

Documentation and maintenance log for polishscores.org and humdrum.nifc.pl

Craig Stuart Sapp

12 July 2022

revised 2 February 2023

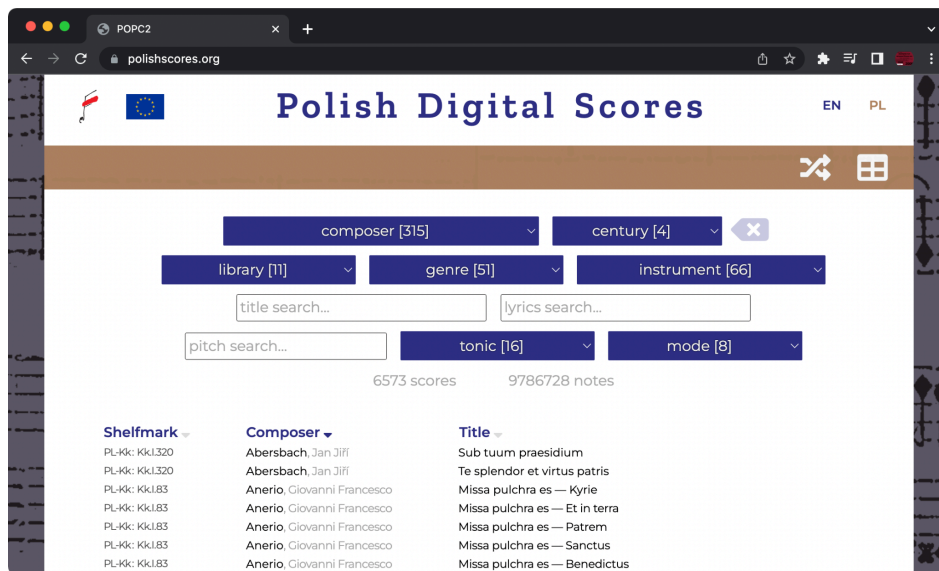
<https://bit.ly/polishscores-maintenance>

This document describes the installation and maintenance for two websites related to the POPC2 project. The two websites are:

<https://polishscores.org> (and alias <https://www.polishscores.org>)

<https://humdrum.nifc.pl>

Polishscores.org is the front-end website for the POPC2 digital scores:



while humdrum.nifc.pl is used to serve Humdrum digital scores and their derivatives for polishscores.org (and is also set up to serve POPC1 digital scores as well in the future), as well as the search and browse indexes for polishscores.org.

On the POPC2 server, all primary resources are stored in [/home/nifc](#), and each resource originates from a git repository:

Directory name in /home/nifc	Description
website-polish-scores	Files for polishscores.org website
nifc-digital-score-server	Website files for humdrum.nifc.pl
humdrum-chopin-first-editions	Digital scores for POPC1
humdrum-polish-scores	Digital scores for POPC2
humlib	Software for processing Humdrum scores
humdrum-tools	Software for processing Humdrum scores
verovio	Software for converting Humdrum scores into SVG images

Here are the online repositories for each resource in [/home/nifc](#):

Directory name	Online repository
website-polish-scores	https://github.com/craigsapp/website-polish-scores
nifc-digital-score-server	https://github.com/craigsapp/nifc-digital-score-server
humdrum-chopin-first-editions	https://github.com/pl-wnifc/humdrum-chopin-first-editions
humdrum-polish-scores	https://github.com/pl-wnifc/humdrum-polish-scores
humlib	https://github.com/craigsapp/humlib
humdrum-tools	https://github.com/humdrum-tools/humdrum-tools
verovio	https://github.com/rism-digital/verovio

Updating polishscores.org (updating website)

Git repository: <https://github.com/craigsapp/website-polish-scores>.

Polishscores.org files are stored in `/home/nifc/website-polish-scores`. It is a github repository, and updates can be downloaded with the commands:

```
cd /home/nifc  
git pull
```

After downloading the updates, the website files need to be compiled to `/home/nifc/website-polish-scores/_site`. Do this with the command within the base directory of the repository:

```
./compile
```

After compiling the website, permissions may need to be set for SELinux to allow the website to access them:

```
chcon -R -t httpd_sys_content_t  
/home/nifc/website-polish-scores_site
```

If you make local changes to the website, then they will have to be committed to the local repository and then pushed back onto Github:

```
git commit -am "Short description of change(s)."  
git push
```

Downloading updates to the website from Github may be automated in the future with a cron job.

Publishing POPC2 digital scores on Github

Digital scores used on polishscores.org are published in the repository:

<https://github.com/pl-wnifc/humdrum-polish-scores>

These files can be downloaded to your local computer with the command:

```
git clone git@github.com:pl-wnifc/humdrum-polish-scores
```

or to download a copy read only:

```
git clone https://github.com/pl-wnifc/humdrum-polish-scores
```

These files originate from the production repository for the POPC2 project:

<https://git.nifc.pl/transkrypcje/popc2-transkrypcje>

and therefore, you should not edit any of these files directory, but rather in the production repository.

