# Traceability Systems Bill Of Process

## SD-1039

ISSUED December 31, 2013
REVISED

© 2013 Steering Solutions Services Corporation

All rights reserved

31DE13

**Larry Morel**
Sr. Staff Controls Engineer

Date

## TABLE OF CONTENTS

PURPOSE ..... 2
PART I ..... 2
PART II - IMPLEMENTATION ..... 5
PART III - CONSTRUCTION OF A TRACEABILITY SYSTEM ..... 6
TABLE 1 - FUNCTION CODES ..... 8
TABLE 2 - ERROR CODES ..... 9
TABLE 3 - DATABASE TABLES ..... 10
TABLE 4 - DATABASE COLUMNS ..... 11

## PURPOSE

This document defines the Nexteer Automotive Traceability Bill of Process. This document should be used to provide guidance to traceability system developers on the following topics:

- Understanding traceability systems.
- Implementing a traceability system.
- Constructing a traceability system.

## PART I - UNDERSTANDING

1. Overview

1.1. The goal of a traceability system is to capture genealogical information about each part that is shipped out of the factory. This information includes component part information, functional test data, and operation verification. This goal is accomplished by tracing the part through the manufacturing process from the time a unique identifier (UID) is assigned to the part. This UID is usually a serial number in the form of a barcode, but may also be a lot or batch code (that would apply to a group of parts). As the part completes each manufacturing step, information about the part is captured and associated with the UID. The traceability system creates a database record containing the information about that part at that step in the process. These records will be collected by enterprise systems for long term storage.

1.2. Capturing the data at each manufacturing step requires the traceability system to interface with the machinery's control system. The level of integration with the control system is dictated by the type of control system that is implemented. As a rule all test results that are to be sent to the traceability system must reside in the machine's PLC. There will be some exceptions such as PC based machines, which may either write directly to the Traceability database, or store CSV files that will be collected by the Traceability computer and stored in its database.

1.3. Electronic Error Proofing (also known as Part Permission or Part Tracking) may also be implemented as part of the Traceability System. This generally consists of confirmation at each station, that the part presented to it successfully completed the previous operation. It may also verify elements like retry counts and elapsed time. In all cases, the error proofing logic will be implemented in the primary machine controller (e.g., PLC) and not the Traceability computer, but may make use of data stored in the Traceability database.

1.4. The key output from a traceability system is a record detailing the status of a specific part when it left the manufacturing process. A secondary usage of the traceability data is for studying the manufacturing process. Deriving production counts and calculating quality statistics may also be done by analyzing traceability records.

