

The code of HYSS and HYPE - overview

HYSS – Hydrological Simulation System

HYPE – Hydrological Predictions for the Environment

Content

1	Introduction to the HYSS and HYPE code	3
1.1	General overview	3
1.2	Overview of the implementation	4
1.2.1	HYSS calculation structure	4
1.2.2	HYPE calculation structure	6
1.2.3	Files and Fortran modules	8
1.3	Variable structure and information management	11
1.3.1	The interface between HYSS and HYPE	11
1.3.2	Global variables.....	12
2	Specific functions of HYSS and HYPE code.....	13
2.1	Input testing.....	13
2.1.1	Structure of the test function	13
2.1.2	Description of available test processes and test cases	13
2.1.3	The test output	14

1 Introduction to the HYSS and HYPE code

1.1 General overview

The HYPE model is a semi-distributed hydrological model for water and water quality. It simulates water and nutrient concentrations in the landscape over time. Its spatial division is related to sub-catchments (in the model called subbasins) and classes (non-located fractions of the subbasin area separated by land use/vegetation, soil type, or elevation). Figure 1 shows an example of a catchment divided into two subbasins, and 4 slc-classes of different colour. Within a subbasin, HYPE simulates different hydrological compartments; snow pack, soil (three layers) including shallow groundwater, rivers and lakes. In addition, it simulates the coupling between subbasins through routing of river flow.

For an introduction to the model and concepts behind it read the Hydrology Research paper Development and testing of the HYPE water quality model for different spatial scales (Lindström, G., C. Pers, J. Rosberg, J. Strömqvist, and B. Arheimer (2010) 41(3-4):295-319). For a thorough and up to date description of the HYPE model get the latest version of the [model description](#).

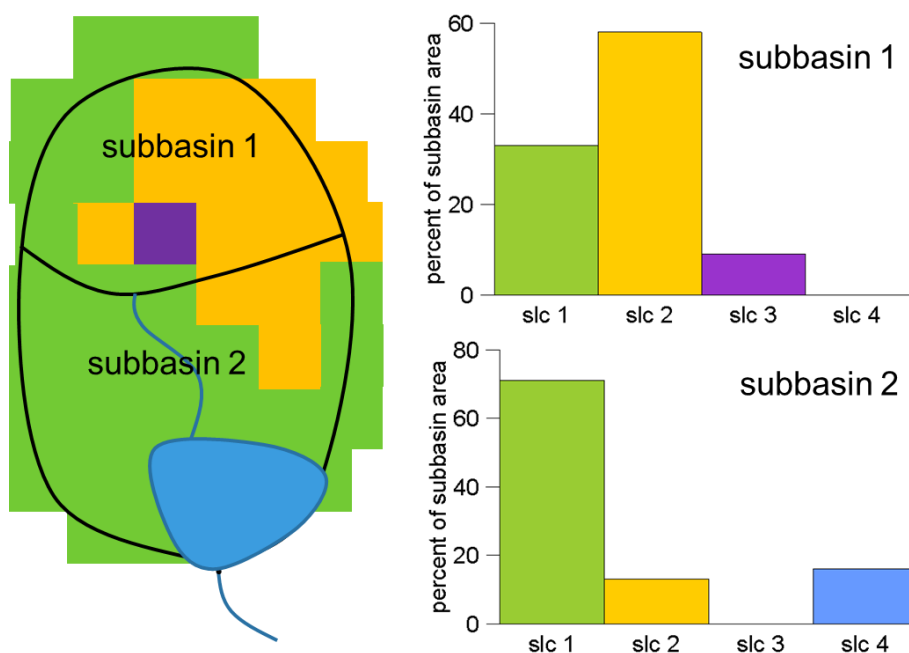


Figure 1. Example of spatial division of a catchment into subbasins and classes (slc).

The hydrological simulation system (HYSS) is used to run the HYPE model. It can be used as the computational framework to run other (hydrological) time stepping models as well. HYSS takes care of input and output to the model. In addition, HYSS provides features for computing evaluation criteria and automatic calibration (optimization) of parameters for the model it runs.

Several performance criteria are calculated in HYSS. Some of these may be included in the objective function to form a multi-criteria optimization. Some criteria are given for each subbasin, while other are collected for the whole simulation. Criteria for separate subbasins include e.g. Nash-Sutcliffe efficiency (NSE) and mean absolute error (MAE). Criteria for whole simulation include e.g. arithmetic mean of NSE over all subbasins.

