

**Version 1.1**

**Date: 2019/03/08**

Abstract ***ePart consist of a significant size of source code, critical to ePart that Engineparts exclusively relies on for its business processes. As such it is mission critical and requires good governance, the how that this document intends addressing***

ePart Versioning

Overview of ePart Source Control and related management

**Table of Contents**

Document approval and distribution list 2

1. Introduction 3

2. Audience 3

3. Objectives 3

4. SVN repository 3

5. Tortoise SVN repository 4

6. Tortoise SVN ***how to*** overview 7

7. SVN specifics 7

8. Acceptance 8

# Document approval and distribution list

|  |  |  |  |
| --- | --- | --- | --- |
| **Description** | **Name / Title** | **Signature** | **Date** |
| **Document Type / purpose** | | | |
| Prepared by |  |  |  |
| Reviewed by |  |  |  |
| Approved by |  |  |  |

# Introduction

ePart consist of a significant basis volume of source code developed over many years aiming at providing well defined requirements. The source code base is critical to ePart that Engineparts exclusively relies on for its business processes and as such it is mission critical and requires good governance.

In any serious developer environment, a version of source control is critical to manage the following aspects:

* Placing all source code in a repository designed purpose designed for this purpose. For ePart the choice was SVN and has been in use for a number of years pre-dating 2007.
* The source code needs regular backing up and is the case of the SVN repository
* Track changes made and by whom in the event of a systems failure post any changes made. This will allow system administrators to rollback to a previous known working state. SVN caters for this
* A user interface (Tortoise SVN) that permits users to manage own copies of the available source code, to acquire the latest source code version, create new versions, detect changes that collide between developers and to apply conflict resolution. SVN supports much of this concept, sufficiently so in the ePart environment where there is one developer currently.

Should there be a compelling need, GitHub should be strongly considered as it performs the same functions as SVN and more.

# Audience

* Management
* ePart Admin
* Developers

# Objectives

To ensure that the ePart source code is protected as required by good corporate governance.

SVN is deployed to ensure good governance of the source code base

Ensure that backups are regularly made

Security is within the realm of good governance.

To introduce a structure that form logically part of the concept of **self**-**documenting** where the structure, the naming conventions and relationships tell a necessary story to any informed reader.

# SVN repository

The SVN repository is a normal file placed within the corporate network where it is suitably available for regular backups and access to by permitted users.

Currently the repository is located on the server known as - EPTBK02.

Associated with the SVN repository is an SVN server designed to serve requests as authorised.

The SVN repository is backed-up using the SVN server functionality and is invoked using a \*.bat file that run daily. The backup file is copied to an filesystem area designated for system wide backup services.

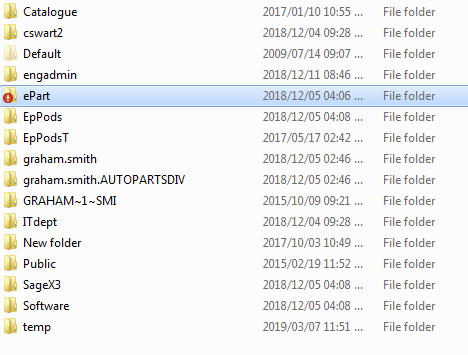
The functioning of the SVN system needs to be audited at regular intervals to ensure disaster recovery.

# Tortoise SVN repository

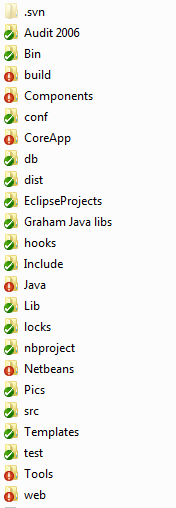
The repository structure follows a logical path mapping to the various business focus points.

The following images provide a high-level view of the descending structure of files to system relationships other than cross-domain relationships that should be inferred by reviewing the actual source code

**Image 5.1 provides a high-level view of where the SVN (*ePart*) repository starts at.**

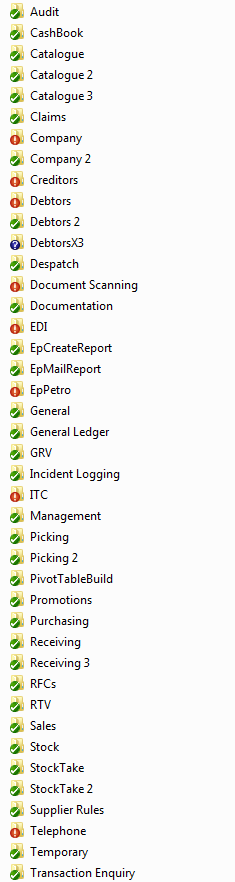


To note is that the ePart sub directory is mapped back to the server-side repository.

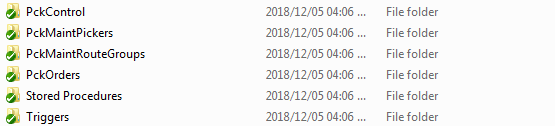
**Image 5.2 provides a high-level view of the various repository sub-sections that map to business / operational / systems centres**

To note are the entries with a red exclamation mark that indicate source code changes that have not been sent to the SVN server. The main entry of concern to the reader is the entry ***CoreApp*** which contains the main set of ALL source code files, whereas the rest are ***non-core*** development, some of which is in production others are more administrative or experimental; the last to ascertain a new developer platform when it still was relevant.

**Image 5.3 provides a high-level view of the various repository sub-sections that map to business / operational / systems centres for the Main / CoreApp of ePart**

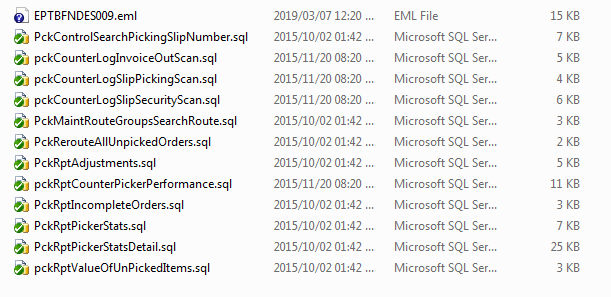


**Image 5.4 provides a high-level view of the various files that make up a sub system functionality**



To note is that all the sub-directories denote either source code for C++ builder or database Stored procedures or Triggers

**Image 5.5 provides a high-level view of the various files that make up the related et of stored procedures**



# Tortoise SVN ***how to*** overview

Tortoise SVN integrates well with Windows Explorer by adding functionality suitable for the manipulation of the SVN structure.

# SVN specifics

|  |  |  |
| --- | --- | --- |
| # | Function | Technical detail |
| 1 | SVN server | URL from internet to re-instate |
| 2 | Server installed on |  |
| 3 | SVN server running | Start-up etc |
| 4 | Daily backup BAT file | used to generate backup and to copy to backup area |
| 5 | Tortoise SVN | Desktop installation and Windows explorer integration |
| 6 | Tortoise SVN user guide | Internet URL |

# Acceptance

I hereby confirm that I have been fully informed of the documents content and, received training to understand how the detailed instructions are to be applied:

Name …………………………………………………………………………….

Job Title ………………………………………………………………………….

Signed ……………………………………………………………………………

Date ………………………………………………………………………………