25)	PartIDs	The name of the Material ID.
6	ColorID	Alphanumeric (25)	PartIDs	The name of the Color ID.
7	CustomerID	Alphanumeric (18)	JobQueue	The name of the Customer.
8	Shift Date	Date	ShiftProd	The date of the shift.
9	Shift Index	Integer	ShiftProd	The shift index for the day. The first shift of the production day would be 1. Values can range from 1-10.
10	CycCnt	Integer	ShiftProd	Total number of machine cycles for the shift.
11	WtCycCnt	Float	ShiftProd	The “weighted” total number of machine cycles for the shift. See Section 3.2.
12	CycOutSpec	Integer	ShiftProd	Total number of out-of-spec cycles for the shift. Any cycle in which one or more process parameters are out-of-spec is an out-of-spec cycle.
13	ExpProdQty	Integer	ShiftProd	Expected number of parts to be produced based on information in the process sheet.
14	CalProdQty	Integer	ShiftProd	The total number of parts produced for the shift as monitored by the MIU.
15	PakProdQty	Integer	ShiftProd	Packed parts reported for the shift.
16	DefectQty	Integer	ShiftProd	Scrap parts reported for the shift.

#	FIELD	TYPE (LENGTH)	DB SOURCE	DESCRIPTION
17	TotTime	Integer	ShiftProd	Total time (run time plus down time) in seconds for the shift.
18	WtTotTime	Float	ShiftProd	The "weighted" total time in seconds for the shift. See Section 3.2.
19	DownTime	Integer	ShiftProd	Down time in seconds for the shift.
20	WtDownTime	Float	ShiftProd	The "weighted" down time in seconds for the shift. See Section 3.2.
21	NumDownTm	Integer	ShiftProd	The number of downtime occurrences for the shift.
22	WtNumDown Tm	Float	ShiftProd	The "weighted" number of downtime occurrences for the shift. See Section 3.2.
23	LaborTime	Integer	ShiftProd	Amount of labor time in seconds calculated for the shift.
24	WtLaborTime	Float	ShiftProd	The "weighted" amount of labor time in seconds calculated for the shift. See Section 3.2.
25	MinCycTm	Float	ShiftProd	Minimum cycle time value that occurred during the shift.
26	MaxCycTm	Float	ShiftProd	Maximum cycle time value that occurred during the shift.
27	SumCycTm	Float	ShiftProd	Sum of the cycle time values that occurred during the shift. Useful in calculating the average for cycle time.
28	SumSqCycTm	Float	ShiftProd	Sum of the squares of the cycle time values that occurred during the shift. Useful in calculating the standard deviation for cycle time.
29	NumCavs	Float	MoldIDs	The total cavities for the tool.
30	RunnerWt	Float	MoldIDs	Runner weight in grams. This value does not get converted to the display unit from System Configuration.
31	PctReg	Float	PartIDs	Percentage regrind of total for the material.
32	PartWt	Float	PartIDs	Part weight in grams. This value does not get converted to the display unit from System Configuration.
33	PcsPerCtn	Integer	PartIDs	Pieces per carton or box.
34	PartCost	Float	PartIDs	Part cost in $ per unit.
35	MatlCost	Float	PartIDs	Material cost in $ per unit.

3.4 SPC History Data Export

The SPC History data export selection allows the user to export information about Automatic SPC. This program can be useful for updating external systems with SPC data that has been collected by an MIU. Data may be exported for the current day, previous day, or a date range for one or all departments.

The following table describes the fields that are exported in the SPC History data export program:

#	FIELD	TYPE (LENGTH)	DB SOURCE	DESCRIPTION
1	MachID	Alphanumeric (6)	MachCon	The name of the Machine Number.
2	JobID	Alphanumeric (18)	JobQueue	The name of the Job Number.
3	Name	Alphanumeric (10)	ParmSet	The name of the Parameter.
4	Sample Date	Date	LongTermEng	The date the sample was taken.
5	Sample Time	Time	LongTermEng	The time the sample was taken.
6	ParmNo	Integer	LongTermEng	The zero-based parameter number.
7	LtCycleIndex	Integer	LongTermEng	The long-term cycle index for each period (0 for the first cycle, up to "N-1", where N is the subgroup size).
8	Value	Float	LongTermEng	The process parameter value.
9	Flag	Alphanumeric (1)	LongTermEng	A flag indicating the status of this particular sample, where: • '.' is in spec; • '-' is below lower spec, • '+' is above upper spec, • 'x' is invalid parameter value (unable to calculate), • '?' is unknown.