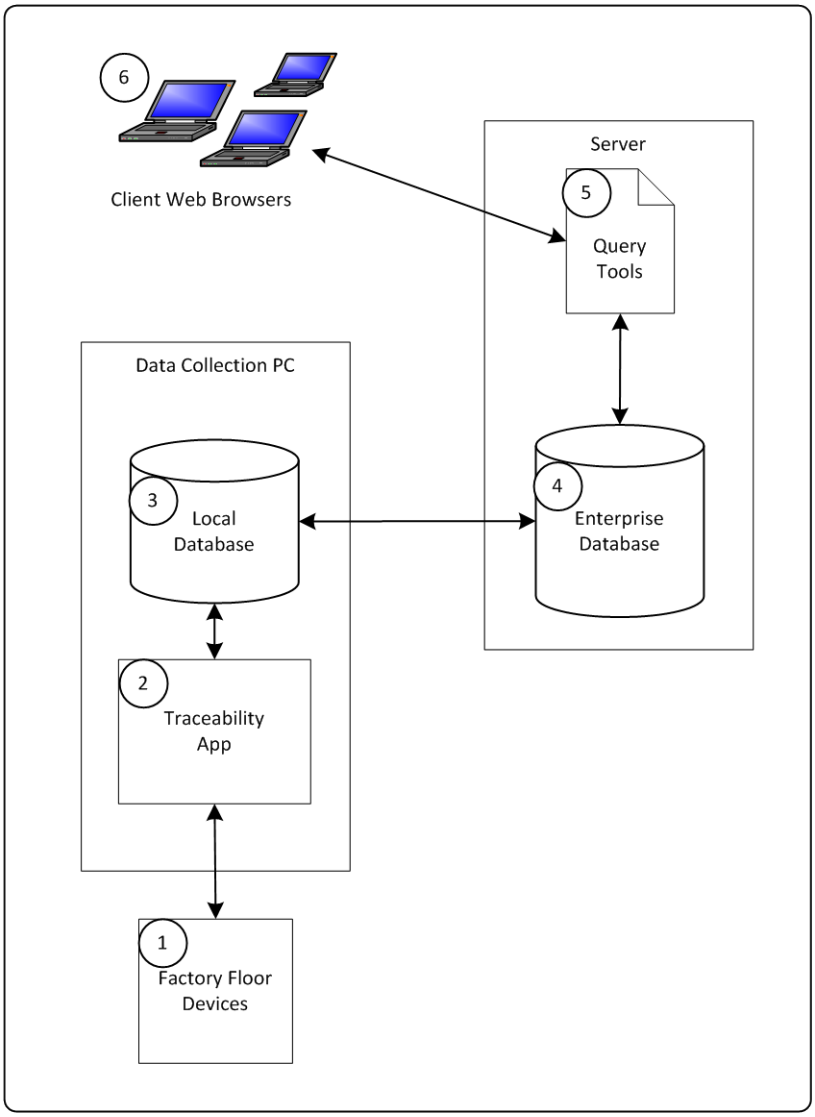


Figure 1 - Basic Traceability Structure

1.5. Traceability System Architecture – Refer to Figure 1

1.5.1. Factory Floor Devices that are part of the manufacturing process provide a Unique Identifier (UID) and the associated data to the traceability system. The Traceability computers will interface only with primary machine controllers, not with hand scanners or instrumentation. The machine controllers will be responsible for formatting and presenting data for the Traceability system.

1.5.2. The Traceability Application is generalized software that interfaces with the factory floor devices and provides an interface to a SQL Server database. The traceability application responds to requests from the machine controller to retrieve or store data from/to its database. Please see SD-1034 and SD-1035 for more detailed information about Traceability implementation in Factory Floor devices.

1.5.3. Local Databases may be local to the factory floor computer, or remote in a data center.

1.5.4. An Enterprise Database is configured to periodically collect the traceability records from each traceability system. Weekly full backups and daily incremental backups should be performed on factory floor databases to minimize the risk of lost data.

1.5.5. Query Tools like web servers and web sites, linked Excel sheets, or dedicated database lookup programs are provided to give the end-user access to the traceability records and any information that is derived from the traceability data.