HYSS features several automatic calibration methods. All are based on optimizing an objective function that is given by the user and composed of some of the available criteria in HYSS. It can be a combination of different criteria and simulated/recorded variables, e.g. NSE for discharge and MAE for lake water level. The calibration methods include among others MonteCarlo simulations with random model parameters within a parameter space and differential evolution Markov Chain method. Directional optimization methods are also implemented, e.g. the Brent method of optimizing one model parameter at the time.

For more information about the HYSS and HYPE code look at the [documentation on the web](#). For information on how to run the model see the tutorial on the wiki '[Quick guide on using HYPE](#)'.

1.2 Overview of the implementation

The model code is written in Fortran 95 and can be compiled for Windows or Linux with gfortran or intel compilers. The source code is published under the GNU Lesser General Public License and publicly available at hypecode.smhi.se. The code is divided into two main parts (Figure 2); the hydrological simulation system (HYSS) and the actual hydrological model (HYPE). HYSS handles input and output routines as well as and time stepping procedures. It also features a set of routines for automated model calibration. The HYPE part of the code contains routines for simulating different hydrological compartments, but also defines parameters and output variables available in the model.

The HYSS simulation system can be used as a computational framework component to run other (hydrological) time stepping models as well.

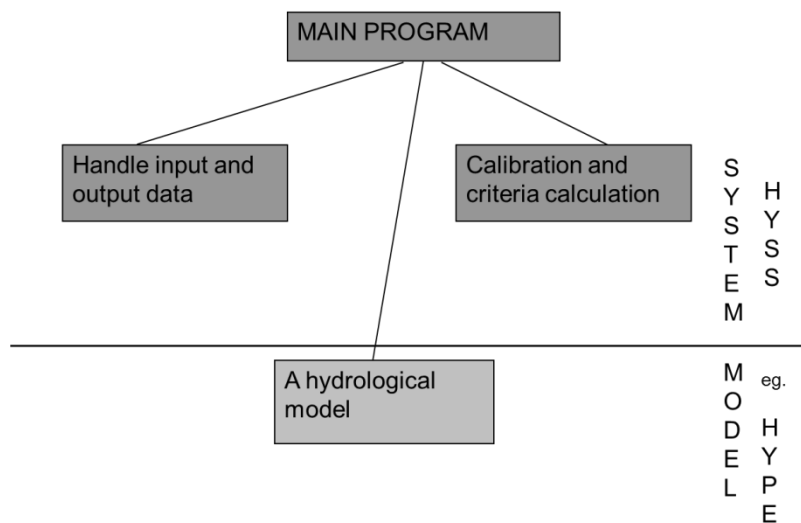


Figure 2. Illustration of the structural division of the code into two parts.

1.2.1 HYSS calculation structure

The Hydrological Simulation System handles input data, both regarding the model set-up and the settings for the specific simulation. Depending on simulation settings different type of output files are written, e.g. time-series of different time steps with several different variables for one subbasin or time-series of one output variable for all subbasins. Several algorithms for automatic calibration of model parameters are implemented in HYSS, as well as a range of criteria that may be combined to a multi-criteria objective function to optimize.

The following are the main computational steps performed by HYSS (Figure 3):

- Read and prepare all input data for the model set-up and simulation. The preparations may include testing of the input data. (INPUT DATA testing)
- Parameter optimization, if that option is chosen. The optimization contains several simulations with different parameters to find the ones that gives the minimum value of the objective function. The result will be one or several sets of (optimal) model parameters. (Optimization loop)
 - Calculate parameters for simulation (New parameters)
 - Simulation with one set of model parameters (Time loop)
 - Get forcing data, may be read from file (Daily FORCING)
 - Simulate the model (MODEL)
 - Accumulate data for performance criteria calculation (Prepare CRITERIA)
 - Calculate value of objective function to determine what next step or the optimization loop will be (CRITERIA)

- Actual model simulation. This includes getting state variables initialized. If optimization has been done the optimized parameters will be used, otherwise the parameter values of input data is used. (Simulation)
 - The model simulation iterates through a series of tasks for each time step (Time loop):
 - Get forcing data, may be read from file (Daily FORCING)
 - Simulate the model (MODEL)
 - Accumulate data for performance criteria calculation (Prepare CRITERIA)
 - Accumulate data for or writing time-series output (Prepare and write OUTPUT)
- Other outputs than time-series files, e.g. subbasin performance criteria and map-files, are calculated and written after the time-simulation-loop is finished (Write RESULT and CRITERIA).

