

Blacknest Data System (BDS)

Real-time import changeover

| | |
|------------------|-----------------------------|
| Project | Blacknest |
| Date | 2021-03-25 |
| Reference | blacknest/bdsImportRealtime |
| Author | Dr Terry Barnaby |

Table of Contents

| | |
|----------------------|---|
| 1. Introduction..... | 1 |
| 2. Overview..... | 1 |
| 3. Work..... | 1 |
| 4. Notes..... | 2 |

1. Introduction

This document provides ideas on the BDS changeover from the old SCREAM and CD1.x file importers to the new realtime importers.

2. Overview

The requirement is to change to use the new stream importers without losing data and with the minimum of manual intervention work.

The main issues with this is that there will be duplicate data during the switch over and how to handle this.

The bdsImportScream and bdsImportCd programs accept data from files about 1 day after the live streams.

The bdsImportStreamGcf and bdsImportStreamCd accept data live but will backfill as much data as is in the upstream network servers by default.

3. Work

1. Choose a time for the switch over for each of the SCREAM and CD1.X systems, probably just after the hour at midday on a certain date although just before midnight might be best as that is the start time for the real-time files. Would be worth doing one service at a time.
2. Set up the configuration files for bdsImportStreamCd and bdsImportStreamGcf to import the needed streams based on the test BdsServer's files.
3. At the switch over time stop the respective old importer running. Either "systemctl stop bdsImportScream" or "systemctl stop bdsImportCd". Run the bdsImportStreamGcf or bdsImportStreamCd daemons, as appropriate, manually with the "-f -c -d 0x03" flags. This will start them such that they import data from the time they started, ie they won't backfill, run in the foreground and produce debug messages. Leave them running for say 30 minutes to make sure they are working and are importing data. You should see BDS data files being created in the BDS data store and the debug messages from the commands should reveal no errors.
4. Kill the bdsImportStreamGcf or bdsImportStreamCd, as appropriate, daemon (CTRL-C) and

start properly with either: “systemctl enable bdsImportStreamGcf” or “systemctl enable bdsImportStreamCd” followed by either: “systemctl start bdsImportStreamGcf” or “systemctl start bdsImportStreamCd”.

5. Restart the old file based importer to fill the data up to when the new importers were started. either: “systemctl start bdsImportScream” or “systemctl start bdsImportCd”.
6. Now both old and new systems will be importing. The old file based streamers will still be one day behind.
7. After about two days check that the old importers have filled in the data up to when the realtime importers started and stop these processes with either: “systemctl stop bdsImportScream” or “systemctl stop bdsImportCd” and then either: “systemctl disable bdsImportScream” or “systemctl disable bdsImportCd”.
8. There will be some duplicate data which the BDS should handle (data may come from either of the two sets of BDS files present). This duplicate data (the old hourly files) can be deleted if wanted.
9. There may be some data missing if stations were offline and/or backfilling. This could be manually imported.

4. Notes

1. I believe the BDS system will allow both imports to continue as the data file start times for the old file based importers won't match the new streamers file start times or if they do the data is already present so the old file importers can fail safely at this point.
2. The BDS data availability system or dataFiles/dataChannels system can be used to determine which data is present or in processes of being imported. Note that if the stream importers are started at midday the system will report data is present over the whole day until some time after midnight on that day when the data files start times will be adjusted in the database when the day file is closed.