1.5.6. Client Web Browsers are not covered in this document.

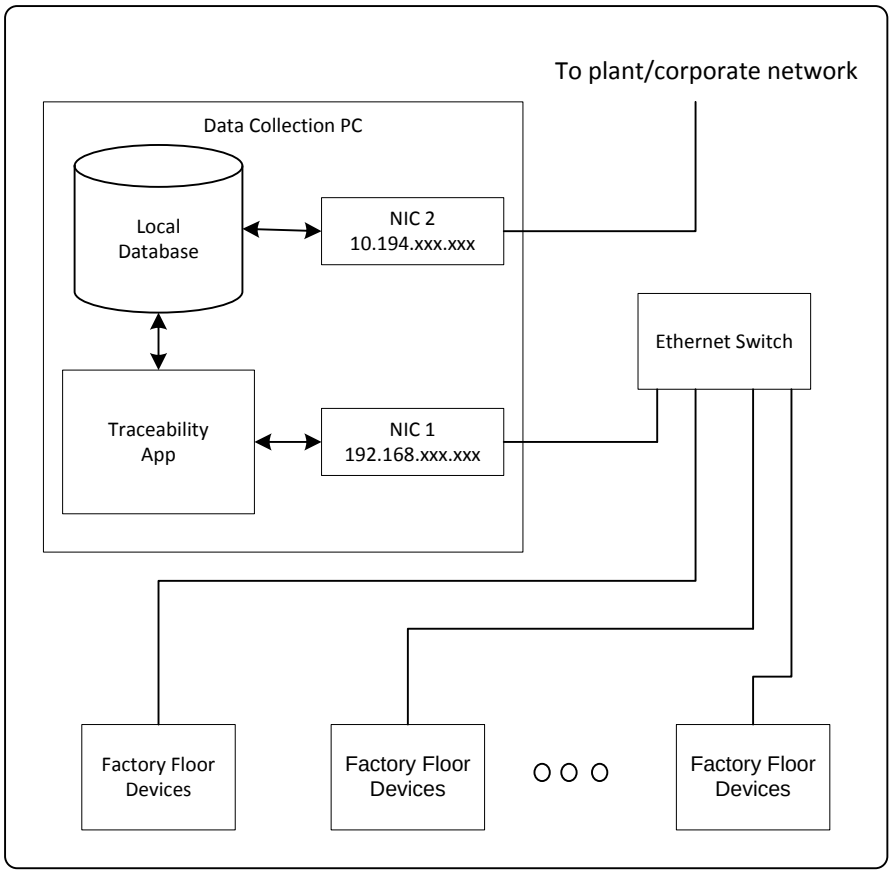


Figure 2 – Networking Within a Cell

1.6. Networking within a Cell – Refer to Figure 2.

1.6.1. Factory floor devices are connected to the computer on a private network and the computer is connected to the plant network – Refer to Figure 2

1.6.2. One Traceability Computer is typically integrated on an assembly line or cell. Multiple assembly lines or cells may be integrated using higher level applications to provide traceability for an entire program.

## PART II – IMPLEMENTATION

2. IMPLEMENTATION

2.1. What Should Go In The T-Spec?

2.1.1. Who is doing the Implementation?
- 2.1.1.1. Preferred Option: Nexteer CSE will provide pre-configured Traceability computers.
- 2.1.1.2. Secondary option: Machine builder purchases Traceability computer(s) and sends them to Nexteer CSE for configuration.
- 2.1.1.3. Third option (if driven by tariffs or regional restrictions): In-Region Nexteer CSE provides pre-configured Traceability computers.
- 2.1.1.4. Fourth Option (with prior Nexteer CSE approval only): Machine builder purchases and configures Traceability computer(s). Note that this is only feasible for highly machine builders that are highly skilled at building PCs and PC applications.

2.1.2. What should the machine builder to do?
- 2.1.2.1. Logic/functional requirements for Traceability and Electronic Error Proofing.
- 2.1.2.2. The machine builder needs to be made aware of Traceability requirements such as data storage, electronic error proofing, and genealogy.
- 2.1.2.3. The machine builder is to be responsible for PLC programming to interface the machine control application with the Nexteer Traceability PLC modules.
- 2.1.2.4. If machine is computer-controlled and not PLC controlled, machine builder will provide data in a CSV (comma separated value) format as specified by Nexteer CSE for collection by Nexteer Traceability programs.
- 2.1.2.5. Consider PLC Selection Guidelines (ref: SD1035)
- 2.1.2.6. Consider PC hardware and software requirements

2.1.3. What has to be provided in the machines?
- 2.1.3.1. Bar code scanners (hand scanner vs fixed scanner)
- 2.1.3.2. Network accessibility
- 2.1.3.3. Possible HMI requirements

2.2. MQ1 Testing Considerations
- 2.2.1. All Traceability functions should be tested and approved at MQ1.
- 2.2.2. Will we test with the actual Traceability computer or with a laptop?
- 2.2.3. Will we have to take data from one machine builder to another to complete Traceability testing or can we simulate data?

