Contents

[Introduction 2](#_Toc514235390)

[System Overview 2](#_Toc514235391)

[Design Consideration 2](#_Toc514235392)

[Assumption and Dependencies 3](#_Toc514235393)

[Goals and Guidelines 3](#_Toc514235394)

[Architecture Strategies 3](#_Toc514235395)

[System Architecture 3](#_Toc514235396)

[Architecture Template 3](#_Toc514235397)

[Component View 4](#_Toc514235398)

[Team Responsibilities 5](#_Toc514235399)

[Requirement Summary 5](#_Toc514235400)

[Technical Design Specifications 5](#_Toc514235401)

[Data Flow 5](#_Toc514235402)

[System Operation Procedure 6](#_Toc514235403)

[Data Dictionary 7](#_Toc514235404)

[TDMS Database 7](#_Toc514235405)

[Config Database 7](#_Toc514235406)

[Job Database 7](#_Toc514235407)

[Data Management Policy 7](#_Toc514235408)

[Communication Protocol 8](#_Toc514235409)

[Error Handling 8](#_Toc514235410)

[eService Express 8](#_Toc514235411)

[SCM Data Collector 8](#_Toc514235412)

[Project Management 8](#_Toc514235413)

[Task Breakdown 8](#_Toc514235414)

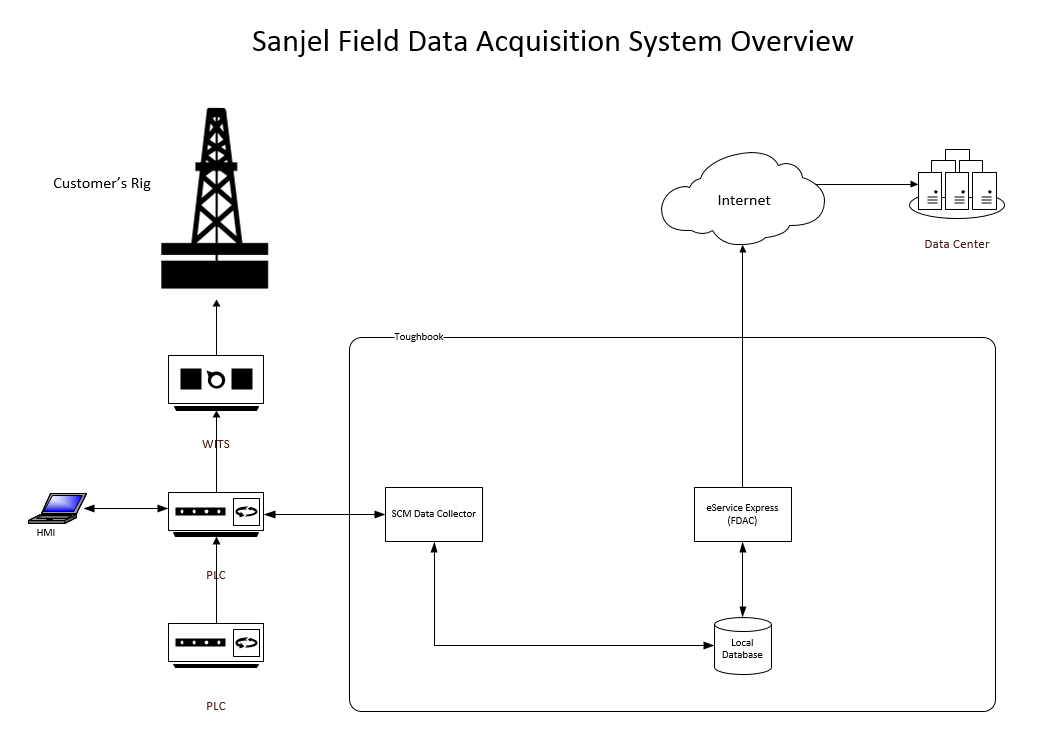
[Timeline 9](#_Toc514235415)

Sanjel Field Data Acquisition System Specification

# Introduction

Sanjel Field Data Acquisition System (aka. SFDAS) is a replacement of legacy FADS application. It not only provides the important functionalities of legacy FADS, it also evolves with current technologies and provides a better framework for future evolution.