3.5 Job Descriptor Data Export

The Job Descriptor data export selection allows the user to export information about all jobs that have been run. Data for running, suspended, and completed jobs is included in the data export. This program can be useful for updating external systems with production-related information for jobs, including the total number of parts produced and the total number of packed parts.

The Job History data export is very similar to the Production History data export. The Job History selection exports data for the entire job, whereas the Production History selection exports data by shift.

The following table describes the fields that are exported in the Job Descriptor data export program:

#	FIELD	TYPE (LENGTH)	DB SOURCE	DESCRIPTION
1	JobID	Alphanumeric (18)	JobQueue	The name of the Job Number.
2	MachID	Alphanumeric (6)	MachCon	The name of the Machine Number.
3	PartID	Alphanumeric (18)	PartIDs	The name of the Part Number.
4	MoldID	Alphanumeric (18)	MoldIDs	The name of the Tool Number.
5	MatlID	Alphanumeric (25)	PartIDs	The name of the Material ID.
6	ColorID	Alphanumeric (25)	PartIDs	The name of the Color ID.
7	CustomerID	Alphanumeric (18)	JobQueue	The name of the Customer.
8	StartDate	Date	JobQueue	The job's actual start date.
9	StartTime	Time	JobQueue	The job's actual start time in the format hh:mm.
10	StopDate	Date	JobQueue	The job's actual stop date. If the job has not ended yet, the value 0/0/0 will be used.
11	StopTime	Time	JobQueue	The job's actual stop time in the format hh:mm or hh.mm. If the job has not ended yet, the value 0:0 will be used.
12	SchedQty	Integer	JobQueue	The lot size.
13	CycCnt	Integer	JobProd	Total number of machine cycles for the job.
14	WtCycCount	Float	JobProd	The “weighted” total number of machine cycles for the job.
15	CycOutSpec	Integer	JobProd	Total number of out-of-spec cycles for the job. Any cycle in which one or more process parameters are out-of-spec is an out-of-spec cycle.
16	ExpProdQty	Integer	JobProd	Expected number of parts to be produced based on information in the process sheet.
17	CalProdQty	Integer	JobProd	The total number of parts produced for the job as monitored by the MIU.

#	FIELD	TYPE (LENGTH)	DB SOURCE	DESCRIPTION
18	PakProdQty	Integer	JobProd	Packed parts reported for the job.
19	DefectQty	Integer	JobProd	Scrap parts reported for the job.
20	TotTime	Integer	JobProd	Total time (run time plus down time) in seconds for the job.
21	WtTotTime	Float	JobProd	The "weighted" total time in seconds for the job.
22	DownTime	Integer	JobProd	Down time in seconds for the job.
23	WtDownTime	Float	JobProd	The "weighted" down time in seconds for the job.
24	NumDownTm	Integer	JobProd	The number of downtime occurrences for the job.
25	WtNumDown Tm	Float	JobProd	The "weighted" number of downtime occurrences for the job.
26	LaborTime	Integer	JobProd	Amount of labor time in seconds calculated for the job.
27	WtLaborTime	Float	JobProd	The "weighted" amount of labor time in seconds calculated for the job.
28	MinCycTm	Float	JobProd	Minimum cycle time value that occurred during the job.
29	MaxCycTm	Float	JobProd	Maximum cycle time value that occurred during the job.
30	SumCycTm	Float	JobProd	Sum of the cycle time values that occurred during the job. Useful in calculating the average for cycle time.
31	SumSqCycTm	Float	JobProd	Sum of the squares of the cycle time values that occurred during the job. Useful in calculating the standard deviation for cycle time.
32	NumCavs	Float	MoldIDs	The total cavities for the tool.
33	RunnerWt	Float	MoldIDs	Runner weight in grams. This value does not get converted to the display unit from System Configuration.
34	PctReg	Float	PartIDs	Percentage regrind of total for the material.
35	PartWt	Float	PartIDs	Part weight in grams. This value does not get converted to the display unit from System Configuration.
36	PcsPerCtn	Integer	PartIDs	Pieces per carton or box.
37	PartCost	Float	PartIDs	Part cost in $ per unit.
38	MatlCost	Float	PartIDs	Material cost in $ per unit.
39	Job Status	Alphanumeric (4)	JobQueue	The job's status, PEND (Pending), RUN (Running), SUSP (Suspended), or COMP (Completed).

