RESERVOIR ASSESSMENT TOOL (RAT)

RATATHON

Ultimate Guide to RAT 3.0

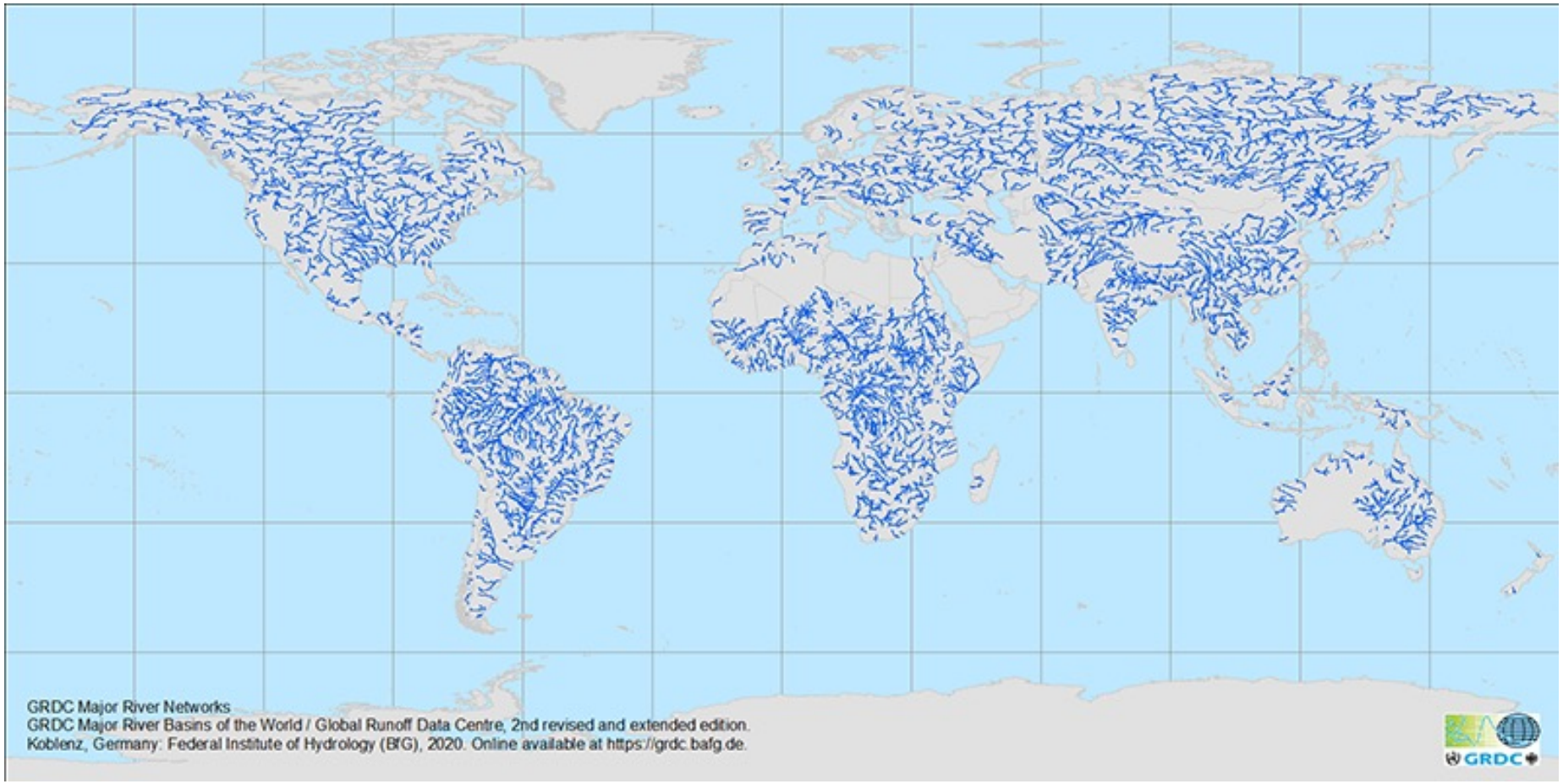
By Sanchit Minocha

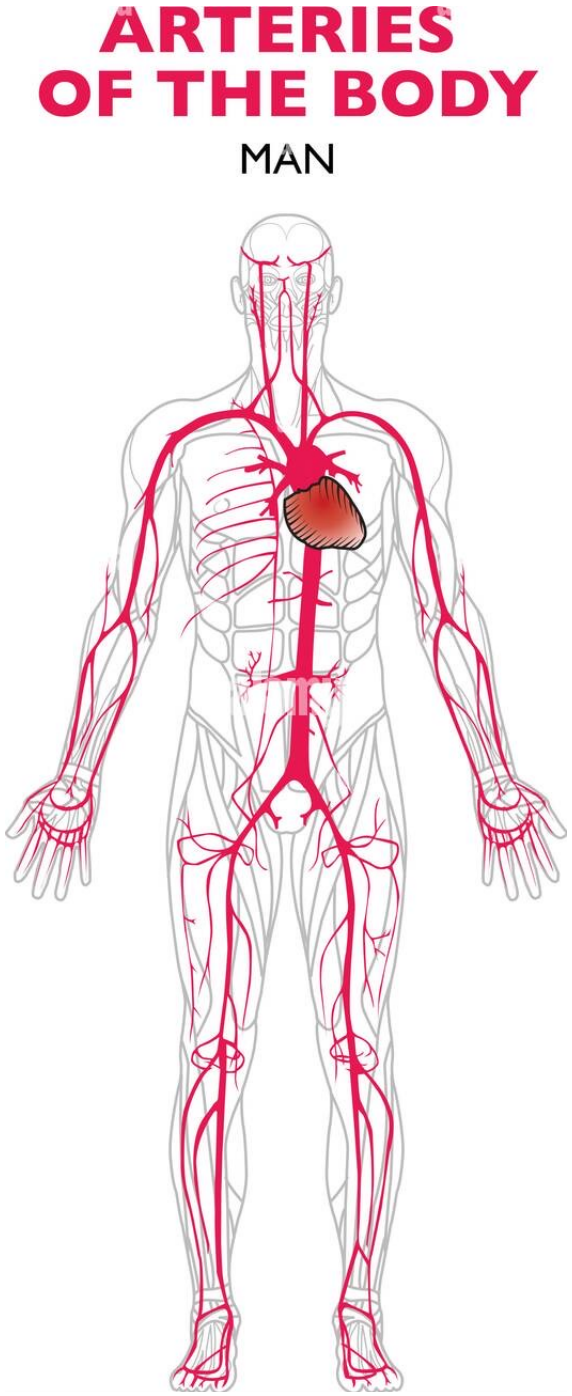
with contributions from Pritam Das,
SASWE Group, University of Washington

ANACONDA®

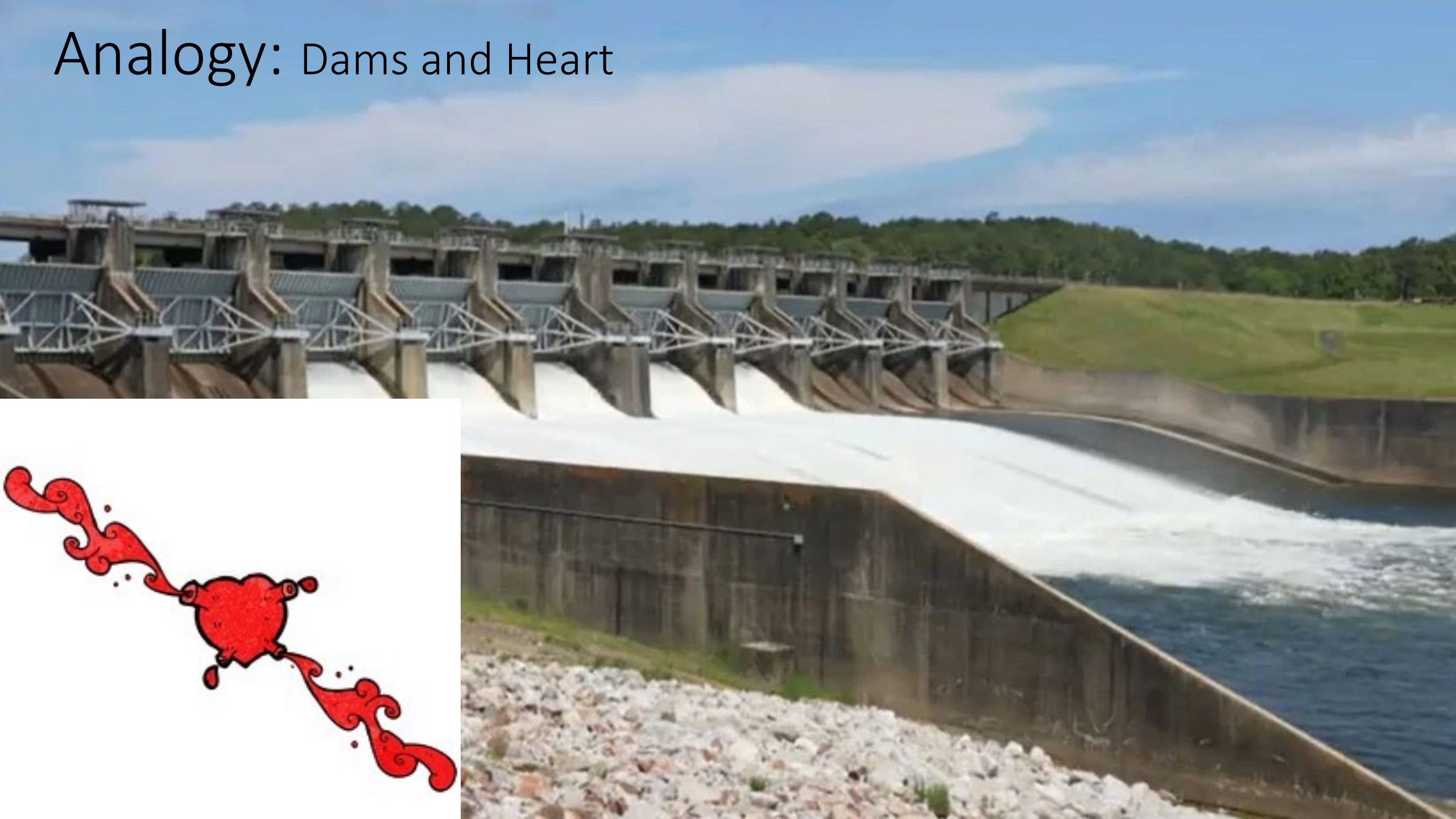
www.satellitedams.net

Analogy: Rivers and Arteries

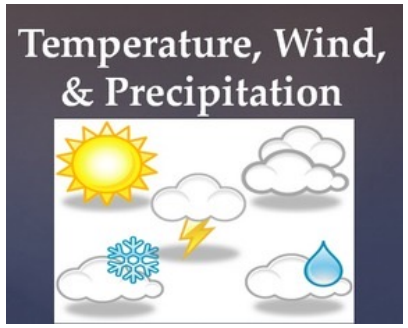




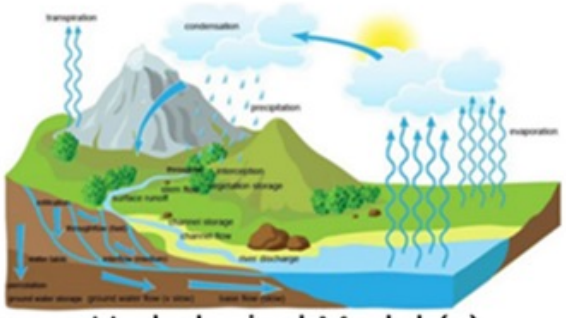
Analogy: Dams and Heart



Big Picture: How RAT Works?



Meteorological data (f)

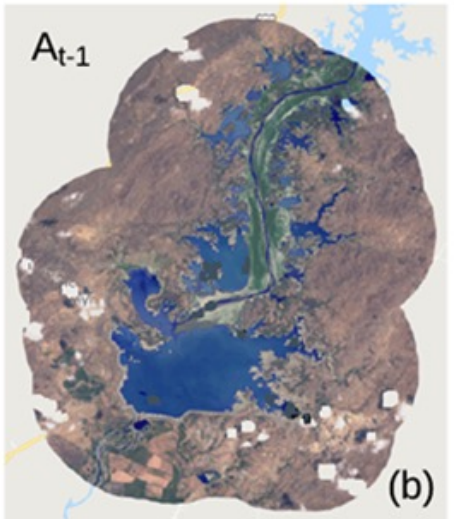


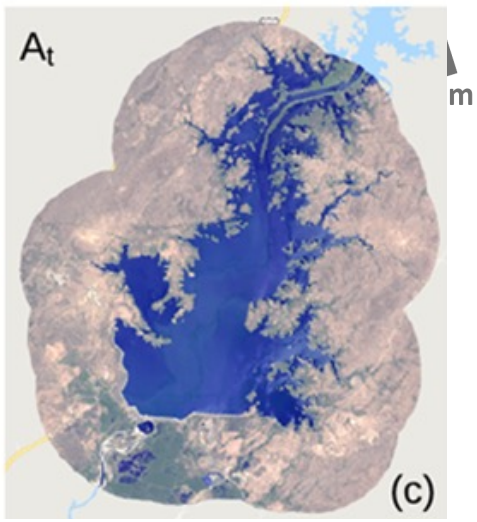
Hydrological Model (e)

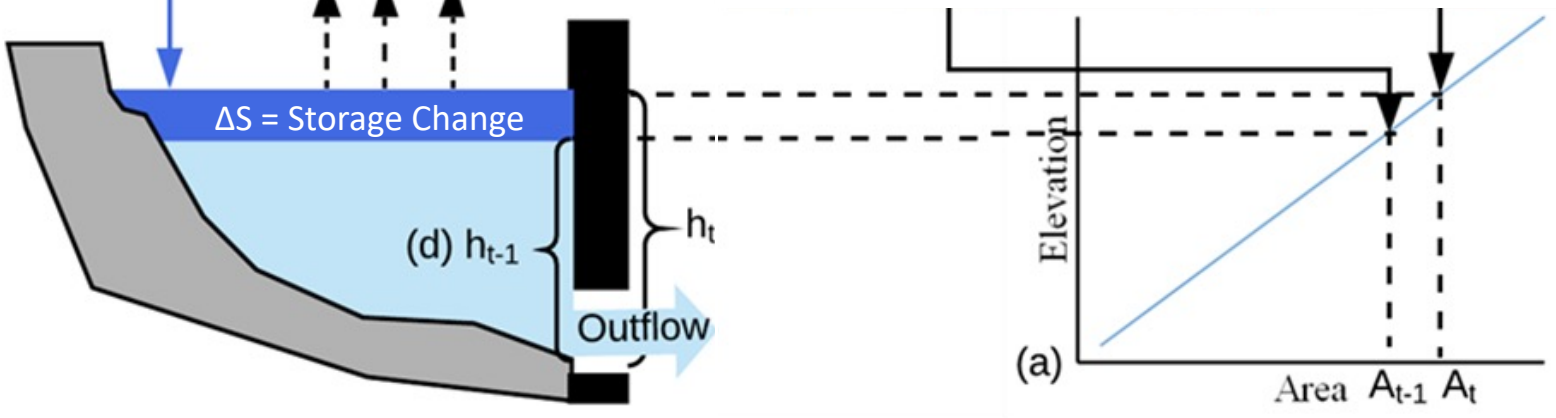
(by Penman Equation)

Inflow

Evaporation







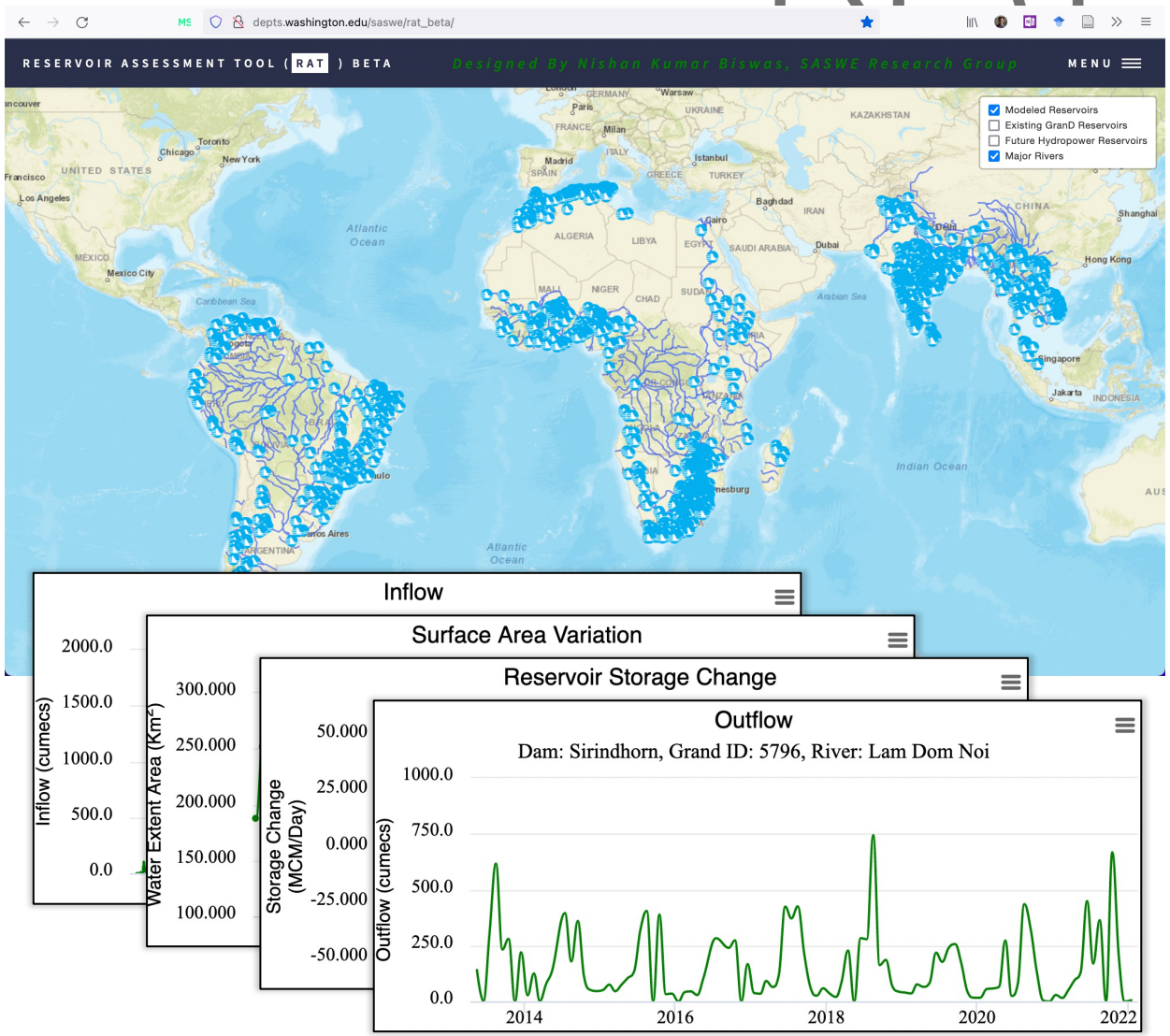
Outflow = Inflow – Evaporation – Storage Change

Reservoir Assessment Tool 1.0

- Reservoir operations monitoring using **publicly available satellite data** and **hydrological modeling**

Biswas et al. (2021)

- **Monthly Estimates**:
  - **Inflow** (I): Modeled using VIC
  - **Surface Area** (A): Landsat 7, 8
  - **Storage Change** (ΔS): A and Area-Elevation Curve (AEC)
  - **Outflow** (O): I and ΔS



Reservoir Assessment Tool 2.0

- More Sensors
  - ***Optical*** *– Landsat-8, Sentinel-2*
  - ***SAR*** *– Sentinel-1*
  - ***Altimetry*** *– JASON-3*
- Weekly Observations
  - ***3-7*** *Days surface area*
  - ***10*** *Days altimetry*
- Better Performance using parallel processing

BUT ….