# System Overview



# Design Consideration

* Recognize modern technology trends
* Compliance with company development strategy
* Following industry best practices

# Assumption and Dependencies

* Our goal is to improve overall field operation system, not to rebuild.
* System is built by EOS(Engineering Operation Support) and IT (Information Technology) together.
* System will be compatible with local infrastructure development roadmap.
* Toughbook system requirement:
  + Windows 7 and Windows 10 64-bit
  + .Net Framework 4.6.1 and .Net Core 2.0

# Goals and Guidelines

The major goals of SFADS are:

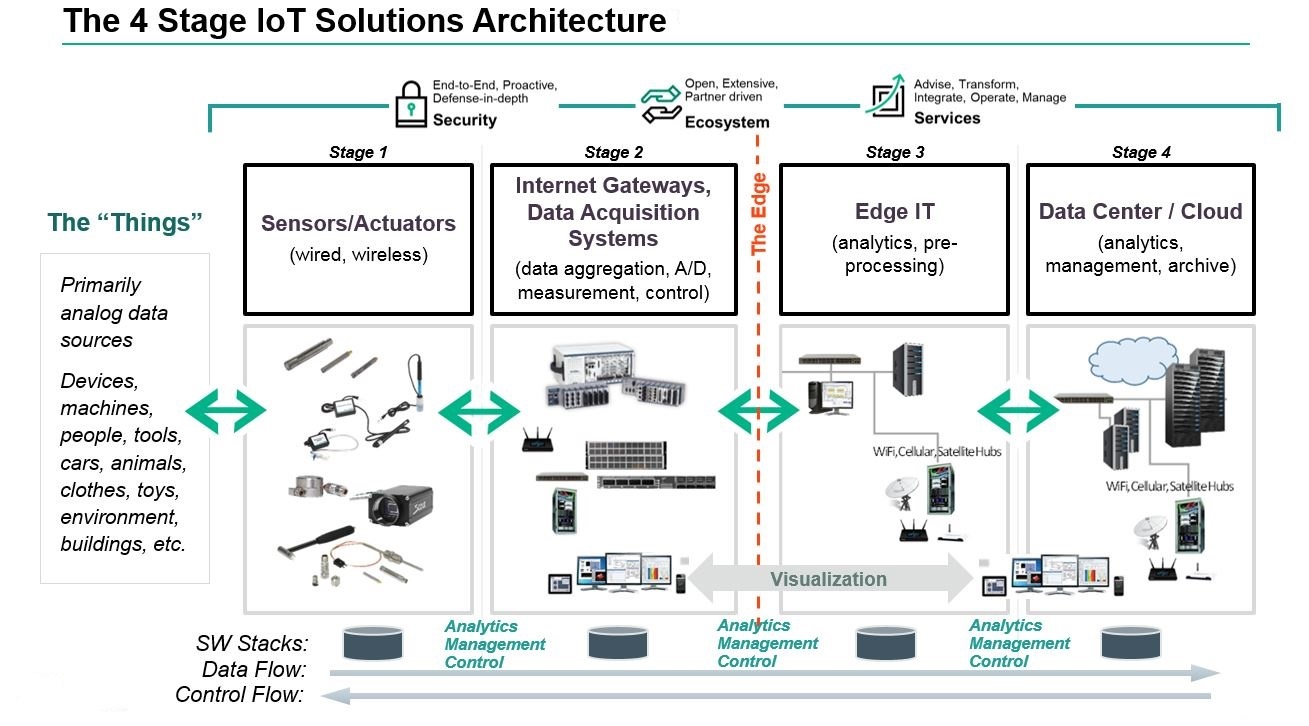
* Simple and intuitive to use.
* Maintain data integrity through the whole process.
* Maintenance free as possible.

# Architecture Strategies

* SFADS is divided to major sub-systems to separate the concerns of environmental factors.
* Subsystems are designed by following similar interfaces to avoid the re-do efforts.
* Adopt mature technologies to reduce the risks.

# System Architecture

## Architecture Template



## Component View

Field Network Infrastructure, includes truck unit based Local Area Network (LAN) , Internet gateway, WIFI hotspot, device connectivity in the scope of field network, etc.

Enterprise Network Infrastructure, includes camp/office based Local Area Network(LAN), Internet gateway, Enterprise domain controller, data center connectivity, etc.

Data Acquisition Component, includes PLC data acquisition and control, local data gateway (SCM Data Collector).

Data Processing Component, includes data aggregation and analytics, real-time charting, local reporting and operation log generation, local database management, etc. (eService Express - FDAC)

Data Transmission Component, includes real-time data streaming to data center, message queue while offline, data retrieval from data center, etc. (eService Express).

Data Analytics Component, includes archive data store, data warehouse for analytics needs.

## Team Responsibilities

General speaking, EOS Team covers stage 1 & 2, IT Team covers stage 3 & 4.

EOS Team is responsible to build Field Network Infrastructure and Data Acquisition Component.