3.6 Process Sheet Data Export

The Process Sheet data export selection allows the user to export information about process sheets. This program can be useful for updating external systems with process parameter-related information from process sheets.

The following table describes the fields that are exported in the Process Sheet data export program:

#	FIELD	TYPE (LENGTH)	DB SOURCE	DESCRIPTION
1	MachID	Alphanumeric (6)	MachCon	The name of the Machine Number.
2	PartID	Alphanumeric (18)	PartIDs	The name of the Part Number.
3	MoldID	Alphanumeric (18)	MoldIDs	The name of the Tool Number.
4	SampID	Alphanumeric (18)	QCSheet	The name of the Sample ID.
5	Name	Alphanumeric (10)	ParmSet	The name of the Parameter.
6	ParmNo	Integer	PSEng	The zero-based parameter number.
7	UpperLim	Float	PSEng	The Engineering upper specification limit.
8	NomVal	Float	PSEng	The Engineering nominal value.
9	LowerLim	Float	PSEng	The Engineering lower specification limit.
10	ExpPctUp	Float	PSProd[1]	Standard percent uptime.
11	ExpCycTm	Float	PSProd	Standard cycle time.
12	ExpGood	Float	PSProd	Standard percent good parts.
13	HistPctUp	Float	PSProd	Historical percent uptime.
14	HistCycTm	Float	PSProd	Historical cycle time.
15	HistGood	Float	PSProd	Historical percent good parts.
16	LaborFactor	Float	PSProd	Standard labor factor.
17	LaborCost	Float	PSProd	Cost in $ per unit time of an operator. Used as a multiplier of labor time to produce a cost of direct labor for a period of time.
18	NPL	Float	PsProd	Non-Production Limit, specified in seconds.

[1] Fields from the PSProd table do not vary as the process parameter (ParmNo field in PSEng database table) varies. They are included in this data export file to simplify the process of taking process sheet information to an external system.

3.7 Part ID Data Export

The Part ID data export selection allows the user to export information about parts. This program can be useful for updating external systems with information specific to a produced part or item.

The following table describes the fields that are exported in the Part ID data export program:

#	FIELD	TYPE (LENGTH)	DB SOURCE	DESCRIPTION
1	PartID	Alphanumeric (18)	PartIDs	The name of the Part Number.
2	PartDesc	Alphanumeric (20)	PartIDs	A description for the part.
3	MatlID	Alphanumeric (25)	PartIDs	Material ID.
4	PctReg	Float	PartIDs	Percent regrind of total material requirement.
5	PartWt	Float	PartIDs	Part weight in grams. This value does not get converted to the display unit from System Configuration.
6	PcsPerCtn	Integer	PartIDs	Pieces per container.
7	PartCost	Float	PartIDs	Part cost.
8	MatlCost	Float	PartIDs	Material cost.
9	ColorID	Alphanumeric (25)	PartIDs	Color ID.
10	MiscInfo1	Alphanumeric (20)	PartIDs	Part description 1.
11	MiscInfo2	Alphanumeric (20)	PartIDs	Part description 2.
12	PartInfo	Alphanumeric (40)	PartIDs	Part information.

3.8 Tool ID Data Export

The Tool ID data export selection allows the user to export information about tools. This program can be useful for updating external systems with information for a physical tool, mold, die, fixture, or other apparatus that is used in conjunction with a machine to produce a certain part or a group of parts.

The following table describes the fields that are exported in the Tool ID data export program:

#	FIELD	TYPE (LENGTH)	DB SOURCE	DESCRIPTION
1	MoldID	Alphanumeric (18)	MoldIDs	The name of the Tool Number.
2	MoldDesc	Alphanumeric (20)	MoldIDs	A description of the tool.
3	MachReq	Alphanumeric (20)	MoldIDs	User-defined description field for machine type.
4	NumCavs	Integer	MoldIDs	Total cavities for the tool.
5	ActCavs	Integer	MoldIDs	Active cavities for the tool. May not exceed the total cavities.
6	Maker	Alphanumeric (20)	MoldIDs	User-defined description field for describing the tool's maker.
7	Location	Alphanumeric (20)	MoldIDs	User-defined description field for describing the tool's current location.
8	RunnerWt	Float	MoldIDs	Runner weight in grams. This value does not get converted to the display unit from System Configuration.
9	MiscInfo1	Alphanumeric (20)	MoldIDs	Tool description 1.
10	MiscInfo2	Alphanumeric (20)	MoldIDs	Tool description 2.
11	Setup Time	Integer	MoldIDs	The setup time in minutes for the tool.