## PART III – CONSTRUCTION OF A TRACEABILITY SYSTEM

3. CONSTRUCTION

3.1. Minimum Hardware Requirements (exceptions allowed upon approval of Nexteer CSE)

3.1.1. Standard Dell Optiplex PC (desktop or mini-tower) (e.g., Optiplex 9010)

3.1.1.1. Multi-processor

3.1.1.2. 4 GB RAM memory

3.1.1.3. 2x 100+ GB Solid State hard drives in RAID-1 config (RAID-1 REQUIRED)

3.1.1.4. 2x Ethernet Ports

3.1.1.5. High efficiency power supply

3.1.1.6. 19” LCD Monitor

3.2. Managed Ethernet Switch for connections between Traceability computer and PLCs

3.2.1. MOXA EDS-516A or equivalent (note: Managed switches are necessary for >6 connections or if any multicast communications are present in the connected PLCs to provide a deterministic communications environment and avoid communications-related timing issues.)

3.2.2. Other switches are acceptable upon approval by Nexteer CSE.

3.3. Machine control when the machine will connect to the Traceability system

3.3.1. Preferred: Allen Bradley Logix 5000 series

3.3.2. Acceptable if only minimal functionality required: Allen Bradley Logix 500

3.3.3. All other types must be approved by Nexteer CSE.

3.3.3.1. See SD-1034, Traceability Interface

3.3.3.2. See SD-1035 PLC Type Traceability Considerations

3.3.4. Machine builder may be required to provide an Allen Bradley Logix 5000 Supervisory PLC to collect Traceability data, if other control system is used.

3.3.5. Computer-based controllers should create CSV files with Traceability data in Nexteer-specified format. Files will be deleted by Nexteer’s Traceability system after upload.

3.4. Software Requirements

3.4.1. Operating System & standard system requirements

3.4.1.1. Windows 7 Professional 32 bit– Current, preferred; 64 bit Windows or other operating systems require prior Nexteer CSE approval.

3.4.1.1.1. Single Windows user defined: Steering (no password)

3.4.1.1.2. Computer BIOS configured to reboot at power recovery

3.4.1.1.3. Firewall and power management disabled

3.4.1.1.4. DNS and Timeserver configuration per Nexteer regional standards.

3.4.1.1.5. Remote Desktop enabled
3.4.1.1.6. Network discovery turned off
3.4.1.1.7. .NET 4 runtime environment
3.4.1.2. SQL Server Express 2008 R2
3.4.1.2.1. Database Instance name: CITADEL
3.4.1.2.2. Main Traceability database name: Traceability
3.4.1.2.3. Maximum Server Memory setting should be set to 1024MB, rather than the default 2.1 petabytes
3.4.1.2.4. Shared Memory, TCP/IP and Named Pipes protocols enabled
3.4.1.2.5. Please contact Nexteer CSE for SQL script files to create the database, tables and triggers. Ref: Tables 3 & 4
3.4.1.3. McAfee Application Control/Solidcore Antivirus
3.4.1.4. UltraVNC installed for remote support

3.4.2. Traceability Application
3.4.2.1. Nexteer VB.Net Traceability application as supplied by Nexteer CSE
3.4.2.2. CSV File Format (if data is uploaded from a PC instead of a PLC):
3.4.2.2.1. One file per part produced
3.4.2.2.2. Each file has two rows:
3.4.2.2.3. Top row contains these field names:
3.1.1.1.1.1. Timestamp (yyyy-mm-dd HH:MM:ss)
3.1.1.1.1.2. PLCID,(SD # of process controller & ‘X’)
3.1.1.1.1.3. StationID (PLCID & station # like 01, 02, etc.)
3.1.1.1.1.4. DataFunction (14 to signify upload from CSV; Ref. Table 1)
3.1.1.1.1.5. Serial (unique identifier – typically a part bar code)
3.1.1.1.1.6. Status (9999=good; 1xxx = bad; Ref Table 2)
3.1.1.1.1.7. Model (optional part model ID)
3.1.1.1.1.8. StationName (like ”Test Cell”)
3.1.1.1.1.9. Name000, Data000 to Namexxx, Dataxxx) Process, part or station data; contents of Name fields tell what the data is; Data fields hold the values.
3.1.1.1.1.10. Ex: Name000 = “Global Pass-Fail”;
Data000 = “Pass”