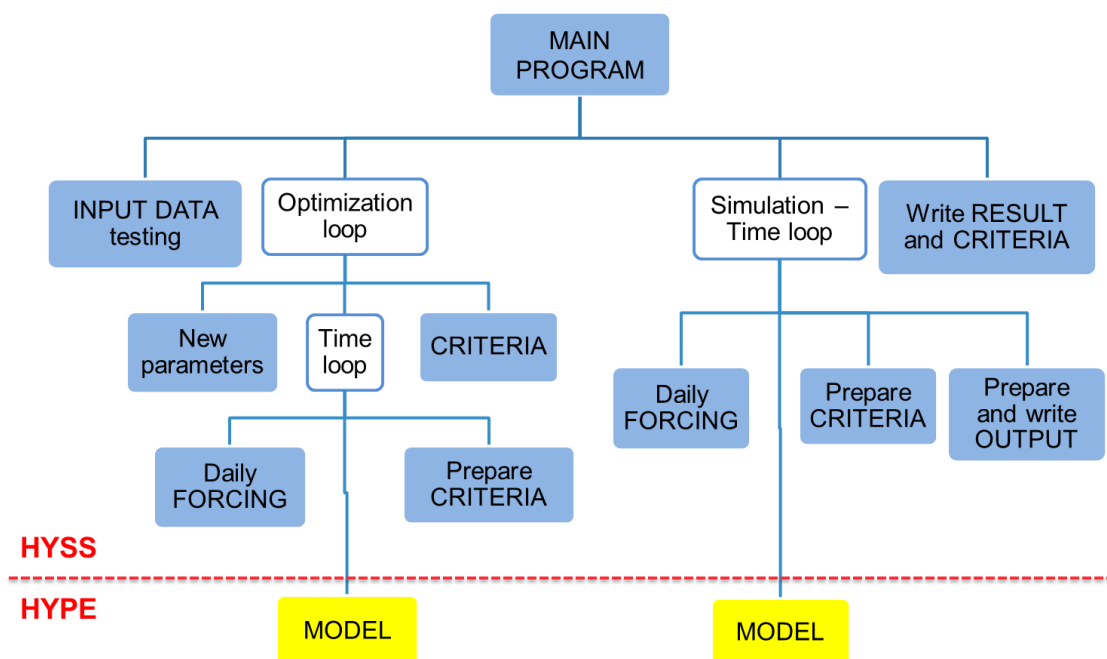


Figure 3. Simplified scheme over HYSS calculation structure.

The criteria calculations are performed in several steps in HYSS. During the time-loop, preparations are made for criteria calculations. These include:

- Accumulation of calculated and observed values to the appropriate criteria time step. Usually the simulation time step and criteria time step is daily, but criteria may be calculated on e.g. monthly accumulated or averaged values.
- Accumulation of sum of simulated and observed values that later will be used to calculate criteria. Also maximum values are accumulated.
- Save all simulated and observed data point pairs. These are needed to calculate some criteria, e.g. Kendall's tau.

Most of the criteria are calculated from sums accumulated during the time simulation loop (second bullet above). The following sums are accumulated for selected variables and all subbasins:

- sum of observed values
- sum of the square of the observed values

- sum of the square of the simulated values
- sum of observed value times simulated value
- sum of difference between simulated and observed value
- sum of absolute difference between simulated and observed value

Criteria are calculated after the time simulation loop is finished. There are two types of criteria calculations; those that are done for each subbasin separately, and those that collect the performance of the whole simulation (all subbasins). If criteria, e.g. Nash-Sutcliffe efficiency (NSE), for separate subbasins are to be calculated (i.e. to written to file), they are calculated first so that they may be used by some of the whole simulation criteria. Criteria for whole simulation are of four general types; arithmetic mean, median, “regional” and “spatial”. Arithmetic mean and median is applied on the criteria already calculated for subbasins. The “regional” criteria type means that all simulated and observed values (from all subbasins) are treated as if they come from one single data set. This is accomplished by summing the sums above for all subbasins before the criterion is calculated. The “spatial” criteria type means that the criterion is calculated with subbasins’ whole simulation period average of simulated and observed data as the data set. Thus here the criterion is calculated for the spatial difference in average value between subbasins. The types may be applied on different criteria, e.g. arithmetic mean of NSE of all subbasins, regional relative error, or spatial root mean square error. Last, the multi-criteria objective function is calculated as a weighted sum of selected whole simulation criteria.