Once the files on the POPC2 repository have been prepared/updated, they are transferred to the humdrum-polish-scores repository by typing the command

`make update`

in the base directory of humdrum-polish-scores. The two repositories must be in the same directory, and the popc2-transkrypcje repository must be called "production-polish-scores".

Here are make targets for humdrum-polish-scores:

<code>git pull</code>	Download the latest version of the repository.								
<code>make update</code>	Update files from popc2-transkrypcje repository.								
<code>make count-notes</code>	Counts the number of notes for each library. This is a good basic test that the scores are valid (and the note counts should increase rather than decrease). Here is an example count of notes by library: <table><tr><td>3337504</td><td>pl-cz</td></tr><tr><td>175343</td><td>pl-gd</td></tr><tr><td>73359</td><td>pl-kc</td></tr><tr><td>80110</td><td>pl-kj</td></tr></table>	3337504	pl-cz	175343	pl-gd	73359	pl-kc	80110	pl-kj
3337504	pl-cz								
175343	pl-gd								
73359	pl-kc								
80110	pl-kj								

	1320292	pl-kk
	84620	pl-kozmzk
	1576517	pl-sa
	379091	pl-stab
	742223	pl-wn
	40248	pl-wnifc
	2232117	pl-wtm
	85576	pl-wumfc
	10127000	SUM
make readme	This is done at the same time as "make update", so optional. Update the readme. It is useful to do "git diff README.md" to check for changes in notecounts (which should increase rather than decrease).	
git status	List files that have been changed, deleted or need to be added.	
git add pl-gb	If there are any files to be added from the "git status" command, then use "git add" to add them. The best way is to give the library name that contains any files that need to be added.	
git commit -am "description"	Save changes to the local repository, providing a description of what has changed in the quotation marks.	
git push	Upload the changes to the repository to github.	

Preparing POPC2 scores in popc2-transkrypcje repo

There are four subdirectories for preparing Humdrum digital scores in the popc2-transkrypcje repository:

<https://git.nifc.pl/transkrypcje/popc2-transkrypcje>

k rn-diplomatic	This is the directory when the initial encodings for the Humdrum digital scores are stored. "Diplomatic" means an exact representation of the original scanned edition for the music. The "out" subdirectory contains the final scores in the diplomatic format.
k rn-iiif	This is a side project for adding IIIF bounding boxes in the digital scores to the musical content in the scans. See https://iiif.humdrum.org . Files in k rn-iiif/out are to be moved to k rn-diplomatic/out or k rn-final/outf after IIIF information has been added to the scores.
k rn-modern	This is a side project for creating modern scores from the diplomatic ones. The files in k rn-modern/out are to be moved to to k rn-final/out
k rn-final	This is the storage location of files after they have been processed in k rn-diplomatic and k rn-modern, so files in this directory should have diplomatic encodings and have instructions for creating modern editions from the diplomatic ones.

Once files have been placed into krn-diplomatic/out, they are suitable for publishing on <https://github.com/pl-wnifc/humdrum-polish-scores>. In addition, files in krn-final/out are also ready for publication (after modernization instructions are added to the scores). Here is a list of makefile targets that can be run in the base directory of the popc2-transkrypce repository to prepare scores for publication, in the approximate order that they should be done.

<code>git pull</code>	Get the most recent files from the https://git.nifc.pl/transkrypce/popc2-transkrypce repository.
<code>make final</code>	This action moves any files in krn-iiif/out to either krn-diplomatic/out or krn-final/out, then it moves any files in krn-modern/out to krn-final/out.
<code>make cpH > /tmp/report</code>	This action checks the metadata for scores in krn-diplomatic/out and krn-final/out. This action is used to identify scores in krn-diplomatic/out that need to be adjusted from their initial submission states, and (2) check for basic problems in the scores that need fixing. The /tmp/report file should be reviewed to see the list of problems and the script used to update the metadata fields. "cpH" means "check problems, except Humdrum formatting".
<code>chmod 0755 /tmp/report</code> <code>/tmp/report</code>	If there are any metadata updating commands to run, then these commands will run those commands (using bin/adjustReferenceRecords script).
<code>make prepare</code>	This command will update the