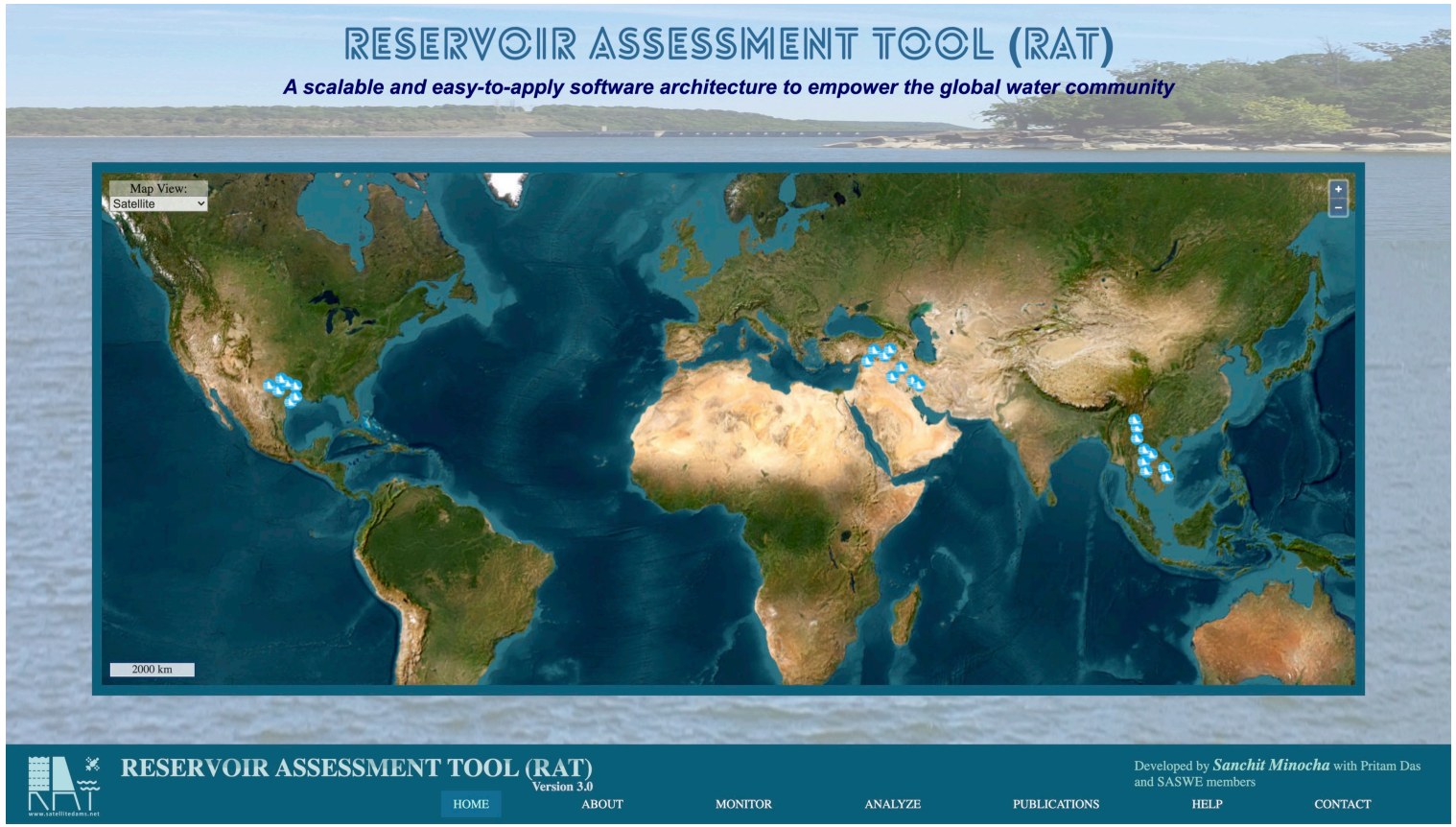
- Limited to Mekong Region
- Not open source
- Not easy to use
- Memory Inefficient

Das et al. (2022)

Reservoir Assessment Tool 3.0

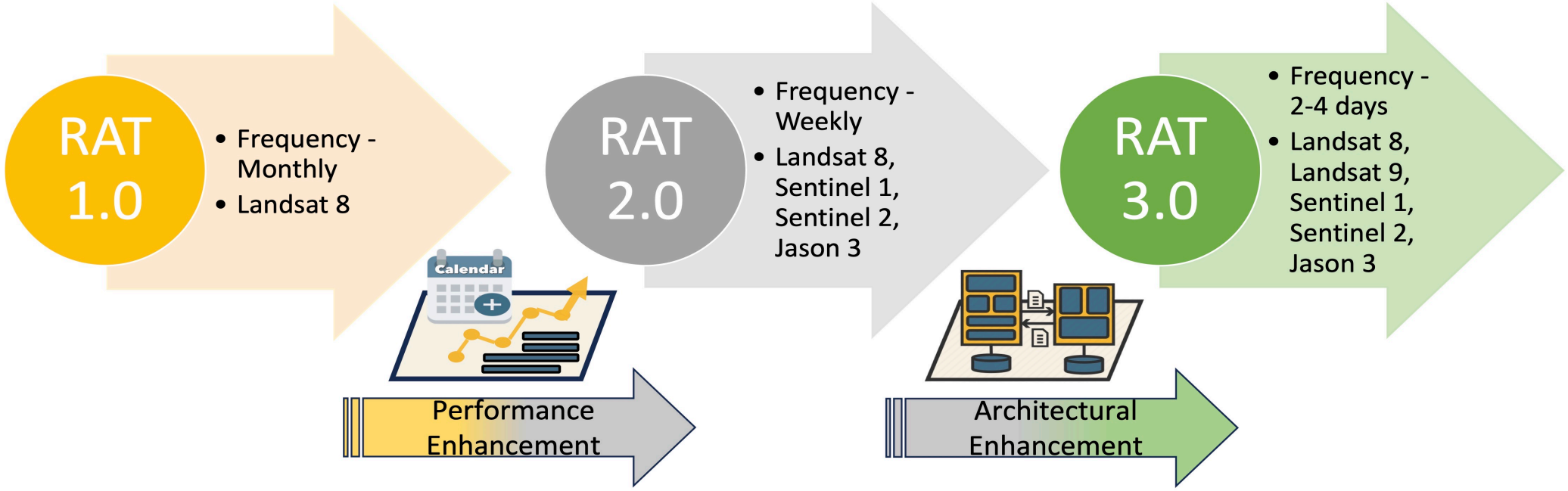
www.satellitedams.net

- More Sensors
  - ***Optical*** *– Landsat-8 and 9, Sentinel-2*
  - ***SAR*** *– Sentinel-1*
  - ***Altimetry*** *– JASON-3*
- Sub-weekly Observations
  - ***2-4*** *Days surface area*
  - ***10*** *Days altimetry*
- Globally applicable
- Efficient
- User-Friendly
- User-Focused Web Application

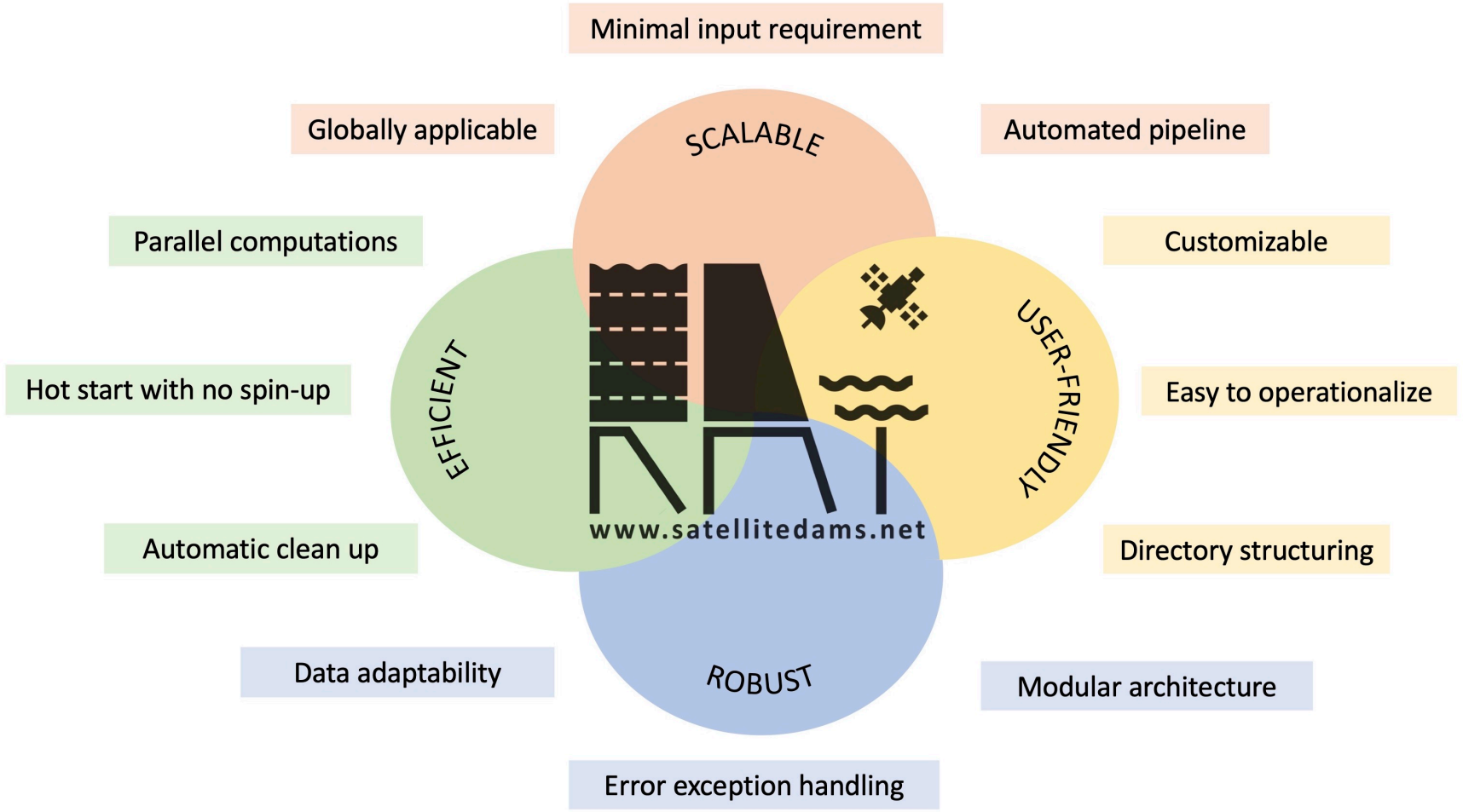


Technological Evolution of RAT 3.0

www.satellitedams.net



Advancements in RAT 3.0



RAT 3.0 : Requirements

1. A Linux-based system with Python installed and preferably VS Code or any code editor that you like.
   a) You can download VS Code here.
   b) Make sure you have downloaded the following utilities in Linux OS:
      - gfortran
      - make
      - miniconda or conda

2. Login credentials for Aviso (for reservoir height data)
   https://www.aviso.altimetry.fr/en/data/data-access/registration-form.html

3. Login credentials for Imerg (for pulling precipitation data for hydrologic model)
   https://registration.pps.eosdis.nasa.gov/registration/

4. Login credentials for Earth Engine using service account (for reservoir storage change calculation)
   https://developers.google.com/earth-engine/cloud/earthengine_cloud_project_setup

Note: Next section details on how to get login credentials for requirements 2,3 and 4. Follow the instructions and keep the credentials(or password file) handy while installing and initializing RAT 3.0.

IMERG Account Set up

https://registration.pps.eosdis.nasa.gov/registration/

Step 1

PPS Registration

Click on ***"Register"*** to get access to PPS Products.

Fill out the form and click on "Save".
You will get a confirmation e-mail and use it to complete the process.

If you don't receive this e-mail in one hour, please check in your spam fol

Once you are registered, you can edit your information by entering in you
contained in the automated email to complete the process.

Please note that by registering to get access to GPM data through PPS, y
do not wish to receive system status emails then please do not register fo

We do not accept email addresses that require us to take a manual actior
gmail address instead.

If you plan to use Near-Real Time (NRT) data stored on jsimpsonftps.pps
Otherwise, your account will only allow access to production data on arthu
for NRT access. You can add/remove NRT access using the "Verify Email

Please note that your Email will be converted to lower case. Once registra
retrieve data from our FTP archives or place orders through STORM.

NEVER reply to an email that is sent to you. If you have questions, please

Register

OR

Enter registered email:

Verify Email or Update Info **Remove from access to PPS**

Make sure to check NRT Products

Step 2

Please note that your Email will be converted to lower case. Once registration is completed use
We do not accept email addresses that require us to take a manual action (Boxbe, etc.). Please

Email *
Confirm Email: *

First Name *
Last Name *

Checking the following box will enable access to the jsimpsonftps.pps.eosdis.nasa.gov FTPS portal.
Please select this option only if you intend to access these products immediately!
Note that, as of 18 June 2020, only FTPS and HTTPS can be used to access NRT data. Standard

FTPS Access
in addition to
Standard Research Products

☐ Near-Realtime Products

Please enter a valid organization below. PPS will perform random checks of organizations and

Organization *
Address

Organization Type *
- ○ Governmental
- ○ Non-Governmental Organization
- ○ International Organization
- ○ University / College
- ○ Other Education
- ○ Research Institute
- ○ Commercial Weather
- ○ Insurance
- ○ Other Commercial
- ○ Private Individual

Area of Interest
- Data
- Merged RT Products
- Tools
- TKIO

Check the box below saying that you agree to allow PPS to send you emails relating to system

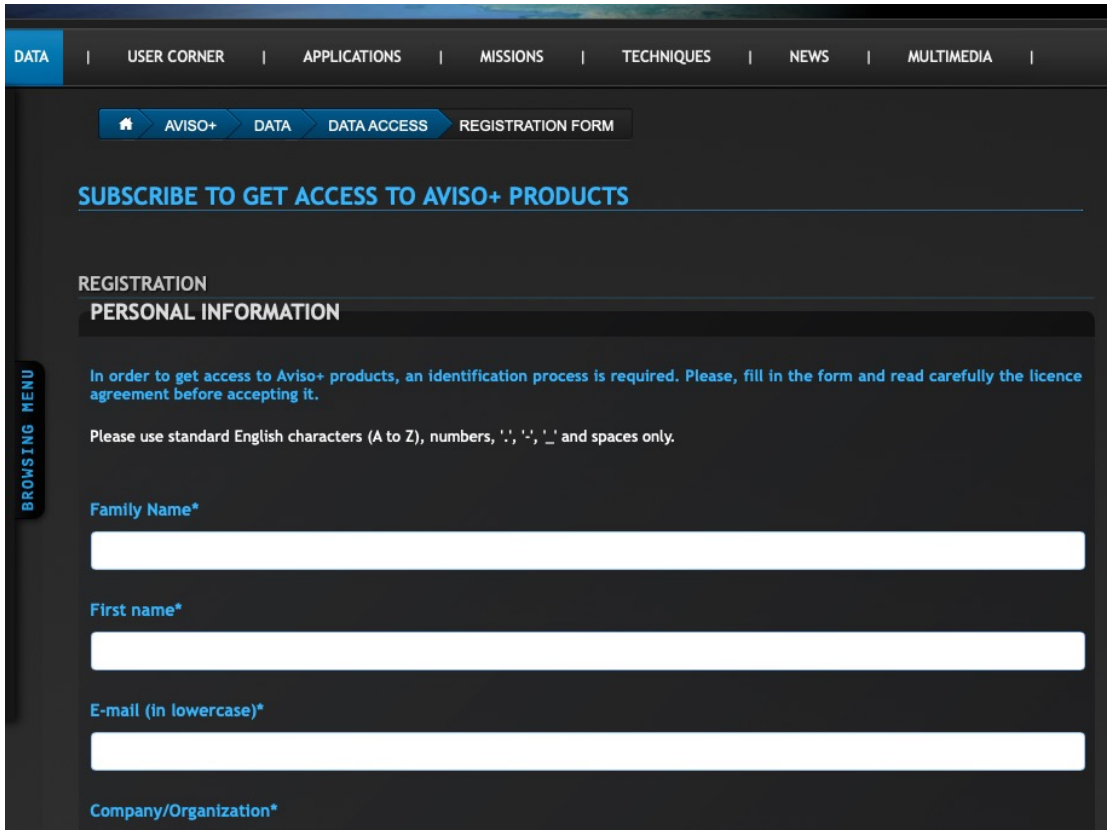
Allow Emails * ☐

Save

AVISO Account Set up

https://www.aviso.altimetry.fr/en/data/data-access/registration-form.html

Step 1: Fill out the form.



Step 2: Select '**GDR/IGDR (Geophysical Data Records)'**
In the Product Selection section.

Step 3: Accept the terms and conditions.

Step 4: Submit the form.

Google Earth Engine (GEE) Set up

User Account	VS	Service Account
➢ User accounts are normal google account being used by us in day-to-day tasks.		➢ A service account is a special type of Google account intended to represent a non-human user that needs to authenticate and be authorized to access data in Google APIs.
➢ saswegee@gmail.com is a user account.		➢ sanchit-rat@globalrat.iam.gserviceaccount.com is a service account.

Google Earth Engine (GEE) Set up

When to use Service Account?

- Running workloads on virtual machines (VMs).
- Running workloads on on-premises workstations or data centers that call Google APIs.
- Running workloads which are not tied to the lifecycle of a human user.

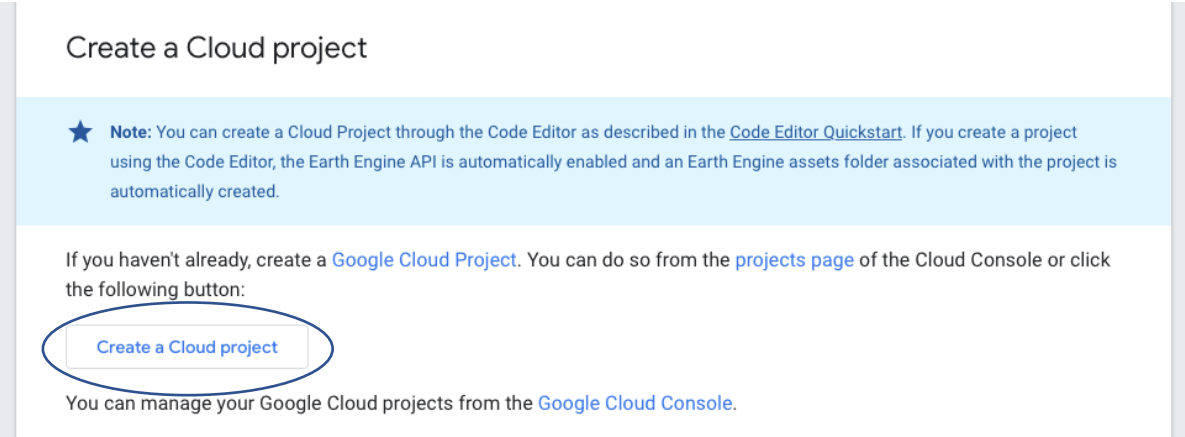
Google Earth Engine (GEE) Set up

www.satellitedams.net

Task 1- Create a google cloud project

https://developers.google.com/earth-engine/cloud/earthengine_cloud_project_setup

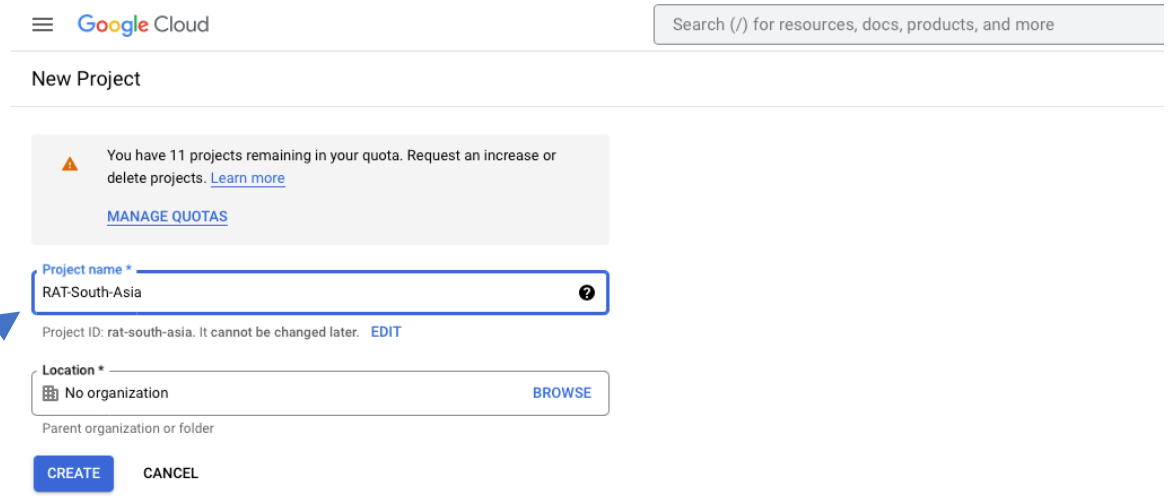
Step 1



Click on ’Create a Cloud project’

Tip: Project name can be something like ‘RAT-SE-Asia’ or ‘RAT Mekong’.

Step 2 Enter a Project name and Click on ’CREATE’.



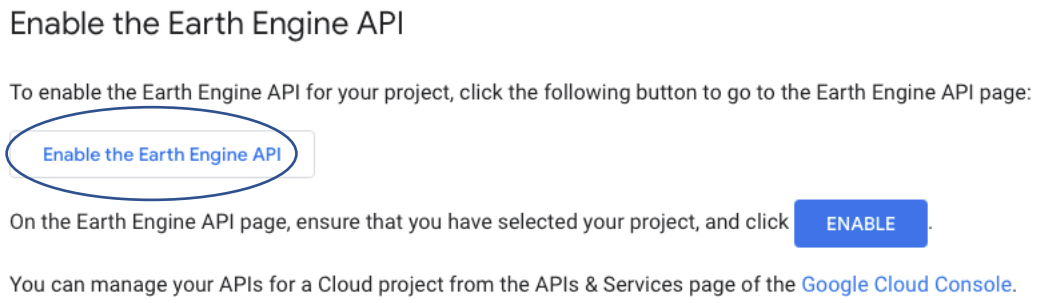
Note: Do not close the project window which will open up.

Google Earth Engine (GEE) Set up

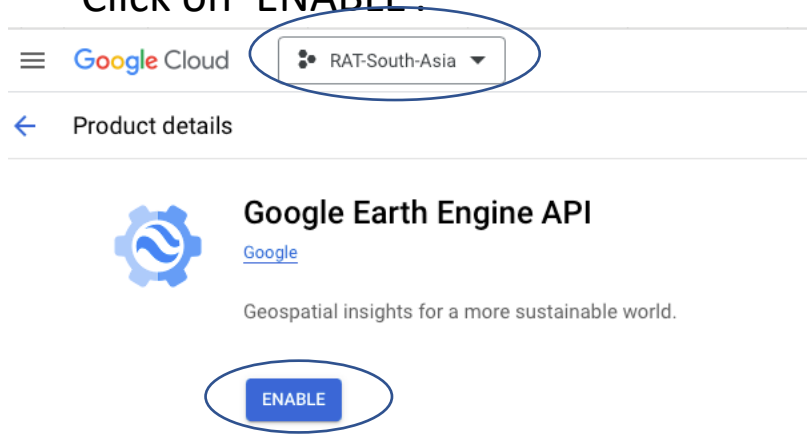
Task 2- Enable Earth Engine API

https://developers.google.com/earth-engine/cloud/earthengine_cloud_project_setup

Step 1 Click on 'Enable the Earth Engine API'



Step 2 Make sure the right project is selected and Click on 'ENABLE'.