1.2.2 HYPE calculation structure

The calculation order for the HYPE model is primarily based on the water flow path (Figure 4). The water flow through soil and the local runoff from soil are calculated before any of the surface water flows. The soil runoff water continues as local water flow through streams and internal lakes (ilake) within the sub-catchment. The local water flow then feeds into the regional routing through main rivers and outlet lakes (olake) which links individual subbasins within the model domain. The calculations within the local and main river are mostly the same as is the calculations done for the local and outlet lakes. The exception is in the lake outflow; an outlet lake may be regulated by different regulation schemes, while ilakes use the same rating curve.

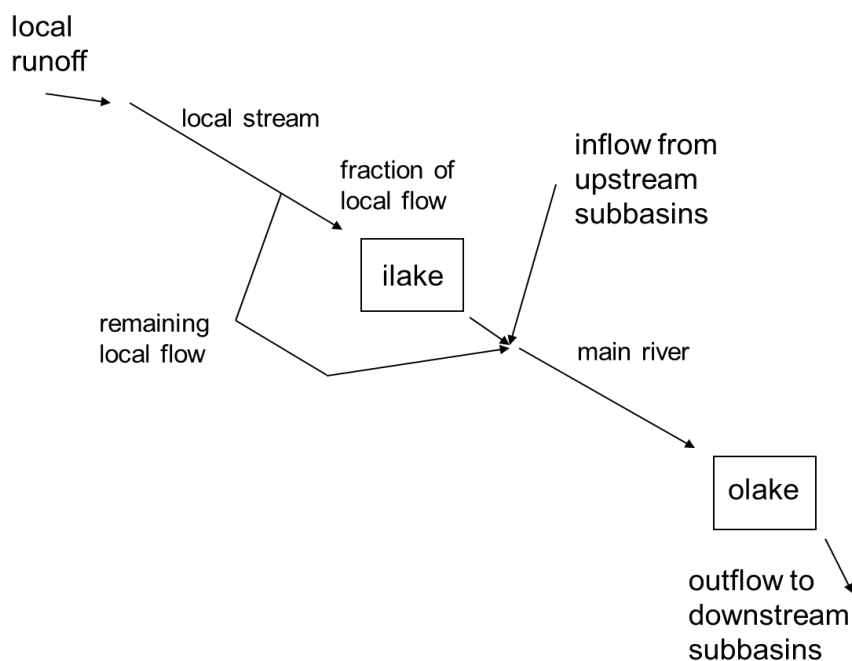


Figure 4. Simplified scheme of water flow within a subbasin. The calculation order starts from soil local runoff via local streams and lakes to the main river and outlet lake, and the coupling of upstream subbasin flow.

The HYPE model subroutine calculates all processes for each time step for one subbasin at a time. The calculation order of subbasins is determined by the water flow order and given in the input data. Thus, the main part of calculations within the model-subroutine is done in a subbasin-loop. Here follows the main computational steps of the MODEL-subroutine for HYPE (Figure 5):

- Calculate the regional groundwater outflow from each subbasin and regional floodplain flows. These flows do not necessarily follow the order of subbasins, and are therefore not possible to calculate within the main subbasin-loop. (Regional groundwater flows)
- Get and/or calculate the current forcing data.
- Main subbasin loop (Subbasin loop):
 - Calculate moving average of air temperatures and daylength for use in the model processes, and set concentration of precipitation.
 - The soil processes are the first processes calculated in the subbasin-loop. More than one soil model exist and they can be used for different classes. All classes with land are calculated. This include glaciers, floodplains and wetlands (Soil calculations, Class loop):
 - The soil processes include for example calculation of snow (Snow), percolation of water in soil, runoff from ground water (Soil runoff) and irrigation, surface runoff, ground water table, denitrification and other nutrient transformation processes in the soil (Transformation). The processes are calculated one after each other.
 - Soil output variables, e.g. soil moisture, are accumulated over classes for calculation of a subbasin average at the end of the class-loop. (Subbasin average)
 - The river network is calculated after local runoff from soil has been calculated. This is referred to as routing. The processes in internal wetlands, local stream and internal lakes are calculated first and then the processes in main rivers and outlet lakes and wetlands. (Routing (river, lake, wetland)):
 - Flows through internal wetland are calculated.
 - Local river inflow, processes and outflow are calculated (River flow, Transformation)
 - Lake inflow, processes and outflow of internal lake are calculated. (Examples: Precipitation and atmospheric deposition, Transformation, Outflow from lake)
 - Main river inflow (including upstream inflow), interaction with floodplain, processes and outflow are calculated (River flow, Transformation)
 - Outlet lake inflow, interaction with floodplain, processes and outflow are calculated. (Examples: Precipitation and atmospheric deposition, Transformation, Outflow from lake)
 - Flows through outlet wetland are calculated, if an outlet wetland is simulated instead of an outlet lake.
 - The calculated outlet lake water level and outflow may be updated (replaced or adjusted) with observations (Updating (QW)). Other updating is also possible.
 - Subbasin average values are calculated and the subbasin results are transferred to output variables available to HYSS. Water balance output may be written to files. (Prepare for output)
- Additional output preparations is done after the main subbasin loop as well as calculation of aquifers (Prepare for output, Aquifer)