	metadata fields from the spreadsheet https://bit.ly/popc2-metadata The contents of this spreadsheet are read from JSON or ATON files created by a Google Apps Script containing the contents of the spreadsheet, as explained on the first workbook of the spreadsheet. This script will also fix common problems with the digital scores, such as missing newlines at the ends of the files.
<code>make iiif</code>	The files in <code>krn-iiif/to-be-done</code> are copies of files in <code>krn-diplomatic/out</code> and <code>krn-final/out</code>. This make-target will update these files so that any changes after the initial copying to <code>krn-iiif</code> are not lost when these files are returned to their main directories.
<code>git status</code>	List files that have been changed or deleted (usually digital score files are not added, but scripts or other support files may need to be added).
<code>git diff</code>	It is useful to run "git diff" to review changes that have been made to the files when updating the reference records.
<code>git commit -am "description"</code>	Save changes to the local repository, providing a description of what has changed in the quotation marks.
<code>git push</code>	Store the updated files back on the <code>popc2-transkrypcje</code>. You may have to first repeat "git pull" if changes to the repository have been made while you were preparing the digital scores.

Updating humdrum.nifc.pl (publishing new scores)

Git repository: <https://github.com/craigsapp/nifc-digital-score-server>

Humdrum.nifc.pl files are stored in `/home/nifc/nifc/nifc-digital-score-server`. First pull the most recent contents of the humdrum-chopin-first-editions and humdrum-polish-scores repositories:

```
cd /home/nifc/humdrum-chopin-first-editions && git pull
```

```
cd /home/nifc/humdrum-polish-scores && git pull
```

Then run the updating command:

```
cd /home/nifc/nifc-digital-score-server
```

```
make un
```

The file `/home/nifc/nifc-digital-score-server/nohup.out` contains a log of the updating procedure (and any errors that may have happened).

This process will probably be automated in the future with a cron job script.

Updating digital score server files

Occasionally it may be necessary to update files for running the digital score server (not updating the digital scores hosted by the score server) by doing these commands:

```
cd /home/nifc/nifc-digital-score-server
```

```
git pull
```

If you make local changes to the server files, then they should be pushed back to Github with these commands:

```
git commit -am "Reason for changes."
```

```
git push
```

Updating support software

It may occasionally be necessary or otherwise a good idea to update support software for humdrum.nifc.pl. This section describes how to update this software.

Updating humlib

```
cd /home/nifc/humlib  
make update  
make  
make install
```

Updating humdrum-tools

```
cd /home/nifc/humdrum-tools  
make update  
make
```

Updating verovio

```
cd /home/nifc/verovio/tools  
git pull  
make  
make install
```

If there are problems updating verovio (such as when large changes have been made to verovio), the makefile may need to be regenerated first:

```
cd /home/nifc/verovio/tools  
git pull  
./configure  
make  
make install
```

Installing polishscores.org

Git repository: <https://github.com/craigsapp/website-polish-scores>.

First, jekyll needs to be installed (see maintenance section below for instructions for installation).

[20220712] Download humdrum-polish-scores repository

Download polishscores.org digital files from
<https://github.com/pl-wnifc/humdrum-polish-scores>:

```
cd /home/nifc
```

```
git clone git@github.com:pl-wnifc/humdrum-polish-scores.git
```

These files are the published Humdrum files for POPC2 (not the POPC2 working repository on gitea).

[20220712] Download polishscores.org website files

Download the Jekyll files for polishscores.org:

```
cd /home/nifc
```

```
git clone git@github.com:craigsapp/website-polish-scores.git
```

Then install necessary jekyll packages:

```
gem install bundler
```

```
gem install bundler
```

(needed to run twice for some reason, maybe for cross-dependencies).

To compile the website:

```
./compile
```

This will compile the website to `/home/nifc/website-polish-scores/_site`.

Installing humdrum-chopin-scores

Git repository: <https://github.com/pl-wnifc/humdrum-chopin-first-editions>

Chopin scores (POPC1) are not strictly needed, but humdrum.nifc.pl expects it to be installed (and humdrum.nifc.pl can be used to serve POPC1 files as well as POPC2 files).

```
cd /home/nifc
```

```
git clone git@github.com:pl-wnifc/humdrum-chopin-first-editions.git
```

Installing humdrum.nifc.pl

[20220712] Download humdrum.nifc.pl server website

Repository: <https://github.com/craigsapp/nifc-digital-score-server>

```
cd /home/nifc
```

```
git clone git@github.com:craigsapp/nifc-digital-score-server.git
```

Install support software for creating derivative files on humdrum.nifc.pl

This section describes the installation process for software that is needed to prepare data for humdrum.nifc.pl. These programs typically only need to be installed once, although occasionally they should be updated (see updating humdrum.nifc.pl section above for maintenance).