Google Earth Engine (GEE) Set up

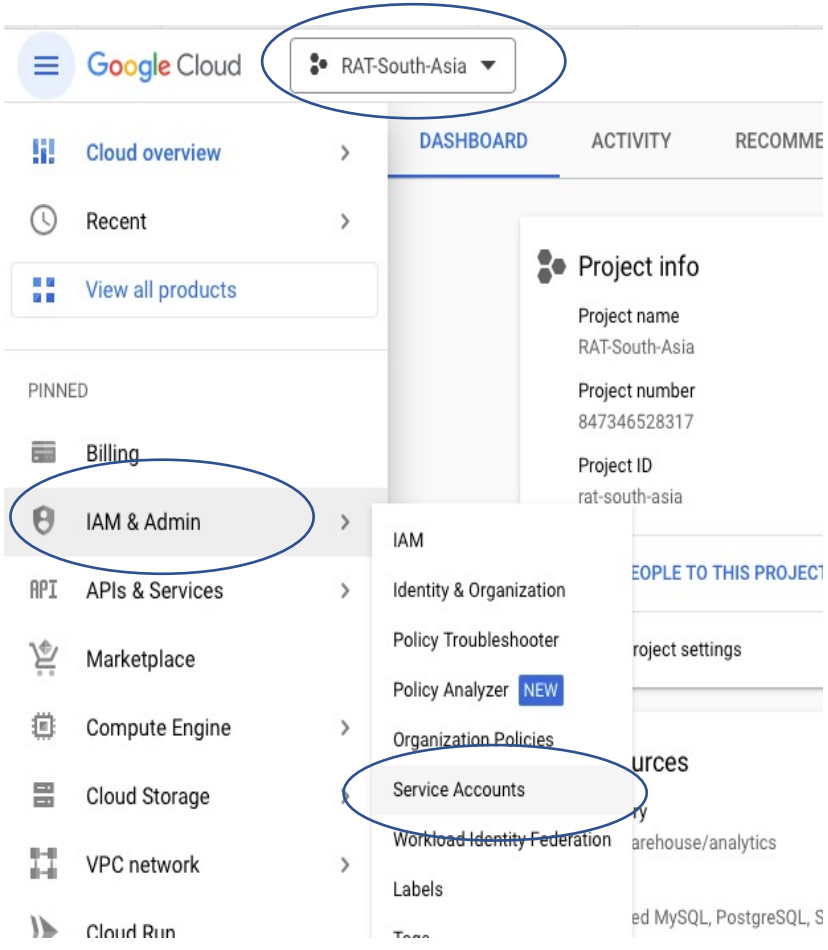
Task 3-Create a service account

Note: In the project window which was opened up while following instructions in the second last page.

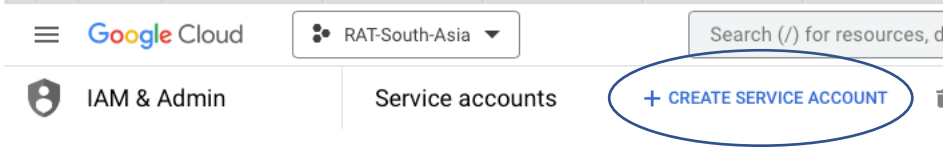
Tip: (1) Service account name can be something like 'rat-Mekong-YOUR_NAME' ('rat-mekong-sanchit').
(2) Description can be who will be using this service account and for what.

Step 1

Make sure the right project is selected and Click on 'IAM & Admin' > 'Service Accounts'

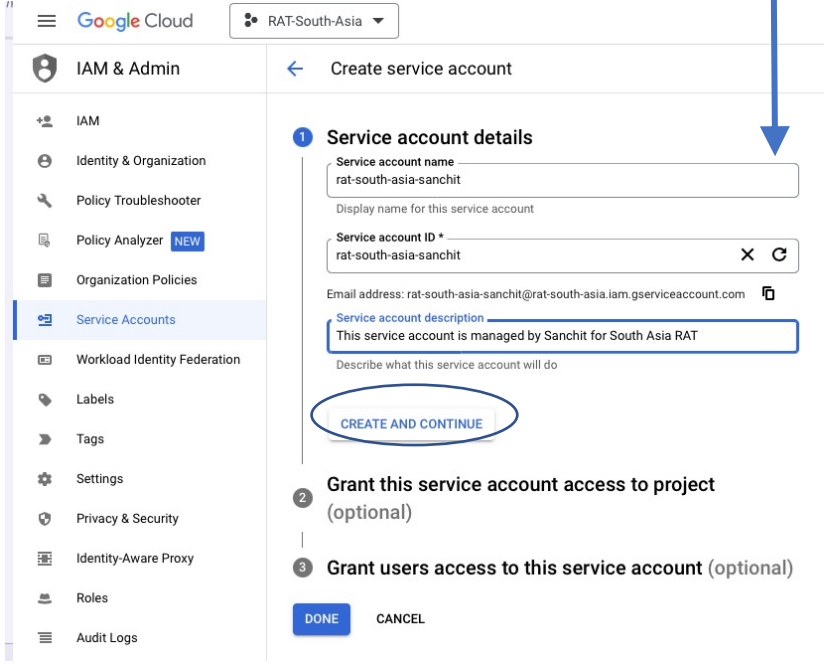


Step 2 Click on 'CREATE SERVICE ACCOUNT'



Step 3

Click on 'CREATE AND CONTINUE'



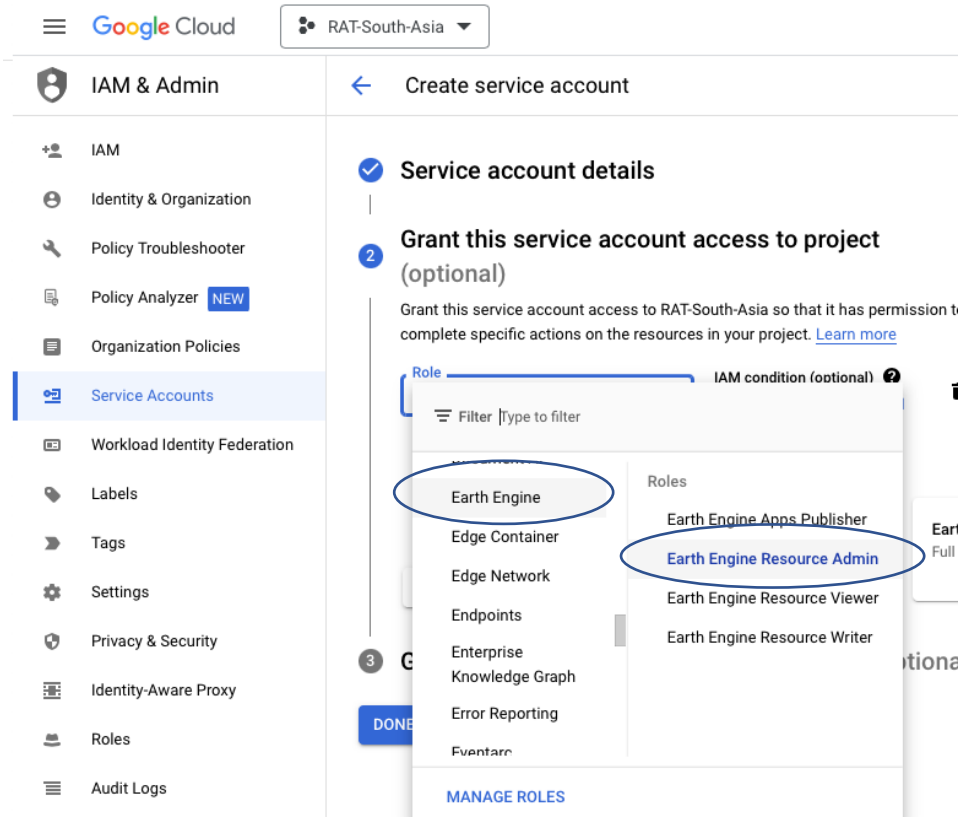
Google Earth Engine (GEE) Set up

Task 3-Create a service account

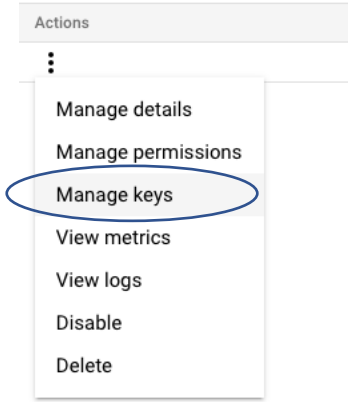
www.satellitedams.net

Step 4

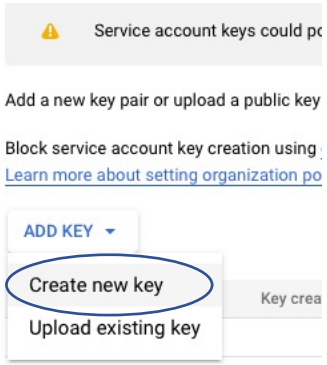
Choose ‘Earth Engine’ > ‘Earth Engine Resource Admin’ and click on ‘CONTINUE’. **After that click on ‘Done’.**



Step 5 Click on ‘Actions’ > ‘Manage Keys’ for the service account you created.



Step 6 Click on ‘Add KEY’ > ‘Create new key’ and select json



Google Earth Engine (GEE) Set up

www.satellitedams.net

Task 4- Register the service account to use Earth Engine

https://developers.google.com/earth-engine/guides/service_account#register-the-service-account-to-use-earth-engine

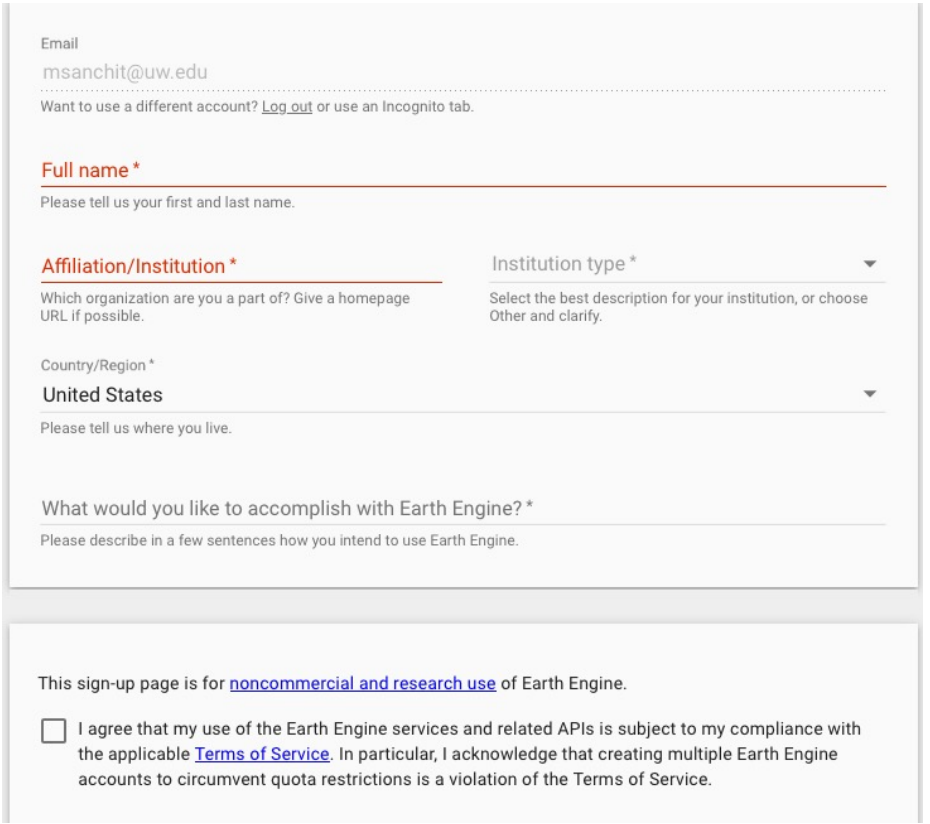
Step 1: Click on 'this page' and select your email address.

Register the service account to use Earth Engine

★ If you use a Cloud project to access Earth Engine, all service accounts in that project with the correct permissions have access to the EE API, and there's no need to register them separately.

If you don't use a registered Cloud project to access Earth Engine, you can use this page to register your service account for use with the Earth Engine API. Once you've successfully registered your service account, follow the instructions that appear on the confirmation screen to enable access to the Earth Engine API.

Step 2: Fill out the form and accept the terms.



Step 3: Click on submit.

Step 4: Again click on the link mentioned above and follow step 1.

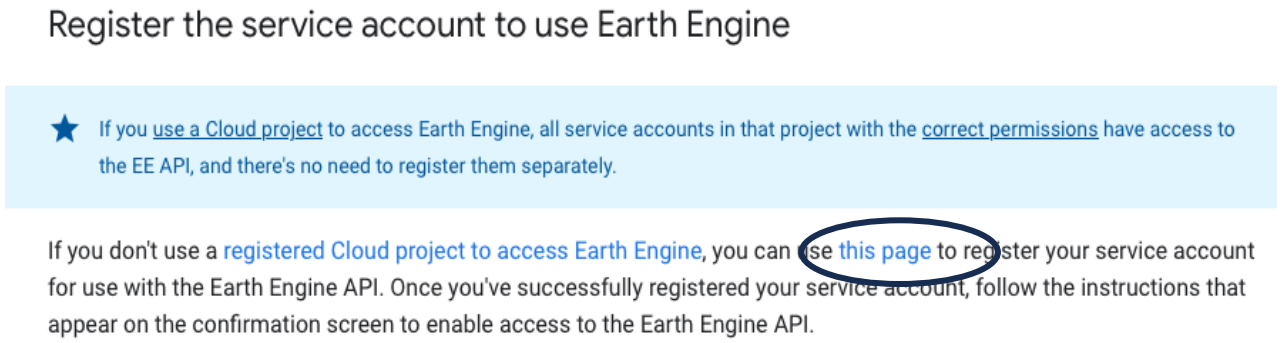
Google Earth Engine (GEE) Set up

www.satellitedams.net

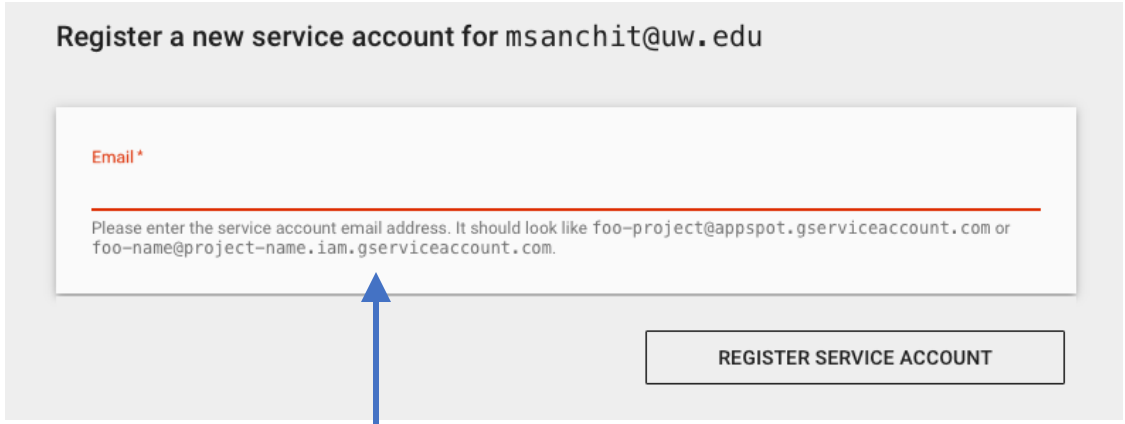
Task 4- Register the service account to use Earth Engine

https://developers.google.com/earth-engine/guides/service_account#register-the-service-account-to-use-earth-engine

Step 4: Again click on the link mentioned above and repeat step 1 by clicking on 'this page'.



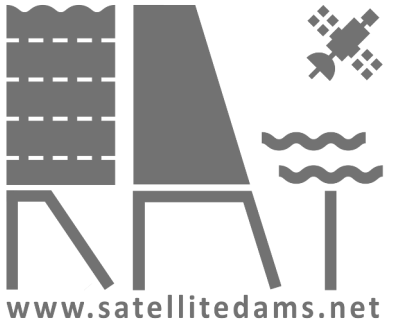
Step 5: Enter the created service account's email address.



Step 6: Click on 'REGISTER SERVICE ACCOUNT'.

Tip: Service account's email address will look something like "foo-name@project-name.iam.gserviceaccount.com"

RAT 3.0 Installation



1. Create an empty project directory and move into it. Ex - ‘rat_project’

```
mkdir rat_project
cd ./rat_project
```

2. Create a conda environment directory inside project directory. Ex - ’.rat_env’

```
mkdir ./.rat_env
```

3. Create a conda environment using directory created above

```
conda create --prefix ./.rat_env
```

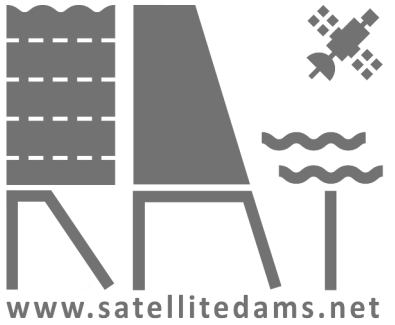
4. Activate this environment using conda

```
conda activate ./.rat_env
```

5. Install RAT 3.0 using conda

```
conda install rat -c conda-forge
```

RAT 3.0 Initialization



Initialize RAT using '**rat init**' command.

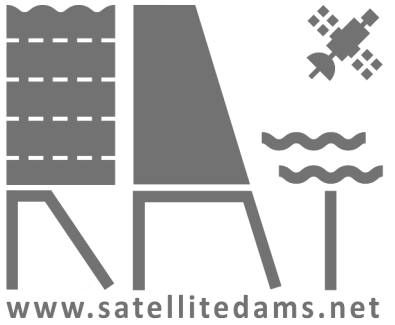
- ✓ Installs Metsim
- ✓ Installs VIC
- ✓ Download & Compile Routing
- ✓ Download Parameter files
- ✓ Download Global Data (If desired)
- ✓ Automatically Populates Config file

```
rat init -d . -g -gp <PATH_TO_DOWNLOAD/OF_DOWNLOADED_GLOBAL_DATA> -s <PATH_TO_SECRETS_FILE> -dr google
```

-d or --dir	Put project directory path	OPTIONAL
-g or --global_data	Whether to download global data	OPTIONAL
-gp or --global_data_dir	Path of global database (to download or if existing)	OPTIONAL
-s or --secrets	Put secrets file path (if you have)	OPTIONAL
-dr or --drive	Which drive to download data from (google/dropbox)	OPTIONAL

Note: A 'rat_config.yaml' file is prepared using 'rat_config_template' in PROJECT_DIRECTORY >> 'Params' which is created after initializing RAT.

RAT Credentials File

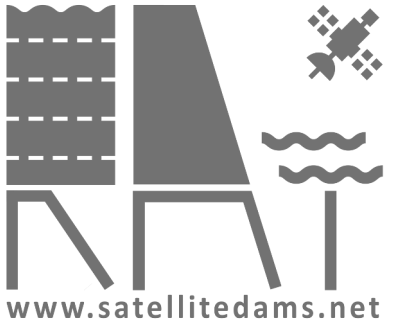


Create a ‘secrets.ini’ file using a text editor with the following content :

```
#Enter user credentials (email, password) for aviso
[aviso]
username=
pwd=

#Enter user credentials (email, password) for imerg
[imerg]
username=
pwd=

#Enter service account (account_email, path to private json key) for google earth engine
[ee]
service_account=
key_file=
```

Note: A ‘secrets_template.ini’ file is provided in PROJECT_DIRECTORY >> ‘Params’ which is created after initializing RAT. Fill in the credentials, rename the file to ‘secrets.ini’ and save it to PROJECT_DIRECTORY >> ‘secrets’ or any safe place.

RAT 3.0 Testing



Test RAT using ‘rat test’ command.

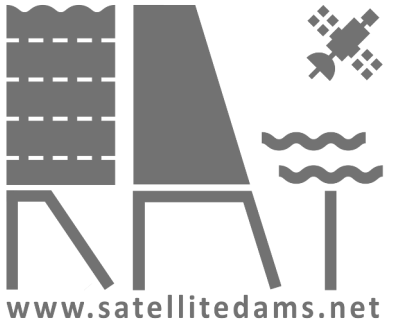
```
rat test -d . -b NUECES -s ./secrets/secrets.ini -dr google
```

- ✓ Download Test data
- ✓ Create test_config.yml file
- ✓ Run RAT for Test data
- ✓ Verifies if RAT has produced outputs as expected.

-d or --dir	Put project directory path	REQUIRED
-b or --basin	Which basin to test RAT on? (NUECES/GUNNISON)	REQUIRED
-s or --secrets	Put secrets file path (if you have)	REQUIRED
-dr or --drive	Which drive to download data from (google/dropbox)	OPTIONAL

Note: A ‘test_config.yaml’ file is prepared using ‘rat_config_template’ in PROJECT_DIRECTORY >> ‘Params’ which is created after initializing RAT.

RAT 3.0 Configuring



Autofill RAT configuration file using ‘rat configure’ command.

```
rat configure -d . -p ./params/rat_config.yaml -gp <GLOBAL_DATA_DIRECTORY_PATH> -nc <NO_CORES> -s ./secrets/secrets.ini
```

-d or --dir	Put project directory path	REQUIRED
-p or --param	Path of the rat configuration file to update	REGUIRED
-gp or --global_data_dir	Path of downloaded global database	OPTIONAL
-nc or –n_cores	Enter number of cores for RAT to use	OPTIONAL
-s or --secrets	Put secrets file path (if you have)	OPTIONAL

- ✓ Update provided parameter file.
- ✓ Does not prepare a new file.

Note: It is not advised to use configure command with ‘rat_config_template.yml’. Make a copy of it and then use the copy to configure.