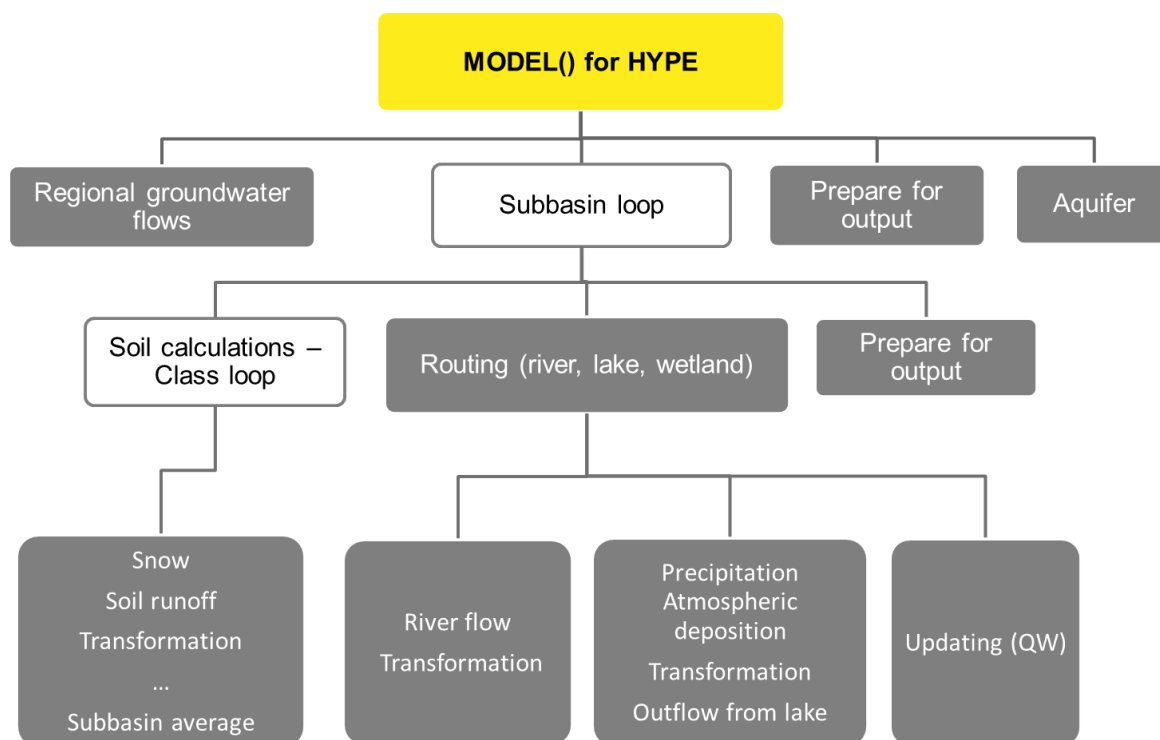


Figure 5. Simplified scheme over HYPE calculation structure for one time step.

1.2.3 Files and Fortran modules

The HYSS/HYPE model code is composed of 30 modules (in separate files) and one main file (Table 1 and 2). 19 modules make up HYPE, one of which is the important `modelmodule` that communicates with HYSS. The modules have a hierarchy dependent on their use (Figure 6).

Table 1. HYSS Fortran modules and files grouped after area of use.

File name	Fortran module name (short name)	Description
Files/modules for running a simulation		
main.f90	-	the file contains the main program for running a simulation
hype_tests.f90	model_test_routines (h_test)	the module contains routines for tests and writing result of tests to log-file
optim.f90	optimization (optim)	the module contains procedures for automatic calibration
Modules for declaring and handling global variables and types		
modvar.f90	modvar	the module contains global variables and types that HYSS provides for the model, and procedures for allocating, setting and getting these variables
worldvar.f90	worldvar	the module contains global variables and types for HYSS, and procedures for allocating, setting and getting these variables, e.g. setting simulation information, handling memory, and getting single observation values
Modules for handling input and output		

compout.f90	compout	the module contains procedures for accumulation output data, for computing output for writing, and for calculation of criteria
data.f90	datamodule (data)	the module contains procedures for observation time-series, for reading and preparing input data, and for saving result files
readwrite.f90	readwrite_routines (readwrite)	the module contains procedures for reading from and writing to file
statedata.f90	state_datamodule (statedata)	the module contains procedures for loading and saving initial states
Modules for general use		
convert.f90	convert	the module contains procedures for transforming between different variable types and between upper and lower case
libdate.f90	libdate	the module contains a variable type for dates and procedures for using this date type
time.f90	timeroutines (time)	the module contains procedures relating to time calculations