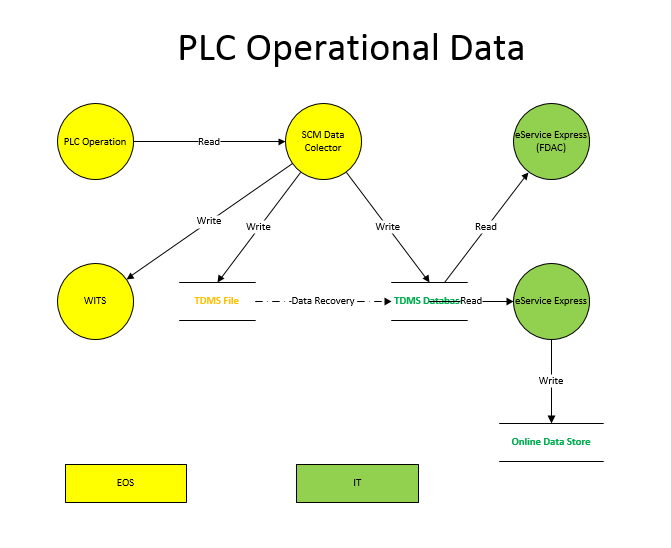
IT Team is responsible to build Enterprise Network Infrastructure, Data Processing Component, Data Transmission Component and Data Analytics Component.

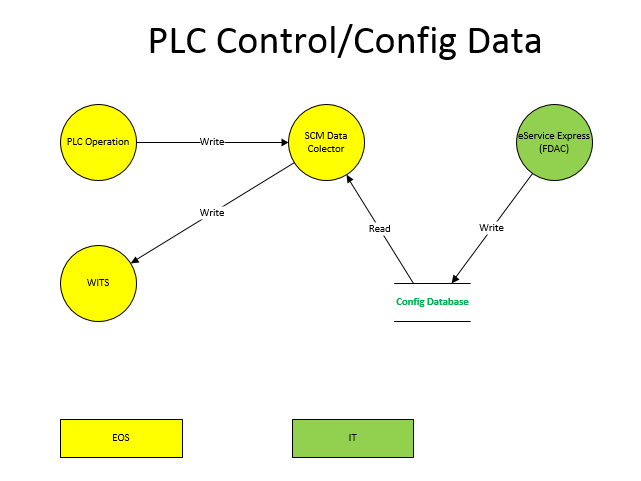
Both teams work together on the communication between components and collaborate closely.

# Requirement Summary

# Technical Design Specifications

## Data Flow





## System Operation Procedure

1. Truck Unit Operation Procedure
   1. Start
   2. ….
   3. …
2. Toughbook Operation Procedure
   1. Toughbook powers on
   2. Windows starts up
   3. eService Express starts up as a service
      1. eService Express looks up TDMS database for unsent data
         1. If yes, send to data center when internet connection is available
      2. eService Express looks up job data file for unsent data
         1. If yes, send to data center when internet connection is available
   4. SCM Data Collector starts up and keep running in background.
      1. SCM Data Collector checks TDMS database for write privilege
         1. If error, send out notification
      2. SCM Data Collector reads PLC data and write to TDMS database @ 1HZ
      3. …..
3. Field Supervisor Operation Procedure
   1. Field Supervisor opens eService Application.
   2. Field Supervisor import call sheet and create job package.
   3. Field Supervisor opens eService Express in web browser
   4. Field Supervisor enters Job Dash Board page.
      1. Field Supervisor enters configuration information of the current Job
      2. eService Express write configuration information to Config database
   5. Field Supervisor enters Job Monitor Page
      1. Job Monitor Page display the job operation charts with the data fed in TDMS database
   6. Field Supervisor completes job package.
   7. Field Supervisor sends job package to server.

## Data Dictionary

### Data Location

Production: C:\EserviceR6Data\Database

Support: C:\EserviceR6DataSupport\Database

### TDMS Database

Database Name: SCM\_PLC

Table: Data

|  |  |  |
| --- | --- | --- |
| Column Name | Date Type | Comments |
| TimeStamp | ‘TimeStamp’ TEXT | ISO 8601 |
| UnitID | TEXT | Numeric character only |
| JSON | TEXT | Serialized changed data |

### Config Database

Database Name: Config

Table: WITS\_SETTING

|  |  |  |
| --- | --- | --- |
| Column Name | Date Type | Comments |
| ID | INTEGER |  |
| TimeStamp | TEXT |  |
| JSON | TEXT | JSON format contains WITS parameters on/off |
| Version | INTEGER | Not used. |