Directory Structure after RAT Initialization

```
.
├── data
├── models/
│   ├── metsim/
│   ├── routing/
│   └── vic/
├── params/
│   ├── metsim/
│   ├── vic/
│   ├── routing/
│   ├── basins_metadata_sample.csv
│   ├── rat_config.yml
│   ├── rat_config_template.yml
│   └── secrets_template.ini
├── global_data/
│   ├── global_altimetry/
│   ├── global_basin_data/
│   ├── global_dam_data/
│   ├── global_drt_flow_file/
│   ├── global_elevation_data/
│   ├── global_grand_data/
│   ├── global_reservoir_data/
│   └── global_vic_params/
```

data : All files processed and created by running RAT will be in this directory

models : Metsim, VIC and routing models are saved here

params : Metsim, VIC , routing parameter files and RAT configuration files are updated here. Template files for secrets.ini, basins_metadata.csv and rat_config are also provided.

Global_data : All default global data provided along with RAT 3.0 is stored here. Once RAT has been run successfully for a basin ‘global_data’ can be deleted except ‘global_basin_data’

Outputs of RAT

data

Contains parameter files for test basins

test_data

Replication of data for test basins.

test_output

<REGION_NAME>

Contains level 1 log for each RAT run.

runs

Contains downloaded raw global data

raw

Log files for each basin

logs

basins

vic_logs

vic log files

<BASIN_NAME>

Pre_processing

Basin_grid_data

metsim

vic

ro

gee

altimetry

post_processing

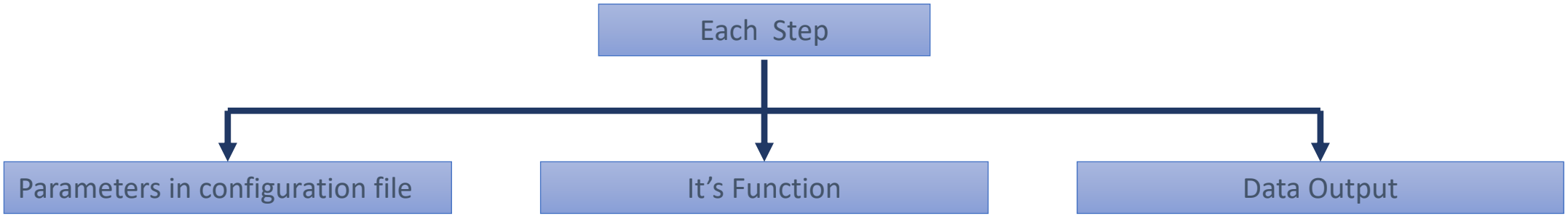
rat_outputs

final_outputs

Steps in RAT 3.0

www.satellitedams.net

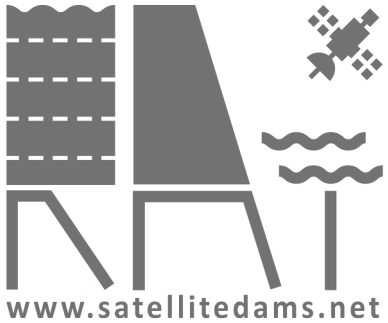
- 2 mandatory and 14 optional steps in RAT 3.0
- Though optional, they should all be run logically (example: Metsim should run before VIC)
- Optionality gives advantage to skip a step if it's run already for the same dates for that basin
- If any mandatory step fails, RAT will stop running for that basin
- If any optional step results in an error, RAT will still try to execute next step given by user



Steps in RAT 3.0

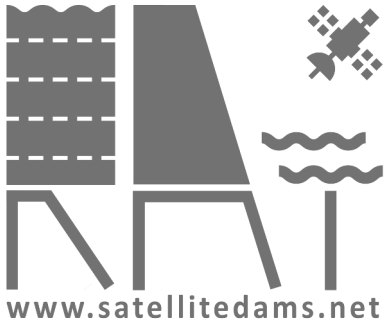
Step Number	Step Name
-1	Reading Configuration settings to run RAT
0	Creating required directory structure for RAT
1	Downloading and Pre-processing of meteorological data
2	Pre-processing of data and preparation of MetSim Input
3	Preparation of MetSim Parameter Files
4	Running MetSim & preparation of VIC input
5	Preparation of VIC Parameter Files
6	Running of VIC and preparation of Routing input
7	Preparation of Routing Parameter Files
8	Running Routing
9	Preparation of parameter files for Surface Area Calculation
10	TMS-OS Surface Area Calculation from GEE
11	Elevation extraction from Altimeter
12	Generating Area Elevation Curves for reservoirs
13	Calculation of Outflow, Evaporation, Storage change and Inflow
14	Conversion of output data to final format as time series

Steps in RAT 3.0 – Step (-1)



Step Number	Step Name
-1	Reading Configuration settings to run RAT
0	Creating required directory structure for RAT
1	Downloading and Pre-processing of meteorological data
2	Pre-processing of data and preparation of MetSim Input
3	Preparation of MetSim Parameter Files
4	Running MetSim & preparation of VIC input
5	Preparation of VIC Parameter Files
6	Running of VIC and preparation of Routing input
7	Preparation of Routing Parameter Files
8	Running Routing
9	Preparation of parameter files for Surface Area Calculation
10	TMS-OS Surface Area Calculation from GEE
11	Elevation extraction from Altimeter
12	Generating Area Elevation Curves for reservoirs
13	Calculation of Outflow, Evaporation, Storage change and Inflow
14	Conversion of output data to final format as time series

Configuration Parameters

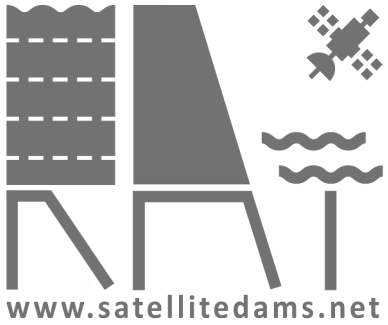


Update parameters in ‘rat_config’ file in ‘Project Directory > params as follows:

```
GLOBAL:

steps: [1,2,3,4,5,6,7,8,9,10,11,12,13,14]
# List of step numbers that needs to be executed by RAT. Default is all steps from 1 to
14.

project_dir: # Main directory path for all RAT models, parameters and global data. RAT
Output Data can have a separate directory

data_dir: # Data Directory path for RAT output and intermediary files. It can be inside
project directory.

basin_shpfile: # Basin Shapefile in json/shapefile format which can have multiple basins
with unique id.

basin_shpfile_column_dict: #example- {'id': 'MRBID'} # Dictionary of column names in
basin_shpfile defined above (must have a unique id column)
```

-1
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14

Configuration Parameters

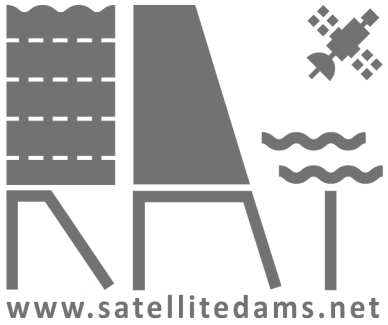


Update parameters in ‘rat_config’ file in ‘Project Directory > params as follows:

```
BASIN:

region_name: # Name of the region – used to cluster data for multiple basins within one
region

basin_name: # Basin name – used for saving data related to that basin

basin_id: # Basin id to map the basin in ‘GLOBAL:basin_shpfile’ # must match with the ‘id’
column specified through dictionary of column names in ‘GLOBAL’
```

-1
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14

Configuration Parameters

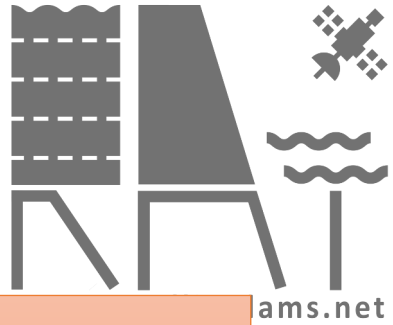


Update parameters in ‘rat_config’ file in ‘Project Directory > params as follows:

```
BASIN:

spin_up: False # True if vic spin-up is required, otherwise False # True is recommended if it’s the first RAT run for a particular basin

start: #example- 2022-08-01 #yyyy-mm-dd #Start date for RAT excluding any spin-up-time

end: #example- 2022-08-31 #yyyy-mm-dd #End date for RAT

vic_init_state_date: #example- 2022-08-18 #yyyy-mm-dd #Date of which initial state of VIC for the particular basin exists #Ignored if first_run is True #It can be left blank if no initial state file exists even when first_run is False
```

-1
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14

Function: Reading Configuration

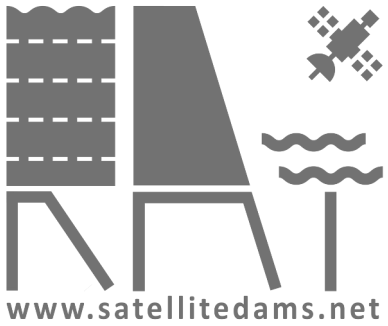


- **Basin Name**, Region and **ID**
- **start**, **end**, **spin_up**, **vic_init_state_date**
- Mostly ‘BASIN’ section parameters
- **Basin Shapefile** in ‘GLOBAL’ section
- **project_dir** and **data_dir** in ‘GLOBAL’ section
- `['CLEAN_UP']['clean_previous_outputs’]`
- **clean_previous_outputs** in ‘CLEAN_UP’ section

- ✓ Reading these parameters
- ✓ Declaring them globally
- ✓ Starts internal-detailed logging
- ✓ Cleans Previous Outputs (if desired)
- ✓ No Output

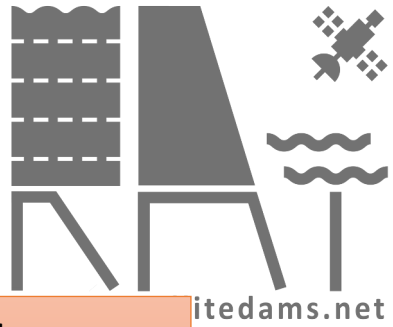
-1
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14

Steps in RAT 3.0 – Step (0)



Step Number	Step Name
-1	Reading Configuration settings to run RAT
0	Creating required directory structure for RAT
1	Downloading and Pre-processing of meteorological data
2	Pre-processing of data and preparation of MetSim Input
3	Preparation of MetSim Parameter Files
4	Running MetSim & preparation of VIC input
5	Preparation of VIC Parameter Files
6	Running of VIC and preparation of Routing input
7	Preparation of Routing Parameter Files
8	Running Routing
9	Preparation of parameter files for Surface Area Calculation
10	TMS-OS Surface Area Calculation from GEE
11	Elevation extraction from Altimeter
12	Generating Area Elevation Curves for reservoirs
13	Calculation of Outflow, Evaporation, Storage change and Inflow
14	Conversion of output data to final format as time series

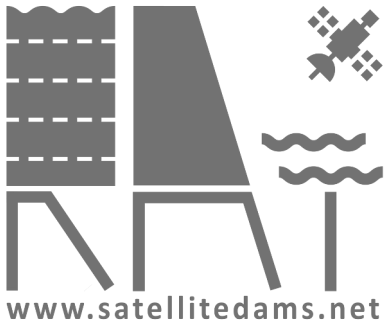
Function: Directory Structure



- ❑ Reading paths in the configuration
- ❑ Defining paths globally
- ❑ Creating directories if do not exist

- ✓ Reading some parameters
- ✓ Declaring them globally
- ✓ No Output

Steps in RAT 3.0 – Step (1)

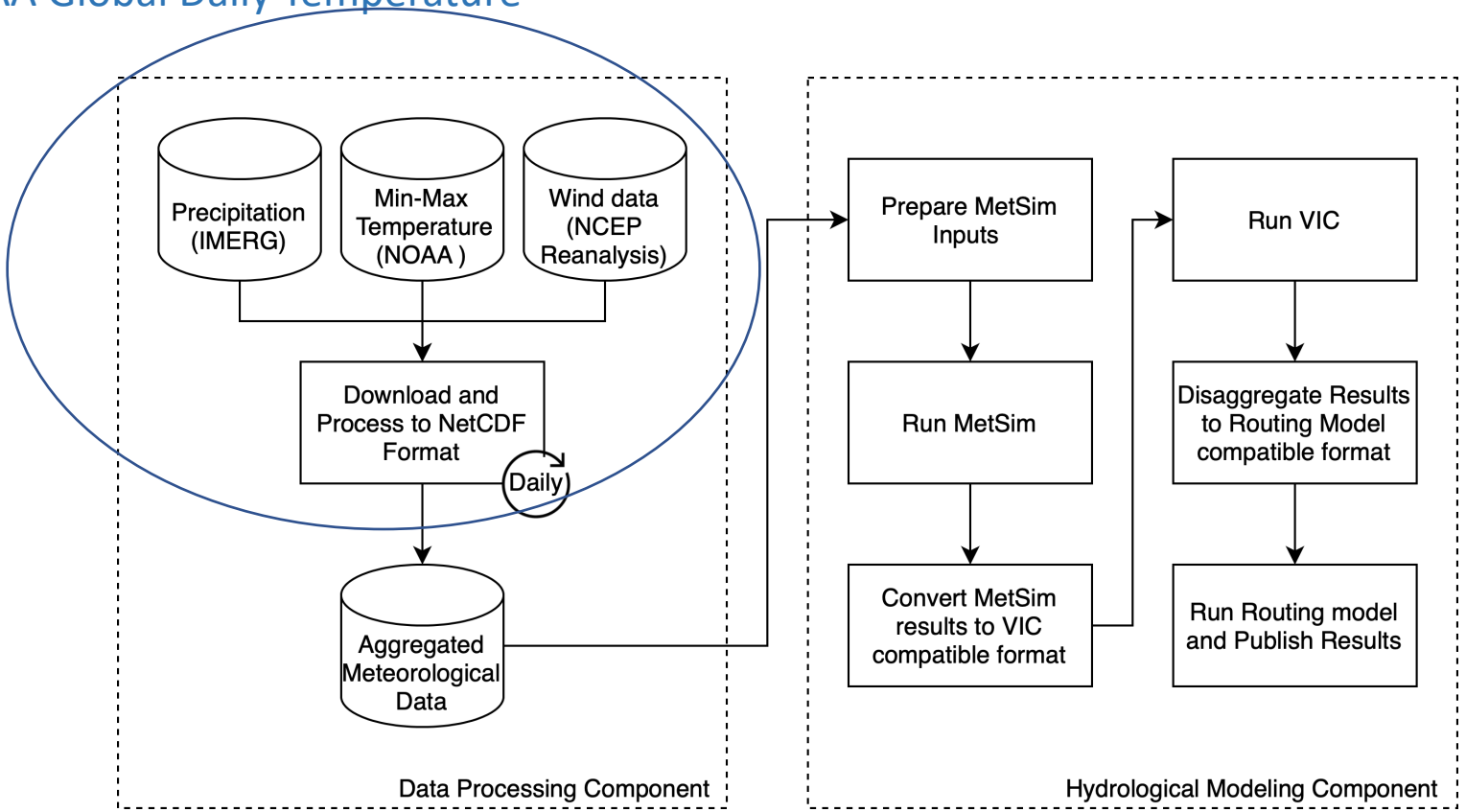


Step Number	Step Name
-1	Reading Configuration settings to run RAT
0	Creating required directory structure for RAT
1	Downloading and Pre-processing of meteorological data
2	Pre-processing of data and preparation of MetSim Input
3	Preparation of MetSim Parameter Files
4	Running MetSim & preparation of VIC input
5	Preparation of VIC Parameter Files
6	Running of VIC and preparation of Routing input
7	Preparation of Routing Parameter Files
8	Running Routing
9	Preparation of parameter files for Surface Area Calculation
10	TMS-OS Surface Area Calculation from GEE
11	Elevation extraction from Altimeter
12	Generating Area Elevation Curves for reservoirs
13	Calculation of Outflow, Evaporation, Storage change and Inflow
14	Conversion of output data to final format as time series

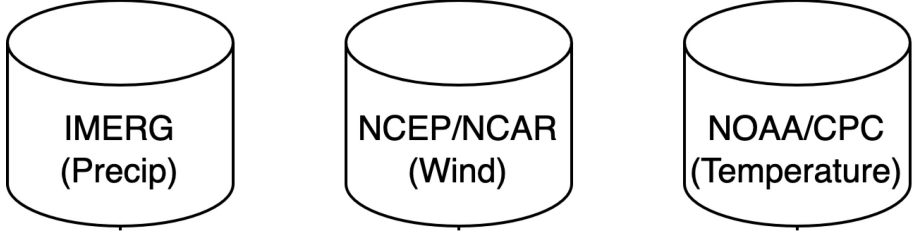
Function: Downloading

- Precipitation is downloaded as IMERG late product
  - In Ascii format
- Temperature products are from NOAA Global Daily Temperature
  - In netcdf format
- Wind data is from NCEP/NCAR
  - In netcdf format

✓ Downloading raw data from servers
✓ Output in **raw**



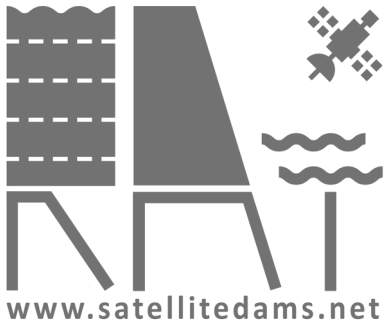
Function: Preprocessing



- Data Downloaded from providers
- Scaled
- Aligned
- Clipped
- Stored for further use

✓ Pre-process downloaded data
✓ Output in pre_**processing**

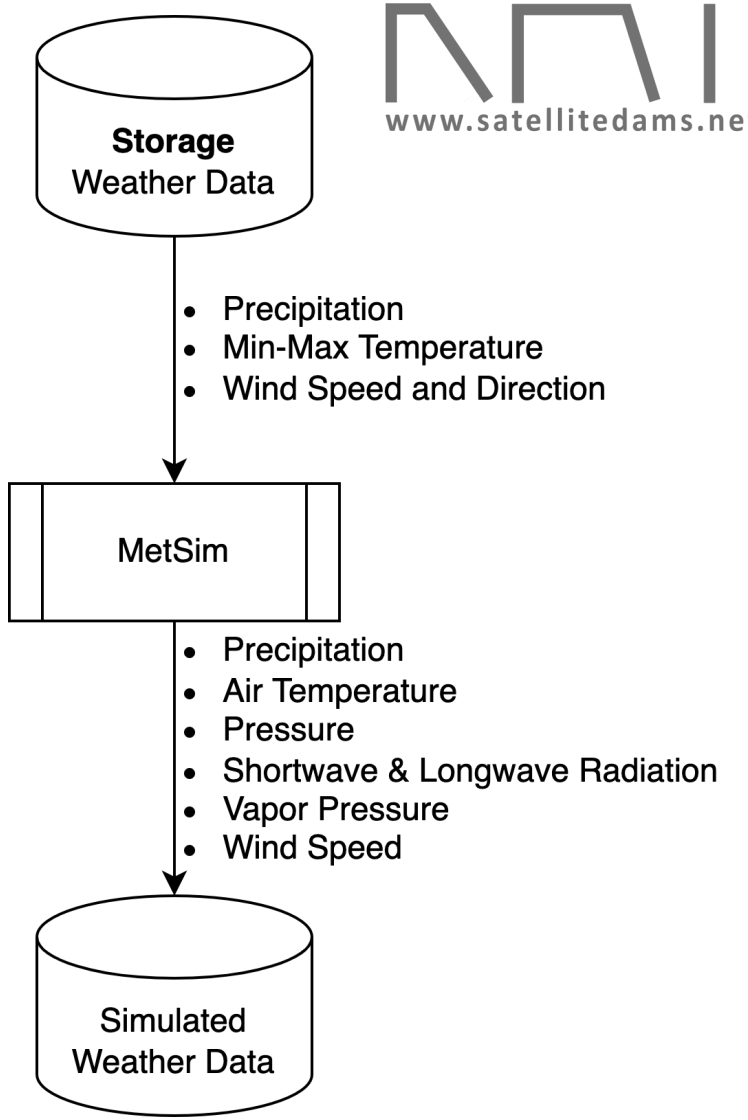
Steps in RAT 3.0 – Step (2)



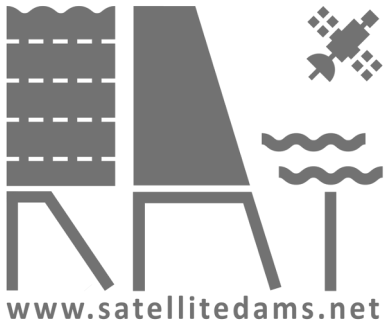
Step Number	Step Name
-1	Reading Configuration settings to run RAT
0	Creating required directory structure for RAT
1	Downloading and Pre-processing of meteorological data
2	Pre-processing of data and preparation of MetSim Input
3	Preparation of MetSim Parameter Files
4	Running MetSim & preparation of VIC input
5	Preparation of VIC Parameter Files
6	Running of VIC and preparation of Routing input
7	Preparation of Routing Parameter Files
8	Running Routing
9	Preparation of parameter files for Surface Area Calculation
10	TMS-OS Surface Area Calculation from GEE
11	Elevation extraction from Altimeter
12	Generating Area Elevation Curves for reservoirs
13	Calculation of Outflow, Evaporation, Storage change and Inflow
14	Conversion of output data to final format as time series

METSIM

- Python Package – Meteorlogical Simulator and forcing disaggregator
- Simulates weather data
- Input is Meteorological data : (i) Current data (**metsim_input.nc**)
  (ii) 3 months historic data (**state.nc**)
- Requires 2 parameter files : **domain.nc** and **params.yaml**
- It's output is required by VIC (**6h_VIC_<start>-<end>.nc**)