Table 2. HYPE Fortran modules and files grouped after HYPE MODEL-subroutine calculation structure (Figure 5).

File name	Fortran module name (short name)	Description
Modules for multiple use		
irrigation.f90	irrigation_module	the module contains variables and procedures for calculation of irrigation
model_hype.f90	modelmodule	the module contains procedures for model calculations during one time step, for initiation of model, for definition of model parameters and output variables, for handling state variables defined in model and procedures for updating
regional_groundwater.f90	regional_groundwater_module (reg_gw)	the module contains variables and procedures for calculation of regional groundwater flow
t_proc.f90	tracer_processes (t_proc)	the module contains procedures for tracer processes (in soil and surface water)
Soil calculations		
atm_proc.f90	atmospheric_processes (atm)	the module contains the subroutine for calculation of precipitation, air temperature and other forcing data for HYPE
glacier_soilmodel.f90	glacier_soilmodel (soil3)	the module contains the subroutine for initiation and calculation the soil processes of glacier class of HYPE
npc_soil_proc.f90	npc_soil_processes (npc_soil)	the module contains procedures for nitrogen, phosphorus and organic carbon processes in soil water of HYPE (the module is also used for Routing (river and lake))
soilmodel0.f90	soilmodel_default (soil0)	the module contains the subroutine for calculation the soil processes of the default soil model of HYPE

soilmodel4.f90	floodplain_soilmodel (soil4)	the module contains the subroutine for calculation the soil processes of the floodplain soil model of HYPE
soil_proc.f90	soil_processes (soil_proc)	the module contains procedures for soil water processes water in HYPE
Routing (river and lake)		
npc_sw_proc.f90	npc_surfacewater_processes (npc_sw)	the module contains procedures for nitrogen, phosphorus and organic carbon processes in surface water of HYPE
sw_proc.f90	surfacewater_processes (s_water)	the module contains procedures for surface water processes in HYPE
update.f90	updating (upd)	the module contains procedures for updating of subbasin outflow, lake water level and nutrient of outflow of HYPE
Modules for general use		
general_func.f90	general_functions (general_func)	the module contains procedures for common equations used by hydrological models
general_wc.f90	general_water_concentration (general_wc)	the module contains procedures for handling water with concentration
hypetypes.f90	statetype_module (hypetype)	the module contains types for state-variables of HYPE and procedure for handling them
hypevar.f90	hypevariables (hypevar)	the module contains global variables for HYPE
Modules for handling input and output		
hype_indata.f90	hype_indata (h_indata)	the module contains procedures for handling and loading input data special for HYPE
hype_wbout.f90	hype_waterbalance (h_wb)	the module contains procedures for handling files for water balance output and saving these results