3.9 Operator Tracking Data Export

The Operator Tracking data export selection allows the user to export information about all machine operators who have logged in for Operator Efficiency or Operator Tracking. This program can be useful for updating external time tracking systems.

The following table describes the fields that are exported in the Operator Tracking data export program:

#	FIELD	TYPE (LENGTH)	DB SOURCE	DESCRIPTION
1	OperatorID	Alphanumeric (18)	OperatorLogin	The Operator's ID.
2	MachID	Alphanumeric (6)	MachCon	The name of the Machine Number the operator logged in to.
3	LoginDate	Date	OperatorLogin	The login date.
4	LoginTime	Time	OperatorLogin	The login time in the format hh:mm.
5	LogoutDate	Date	OperatorLogin	The logout date. If the operator has not logged out yet, the value 0/0/0 will be used.
6	LogoutTime	Time	OperatorLogin	The logout time in the format hh:mm. If the operator has not logged out yet, the value 0:0 will be used.
7	ElapsedTime	Integer	OperatorLogin	The elapsed login time specified in seconds. If the operator has not logged out yet, the value 0 will be used.
8	Description	Alphanumeric (20)	OperatorLoginTransType	A description of the login type, usually "Operator Efficiency" or "Operator Tracking".

3.10 DTR Data Export

The DTR data export selection is an optional feature.

The DTR Data Export option for ProHelp® EPM allows the ProHelp® EPM system to be easily integrated with DTR Software International's *The Manufacturing Manager*™ (TMM) software. It allows ProHelp® EPM shift history data to be exported using the *TMM Standard* format (revised February 3, 1999).