Configuration Parameters

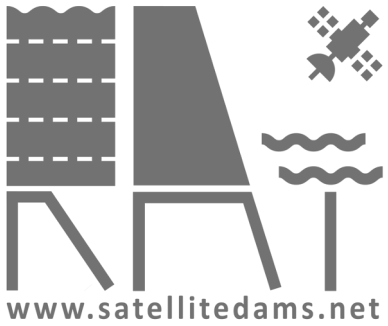


Update parameters in ‘rat_config’ file in ‘Project Directory > params as follows:

```
METSIM:

metsim_env: # Path of Metsim python environment

metsim_param_file: # Default Metsim configuration parameter file

metsim_domain_file: # Path of Metsim domain parameter file #Required if elevation file is not
provided else will be automatically created

historical_precipitation: # Path of dataset which has historical precipitation (>=3 years)
for climatology based correction of satellite precipitation. Optional.
```

-1
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14

Configuration Parameters



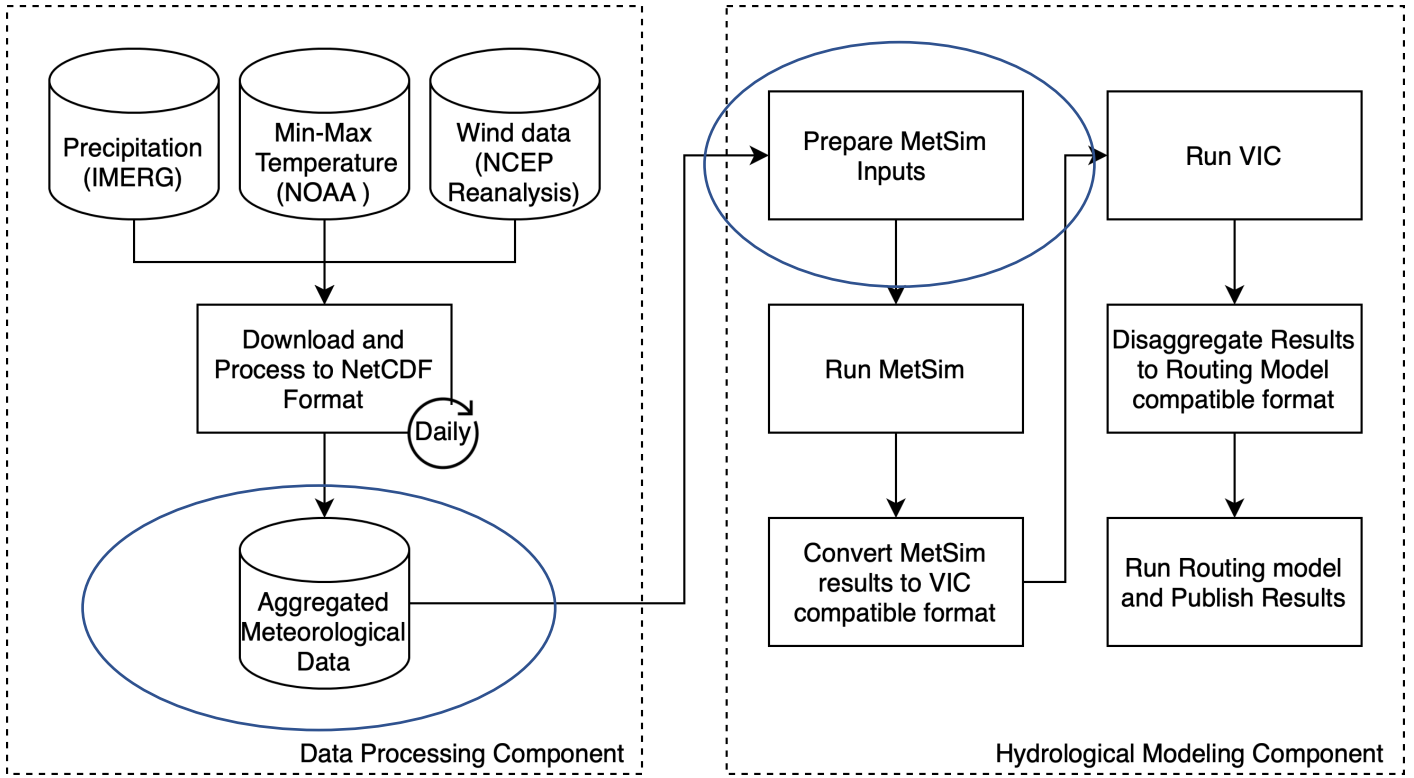
Update parameters in ‘rat_config’ file in ‘Project Directory > params as follows:

```
GLOBAL:

elevation_tif_file: # Elevation raster file in tif format. It should be having elevations in
meter and in WGS84 crs.

multiprocessing: # How many CPU cores to use

multiple_basin_run: False # True if running RAT for multiple basins in one go, will require
basins_metadata(in csv format) to update any basin particular parameter in this configuration
file. This will allow you to run RAT for multiple basins with just one configuration file.
```

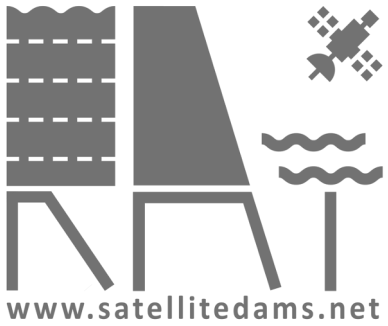
Function: Merging Data & Metsim Input

- Basin is gridded at a resolution of 0.0625°.
- Basin grid file is created(<basin_name>_grid_mask.tif)
- All preprocessed data is combined into single file (**combined.nc**)
- Metsim inputs:
  - Data from start to end (**metsim_input.nc)**
  - Past 90 days data (**state.nc**)

- ✓ No extra parameter from configuration
- ✓ **Basin grid file** created
- ✓ Data Aggregation
- ✓ **Metsim Input Data**



Steps in RAT 3.0 – Step (3)

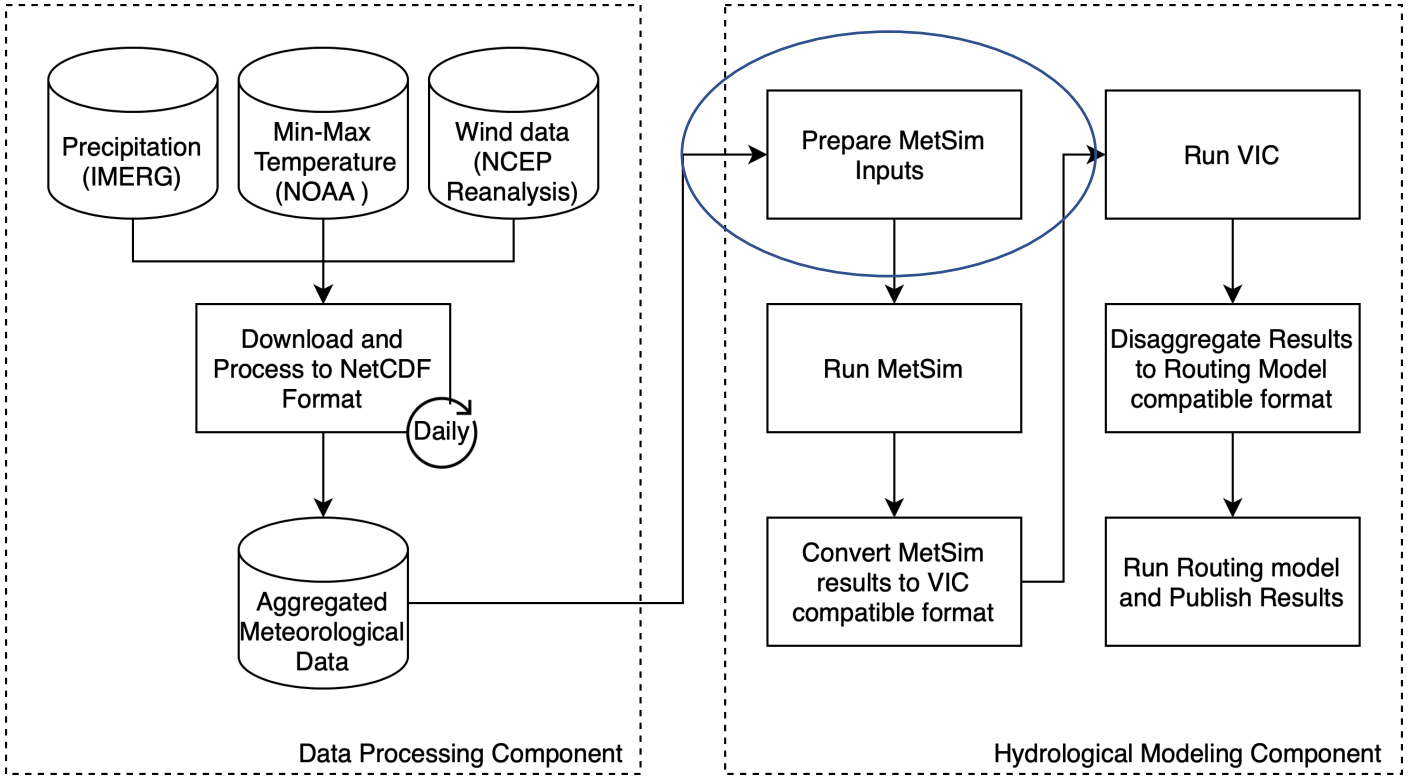


Step Number	Step Name
-1	Reading Configuration settings to run RAT
0	Creating required directory structure for RAT
1	Downloading and Pre-processing of meteorological data
2	Pre-processing of data and preparation of MetSim Input
3	Preparation of MetSim Parameter Files
4	Running MetSim & preparation of VIC input
5	Preparation of VIC Parameter Files
6	Running of VIC and preparation of Routing input
7	Preparation of Routing Parameter Files
8	Running Routing
9	Preparation of parameter files for Surface Area Calculation
10	TMS-OS Surface Area Calculation from GEE
11	Elevation extraction from Altimeter
12	Generating Area Elevation Curves for reservoirs
13	Calculation of Outflow, Evaporation, Storage change and Inflow
14	Conversion of output data to final format as time series

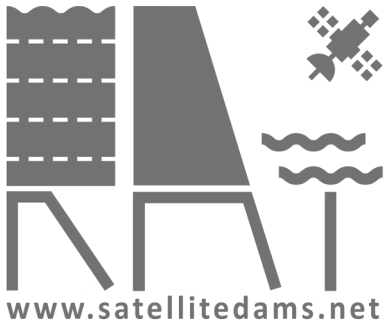
Function: Metsim Parameter

- Requires elevation file in 'GLOBAL' section
  - In Tif format
- Output is metsim domain file (**domain.nc**)

✓ **Metsim domain file** created



Steps in RAT 3.0 – Step (4)

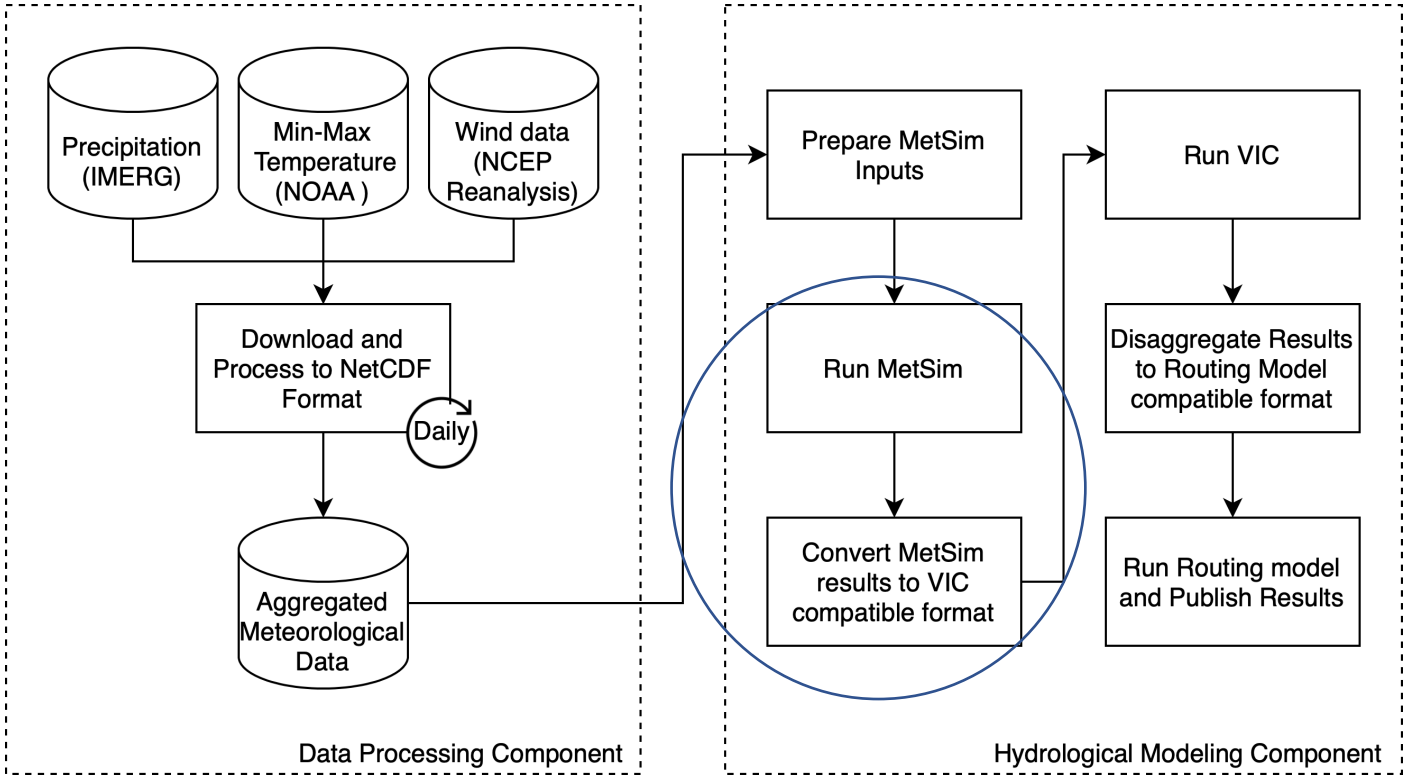


Step Number	Step Name
-1	Reading Configuration settings to run RAT
0	Creating required directory structure for RAT
1	Downloading and Pre-processing of meteorological data
2	Pre-processing of data and preparation of MetSim Input
3	Preparation of MetSim Parameter Files
4	Running MetSim & preparation of VIC input
5	Preparation of VIC Parameter Files
6	Running of VIC and preparation of Routing input
7	Preparation of Routing Parameter Files
8	Running Routing
9	Preparation of parameter files for Surface Area Calculation
10	TMS-OS Surface Area Calculation from GEE
11	Elevation extraction from Altimeter
12	Generating Area Elevation Curves for reservoirs
13	Calculation of Outflow, Evaporation, Storage change and Inflow
14	Conversion of output data to final format as time series

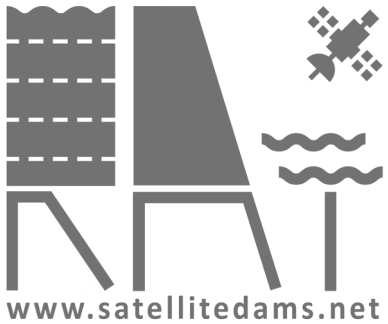
Function: Metsim Run

- Updates Metsim Parameter file (**params.yaml)**
- Runs Metsim using <n_core> parallel processing
  - <n_core> defined in 'GLOBAL' section
- Metsim Output (**6h_VIC_<start>-<end>.nc**)
- Vic Input (**forcings_<year>.nc**)

- ✓ Runs Metsim
- ✓ **Metsim Output**
- ✓ Prepares **Vic Input**



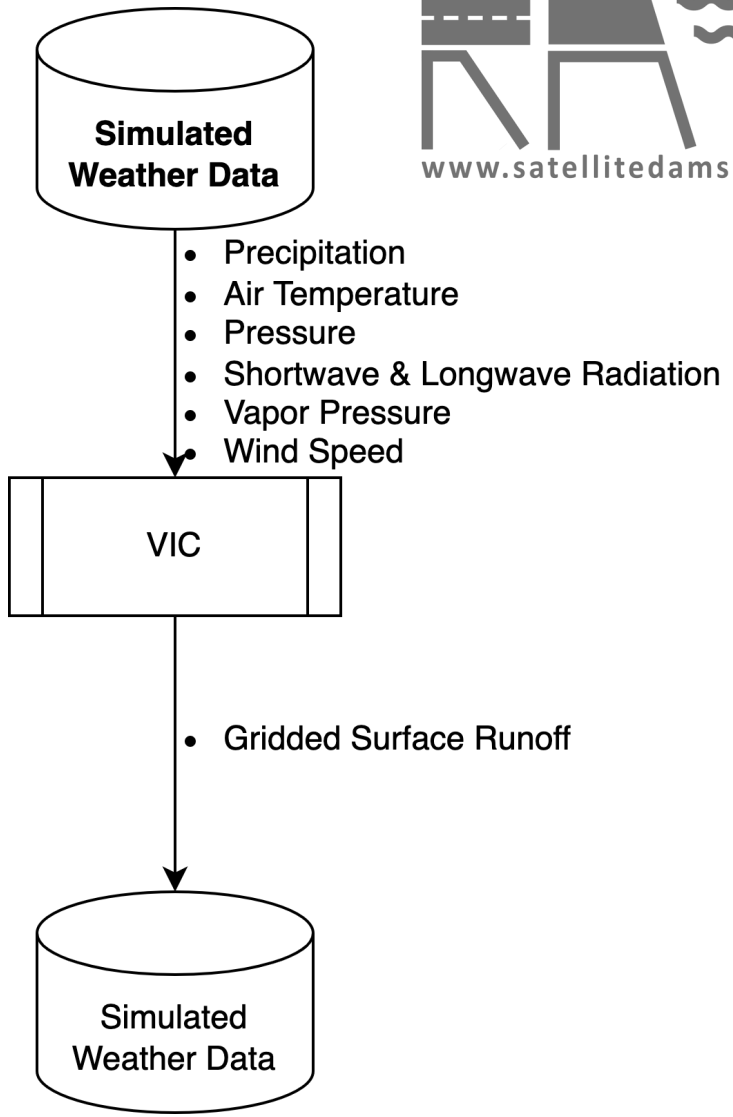
Steps in RAT 3.0 – Step (5)



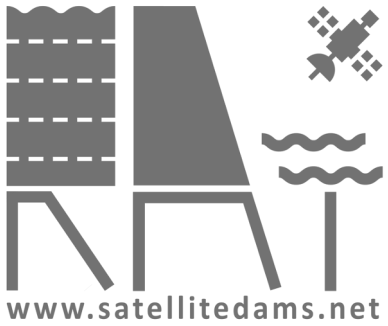
Step Number	Step Name
-1	Reading Configuration settings to run RAT
0	Creating required directory structure for RAT
1	Downloading and Pre-processing of meteorological data
2	Pre-processing of data and preparation of MetSim Input
3	Preparation of MetSim Parameter Files
4	Running MetSim & preparation of VIC input
5	Preparation of VIC Parameter Files
6	Running of VIC and preparation of Routing input
7	Preparation of Routing Parameter Files
8	Running Routing
9	Preparation of parameter files for Surface Area Calculation
10	TMS-OS Surface Area Calculation from GEE
11	Elevation extraction from Altimeter
12	Generating Area Elevation Curves for reservoirs
13	Calculation of Outflow, Evaporation, Storage change and Inflow
14	Conversion of output data to final format as time series

VIC

- Python Package – Variable Infiltration Capacity Model
- Macroscale Hydrologic Model to get gridded surface runoff
- Input is Meteorological forcings : Current data (**forcings_<year>.nc**)
- Requires 3 parameter files : (i) **vic_domain.nc**
  (ii) **vic_soil_param.nc**
  (iii) **vic_params.txt**
- It's output is required by Routing **(nc_fluxes.<start>.nc)**



Configuration Parameters

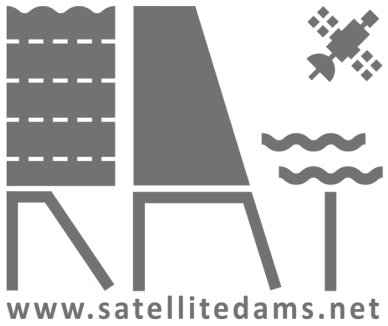


Update parameters in ‘rat_config’ file in ‘Project Directory > params as follows:

```
VIC:

vic_env: # Path of Vic python environment

vic_param_file: # Default vic global configuration parameter file. OPTIONAL. If passed, this
file will be used to initialize, but will be over-riden by any options specified in this
config file. If not passed, all necessary options must be specified in this file in 'VIC
PARAMETERS'.
```

-1
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14

Configuration Parameters

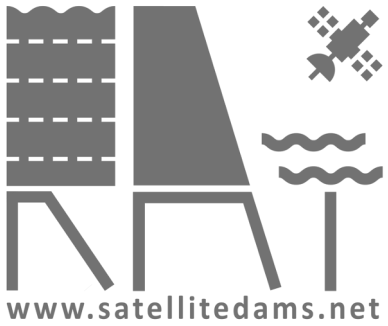


Update parameters in ‘rat_config’ file in ‘Project Directory > params as follows:

```
VIC:

vic_global_data: # True if vic global (relative to basin) parameter information is available
and should be used

## If global (relative to basin) vic parameter information is available and needs to be
cropped for the basin [vic_global_data must be True], provide the following three parameters

vic_global_param_dir: # Directory of global vic soil and domain parameters # Only required if
vic_global_data=True, otherwise ignored

vic_basin_continent_param_filename: # Name of the global vic soil parameter file (in netcdf
format) # Only required if vic_global_data=True, otherwise ignored

vic_basin_continent_domain_filename: # Name of the global vic soil parameter file (in netcdf
format) # Only required if vic_global_data=True, otherwise ignored
```