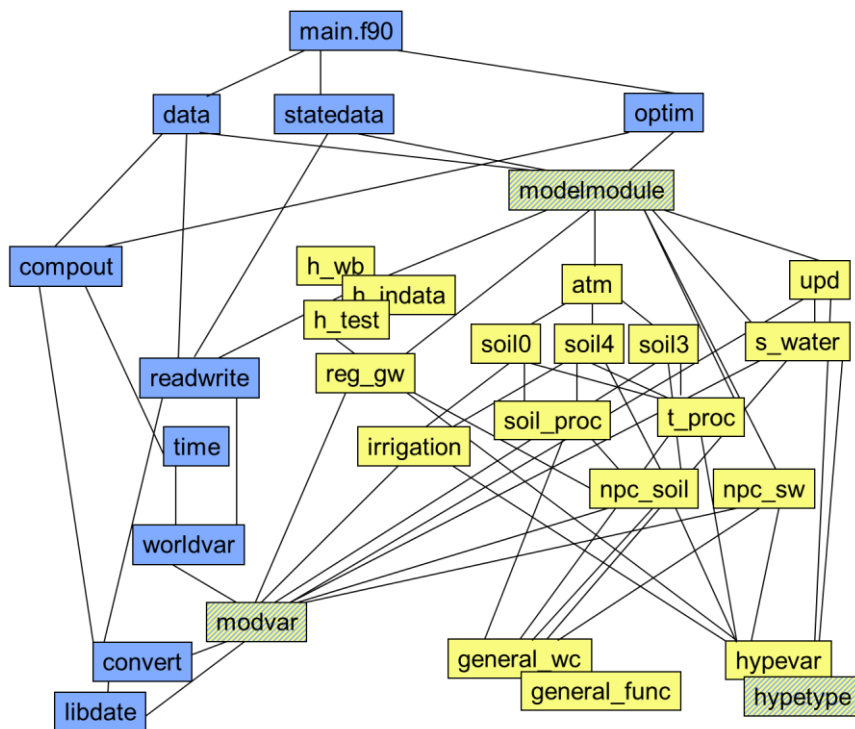


Figure 6. HYSS (blue) and HYPE (yellow) module hierarchy. Explanation of the shortened module names are found in Table 1 and 2.

1.3 Variable structure and information management

1.3.1 The interface between HYSS and HYPE

Three Fortran-modules provide the primary interface between the surrounding simulation system and the model. They are the modules `modvar`, `statetype_module` and `modelmodule`. The module `modvar` holds global variables and arrays that HYSS makes available to the model-programmer, e.g. input data of precipitation and temperature. The module `statetype_module` contains type declarations for state variables. The module `modelmodule` contains the subroutines that the model needs to provide to HYSS, e.g. the subroutine `MODEL` that does all calculation necessary in a time step. In addition the module `readwrite_routines` is available for model defined reading and writing, and the module `model_test_routines` for testing indata for a sane model.

`modvar` provides HYPE with:

- variables for holding settings of the model run, e.g. size of model set-up, updating instructions
- variables and types to hold static input data, e.g. information about subbasins and classes, nutrient point sources, etc.
- variables with data on current time step, e.g. current date, precipitation and temperature
- variables to hold information about all model parameter and values for these model parameters
- variables to hold information about all output variables and values for the output for the current time step
- constants, e.g. the code used for missing value
- subroutines to find information of input data

`statetype_module` provides HYSS and HYPE with:

- six types for state variables that should be able to hold all states for hydrological models
- procedures for allocating, deallocating and initializing state variables
- procedures for initializing a state variables for a submodel
- procedures for transforming states to array for handling states on file

modelmodule provides HYSS with:

- subroutine for calculations during one time step
- subroutines for initial calculations before a simulation starts, e.g. set initial values for state variables and set model configurations
- subroutines for definitions of all model parameters and output variables for the model
- subroutines for initiation of reading HYPE special input and writing HYPE special output.

1.3.2 Global variables

Global variables are declared in the modules worldvar, modvar and hypevariables. In addition the modules irrigation_module, regional_groundwater_module and model_test_routines have module variables that are private in that module. Global variables in hypevariables are used in all HYPE model modules except general_water_concentration, general_functions, statetype_module, hype_indata and hype_wbout. They are not used in any HYSS modules (Figure 7). Global variables in worldvar are used in most HYSS modules and files, but also in the HYPE modules for input and output (Figure 8). Global variables in modvar are used in most HYSS and HYPE modules and files (Figure 8).

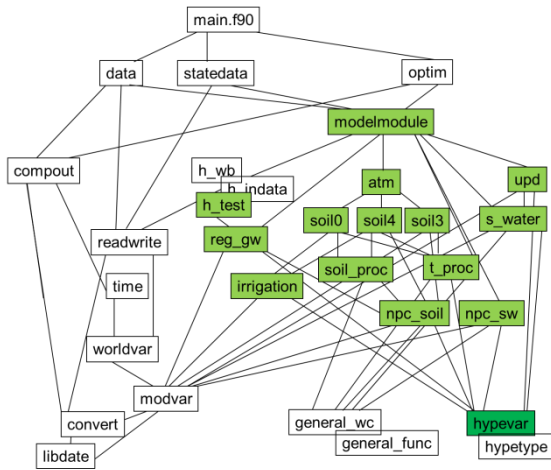


Figure 7. Modules using hypevariables.