[20220712] Install humlib

Note gcc/g++/make (development tools need to be installed first, see further below in the "computer software maintenance log").

```
cd /home/nifc
git clone git@github.com:craigsapp/humlib.git
cd humlib
make
make install
```

(compiled programs copied to `/usr/local/bin` by the last command)

[20220712] Install Humdrum Tools and humextra

```
cd /home/nifc
git clone git@github.com:humdrum-tools/humdrum-tools.git
cd humdrum-tools
make update
make
```

Then copy the following contents to `/etc/profile.d/humdrum.sh`:

```
# Install humdrum-tools into the command path
PATH=/home/nifc/humdrum-tools/humextra/bin:$PATH
PATH=/home/nifc/humdrum-tools/humdrum/bin:$PATH
```

This will install humdrum-tools for all users when using the bash shell (default shell on polishscores.org).

[20220712] Install ImageMagick

ImageMagick is a set of command-line image editing and processing tools (which are probably needed for working with keyscape images).

```
dnf install ImageMagick
```

Check if imagemagick is installed by typing:

```
which convert  
/usr/bin/convert
```

[20220712] Install npm/aton2json

aton2json is needed to convert between ATON and JSON files. This program is written in Javascript and runs via node.

```
sudo dnf install npm -y  
sudo npm install -g aton  
sudo npm install -g posix-argv-parser  
sudo cp /usr/local/lib/node_modules/aton/example/cli/* /usr/local/bin
```

Place the following contents into /etc/profile.d/aton.sh:

```
export NODE_PATH=/usr/local/lib/node_modules
```

[20220712] Install verovio

```
git clone git@github.com:rism-digital/verovio.git  
cd verovio/tools  
git checkout develop-humdrum  
./configure  
make  
make install
```

To check if all support software is available, type:

```
make check-programs
```

in the base directory of the Humdrum data server repository. If all software is installed, you should see this text returned by the program:

aton2json	INSTALLED:	/usr/local/bin/aton2json
chord	INSTALLED:	/usr/local/bin/chord
convert	INSTALLED:	/opt/homebrew/bin/convert
extractx	INSTALLED:	/usr/local/bin/extractx
gzip	INSTALLED:	/usr/bin/gzip
hum2mid	INSTALLED:	/Users/craig/git-cloud/humdrum-tools/humextra/bin/hum2mid
humcat	INSTALLED:	/Users/craig/git-cloud/humdrum-tools/humextra/bin/humcat
humdrum2musicxml	INSTALLED:	/usr/local/bin/humdrum2musicxml
md5sum	INSTALLED:	/sbin/md5

mkeyscape	INSTALLED:	/Users/craig/git-cloud/humdrum-tools/humextra/bin/mkeyscape
notecount	INSTALLED:	/Users/craig/git-cloud/humdrum-tools/humextra/bin/notecount
perl	INSTALLED:	/usr/bin/perl
pmx2svg	INSTALLED:	/usr/local/bin/pmx2svg
prange	INSTALLED:	/Users/craig/git-cloud/humdrum-tools/humextra/bin/prange
scapeinfo	INSTALLED:	/usr/local/bin/scapeinfo
serialize	INSTALLED:	/Users/craig/git-cloud/humdrum-tools/humextra/bin/serialize
tie	INSTALLED:	/usr/local/bin/tie
tindex	INSTALLED:	/Users/craig/git-cloud/humdrum-tools/humextra/bin/tindex
verovio	INSTALLED:	/usr/local/bin/verovio
wget	INSTALLED:	/opt/homebrew/bin/wget
zcat	INSTALLED:	/usr/bin/zcat

Computer software maintenance log

This section describes software installation and maintenance for the computer OS and software that is not given in any of the above sections. Entries below are in reverse chronological order (newer actions at the top of the section).

[20240630] install jq for json pretty printing

```
sudo dnf install jq -y
```

[20240113] source code for pmx2svg script

Here is the PERL script pmx2svg which converts pmx data generated by the prange tool into an SVG image using a web service to convert via SCORE:

```
#!/usr/bin/perl

use strict;
use HTTP::Request::Common;
use LWP::UserAgent;

my $data;
my $line;
```

```

while ($line = <>) {
    $data .= "$line";
}

# Simplify known Unicode characters from Humdrum scores (popc2):
$data =~ s/ł/s/g;
$data =~ s/ſ/s/g;
$data =~ s/ſ/s/g;
$data =~ s/ſ/s/g;
$data =~ s/v/u/g;
$data =~ s/i/i/g;
$data =~ s/a/a/g;
$data =~ s/t/k/g;
$data =~ s/z/z/g;
$data =~ s/ż/z/g;
$data =~ s/ż/z/g;
$data =~ s/ż/z/g;
$data =~ s/ż/z/g;
$data =~ s/ż/z/g;
$data =~ s/æ/ae/g;

my $ua = LWP::UserAgent->new;
my $response = $ua->request(
    POST 'http://score.sapp.org/cgi-bin/score',
    [
        outputformat => 'svg',
        embedpmx      => 'yes',
        inputdata     => [$data],
    ]
);

if ($response->is_success) {
    print $response->decoded_content;
} else {
    die $response->status_line;
}

```

[20230319] file system size has returned to 15GB

Problem resolved: An older backup of the website was used (from about the end of November) when moving the virtual computer somewhere else.

```

[root@polishscores ~]$ df -h .