-1
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14

Configuration Parameters

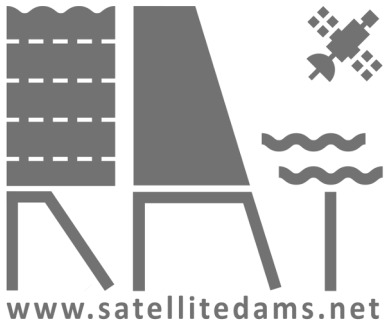


Update parameters in ‘rat_config’ file in ‘Project Directory > params as follows:

```
VIC:

## If basin vic parameter information is available and can be used as it is for basin
[vic_global_data must be False], provide the following two parameters

vic_soil_param_file: #Vic Soil Parameter File Path (in netcdf format) # Only required if
vic_global_data=False, otherwise ignored

vic_domain_file: #Vic Domain File Path (in netcdf format) # Only required if
vic_global_data=False, otherwise ignored
```

-1
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14

Configuration Parameters



Update parameters in ‘rat_config’ file in ‘Project Directory > params as follows:

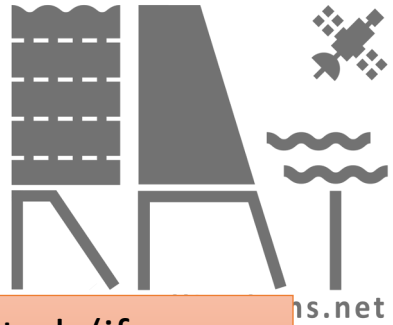
OPTIONAL

 # If vic_param_file is not provided

```
##---- VIC Parameters (Optional; must match the option names of [global parameter file of VIC](https://vic.readthedocs.io/en/master/Documentation/Drivers/Image/GlobalParam/)).
##---- If not provided, then the parameter file defined by `vic_param_file` will be used as the parameter file

VIC PARAMETERS:
  # STARTYEAR: '2001'
  # STARTMONTH: '04'
  # STARTDAY: '02'
  # ENDYEAR: '2021'
  # ENDMONTH: '04'
  # ENDDAY: '01'

  # # Any other option goes here
  # DOMAIN_TYPE:
  #   LAT: lat
  #   LON: lon
  #   MASK: mask
  #   AREA: area
  #   FRAC: frac
  #   YDIM: lat
  #   XDIM: lon
  # FORCING1: Directory path for vic/forcing_
  # FORCE_TYPE:
  #   AIR_TEMP: temp
  #   PREC: prec
  #   PRESSURE: air_pressure
  #   SWDOWN: shortwave
  #   LWDOWN: longwave
  #   VP: vapor_pressure
  #   WIND: wind
  # OUTVAR:             # Specify as list
  #   - OUT_PREC
  #   - OUT_EVAP
  #   - OUT_RUNOFF
  #   - OUT_BASEFLOW
  # RESULT_DIR: Directory path for vic results
```

-1
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14

Function: VIC Parameters

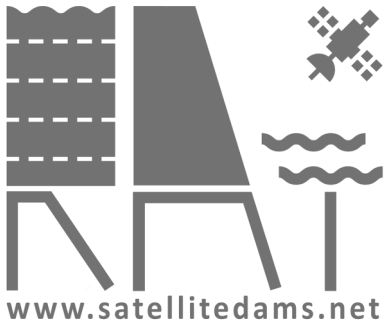


- Requires **vic_soil_param** and **vic_domain** files global/basin in 'VIC' section
  - In Netcdf format
- Outputs are vic domain file (**vic_domain.nc**) and vic soil parameter file (**vic_soil_param.nc**)

✓ **Vic Parameter Files** are created. (if desired)

-1
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14

Steps in RAT 3.0 – Step (6)

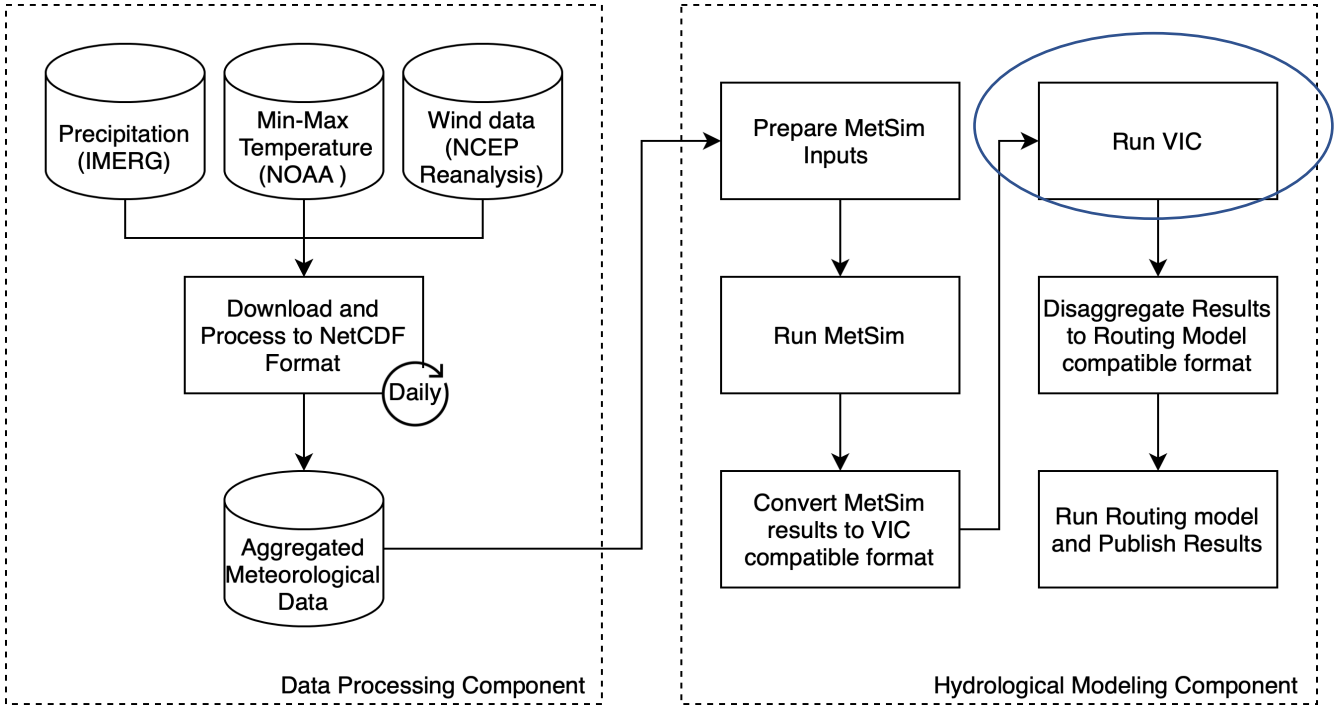


Step Number	Step Name
-1	Reading Configuration settings to run RAT
0	Creating required directory structure for RAT
1	Downloading and Pre-processing of meteorological data
2	Pre-processing of data and preparation of MetSim Input
3	Preparation of MetSim Parameter Files
4	Running MetSim & preparation of VIC input
5	Preparation of VIC Parameter Files
6	Running of VIC and preparation of Routing input
7	Preparation of Routing Parameter Files
8	Running Routing
9	Preparation of parameter files for Surface Area Calculation
10	TMS-OS Surface Area Calculation from GEE
11	Elevation extraction from Altimeter
12	Generating Area Elevation Curves for reservoirs
13	Calculation of Outflow, Evaporation, Storage change and Inflow
14	Conversion of output data to final format as time series

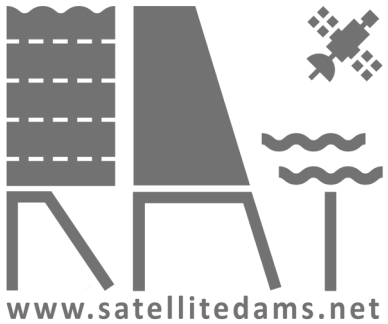
Function: VIC Run

- Updates VIC Parameter file (**vic_params.yaml)**
- Runs VIC using <n_core> parallel processing
  - <n_core> defined in 'GLOBAL' section
- Vic Input (**forcings_<year>.nc**)
- VIC Output (**nc_fluxes_<start>.nc**)
- Routing Input (**fluxes_<lat>_<lon>**)
- Creates VIC Init State for end **(state_<end>.nc)**

- ✓ Runs VIC
- ✓ **VIC Output**
- ✓ Prepares **Routing Input**
- ✓ Creates **Vic Initialisation State**



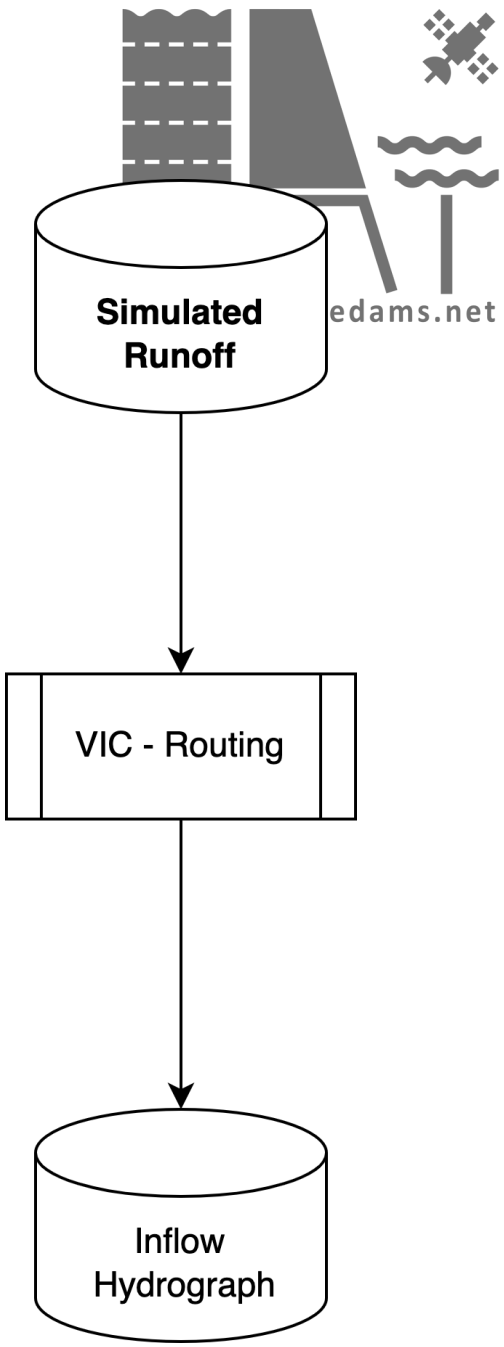
Steps in RAT 3.0 – Step (7)



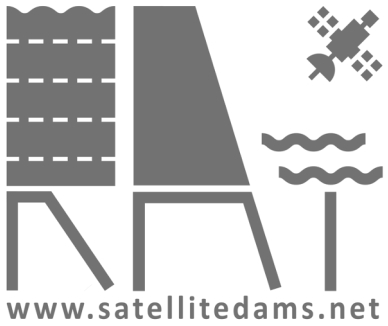
Step Number	Step Name
-1	Reading Configuration settings to run RAT
0	Creating required directory structure for RAT
1	Downloading and Pre-processing of meteorological data
2	Pre-processing of data and preparation of MetSim Input
3	Preparation of MetSim Parameter Files
4	Running MetSim & preparation of VIC input
5	Preparation of VIC Parameter Files
6	Running of VIC and preparation of Routing input
7	Preparation of Routing Parameter Files
8	Running Routing
9	Preparation of parameter files for Surface Area Calculation
10	TMS-OS Surface Area Calculation from GEE
11	Elevation extraction from Altimeter
12	Generating Area Elevation Curves for reservoirs
13	Calculation of Outflow, Evaporation, Storage change and Inflow
14	Conversion of output data to final format as time series

Routing

- Fortran Program
- Routes surface runoff within a grid cell (in the main channel)
- Input is Surface Runoff : Current data (**fluxes_<lat>_<lon>**)
- Requires 4 parameter files : (i) **sta_xy.txt**
  (ii) **fl.asc**
  (iii) **uh.txt**
  (iv) **route_param.txt**
- It's output is daily Inflow (**<station (initial 5)>.day**)
- Output converted to daily inflow in metric scale (< **station (initial 5)>.csv**)



Configuration Parameters

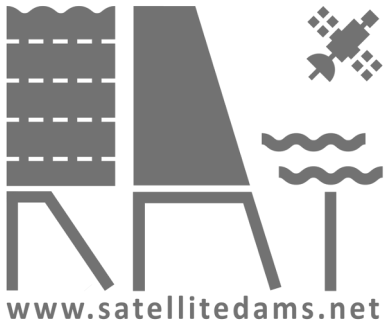


Update parameters in ‘rat_config’ file in ‘Project Directory > params as follows:

```
ROUTING:

route_model: # Path of routing model

route_param_file: # Will be used to initialize routing model. If not passed, every option
must be specified in `ROUTING PARAMETERS` section
```

-1
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14

Configuration Parameters

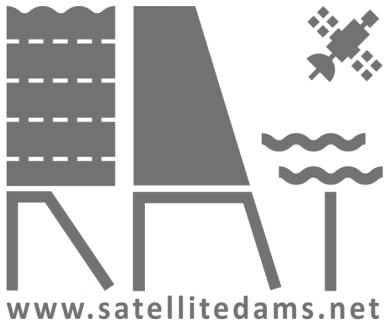


Update parameters in ‘rat_config’ file in ‘Project Directory > params as follows:

```
ROUTING:

## FLow Direction File

## If flow direction file (can be global) is available in tif format
global_flow_dir_tif_file : # flow_direction_file in tif format (global compared to basin, so
can be cropped)

replace_flow_directions : { 1 : 3,
                            4 : 5, # first replace 4 by 5 and then replace 2 by 4
                            2 : 4,
                            8 : 6,                  ## Important Note: If using global flow
                           16 : 7,                  direction file provided with RAT, don't
                           32                       change replace flow directions.
                           64 : 1,
                          128 : 2,
                          255 : 0 }  # If no replace is required, leave it blank
```

-1 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14

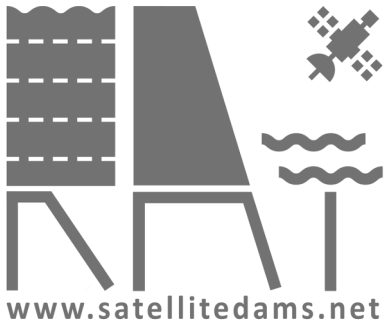
Configuration Parameters



Update parameters in ‘rat_config’ file in ‘Project Directory > params as follows:

```
ROUTING:
## Stations File (Each dam is a station)

## Either provide a global (relative to basin) stations_geojson_file with point geometry
column and define the column names using dictionary or provide station_latlon file path in
csv format

station_global_data : True # True if giving global stations_file and column names, otherwise
false and give station_latlon_path

## If station_global_data is True fill the below variables
stations_vector_file : # Must have geometry(vector) column and separate lat, lon columns
## Must have unique id_column as primary key. name_column values can be left blank, but the
column should exist.
stations_vector_file_columns_dict : {id_column : 'GRAND_ID',
name_column : 'DAM_NAME',
lon_column : 'LONG_DD',
lat_column : 'LAT_DD'}
```

Configuration Parameters

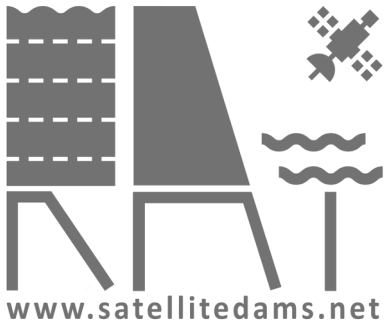


Update parameters in ‘rat_config’ file in ‘Project Directory > params as follows:

```
ROUTING:
## Stations File (Each dam is a station)

## Or provide a stations_csvfile with columns run, name, lat, lon and name of each station
must be unique with no spaces

station_global_data : False # True if giving global stations_file and column names, otherwise
false and give station_latlon_path

## If station_global_data is False fill the below variable
station_latlon_path: # In lat lon format csv format #should have columns run, name, lat, lon
```

-1
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14

Configuration Parameters

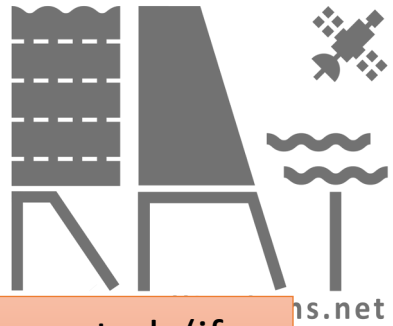


Update parameters in ‘rat_config’ file in ‘Project Directory > params as follows:

```
ROUTING PARAMETERS: # (Optional; if passed, will be given the highest priority and will
override the default ones calculated by scripts; Otherwise provide them in route_param.txt)

# flow_direction_file: # Will be generated automatically from global_flow_dir_tif_file if
                       given, otherwise uncomment and provide the path here
velocity:              1.5
diff:                  800
xmask:                 5000
fraction:              1
# station_file:        null # Will be generated automatically from either `stations_file`
# input_files_prefix: null # Will be generated automatically based on vic output
input_file_precision: 2 # Default. Works fine for 0.0625° grid.
# start_date:          null # Will be automatically populated
# end_date:            null # Will be automatically populated
# output_dir:          # Will be generated automatically for each basin
uh:                    # Unit hydrograph file for routing.
```

-1
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14

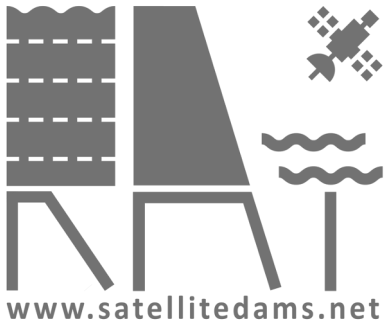
Function: Routing Parameters



- Prepares Basin Station File (**basin_station_latlon.csv**)(If desired)
  - In CSV format
- Prepares Basin grid flow direction file (**fl.asc**)(If desired)
  - In Ascii format

✓ **Routing Parameter Files** are created. (if desired)

Steps in RAT 3.0 – Step (8)

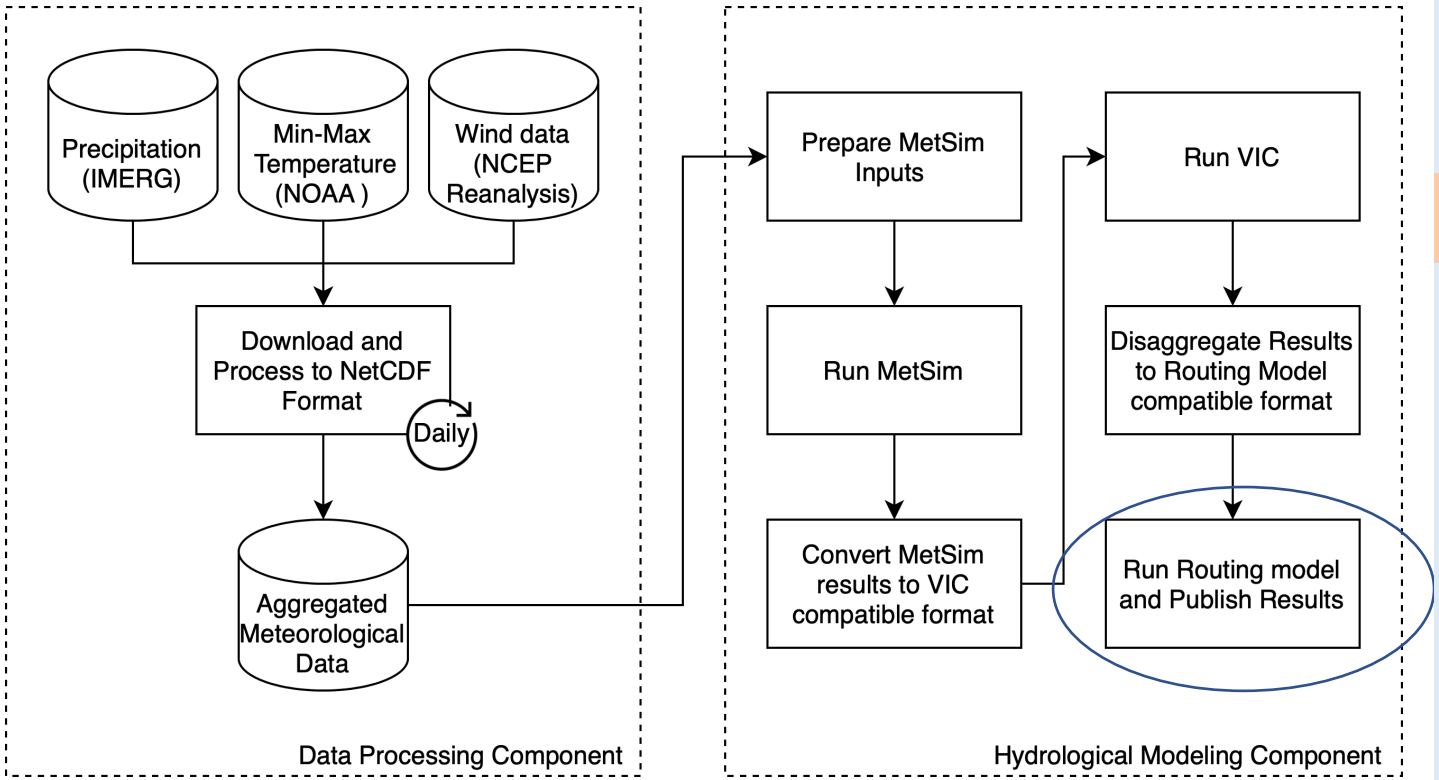


Step Number	Step Name
-1	Reading Configuration settings to run RAT
0	Creating required directory structure for RAT
1	Downloading and Pre-processing of meteorological data
2	Pre-processing of data and preparation of MetSim Input
3	Preparation of MetSim Parameter Files
4	Running MetSim & preparation of VIC input
5	Preparation of VIC Parameter Files
6	Running of VIC and preparation of Routing input
7	Preparation of Routing Parameter Files
8	Running Routing
9	Preparation of parameter files for Surface Area Calculation
10	TMS-OS Surface Area Calculation from GEE
11	Elevation extraction from Altimeter
12	Generating Area Elevation Curves for reservoirs
13	Calculation of Outflow, Evaporation, Storage change and Inflow
14	Conversion of output data to final format as time series