SAMPLE FILE CONTENTS:

```
Timestamp,PLCID,StationID,DataFunction,Serial,Status,Model,StationName,Name000,Data000,Name001
2013-03-08 09:22:03.165,SD800293X,SD800293X01,14,HA130350011,9999,3C,Test Cell A,Global Pass-Fail,Pass,Label,A1130350011
```

| Function | Description | Trace App, SQL Trace, or SQL Plant Server | Send SQL data back to PLC? | Comments |
|---|---|---|---|---|
| 1 | Status Lookup | Trace App | Yes | Most recent entry for serial number |
| 2 | Previous Station Lookup | Trace App | Yes | Most recent Serial number record from previous station |
| 3 | Most recent Part Status + Data | Trace App | Yes | Like #1, but returns Name/Data information also |
| 10 | Normal Results info | SQL Traceability Computer | No | 1) Trace App inserts record in Status table.<br>2) SQL Trigger on Traceability computer inserts record in Summary Table. |
| 11 | Last Station in system | SQL Traceability Computer | No | 1&2 same as NORMAL RESULTS, plus:<br>3) SQL Trigger on Traceability computer INSERTs record in downstrean Traceability computer Status table.<br>If downstream INSERT fails, a RECORD COLLECTION routine in downstream Traceability computer must collect the record. |
| 12 | Marriage Data | SQL Traceability Computer & SQL Plant Server | No | 1&2 same as NORMAL RESULTS, plus:<br>3) SQL Trigger on Plant Server INSERTs record in Marriage table on Plant Server.<br>**First 5 DATA records are reserved for marriage serial numbers.** |
| 13 | Batch Info | SQL Traceability Computer & SQL Plant Server | No | 1&2 same as NORMAL RESULTS, plus:<br>3) SQL Trigger on performs a query to locate other Serial numbers with same batch number, then INSERTs record in Status table for each of these Serial numbers with the batch results info. |
| 14 | File record | SQL Traceability Computer | No | Results are from a CSV file, and are inserted by another application |
| 15 | In Process | SQL Traceability Computer | No | Same as NORMAL RESULTS (Function 10) |
| 16 | SN Batch results | SQL Traceability Computer | No | 1,2&3 same as NORMAL RESULTS, plus:<br>SQL Trigger performs a query to locate other Serial numbers with same batch number, then INSERTs record in Status table for each of these Serial numbers with the batch results info.<br>Batch Number must be in Data000 column. |
| 17 | Batch number results<br><br>(Batch number is used as the Serial Number) | SQL Traceability Computer | No | 1,2&3 same as NORMAL RESULTS, plus:<br>SQL Trigger performs a query to locate other Serial numbers with same batch number, then INSERTs record in Status table for each of these Serial numbers with the batch results info.<br>In this instance the Batch Number must be in the Serial Number column. |
| 20 | Trace App Results info | SQL Traceability Computer | No | Same as NORMAL RESULTS (Function 10) |
| 1xx | Traceability bypassed indication | | | Same as NORMAL RESULTS (Function 10). Triggers should subtract 100 from the Function Number and process as normal. |

**TABLE 1 (reference only – See SD 1034 for master version)**

| | | |
|---|---|---|
| The following are error codes echoed back from the Traceability PC based on the communication request function. | | The following status codes are stored in the database and/or displayed on the screen to indicate permission status and/or completion status |
| | | |
| **Function 1** | | 9999: Good Part Completed |
| 5000 - None or more than 1 record found | | 9000: Lookup OK (equivalent to cycle in process) |
| 5001 - Error requesting data from status table | | 1000: Machine REJECT |
| 5003 - Error inserting record into request log table | | 1001: Prev Station REJECT |
| | | 1002: Wrong Previous Station |
| **Function 2** | | 1010: Retry Count REJECT |
| 5000 - None or more than 1 record found | | 1011: Wait Time error - RETRY |
| 5001 - Error requesting data from status table | | 1012: PartOK=PartNOK ERROR (V0 system only) |
| 5003 - Error inserting record into request log table | | 5000: Permission Record Not Found |
| | | (1xxx series codes are only recommendations |
| **Function 10** | | and may be tailored to the application) |
| 5003 - Error inserting record into request log table | | |
| 5004 - Error inserting record into status table | | |
| 5010 - Parameter PLC ID is missing | | |
| 5011 - Parameter Station ID is missing | | |
| 5012 - Parameter Serial is missing | | |
| 5013 - Parameter Status is missing | | |
| 5020 - Multiple parameters are missing | | |
| | | |
| **Function ..0** | | |
| 5003 - Error inserting record into request log table | | |
| 5015 - Parameter Function not valid | | |

