

# Package: reference.fabric (via r-universe)

December 19, 2024

**Title** Hydrological Reference Fabric Tools

**Version** 1.0.0.9000

**Description** Development tools and `targets` pipeline for generating a national hydrological geospatial reference fabric.

**URL** <https://github.com/lynker-spatial/reference.fabric>

**BugReports** <https://github.com/lynker-spatial/reference.fabric/issues>

**License** GPL (>= 3)

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.3.2

**SystemRequirements** mapshaper (<https://github.com/mbloch/mapshaper/>)

**Depends** R (>= 4.1)

**Imports** archive, arrow, aws.s3, dplyr, hydrofab, nhdplusTools, rmapshaper, sbtools, sf, stats, targets, utils, yyjsonr

**Remotes** mikejohnson51/hydrofab

**Config/pak/sysreqs** cmake libgdal-dev gdal-bin libgeos-dev libglpk-dev libarchive-dev libicu-dev libpng-dev libsecret-1-dev libsodium-dev libxml2-dev libssl-dev libproj-dev libsqlite3-dev libudunits2-dev libnode-dev libx11-dev

**Repository** <https://owp-spatial.r-universe.dev>

**RemoteUrl** <https://github.com/lynker-spatial/reference.fabric>

**RemoteRef** HEAD

**RemoteSha** 0ae6428678e09814f5eb4a469f78c0c67fdb4eeb

## Contents

make_fabric . . . . .	2
rf.targets.burnline_events . . . . .	2
rf.targets.clean_catchment . . . . .	3

rf.targets.clean_flowlines . . . . .	3
rf.targets.clean_waterbodies . . . . .	4
rf.targets.rectify_catchment_borders . . . . .	5
rf.targets.reference_catchments . . . . .	5
rf.targets.reference_flowlines . . . . .	6
rf.targets.reference_waterbodies . . . . .	6

<b>Index</b>	<b>8</b>
--------------	----------

---

make_fabric	<i>Make Reference Fabric</i>
-------------	------------------------------

---

### Description

Make Reference Fabric

### Usage

make\_fabric(...)

### Arguments

... Additional arguments passed to [tar\\_make](#).

---

rf.targets.burnline_events	<i>Burnline Events Processing Target</i>
----------------------------	--

---

### Description

Processes burn line events according to step 4.3.1

### Usage

rf.targets.burnline\_events(nhd\_gdb, rf\_vaa, dir\_data)

### Arguments

nhd_gdb	Path to NHDPlus Seamless GeoDatabase
rf_vaa	NHD Value-added attributes table
dir_data	Data output directory

### Value

Path to outputted burn line events file for CONUS

---

```
rf.targets.clean_catchment
```

*Clean Catchments Target*

---

**Description**

Processes catchments according to steps 4.2.1 - 4.2.3.

**Usage**

```
rf.targets.clean_catchment(cat_info, simplify_keep = 0.2)
```

**Arguments**

cat\_info            data.frame containing path information for catchments and flowlines  
simplify\_keep      Tolerance value for weighted Visvalingam simplification

**Value**

Path to cleaned catchment output (cat\_info\$outfile).

---

```
rf.targets.clean_flowlines
```

*Clean Flowlines Target*

---

**Description**

Processes flowlines according to step 4.3.2

**Usage**

```
rf.targets.clean_flowlines(  
  flowlines_path,  
  rf_cat_paths,  
  rf_ble_path,  
  rf_vaa,  
  rf_enhd,  
  dir_cleaned  
)
```

**Arguments**

flowlines_path	Path to NHD flowlines file
rf_cat_paths	Path to <i>reference</i> catchment outputs
rf_ble_path	Path to CONUS burn line events file
rf_vaa	NHDPlus-VAA table
rf_enhd	E2NHDPlus River Attributes table
dir_cleaned	Cleaned output directory

**Value**

Path to cleaned flowlines

---

`rf.targets.clean_waterbodies`  
*Clean Waterbodies Target*

---

**Description**

Processes waterbodies according to steps 4.1.1 - 4.1.6.

**Usage**

```
rf.targets.clean_waterbodies(waterbodies_path, dir_cleaned)
```

**Arguments**

waterbodies_path	Path to unmodified NHDWaterbody file
dir_cleaned	Directory to output cleaned waterbody geometry

**Value**

Path to cleaned waterbodies

---

rf.targets.rectify\_catchment\_borders  
*Catchment VPU Rectification Target*

---

**Description**

Processes VPU rectification of *cleaned* catchments according to step 4.2.4.

**Usage**

```
rf.targets.rectify_catchment_borders(  
    cat_cleaned_paths,  
    vpu_topology,  
    dir_reference  
)
```

**Arguments**

cat\_cleaned\_paths      Paths to cleaned catchment files  
vpu\_topology      data.frame containing topology definition for VPUs  
dir\_reference      Output directory for reference features

**Value**

List of catchment output paths

**Note**

This process is highly data-dependent, so we must process it sequentially.

---

rf.targets.reference\_catchments  
*Catchments Reference Output Target*

---

**Description**

Processes remaining catchment steps according to step 4.2.5.

**Usage**

```
rf.targets.reference_catchments(rf_cat_path)
```

**Arguments**

rf\_cat\_path      Path to VPU-rectified catchment

**Value**

Path to outputted reference catchment

**FIXME**

Snap to underlying grid with size of .0009?

---

`rf.targets.reference_flowlines`  
*Reference Flowlines Target*

---

**Description**

Processes flowlines according to step 4.3.3

**Usage**

```
rf.targets.reference_flowlines(nhd, vpu, rf_enhd_comid, dir_reference)
```

**Arguments**

<code>nhd</code>	Path to <i>cleaned</i> flowlines
<code>vpu</code>	Current VPU for nhd.
<code>rf_enhd_comid</code>	COMIDs with E2NHDPlus river attributes
<code>dir_reference</code>	Reference output directory

**Value**

Path to reference flowlines

---

`rf.targets.reference_waterbodies`  
*Process Reference Waterbodies Target*

---

**Description**

Processes *cleaned* waterbodies according to steps 4.1.7 - 4.1.9.

**Usage**

```
rf.targets.reference_waterbodies(wb_clean, wb_vpu, wb_reference_dir)
```

**Arguments**

wb\_clean            Path to cleaned waterbodies file  
wb\_vpu             The corresponding VPU of wb\_clean  
wb\_reference\_dir    Directory to output reference waterbodies

**Value**

Path to reference waterbodies

**FIXME**

On/off network waterbodies based on NHD flowlines?

# Index

`make_fabric`, [2](#)

`rf.targets.burnline_events`, [2](#)

`rf.targets.clean_catchment`, [3](#)

`rf.targets.clean_flowlines`, [3](#)

`rf.targets.clean_waterbodies`, [4](#)

`rf.targets.rectify_catchment_borders`,  
[5](#)

`rf.targets.reference_catchments`, [5](#)

`rf.targets.reference_flowlines`, [6](#)

`rf.targets.reference_waterbodies`, [6](#)

`tar_make`, [2](#)