```

Filesystem	Size	Used	Avail	Use%	Mounted on
.					

```
/dev/mapper/fedora_fedora-root    15G    15G    20K 100% /
```

```
[root@polishscores ~]# fdisk -l
Disk /dev/sda: 100 GiB, 107374182400 bytes, 209715200 sectors
Disk model: QEMU HARDDISK
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x2095111a

Device      Boot    Start        End    Sectors  Size Id Type
/dev/sda1   *        2048      2099199    2097152    1G 83 Linux
/dev/sda2                2099200 209715199 207616000    99G 8e Linux LVM
Disk /dev/mapper/fedora_fedora-root: 15 GiB, 16106127360 bytes,
31457280 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk /dev/zram0: 7.57 GiB, 8128561152 bytes, 1984512 sectors
Units: sectors of 1 * 4096 = 4096 bytes
Sector size (logical/physical): 4096 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes
```

```
[root@polishscores ~]# pvdisplay
--- Physical volume ---
PV Name                /dev/sda2
VG Name                fedora_fedora
PV Size                <99.00 GiB / not usable 3.00 MiB
Allocatable            yes
PE Size                4.00 MiB
Total PE               25343
Free PE                21503
Allocated PE           3840
PV UUID                Gj2IFb-tcNc-TTwM-zMev-Hso0-aul3-YlmAET
```

```
[root@polishscores ~]# lvdisplay
--- Logical volume ---
```



```
LV Path                /dev/fedora_fedora/root
LV Name                root
VG Name                fedora_fedora
LV UUID                wbpHeL-UqmJ-Xlbd-rfkD-hTI4-21xM-NUwuVq
LV Write Access        read/write
LV Creation host, time fedora, 2022-07-08 07:53:11 -0400
LV Status              available
# open                 1
LV Size              15.00 GiB
Current LE             3840
Segments               1
Allocation             inherit
Read ahead sectors     auto
- currently set to    256
Block device           253:0
```

Trying to extend and resize the filesystem as on 20230202:

```
[root@polishscores ~]# lvextend -r -l 100%FREE
/dev/fedora_fedora/root
    Couldn't create temporary archive name.
```

[20230202] Increase file system size

Currently there is a single 15GB partition, with ~85GB unused on /dev/sda2:

```
[root@polishscores ~]$ lsblk
```

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINTS
sda	8:0	0	100G	0	disk	
└─sda1	8:1	0	1G	0	part	/boot
└─sda2	8:2	0	99G	0	part	
└─fedora_fedora-root	253:0	0	15G	0	lvm	/
sr0	11:0	1	2.2G	0	rom	
zram0	252:0	0	7.6G	0	disk	[SWAP]

```
[csapp@polishscores ~]$ sudo fdisk -l
[sudo] password for csapp:
Disk /dev/sda: 100 GiB, 107374182400 bytes, 209715200 sectors
Disk model: QEMU HARDDISK
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x2095111a
```

Device	Boot	Start	End	Sectors	Size	Id	Type
/dev/sda1	*	2048	2099199	2097152	1G	83	Linux
/dev/sda2		2099200	209715199	207616000	99G	8e	Linux LVM

```
Disk /dev/mapper/fedora_fedora-root: 15 GiB, 16106127360 bytes,
31457280 sectors
```

```
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

```
Disk /dev/zram0: 7.57 GiB, 8128561152 bytes, 1984512 sectors
```

```
Units: sectors of 1 * 4096 = 4096 bytes
Sector size (logical/physical): 4096 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes
```

```
[root@polishscores ~]$ pvdisplay
--- Physical volume ---
PV Name                /dev/sda2
VG Name                fedora_fedora
PV Size                <99.00 GiB / not usable 3.00 MiB
Allocatable            yes
PE Size                4.00 MiB
Total PE               25343
Free PE                21503
Allocated PE           3840
PV UUID                Gj2IFb-tcNc-TTwM-zMev-Hso0-aul3-YlmAET
```

```
[root@polishscores ~]$ lvs
  --- Logical volume ---
LV Path                /dev/fedora_fedora/root
LV Name                 root
VG Name                 fedora_fedora
LV UUID                 wbpHeL-UqmJ-Xlbd-rfkD-hTI4-2lxM-NUwuVq
LV Write Access         read/write
LV Creation host, time fedora, 2022-07-08 07:53:11 -0400
LV Status                available
# open                  1
LV Size                 15.00 GiB
Current LE              3840
Segments                1
Allocation              inherit
Read ahead sectors      auto
  - currently set to    256
Block device            253:0
```

