Table of Contents

1 Introduction 2

1.1 Purpose of this Document 2

1.2 Scope of this Document 2

1.3 Context of this Issue 2

1.4 Definition of Terms 2

# Introduction

## Purpose of this Document

This document describes the detailed design of the components required for parallel data analysis of the Mantid Framework.

It is based on the design of the Mantid Framework specified in the Architectural Design Document [ADD]

It will form the basis of the development of this aspect of the framework and act as a guide for maintaining the system.

## Scope of this Document

These requirements cover the development of the parallel/batch data analysis aspect of the Mantid Framework and MantidPlot.

## Context of this Issue

This is the first draft of the LED-DDD derived from the ADD and after internal review will be updated and used as a basis for the development of the system.

## Definition of Terms

|  |  |
| --- | --- |
| ADD | The Architectural Design Document (this document), the high level design document for the entire system. |
| URD | The User Requirements Document, records the users’ requirements for the system. |
| SRD | The Software Requirements Document specifies the behaviour of the software system. |
| API | Application Programming Interface defines the interface through which two programs may interact. |

# Detailed Design

## Context

While live data analysis of the neutron data stream will provide users with much needed timely feedback and the ability to make informed decisions on steering subsequent experiments, a number of challenges must be addressed for the Mantid software to carry out analysis of datasets in the 10-100’s of gigabytes or more. These datasets are currently processed in Mantid using big-memory analysis nodes (128-256 GB RAM) on which Mantid currently provides node-level parallelism through the use of OpenMP. As dataset sizes continue to increase, the memory requirements alone will require parallel analysis techniques. In addition to a large memory footprint, many analysis workloads currently require staging datasets from one file system to another, prior to parallel data analysis, in such an environment, the time for this staging will dominate the analysis workflow. At the SNS there are plans to implement a parallel file system to help alleviate this bottleneck. Analysis tasks that are heavily compute or memory footprint bound will be accelerated by developing parallel analysis capabilities within Mantid. In this architecture, Mantid analysis tasks can be run concurrently on an analysis cluster and the local experimental workstation with summary analysis results displayed on the experimenter’s workstation.

## Overview

DAS -> Writes NeXus ->

Once the NeXus raw data file has been written (and is accessible to be read) an automated

## Automated Processing

Pete / Shelly…

## Configuration of Processing Task

Pete / Shelly…

## Splitting up Processing Task

Pete

## Parallel Extensions to Mantid

Russell / Sarp

## Mantid Task Submission

Ross

## Parallel IO

In order to make most efficient use of a parallel file system we may need to implement addition techniques and methodology for some IO tasks within Mantid. Until we have access to a Parallel File System, it is impossible to determine what these changes are and if indeed they are necessary.

This will be revisited once we have done sufficient testing and parameterisation using a parallel file system.