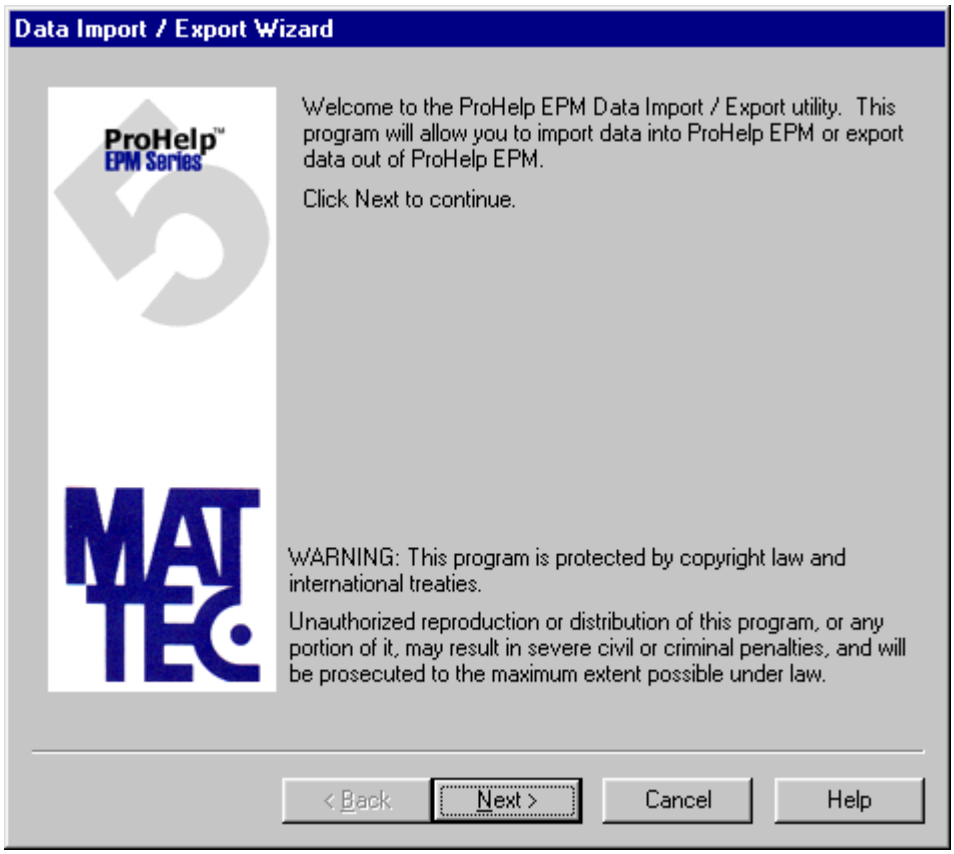
4. Operation

Before executing Data Import, Data Export, or the DTR Data Export you must purchase the Data Import / Export option and/or the DTR Data Export option from Mattec Corporation. For additional information, please contact Mattec's Sales Department at 1-513-683-1802.

To execute the Data Import / Export program, follow these steps:

- Install the ProHelp® EPM software on your computer.
- Click on the **`Start Menu`**.
- Select **`Programs`**.
- Select the **`Mattec`** folder.
- Select the **`Utilities`** folder.
- Select the **`Data Import / Export`** program.

The Data Import / Export Wizard will be displayed:



Follow the onscreen prompts as required. The status of the Data Import, Data Export, or DTR Data Export will be shown, as appropriate.

5. Direct Database Operation (Advanced Users Only)

As noted in Section 1.2, it is a violation of Mattec's copyright to use the Data Import / Export program, DTR Data Export program, database stored procedures, database views, or other related utilities without purchasing the Data Import / Export option or the DTR Data Export option, as appropriate, from Mattec Corporation.

However, if you have purchased the Data Import / Export option or DTR Data Export option, a very knowledgeable database administrator could call the related stored procedures and views to create a sophisticated and/or automated interface between ProHelp® EPM and your ERP system.

Before implementing an advanced interface, it’s important to note that the stored procedures and views are subject to change without notice.

5.1 Exporting Data Using Views (Advanced Users Only)

Mattec utilizes database views to extract data from the database for the purpose of exporting it to external systems. The Data Import / Export program calls the appropriate views to extract data, processes that data, then writes the processed data to an ASCII text file.

It's possible for a knowledgeable database administrator to use these same views to create a sophisticated and/or automated interface between ProHelp® EPM and your ERP system, without the need to create an intermediate ASCII text file.

Following is a brief description of the database views that are used by the Data Import / Export program. For the exact format of data that is extracted, refer to the actual views.

DATABASE VIEW	DESCRIPTION
vExportShiftProd	Provides shift history information. This view is used by the "Production History" export option. These fields are described in Section 3.3.
vExportLongTermSPC	Provides data related to long-term (automatic) SPC history. This view is used by the "SPC History" export option. These fields are described in Section 3.4.
vExportJob	Provides data related to any job that has run. Used by the "Job History" export option. These fields are described in Section 3.5.
vExportProcessSheet	Provides data stored in Process Sheets. This view is used by the "Process Sheets" export option. These fields are described in Section 3.6.
vExportPartID	Provides data stored in Part IDs. This view is used by the "Part IDs" export option. These fields are described in Section 3.7.
vExportToolID	Provides data stored in Tool IDs. This view is used by the "Tool IDs" export option. These fields are described in Section 3.8.
vExportOperatorLogin	Provides data related to operator logins. This view is used by the "Operator Tracking" export option. These fields are described in Section 3.9.

Examples:

```
SELECT * FROM vExportJob

SELECT * FROM vExportShiftProd
WHERE ShiftSeq >= 200108180 AND
      ShiftSeq <= 200108189
```

5.2 Importing Data Using Stored Procedures (Advanced Users Only)

Mattec utilizes database stored procedures to import data into the database. The Data Import / Export program reads an ASCII text file, processes that data, then calls the appropriate stored procedures using that processed data to create or update the appropriate database records.

It's possible for a knowledgeable database administrator to use these same stored procedures to create a sophisticated and/or automated interface between ProHelp® EPM and your ERP system, without the need to create an intermediate ASCII text file.

Following is a brief description of the database stored procedures that are used by the Data Import / Export program. For the exact arguments that are required, refer to the actual stored procedures.