```
[root@polishscores ~]$ lvextend -r -l 100%FREE
/dev/fedora_fedora/root
Size of logical volume fedora_fedora/root changed from 15.00
GiB (3840 extents) to <84.00 GiB (21503 extents).
Logical volume fedora_fedora/root successfully resized.
meta-data=/dev/mapper/fedora_fedora-root isize=512    agcount=4,
agsize=983040 blks
                               =                sectsz=512    attr=2,
projid32bit=1
                               =                crc=1          finobt=1,
sparse=1, rmapbt=0
                               =                reflink=1      bigtime=0
inobtcount=0
data        =                bsize=4096    blocks=3932160,
imaxpct=25
                               =                sunit=0        swidth=0 blks
```

```
naming      =version 2                bsize=4096    ascii-ci=0,
ftype=1
log          =internal log            bsize=4096    blocks=2560,
version=2
            =                        sectsz=512      sunit=0 blks,
lazy-count=1
realtime     =none                    extsz=4096     blocks=0,
rtextents=0
data blocks changed from 3932160 to 22019072
```

```
[csapp@polishscores ~]$ df -h .
Filesystem                                Size  Used Avail Use% Mounted on
/dev/mapper/fedora_fedora-root            84G   14G   70G   17% /
```

```
[csapp@polishscores ~]$ sudo lvdisplay
--- Logical volume ---
LV Path                                /dev/fedora_fedora/root
LV Name                                root
VG Name                                fedora_fedora
LV UUID                                wbpHeL-UqmJ-Xlbd-rfkD-hTI4-21xM-NUwuVq
LV Write Access                        read/write
LV Creation host, time                 fedora, 2022-07-08 07:53:11 -0400
LV Status                              available
# open                                1
LV Size                               <84.00 GiB
Current LE                             21503
Segments                               1
Allocation                             inherit
Read ahead sectors                     auto
- currently set to                     256
Block device                           253:0
```

```
[root@polishscores ~]$ pvdisplay
--- Physical volume ---
PV Name                                /dev/sda2
VG Name                                fedora_fedora
PV Size                                <99.00 GiB / not usable 3.00 MiB
Allocatable                            yes
```

PE Size	4.00 MiB
Total PE	25343
Free PE	3840
Allocated PE	21503
PV UUID	Gj2IFb-tcNc-TTwM-zMev-Hso0-aul3-YlmAET

Now the root filesystem is 84GB.

[20230118] Allow logs to be written

Allow the webserver to write to the NIFC data server access logs:

```
semanage fcontext -a -t httpd_sys_rw_content_t  
'/home/nifc/nifc-digital-score-server/logs'
```

```
restorecon -v '/home/nifc/nifc-digital-score-server/logs'
```

```
ausearch -c 'data-nifc' --raw | audit2allow -M my-datanifc
```

```
semodule -X 300 -i my-datanifc.pp
```

[20220902] Set logrotate for Apache to persist longer

Contents of `/etc/logrotate.d/httpd`:

```
/var/log/httpd/*log {  
    missingok  
    notifempty  
    sharedscripts  
    monthly  
    size 10M  
    compress  
    postrotate  
        /bin/systemctl reload httpd.service > /dev/null 2>/dev/null || true  
}
```

```
endscript  
}
```

[20220902] Setup humdrum.nifc.pl on Apache server

Create a symbolic link to the website files:

```
cd /var/www/websites
```

```
ln -s /home/nifc/nifc-digital-score-server humdrum.nifc.pl
```

Add this contents to `/etc/httpd/conf.d/vhost-humdrum.nifc.pl.conf`:

```
<VirtualHost *:80>  
    ServerName humdrum.nifc.pl  
    DocumentRoot /var/www/websites/humdrum.nifc.pl  
    CustomLog logs/humdrum-nifc.log combined  
    ErrorLog logs/humdrum-nifc-error.log  
    ServerAdmin craig@ccrma.stanford.edu  
    ScriptAlias /cgi-bin "/var/www/cgi-bin"  
    RewriteEngine On  
    RewriteRule ^/([^?]*\?(.))$ /cgi-bin/data-nifc?id=$1&$2 [NC,PT,QSA]  
    RewriteRule ^/([^?]*)$ /cgi-bin/data-nifc?id=$1 [NC,PT,QSA]  
    Header add Access-Control-Allow-Origin "*"   
    <Directory /var/www/websites/humdrum.nifc.pl>  
        AllowOverride All  
        Options Indexes FollowSymLinks  
        IndexOptions FancyIndexing ScanHTMLTitles IconHeight=10 IconWidth=11  
        SuppressDescription NameWidth=45  
        Order allow,deny  
        Allow from all  
    </Directory>  
    <Directory "/var/www/cgi-bin">  
        AllowOverride All  
        Options None  
        Order allow,deny  
        Allow from all  
    </Directory>  
</VirtualHost>
```

Restart Apache:

```
apachectl restart
```

[20220901] Allow CGI scripts in Apache with SELinux

Allow CGI scripts for Apache to run (only needed once):

```
semanage boolean -m --on httpd_enable_cgi
```

Give permission to data-nifc script to run:

```
semanage fcontext -a -t httpd_sys_script_exec_t  
/var/www/cgi-bin/data-nifc
```

Refresh permissions (for some reason):

```
restorecon /var/www/cgi-bin/data-nifc
```

Install CGI.pm PERL module:

```
dnf install "perl(CGI)"
```

Temporarily turn off SELinux:

```
setenforce 0
```

Turn back on

```
setenforce 1
```

Check the state of SELinux:

```
sestatus
```

[20220712] Install humdrum2musicxml

The converter from Humdrum to MusicXML is run in Music 21, and should be installed locally on polishscores.org. For now use the converted installed on the Stanford server via a PERL script that should be installed in `/usr/local/bin/humdrum2musicxml`:

```
#!/usr/bin/perl  
#
```

```
# Programmer:   Craig Stuart Sapp <craig@ccrma.stanford.edu>
# Creation Date: Mon 13 Sep 2021 11:06:35 AM PDT
# Last Modified: Mon 13 Sep 2021 11:06:38 AM PDT
# Program Name: humdrum2musicxml
# Syntax:       PERL 5
# Usage:        humdrum2musicxml file.krn > file.musicxml
# Usage:        cat file.krn | humdrum2musicxml > file.musicxml
#
# Description:   Converts Humdrum data into MusicXML data, using
#               conversion interface at http://data.musicxml.humdrum.org
#
# To use LWP, you may need to run this command or related:
#   perl -MCPAN -e "install Bundle::LWP;"
#   dnf install 'perl(LWP::ConsoleLogger)'
#
```

```
use strict;
use HTTP::Request::Common;
use LWP::UserAgent;
#use LWP::ConsoleLogger::Easy qw( debug_ua );

my $data;
my $line;
while ($line = <>) {
    $data .= "$line";
}

my $url = "https://data.musicxml.humdrum.org";
my $ua = LWP::UserAgent->new();
#debug_ua( $ua );
my $request = POST($url, ['inputdata', [$data]]);
my $content = $ua->request($request)->as_string();
my @lines = split(/\n/, $content);
my $emptyline = 0;
for (my $i=0; $i<@lines; $i++) {
    $line = $lines[$i];
    if (($line eq "") && !$emptyline) {
        $emptyline = 1;
        next;
    }
    next if !$emptyline;
    print "$line\n";
}
```

Also set the permissions so that anyone can run it:

```
sudo chmod 0755 /usr/local/bin/humdrum2musicxml
```

The script is also located in:

/home/nifc/nifc-digital-score-server/cache/bin/humdrum2musicxml

In addition, some PERL packages are needed to be installed to run the above script:

```
sudo dnf install 'perl(LWP::ConsoleLogger)'  
sudo dnf install perl-LWP-Protocol-https
```

[20220902] Install music21/converter21 directly:

```
pip3 install music21  
pip3 install converter21
```

Maybe add daily cron update for these commands to install latest version of converter21.

[20220712] Install cmake

Cmake is needed for compiling verovio.

```
sudo dnf install cmake -y
```

[20220712] SELinux notes

Reference:

<https://www.thegeekdiary.com/how-to-check-whether-selinux-is-enabled-or-disabled>

[20220712] Install apache web server

The web server is needed to host files for poliscres.org and humdrum.nifc.pl.

References:

<https://unixcop.com/how-to-install-lamp-stack-on-fedora-36>

```
sudo dnf install httpd
```

To start the web server:

```
sudo systemctl start httpd
```

To automatically start the server when rebooting computer:

```
sudo systemctl enable httpd
```

To check on the status of the web server:

```
sudo systemctl status httpd
```

If firewall is running, these commands are needed to allow apache to break through the firewall:

```
firewall-cmd --permanent --add-service=http  
firewall-cmd --permanent --add-service=https  
firewall-cmd --reload
```

Link to websites in `/home/nifc`:

```
mkdir -p /var/www/websites  
cd /var/www/websites  
sudo ln -s /home/nifc/website-polish-scores/_site polishscores
```

Allow files to be visible through SELinux:

```
chcon -R -t httpd_sys_content_t  
/home/nifc/website-polish-scores_site
```

https is controlled by a proxy, so no need to set up on the polishscores server.

[20220712] Install jekyll

Jekyll is needed to compile the Github repository files for the website polishscores.org and humdrum.nifc.pl.

References:

<https://jekyllrb.com/docs/installation>

<https://help.github.com/articles/setting-up-your-github-pages-site-locally-with-jekyll>

<https://bundler.io>

```
sudo dnf install ruby ruby-devel openssl-devel redhat-rpm-config  
@development-tools  
gem install bundler
```

[20220712] Install spell-checker (English)

```
dnf install aspell
which ispell
/usr/bin/ispell
```

[20220712] Install gcc, make, and other development tools

```
sudo dnf groupinstall "Development Tools" "Development Libraries"
```

Check if gcc is installed:

```
gcc -v
gcc version 12.1.1 20220507'
```

Still need to install G++ (gcc C++ compiler):

```
dnf install g++
```

Also needed to install these tools for compiling pcre (needed for compiling humextras):

```
dnf install libtool m4 automake
```

[20220712] Create .gitconfig for user

In order to commit git repositories on Github (or gitea) this configuration file is needed. Save this content (or similar for another user) in `~/.gitconfig`:

```
[user]
  name = Craig Stuart Sapp
  email = craigsapp@gmail.com
[init]
  defaultBranch = main
[pull]
  rebase = true
```

[20220712] Create ssh keys

This is required for pushing changes to repositories back onto Github (for each user).

Reference:

<https://docs.github.com/en/authentication/connecting-to-github-with-ssh/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent>

```
ssh-keygen -t ed25519 -C "csapp@polishscores.org"  
eval "$(ssh-agent -s)"
```

Then copy the contents of the file `~/.ssh/id_ed25519.pub` to paste in Github keys on the page <https://github.com/settings/keys>.

[20220712] Install git

Git is needed to download (and upload) files from repositories related to the website and support software.

```
sudo dnf -y install git
```

[20220712] Install automatic updates

Update the OS software automatically to keep it up to date (particularly with security patches). The automatic updates will not upgrade the OS version (Fedora 36 to Fedora 37, for example).

Reference:

<https://docs.fedoraproject.org/en-US/quick-docs/autoupdates>

```
sudo dnf install dnf-automatic  
systemctl enable --now dnf-automatic.timer
```

Check timers for updates:

```
systemctl list-timers dnf-*
```

Updates will be checked for about once every three hours.

[20220712] Update OS software

Update to the most recent software for the the OS (currently Fedora 36):

```
sudo dnf upgrade --refresh
```

This is the manual way of updating all software installed with the OS (the "Install automatic updates" section above will handle this automatically in the future).

[20220712] Create users

Example process for creating a new user for *csapp*:

```
adduser csapp  
passwd csapp
```

Allow csapp to be superuser:

```
usermod -aG wheel csapp
```

Allow csapp to be in the nifc group:

```
usermod -aG nifc csapp
```

This will allow shared working on files in `/home/nifc`.

Check the groups a user is in:

```
groups
```

Check if csapp can be a superuser:

```
sudo -IU csapp
```

How to remove user from wheel:

```
gpasswd -d sapp wheel
```

How to delete a user (including their files in `/home`)

```
userdel -r groupname
```

Add a fake user to store website files

Allow users in the nifc group to edit files in this fake account:

```
mkdir /home/nifc
```

```
chown -R root:nifc /home/nifc
```

Allow users in the nifc group to collaborate on files in this directory:

```
chmod 2775 /home/nifc
```

The "2" in the permissions number means "Set group ID" bit, which allows files and directories to keep the nifc group.

When setgid permission is applied to a directory, files that were created in this directory belong to the group to which the directory belongs. Any user who has write and execute permissions in the directory can create a file there. However, the file belongs to the group that owns the directory, not to the user's group ownership. Files in that directory will have the same group as the group of the parent directory.

Reference:

<https://unix.stackexchange.com/questions/52707/difference-between-chmod-775-and-chmod-2755>

[20220712] Add nifc group

A group called "nifc" was created in order to allow different users to collaborate on editing/maintaining website files and support software. All shared resources are placed into the "nifc" group.

```
groupadd nifc
```

Check if the group was added:

```
less /etc/group | grep nifc
```

How to delete a group:

```
groupdel groupname
```

[20220712] Set the hostname

Reference:

<https://docs.fedoraproject.org/en-US/quick-docs/changing-hostname>

```
hostnamectl set-hostname --static polishscores
```

[20220712] Computer setup

Operating system	Fedora 36
Processors	4
Memory	8 GB
Disk space	15 GB
CPU speed	2.4 GHz
Local IP	10.10.0.62
External Web IP	148.81.200.236

- POPC2 image transfer computer: 192.168.68.113
- Internal network IP address of the virtual computer: 10.10.0.62