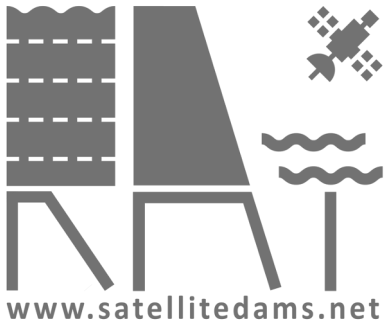
Function: Routing Run

- Updates Routing Parameter file (**route_param.txt)**
- Runs Routing
- Rout Input (**fluxes_<lat>_<lon>**)
- Rout Output (**<station (initial 5)>.day**)
- Inflow (< **station (initial 5)>.csv**)
- Creates Routing Init State for end **(rout_init_state_file<end>.nc)**

- ✓ Runs Routing
- ✓ **Routing Output**
- ✓ Generates **Inflow**
- ✓ **Creates routing initialization state file**

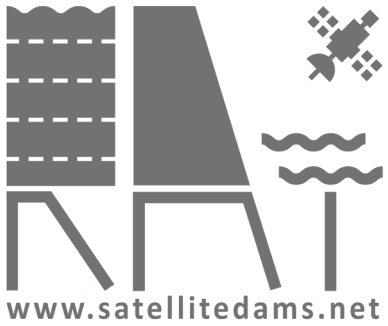


Steps in RAT 3.0 – Step (9)



Step Number	Step Name
-1	Reading Configuration settings to run RAT
0	Creating required directory structure for RAT
1	Downloading and Pre-processing of meteorological data
2	Pre-processing of data and preparation of MetSim Input
3	Preparation of MetSim Parameter Files
4	Running MetSim & preparation of VIC input
5	Preparation of VIC Parameter Files
6	Running of VIC and preparation of Routing input
7	Preparation of Routing Parameter Files
8	Running Routing
9	Preparation of parameter files for Surface Area Calculation
10	TMS-OS Surface Area Calculation from GEE
11	Elevation extraction from Altimeter
12	Generating Area Elevation Curves for reservoirs
13	Calculation of Outflow, Evaporation, Storage change and Inflow
14	Conversion of output data to final format as time series

Configuration Parameters

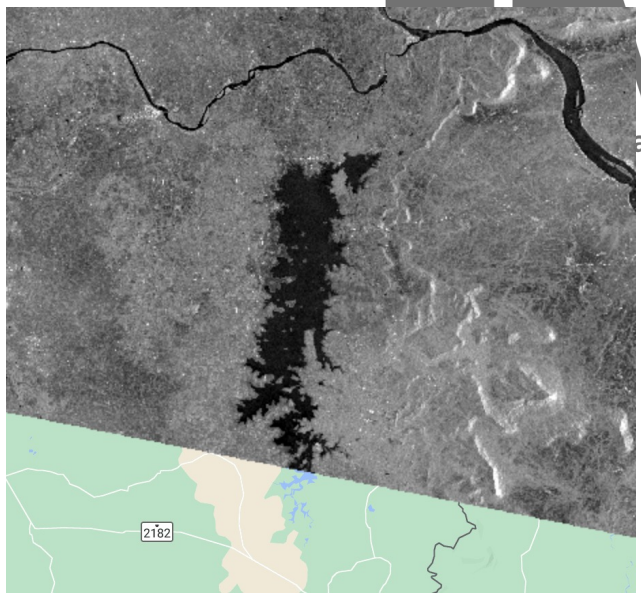


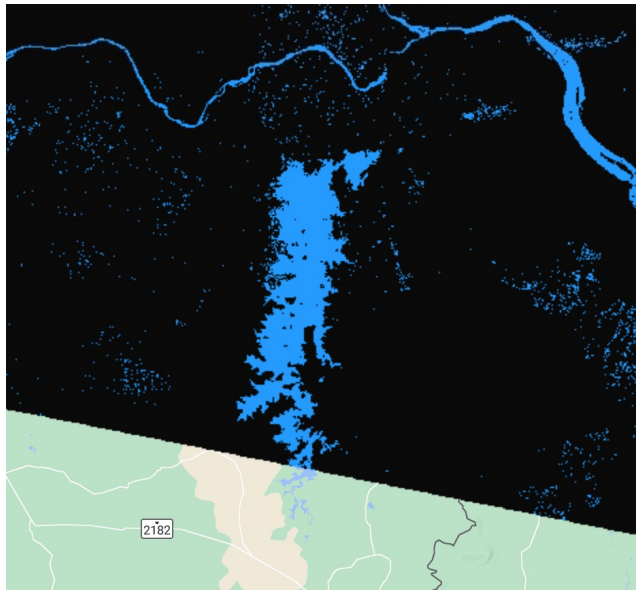
Update parameters in ‘rat_config’ file in ‘Project Directory > params as follows:

```
GEE:

## All reservoir polygons in one shapefile with a column(s) mapping to
station/dams and having dam area in square km
reservoir_vector_file :
## If station_global_data is True, id_column and dam_name_column values should
match with the values of id_column and name_column of stations_vector_file.

## If station_global_data is False, values of dam_name_column should match with
the name column in station_latlon_path or station_file in Routing or Routing
Parameters.
## If station_global_data is False, id_column field is not required below, can
be left blank.

reservoir_vector_file_columns_dict : {id_column : 'GRAND_ID',dam_name_column :
                                      'DAM_NAME',area_column : 'AREA_SKM'}
```

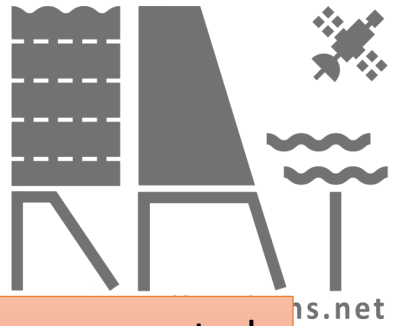
GEE Surface Area

- Google Earth Engine API
- Uses TMS-OS to calculate surface area of a reservoir
- Uses Sentinel-1,Sentinel 2, Landsat 8 & 9
- Do batch wise Extraction (Example 5 days)
- It’s output is surface area timeseries (**<station>.csv**)
- As of now it works for after 2019 time period.





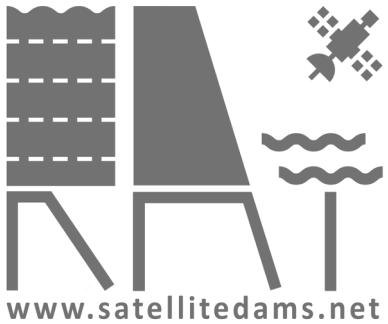
Function: Surface Area Parameters



- ❑ Filters the reservoirs which are within basin
  - ❑ Filtered using <reservoir_vector_file> provided in 'GEE' section
- ❑ Prepares Basin Reservoir Shape File (**basin_reservoirs.shp**)
  - ❑ In Shapefile format

✓ **Surface Area Parameter Files** are created. (if desired)

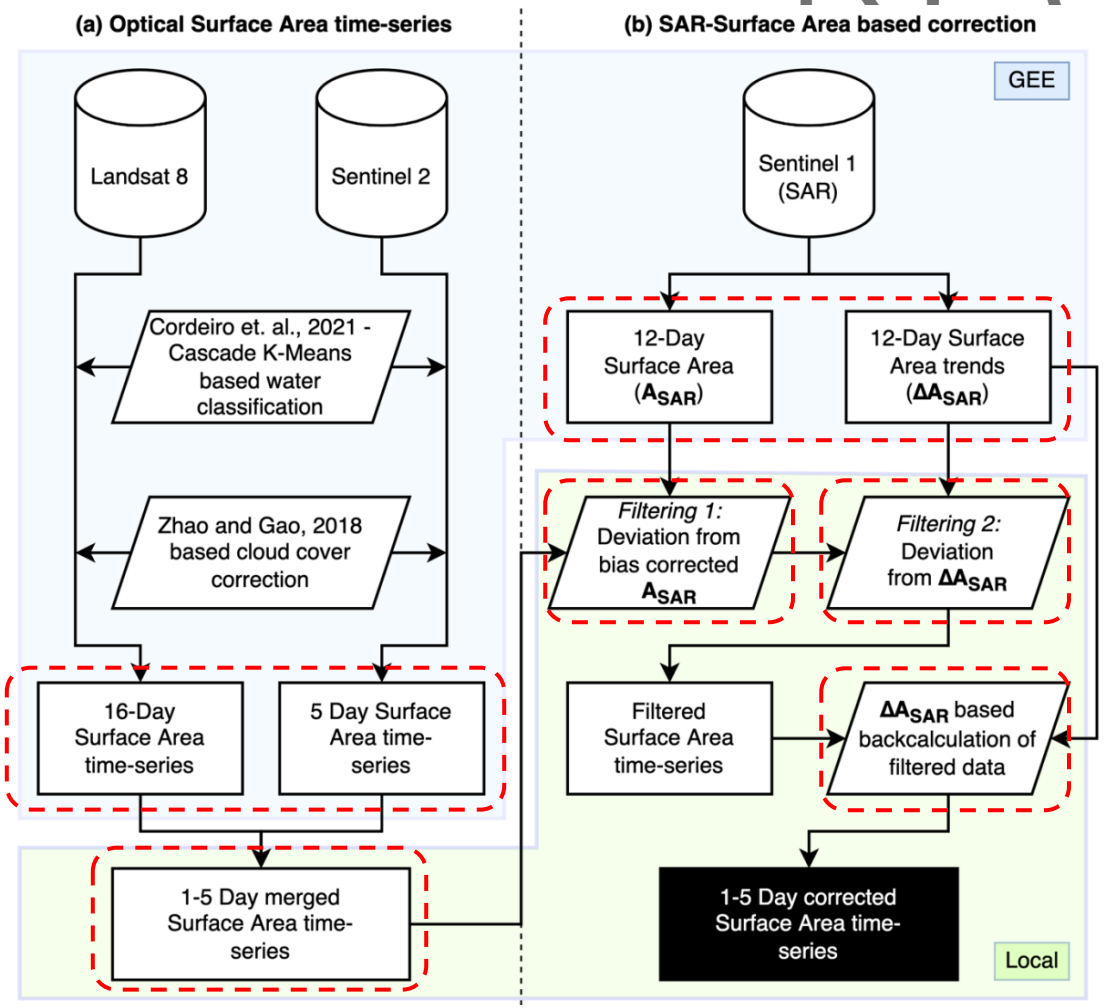
Steps in RAT 3.0 – Step (10)



Step Number	Step Name
-1	Reading Configuration settings to run RAT
0	Creating required directory structure for RAT
1	Downloading and Pre-processing of meteorological data
2	Pre-processing of data and preparation of MetSim Input
3	Preparation of MetSim Parameter Files
4	Running MetSim & preparation of VIC input
5	Preparation of VIC Parameter Files
6	Running of VIC and preparation of Routing input
7	Preparation of Routing Parameter Files
8	Running Routing
9	Preparation of parameter files for Surface Area Calculation
10	TMS-OS Surface Area Calculation from GEE
11	Elevation extraction from Altimeter
12	Generating Area Elevation Curves for reservoirs
13	Calculation of Outflow, Evaporation, Storage change and Inflow
14	Conversion of output data to final format as time series

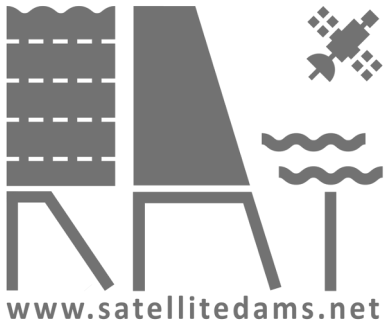
Function: GEE Run

- ❑ Connects to Earth Engine
- ❑ Calculates Surface Area using Sentinel 1,2 & landsat 8,9
- ❑ Applies TMS-OS algorithm
- ❑ Surface area time series Output (**<station>.csv**)



✓ Runs Surface Area Calculation Script
✓ **Surface Area Time Series Output**

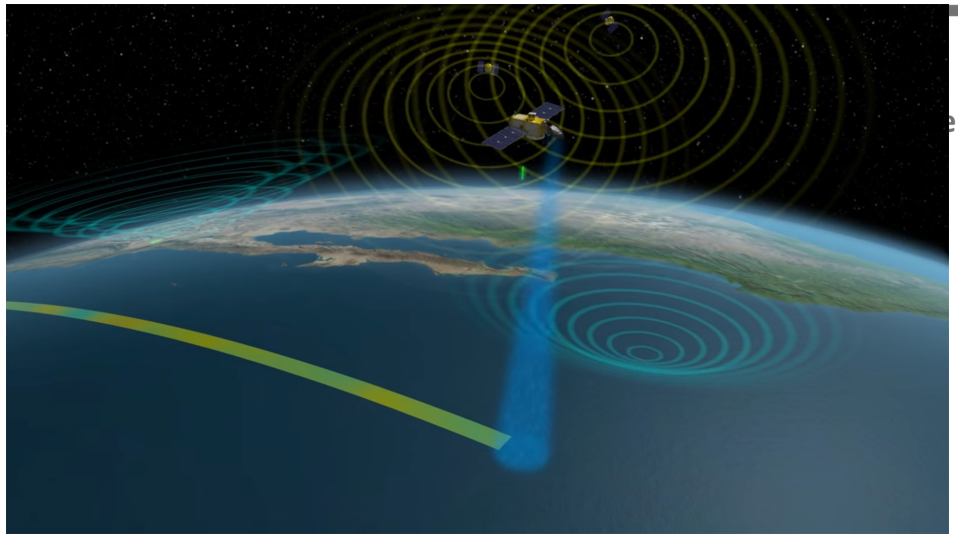
Steps in RAT 3.0 – Step (11)

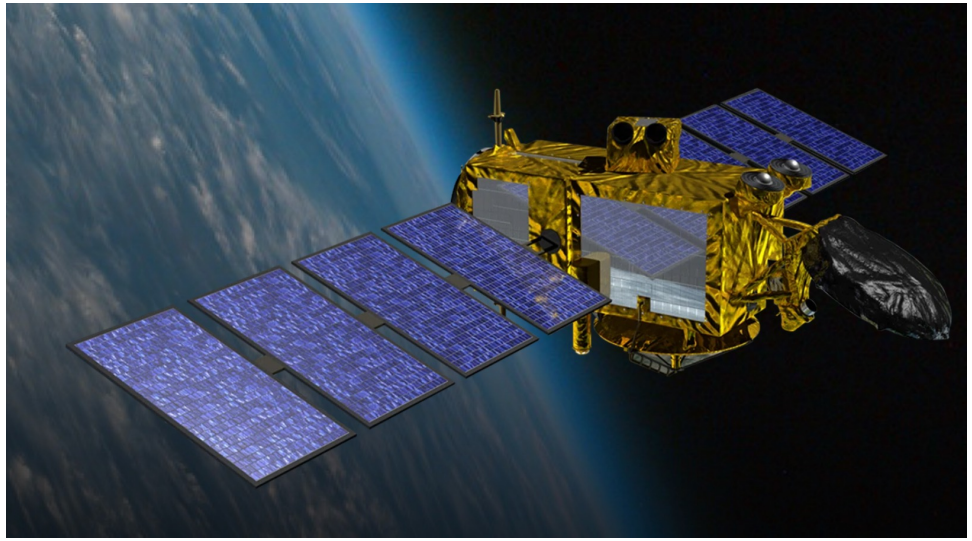


Step Number	Step Name
-1	Reading Configuration settings to run RAT
0	Creating required directory structure for RAT
1	Downloading and Pre-processing of meteorological data
2	Pre-processing of data and preparation of MetSim Input
3	Preparation of MetSim Parameter Files
4	Running MetSim & preparation of VIC input
5	Preparation of VIC Parameter Files
6	Running of VIC and preparation of Routing input
7	Preparation of Routing Parameter Files
8	Running Routing
9	Preparation of parameter files for Surface Area Calculation
10	TMS-OS Surface Area Calculation from GEE
11	Elevation extraction from Altimeter
12	Generating Area Elevation Curves for reservoirs
13	Calculation of Outflow, Evaporation, Storage change and Inflow
14	Conversion of output data to final format as time series

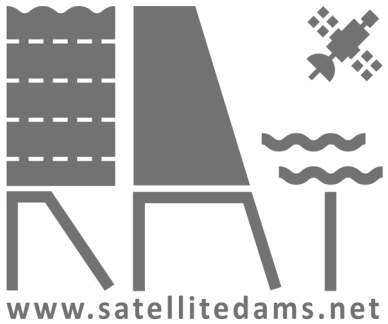
Altimeter Height Extraction

- Radar based technology
  - Active sensor pings the water surface
- Jason-3 data
  - 10 day frequency
- Low spatial Coverage
- Highly Accurate
- Future Missions : SWOT





Configuration Parameters

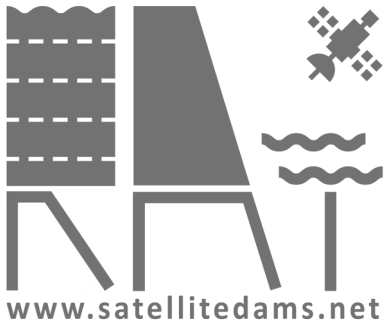


Update parameters in ‘rat_config’ file in ‘Project Directory > params as follows:

```
ALTIMETER:

altimeter_tracks:      ## altimeter track paths in geojson format

geoid_grid:             ## geoid egm grid data in matlab file

last_cycle_number:     226
```

-1
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14

Configuration Parameters

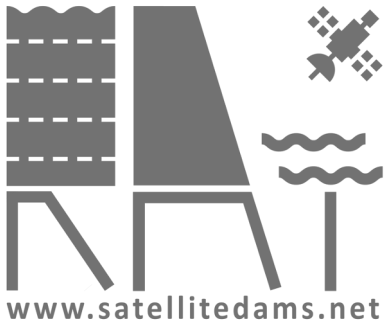


Update parameters in ‘rat_config’ file in ‘Project Directory > params as follows:

```
ALTIMETER:

## Unique ids of reservoirs to process for altimeter for a basin in csv format with column
name 'reservoir_uni_id’

## You can specify min and max latitudes for ROI in this file with column name 'min_lat' and
'max_lat’

## If station_global_data is True, these ids sould match with 'uniq_id' values in
basion_reservoir shapefile generated automatically

## If station_global_data is False, these ids should match with dam_name_column in
reservoir_vector_file

reservoirs_csv_file:

## If you want to use the above file only to specify range of latitude and not the list of
reservoirs to process for altimeter
only_for_range: False
```

-1
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14

Function: Altimetry Run

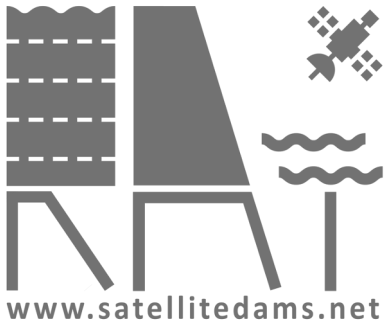


- Downloads raw altimetry data
- Find reservoirs on the altimeter track path
  - Can be provided manually
- Processes data to extract height (depth) of reservoirs
- Outputs depth time series (**<reservoir>.csv**)

✓ Runs Surface Area Calculation Script
✓ **Surface Area Time Series Output**

-1 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14

Steps in RAT 3.0 – Step (12)

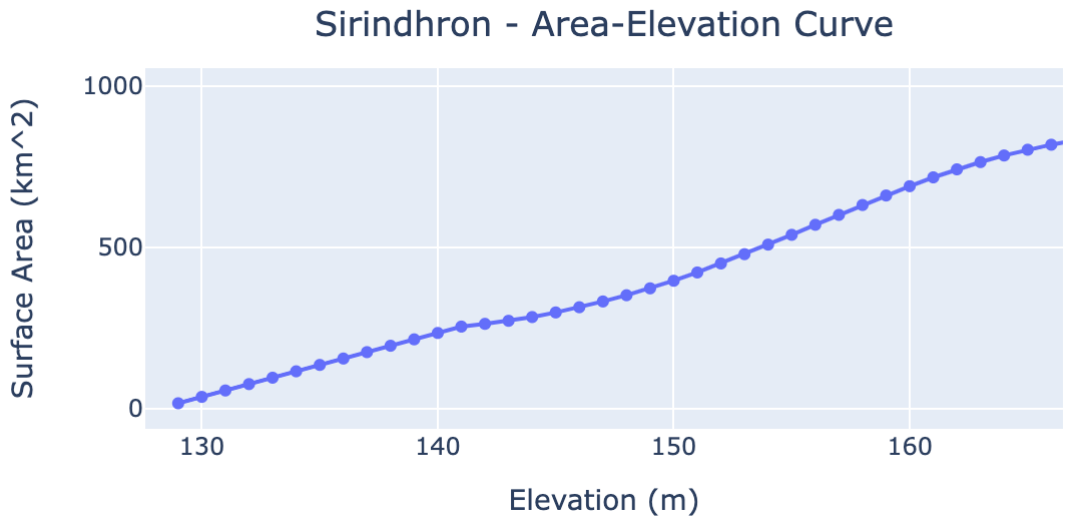


Step Number	Step Name
-1	Reading Configuration settings to run RAT
0	Creating required directory structure for RAT
1	Downloading and Pre-processing of meteorological data
2	Pre-processing of data and preparation of MetSim Input
3	Preparation of MetSim Parameter Files
4	Running MetSim & preparation of VIC input
5	Preparation of VIC Parameter Files
6	Running of VIC and preparation of Routing input
7	Preparation of Routing Parameter Files
8	Running Routing
9	Preparation of parameter files for Surface Area Calculation
10	TMS-OS Surface Area Calculation from GEE
11	Elevation extraction from Altimeter
12	Generating Area Elevation Curves for reservoirs
13	Calculation of Outflow, Evaporation, Storage change and Inflow
14	Conversion of output data to final format as time series