**TABLE 2 (reference only – See SD 1034 for master version)**

- Four Database tables; (_permission table is reserved for future use)

_requestlog: written by Traceability app; holds record of all transactions from/to Traceability app

- ID (int)
- TimeStamp (datetime)
- PLCID (varchar)
- StationID (varchar)
- DataFunction (varchar)
- Serial (varchar)
- Status (varchar)
- Station Name (varchar)
- StationLookupID (varchar)
- Overflow (varchar)

_status: written/read by Traceability app; holds all part status

- ID (Uint)
- Timestamp (datetime)
- PLCID (varchar)
- StationID (varchar)
- Status (varchar)
- Model (varchar)
- StationName (varchar)
- StationLookupID(varchar)
- Name00 (varchar)
- Data00 (varchar)
- Name01 (varchar)
- Data01 (varchar)
- …….Namex x
- …….Data xx
- Overflow (varchar)

_summary: updated by _status Trigger; pulled into server and then erased in Traceability PC

- Same as _status

_permission: written by Trigger on the _status table. Holds just the most recent record for each Serial # to provide quick lookup for status.

- ID (int)
- TimeStamp (datetime)
- PLCID (varchar)
- StationID (varchar)
- DataFunction (varchar)
- Serial (varchar)
- Status (varchar)
- Station Name (varchar)
- StationLookupID(varchar)
- Overflow (varchar)

)

**Table 3 (Four primary tables; _buffer tables,when used, are same format as _status**

| Traceability Database Table Columns | | |
|---|---|---|
| Columns | Data Type | Description |
| [ID] | | |
| Timestamp | datetime | Last time record updated |
| PLCID | varchar(100) | Capital Equipment Tag # of machine with the controlling PLC (like SD123456X) |
| StationID | varchar(100) | Unique PLC ID + station # withing that PLC (like SD123456X01, SD123456X02) |
| DataFunction | varchar(100) | See above |
| Serial | varchar(100) | Serial # of the part for which information is being stored |
| Status | varchar(100) | Returned status from the last transaction (9999 = good; 9000 = in process; everything else indicates some type of error) |
| Model | varchar(100) | Part/Model being run - part/customer specific |
| StationName | varchar(100) | Unique, descriptive name of the station - used to organize the main page of the Traceability display |
| StationLookupID | varchar(100) | Currently reserved in this table; intended to hold specific station ID for Function 3 (or similar) lookup |
| OpType | Varchar(100) | Currently reserved in this table; intended to allow lookup across a group of similar machines (as identified with the OpType field) with a single query |
| Name000 - Namexxx | varchar(1024) | each Namexxx holds a data name identifier for its companion Dataxxx; it will begin with M_ for serial number marriage data (as when a motor serial number joins an assist mechanism) or will begin with D_ for process data. |
| Data000 - Dataxxx | varchar(1024) | Dataxxx fields hold the data for the Name/Data pairs. Data is stored as character data and is specific to the operation it is associated with. |

**Table 4 (columns ID – StationName are common to all tables; _status & _summary have all columns)**

## RECORD OF REVISIONS

| Revision # | Date | Section | Description |
|---|---|---|---|
| 001 | 31DE13 | All | Original Issue |
| 002 | | | |
| 003 | | | |
| 004 | | | |
| 005 | | | |
| 006 | | | |
| 007 | | | |
| 008 | | | |
| 009 | | | |
| 010 | | | |
| 011 | | | |
| 012 | | | |
| 013 | | | |
| 014 | | | |
| 015 | | | |
| 016 | | | |
| 017 | | | |
| 018 | | | |
| 019 | | | |
| 020 | | | |