Table: DC\_FLAGS

|  |  |  |
| --- | --- | --- |
| Column Name | Date Type | Comments |
| ID | INTEGER | PK, Unique |
| Name | TEXT | Unique |
| VALUE | TEXT |  |
| TimeStamp | TEXT | Last Modified Date Time |

Data Definition

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Value | Meaning | Comment |
| WITS\_STATE | 0 | OFF | Data Collector updates WITS connection state to client’s system at certain interval, eService Express will read the state and display on UI |
| 1 | ON |
|  |  |  |  |

Table: ESE\_FLAGS

|  |  |  |
| --- | --- | --- |
| Column Name | Date Type | Comments |
| ID | INTEGER | PK, Unique |
| Name | TEXT | Unique |
| VALUE | TEXT |  |
| TimeStamp | TEXT | Last Modified Date Time |

Data Definition

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Value | Meaning | Comment |
| SCM\_PLC\_DB\_LOCK | 0 | OFF | eSe will lock the SCM\_PLC Database while doing data cleanup, DC will buffer the data and write to database when lock is removed |
| 1 | ON |
|  |  |  |  |

### Job Database



Database Name: LocalData

Table: JOB\_TAG

|  |  |  |
| --- | --- | --- |
| Column Name | Date Type | Comments |
| ID | INTEGER |  |
| JOB\_UNIQUE\_ID | TEXT |  |
| VERSION | INTEGER |  |
| JOB\_START\_TIME | DATETIME |  |
| JOB\_END\_TIME | DATETIME |  |
| IS\_CURRENT\_JOB | INTEGER |  |
| JOB\_MONITOR\_SETTINGS | TEXT |  |
| UNIT\_SELECTION | TEXT |  |
| WITS\_SETTING | TEXT |  |
| SURFACE\_LOCATION | TEXT |  |
| STATUS | TEXT |  |
| CLIENT\_COMPANY | TEXT |  |
| JOB\_NUMBER | TEXT |  |
| JOB\_TYPE | TEXT |  |
| CLIENT\_REP | TEXT |  |
| SUPERVISOR | TEXT |  |
| WELL\_NAME | TEXT |  |
| SERVICE\_POINT | TEXT |  |
| DOWNHOLE\_LOCATION | TEXT |  |
| JOB\_DATE\_TEIME | DATETIME |  |
|  |  |  |
|  |  |  |
|  |  |  |

Table: Upload\_Log

|  |  |  |
| --- | --- | --- |
| Column Name | Date Type | Comments |
| ID | INTEGER |  |
| VERSION | INTEGER |  |
| COMPUTER\_NAME | TEXT |  |
| JOB\_NUMBER | TEXT |  |
| JOB\_UNIQUE\_ID | TEXT |  |
| START\_TIME | DATETIME |  |
| END\_TIME | DATETIME |  |
| PACKING\_TIME | DATETIME |  |
| IS\_RECEIVED\_ON\_SERVER | INTEGER | Write back after the data is transmitted |
| PACKING\_DURATION | TEXT |  |
| PACKING\_SIZE | INTEGER |  |

Database: Reference

Table: ChartProfile – The purpose for this one is to pre-define chart definition by job type – currently no requirement generated

|  |  |  |
| --- | --- | --- |
| Column Name | Date Type | Comments |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## Data Management Policy

eService Express is responsible to manage local databases.

* Local database health check
* Local database rebuild and recovery
* Local data retention
* Maintain data integrity between local copy and data center copy

SCM Data Collector will be involved in local database recovery if needed.

## Communication Protocol

eService Express communicate with SCM Data Collector through shared database. Each database will have only one writer.

## Error Handling

### eService Express

…..

### SCM Data Collector

…..

# Project Management

## Task Breakdown

1. High Level Requirement Analysis
2. Architecture Design
3. Architecture Prototyping
   1. SCM Data Collector Architecture Prototyping
   2. eService Express Architecture Prototyping
4. Edge Interface Design
5. Detail Requirement Analysis
6. Detail Design
7. Design Review and Phase Scope Decision
8. Implementation
   1. SCM Data Collector Implementation
   2. eService Express Implementation
9. Integration Test & Improvement
10. Field Test
11. Production Pilot
12. Production Deployment

## Timeline

May 1- June 1 Both team work on task 1- 7 collaboratively.

June 4 – July 13 Two teams work on task 8 individually.

July 16 – July 27 Both team work together on task 9.

July 30 – August 31 Perform Field Test and Production Pilot if possible

September – Production Deployment