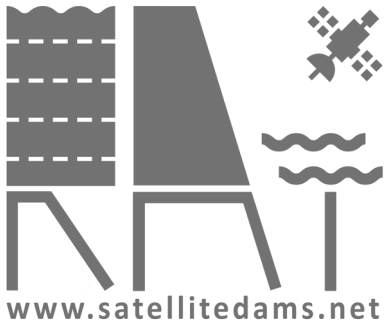
Area Elevation Curve

www.satellitedams.net

- Relation between water level and surface area for a reservoir
- Can be obtained using satellite data
- If possible, get in-situ data



Configuration Parameters



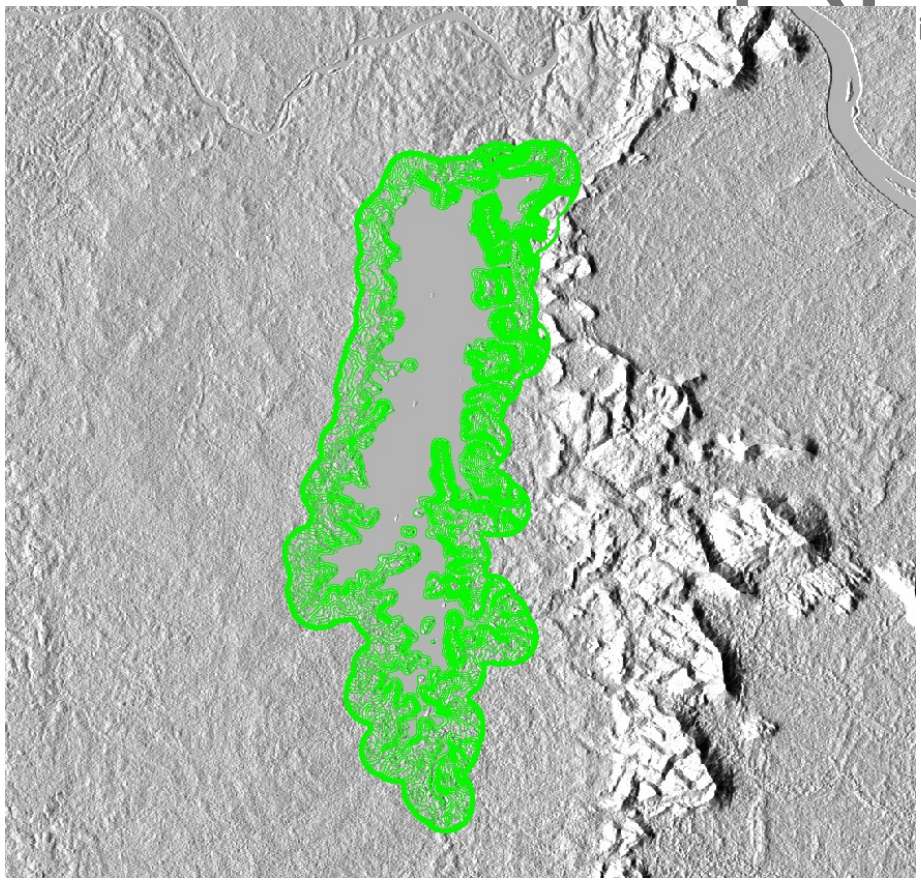
Update parameters in ‘rat_config’ file in ‘Project Directory > params as follows:

```
POST_PROCESSING:

## Folder path containg aec csv files for all reservoirs with the file names should
be unique_identifiers.

## If station_global_data is True, these unique_identiefiers sould match with
'uniq_id' values in basion_reservoir shapefile generated automatically

## If station_global_data is False, these ids should match with dam_name_column in
reservoir_vector_file

## If the folder path is not present or some reservoir aec file is not present, it
will be generated automatically
aec_dir:
```

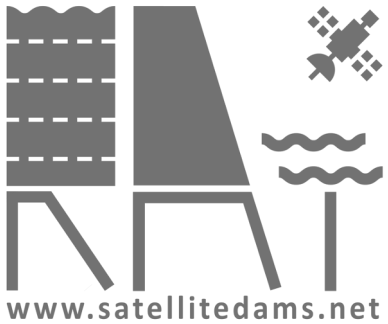
-1
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14

Function: Area Elevation Curve Estimation

- GEE script to extract area elevation curve
  - Uses DEM defined in 'GLOBAL' section
    - SRTM 30 m global resolution
  - Bathymetry gets extrapolated
- Outputs area elevation curve file for each reservoir **(<reservoir>.csv)**



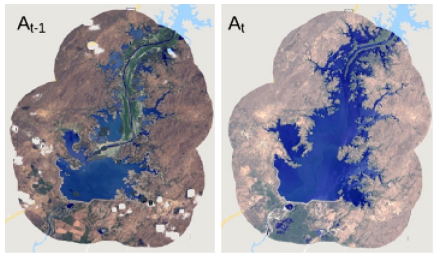
Steps in RAT 3.0 – Step (13)

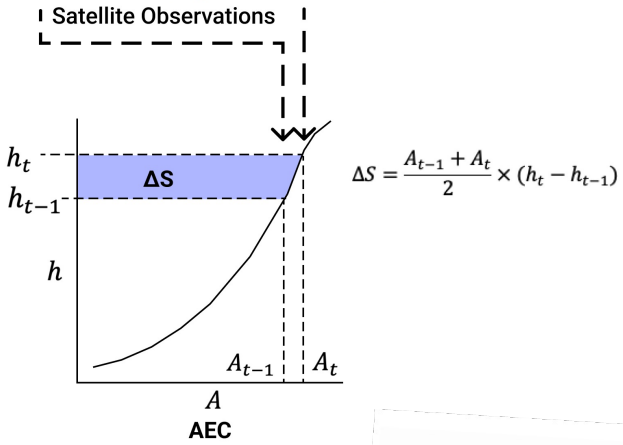


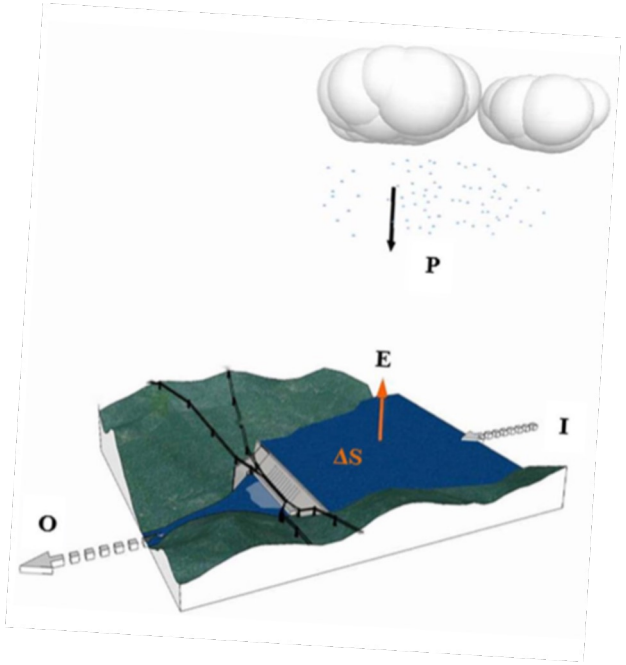
Step Number	Step Name
-1	Reading Configuration settings to run RAT
0	Creating required directory structure for RAT
1	Downloading and Pre-processing of meteorological data
2	Pre-processing of data and preparation of MetSim Input
3	Preparation of MetSim Parameter Files
4	Running MetSim & preparation of VIC input
5	Preparation of VIC Parameter Files
6	Running of VIC and preparation of Routing input
7	Preparation of Routing Parameter Files
8	Running Routing
9	Preparation of parameter files for Surface Area Calculation
10	TMS-OS Surface Area Calculation from GEE
11	Elevation extraction from Altimeter
12	Generating Area Elevation Curves for reservoirs
13	Calculation of Outflow, Evaporation, Storage change and Inflow
14	Conversion of output data to final format as time series

Function: Post Processing

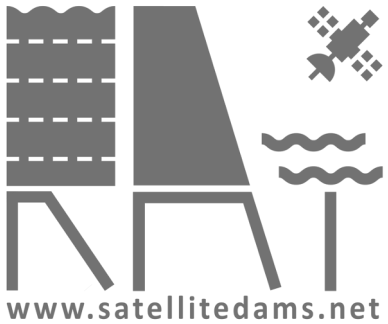
- Storage Change(ΔS) Calculation (Trapezoidal Method)
  - Requires : (i) Area Elevation Curve (**<reservoir>.csv**)
    (ii) Surface Area Time Series (**<reservoir>.csv**)
  - Output : Storage Change (**<reservoir>.csv**)
- Evaporation Calculation (Penman's Equation)
  - Requires : (i) VIC Output – grided surface (**<reservoir>.csv**)
    (ii) Surface Area Time Series (**<reservoir>.csv**)
  - Output : Evaporation (**<reservoir>.csv**)
- Outflow Calculation(Mass Balance Approach)
  - Requires : (i) Inflow (**<reservoir>.csv**)
    (ii) Storage Change (**<reservoir>.csv**)
    (iii) Evaporation (**<reservoir>.csv)**
  - Output : Evaporation (**<reservoir>.csv**)







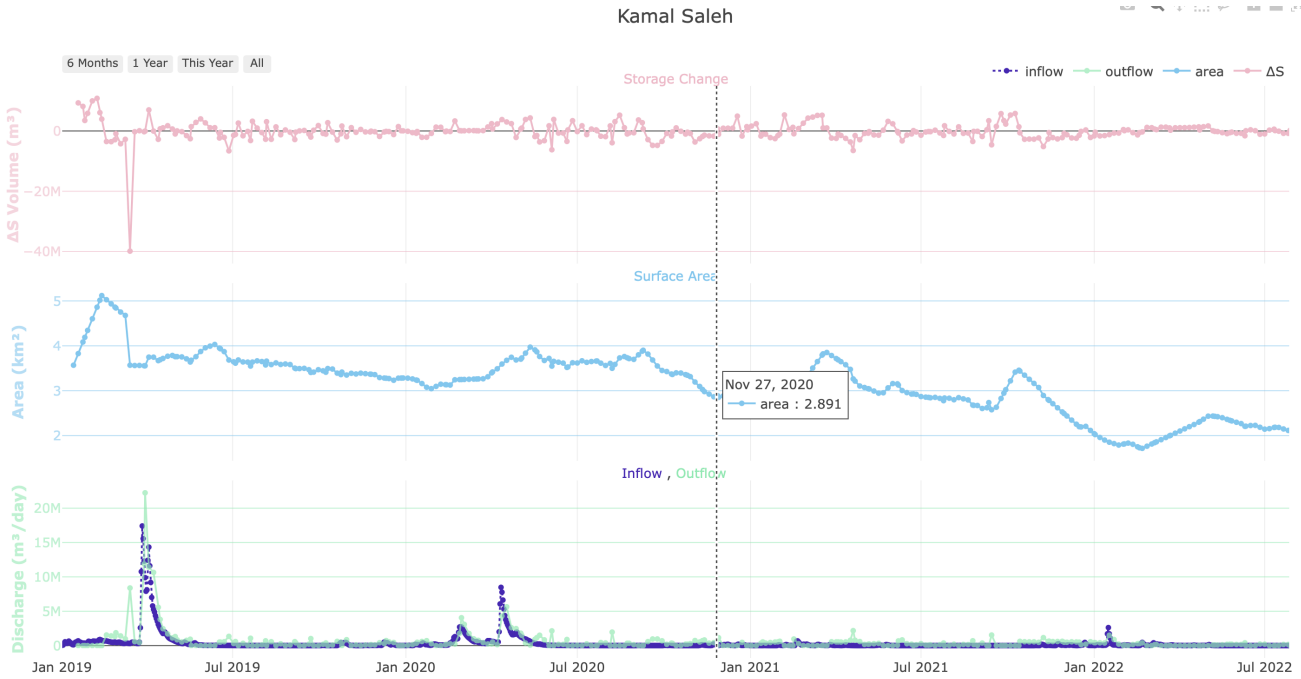
Steps in RAT 3.0 – Step (14)



Step Number	Step Name
-1	Reading Configuration settings to run RAT
0	Creating required directory structure for RAT
1	Downloading and Pre-processing of meteorological data
2	Pre-processing of data and preparation of MetSim Input
3	Preparation of MetSim Parameter Files
4	Running MetSim & preparation of VIC input
5	Preparation of VIC Parameter Files
6	Running of VIC and preparation of Routing input
7	Preparation of Routing Parameter Files
8	Running Routing
9	Preparation of parameter files for Surface Area Calculation
10	TMS-OS Surface Area Calculation from GEE
11	Elevation extraction from Altimeter
12	Generating Area Elevation Curves for reservoirs
13	Calculation of Outflow, Evaporation, Storage change and Inflow
14	Conversion of output data to final format as time series

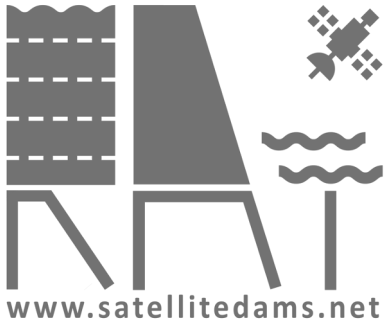
Function: Final Outputs & Cleaning

- Generating Final outputs
  - Website frontend compatible
- Cleans intermediate files that will not be useful next time
  - Saves memory
  - Keeps RAT outputs to append next time



-1
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14

Configuration Parameters



Update parameters in ‘rat_config’ file in ‘Project Directory > params as follows:

```
CLEAN_UP:

clean_preprocessing: False # Deletes pre-processing data except global raw data

clean_metsim: False # Deletes metsim outputs

clean_vic: False # Deletes vic inputs and outputs as well as previous vic_init_states

clean_routing: False # Deletes routing inputs and outputs

clean_gee: False # Deletes unnecessary small chunk files downloaded using gee; not
deleting the combined file, To delete the combined file use clean_previous_outputs

clean_altimetry: False # Deletes raw altimetry data which takes a lot of time to
download; does not delete the extracted altimetry data, to delete extracted altimetry
data use clean_previous_outputs

clean_previous_outputs: False # Deletes previous rat_outputs, routing inflow, gee
extracted area and altimetry extracted heights
```

-1
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14

Multiple Basin Run

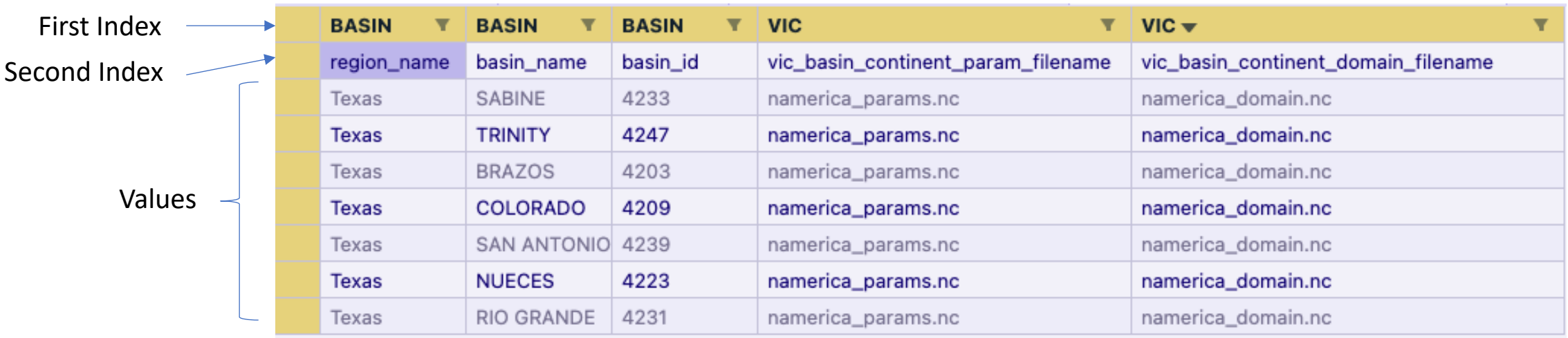
Update parameters in ‘rat_config’ file in ‘Project Directory > params as follows:

www.satellitedams.net

```
GLOBAL:

basins_metadata: # A csv file which is multi-indexed and have same indexing as rat_config.yml
# Only required if multiple_basin_run is true, otherwise ignored

basins_to_process: #example- ['basin1','basin2'] # List of basin names to process if
multiple_basin_run is true, otherwise ignored (must match with index 'BASIN: basin_name:'
defined in basins_metadata csv file)
```

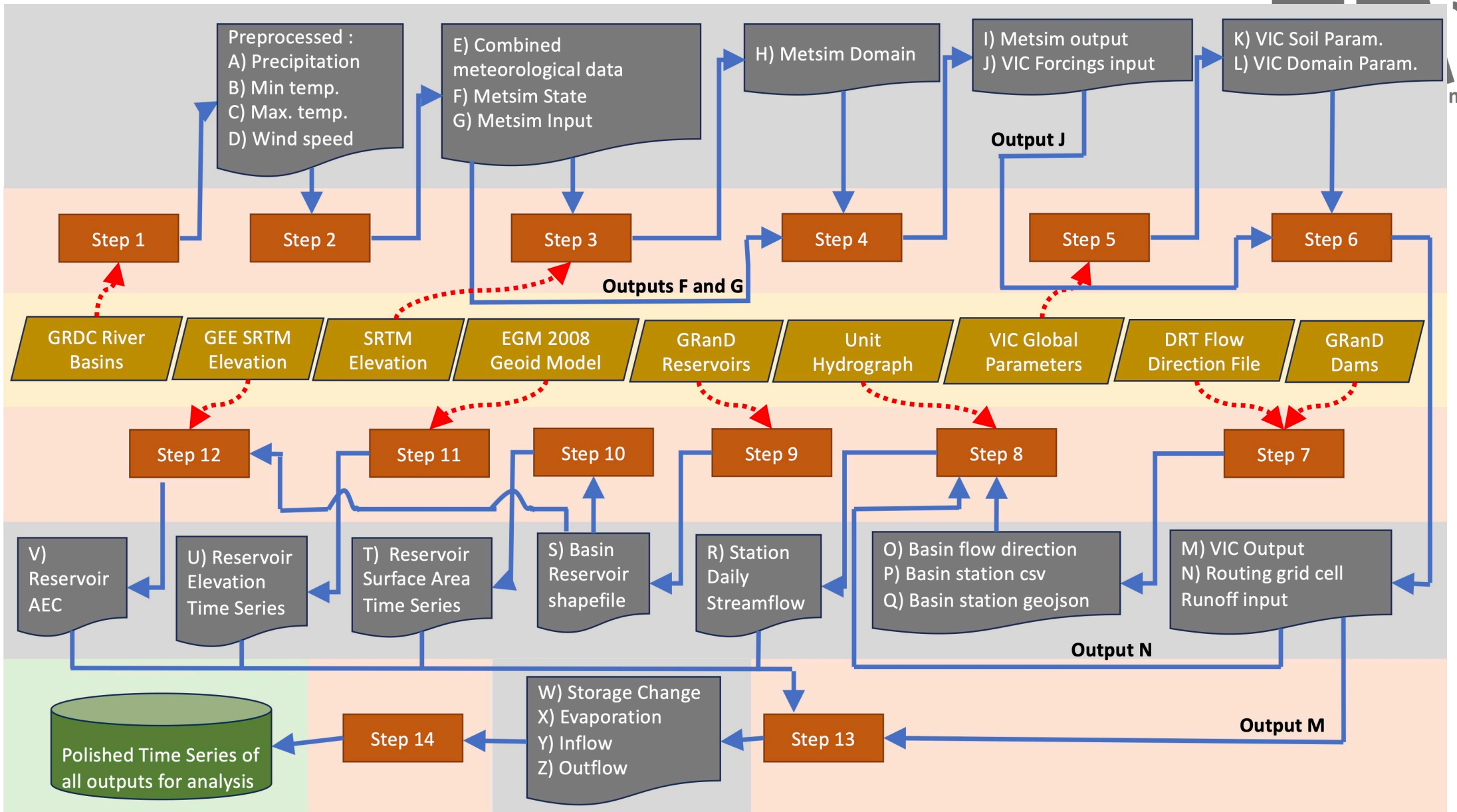
Multiple Basin Run

This is a sample for basins_metadata file



	BASIN	BASIN	BASIN	VIC	VIC
	region_name	basin_name	basin_id	vic_basin_continent_param_filename	vic_basin_continent_domain_filename
	Texas	SABINE	4233	namerica_params.nc	namerica_domain.nc
	Texas	TRINITY	4247	namerica_params.nc	namerica_domain.nc
	Texas	BRAZOS	4203	namerica_params.nc	namerica_domain.nc
	Texas	COLORADO	4209	namerica_params.nc	namerica_domain.nc
	Texas	SAN ANTONIO	4239	namerica_params.nc	namerica_domain.nc
	Texas	NUECES	4223	namerica_params.nc	namerica_domain.nc
	Texas	RIO GRANDE	4231	namerica_params.nc	namerica_domain.nc

Summarized workflow of RAT 3.0



Run RAT 3.0

Command Line

Test RAT using 'rat test' command.

```
rat run      –p ./rat_project/params/rat_config.yml          –o  3
```

Put rat configuration file path

REQUIRED

Number of days for operational latency

OPTIONAL

Should be provided only if RAT has already run once for a particular basin

- ✓ Runs RAT either once or operationally

Python

```
from rat.run_rat import run_rat
run_rat(config_fn = './rat_project/params/rat_config.yml',      operational_latency=None)
```

Thank You

"Let RAT help you TRACK"

www.satellitedams.net