DATABASE STORED PROCEDURE	**DESCRIPTION**
pImportAppendJobToSchedule	Schedules an existing, unscheduled job. The job must exist in the Job Descriptor table. This stored procedure is used by the "Create or Append to Machine Schedule (SA)" action request in Section 2.8.
pImportJob	Use to create a new job and place it in the Job Descriptor table or to update an existing job. This stored procedure is used by the "Create or Modify a Job Descriptor (JB)" action request in Section 2.6.
pImportPart	Used to create or modify a Part ID. This stored procedure is used by the "Create or Modify a Part ID (PI)" action request in Section 2.3.
pImportPSEng (*)	Used to create or modify the process-specific information for a Process Sheet. This stored procedure is used by the "Create or Modify a Process Sheet (Process-specific Information) (PE)" action request in Section 2.5.
pImportPSProd (*)	Used to create or modify the basic information for a Process Sheet. This stored procedure is used by the "Create of Modify a Process Sheet (Basic Information) (PP)" action request in Section 2.4.
pImportScheduleDelete	Used to delete a machine's schedule. This stored procedure is used by the "Delete a Machine Schedule (SD)" action request in Section 2.7.
pImportTool	Used to create or modify a Tool ID. This stored procedure is used by the "Create of Modify a Tool ID (TI)" action request in Section 2.2.

* - If the pImportPSEng and pImportPSProd stored procedures are not done properly, the user will not be able to edit the Process Sheet via Edit Facilities. The Process Sheet record must be corrected manually (using pImportPSEng and pImportPSProd) before Edit Facilities will work properly (for this Process Sheet). Reference Section 2.5 for additional information.

Examples:

```
pImportJob 'Job1', 'Mach1', 'Tool1', 'Part1', 998111111, 1009300000, 1000, 998111111,
cust1, comment1, c2

pImportScheduleDelete 'Mach1'

pImportAppendToSchedule 'Job1', 'Mach1'
```

Note: There are other database stored procedures that have the name **pImportXXX**. Many of those stored procedures are used by the Data Conversion program and are not appropriate for use when importing data from an external ERP system.

5.3 Sample Programs (Advanced Users Only)

A knowledgeable user could create a custom stored procedure to import data directly into ProHelp® EPM from a third-party system and/or to export data from ProHelp® EPM to a third-party system. This stored procedure could then be scheduled to run automatically (using the SQL Server agent). This stored procedure could also execute without the need for an intermediate ASCII text file.

Before connecting to an external database system, it may be necessary to add a "New SQL Server Registration" to the external system on the ProHelp® EPM server. Similarly, it may be necessary to execute the following in the SQL Server Query Analyzer:

```
sp_addlinkedserver MYSERVER
```

Where ***MYSERVER*** is the name of an external database server. This command only needs to be executed one time.

Reference the SQL Server documentation that came with your system for additional information.

5.3.1 Example 1 – Data Import

Assume the following sample stored procedure exists in the same database as ProHelp® EPM. It will extract a machine schedule from an external system (computer "MYSERVER", database "MYDATABASE", table "MyTable") and import the schedule into ProHelp® EPM using the pImportAppendToSchedule stored procedure.

```
CREATE PROCEDURE SampleCustomDataImport AS
BEGIN

  DECLARE @varJobID  VARCHAR(18)
  DECLARE @varMachID VARCHAR(5)

  /* Create a cursor to hold the data from the external system */
  DECLARE JobListCursor CURSOR FOR
        SELECT JobId, MachId FROM MYSERVER.MYDATABASE.dbo.MyTable

  /* Open the cursor */
  OPEN JobListCursor

  /* Loop through the job list cursor */
  FETCH JobListCursor INTO @varJobID, @varMachId
  WHILE (@@fetch_status >= 0)
  BEGIN

        /* Add the Job to ProHelp EPM's schedule */
        EXEC pImportAppendJobToSchedule @varJobId, @varMachId

        /* Fetch the next row from the cursor */
        FETCH JobListCursor INTO @varMachId, @varJobID

  END

  /* Clean Up */
  DEALLOCATE JobListCursor

END
```

For your convenience, the above stored procedure actually exists in the MATTEC_PROHELP database on your ProHelp® EPM system.

5.3.2 Example 2 – Data Export

Assume the user wishes to select Production History data from ProHelp® EPM and insert the data into an external system. The user could select the data by executing the following:

```
SELECT * FROM vExportShiftProd
```

The SQL statement above will select all Production History data that is stored online. To limit this data to a single day, you could execute the following:

```
SELECT * FROM vExportShiftProd
WHERE ShiftSeq >= 200108180 AND
      ShiftSeq <= 200108189
```

In the example above, you'll notice that the ShiftSeq (Shift Sequence Number) 200108180 is used as a starting point and 200108189 is used as an ending point. The first 8 digits of the ShiftSeq represent the date in YYYYMMDD format. In this particular example, year = 2001, month = 08, and day = 18. The last digit of the ShiftSeq represents the shift number, where 0 is the first shift number in a day and 9 is the last possible shift number in a day.

To set variables equal to today's first ShiftSeq and today's last ShiftSeq, you could execute the following:

```
DECLARE @StartingShiftSeq CHAR(9)
DECLARE @EndingShiftSeq   CHAR(9)

/* Set the StartingShiftSeq to today */
SET @StartingShiftSeq = (SELECT datepart(year,  getdate()) * 100000
                              + datepart(month, getdate()) * 1000
                              + datepart(day,   getdate()) * 10
                              + '0')

SET @EndingShiftSeq = (SELECT datepart(year,  getdate()) * 100000
                            + datepart(month, getdate()) * 1000
                            + datepart(day,   getdate()) * 10
                            + '9')
```

Again, recall that '0' is the first shift number of the day and '9' is the last shift number of the day.

However, you are probably more interested in determining yesterday's first ShiftSeq and yesterday's last ShiftSeq. To do that, execute the following:

```
SET @StartingShiftSeq = (SELECT datepart(year, dateadd(day, -1, getdate())) * 100000
                          + datepart(month, dateadd(day, -1, getdate())) * 1000
                          + datepart(day,   dateadd(day, -1, getdate())) * 10
                          + '0')
```

```
SET @EndingShiftSeq = (SELECT datepart(year,  dateadd(day, -1, getdate())) * 100000
                            + datepart(month, dateadd(day, -1, getdate())) * 1000
                            + datepart(day,   dateadd(day, -1, getdate())) * 10
                           + '9')
```

In the example above, you'll notice that we use the **`dateadd`** function to subtract 1 from today in order to determine the date for "yesterday".

Finally, we are ready to consider a simple example.

Assume the following sample stored procedure exists in the same database as ProHelp® EPM. It will extract Production History data from yesterday and copy the data to an external system (computer “MYSERVER”, database “MYDATABASE”, table “MyOtherTable”, fields “MyMachNo”, “MyJobId”, “MyShiftSeq”, and “MyProdQty”).

```
CREATE PROCEDURE SampleCustomDataExport AS
BEGIN

    /* Declare variables for the starting ShiftSeq and the ending ShiftSeq */
    DECLARE @StartingShiftSeq CHAR(9)
    DECLARE @EndingShiftSeq  CHAR(9)

    /* Determine the first ShiftSeq for "yesterday" */
    SET @StartingShiftSeq = (SELECT datepart(year,  dateadd(day, -1, getdate())) * 100000
                               + datepart(month, dateadd(day, -1, getdate())) * 1000
                               + datepart(day,   dateadd(day, -1, getdate())) * 10
                               + '0')

    /* Determine the last ShiftSeq for "yesterday" */
    SET @EndingShiftSeq = (SELECT datepart(year,  dateadd(day, -1, getdate())) * 100000
                             + datepart(month, dateadd(day, -1, getdate())) * 1000
                             + datepart(day,   dateadd(day, -1, getdate())) * 10
                             + '9')

    /* Select the Production History data from ProHelp EPM and write */
    /* it to a table in an external system */
    INSERT INTO MYSERVER.MYDATABASE.dbo.MyOtherTable(MyMachNo,
                                                     MyJobID,
                                                     MyShiftSeq,
                                                     MyProductionQty)
    SELECT MachId, JobId, ShiftSeq, CalProdQty
        FROM vExportShiftProd
        WHERE ShiftSeq >= @StartingShiftSeq AND
              ShiftSeq <= @EndingShiftSeq

END
```

In the example above, you’ll notice that the data is selected from ProHelp® EPM (the last **`SELECT`** statement) and copied to the external system (the **`INSERT`** statement) as a single command.

For your convenience, the above stored procedure actually exists in the MATTEC_PROHELP database on your ProHelp® EPM system.