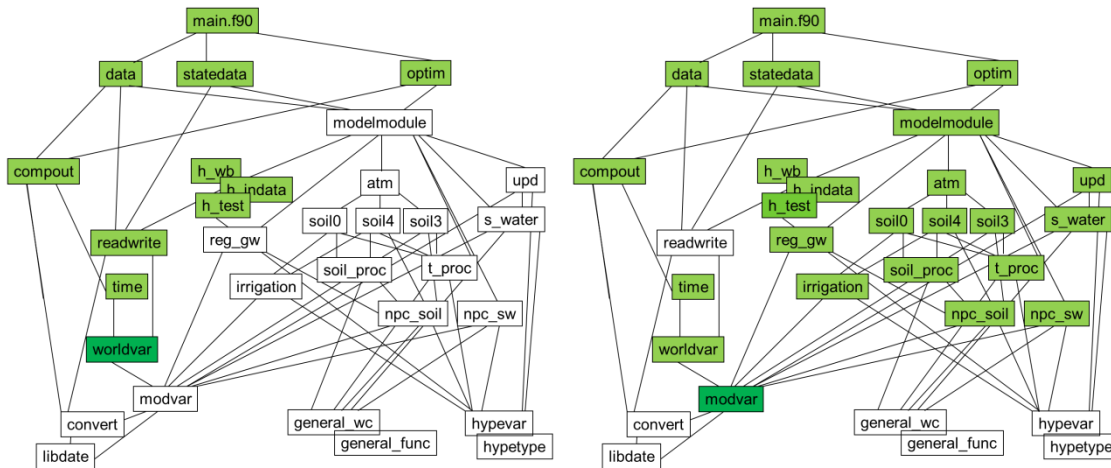


Figure 8. Modules using worldvar (left) and modvar (right).

The module libdate has no global variables, but introduces new types and operators. These are used in all HYSS modules and files, and also in the HYPE modules for input and output (Figure 9).

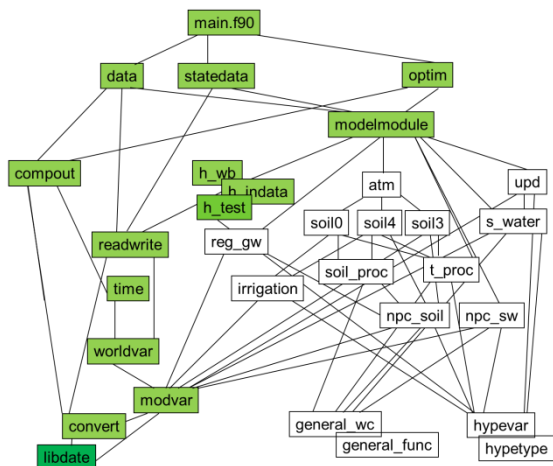


Figure 9. Modules using libdate.

2 Specific functions of HYSS and HYPE code

2.1 Input testing

The HYPE model has a special function for defining and performing tests on input data to a simulation. The test routine will check for common mistakes and mandatory data missing from the model set-up before the actual simulation begins. The routine does not test the complete model and captures only mistakes programmed to be checked for. The tests are only done when asked for by the user.

The results of the tests are written to a special log-file named similar to the hyss-log; test_SEQNR_DATE_TIME.log. The output in the log file can be minimal showing only passed or failed for each test case. More information can be asked for to show what was tested in each test case or extended further to include also tested values.

2.1.1 Structure of the test function

The test function is divided into **test processes** which each cover one aspect of a model simulation. The division is primary grouped under hydrological processes, but generic tests belonging to an input file are also grouped into a test process. Examples of test processes in HYPE are: soil water, observations, generic test for GeoData.txt, glaciers, nutrient sources. A test process contains **test cases** within that area. Test cases can be testing the input for a specific hydrological or biogeochemical process, a specific file or a model option. For example the test process observations have a test case for each observation file, while the test process evaporation has test cases for potential evapotranspiration model and calculation of actual evaporation.

A test case can contain many smaller test cases; e.g. individual parameter values, forcing data and input data may be required for a specific test case to be passed. These individual tests are also **test cases**. A test process can contain up to 64 test cases. Many test cases use the same **tolerance**, e.g. all values > 0. The HYSS/HYPE code therefore has a common function to check pre-specified tolerances. There are four **kinds** of test cases; parameter, input, forcing and generic. A generic kind of test case can be formulated freely and may be composed of the result of several test cases.

2.1.2 Description of available test processes and test cases

Description of test cases can be found on the [web](#) among the description of the HYSS and HYPE code. Below are tabulated current test processes and their major test cases.

Test process	Test cases
Observations	ForcKey.txt, Pobs.txt, Tobs.txt, TMINobs.txt,

	TMAXobs.txt, RHobs.txt, SFobs.txt, SWobs.txt, SWobs.txt, Uobs.txt, Qobs.txt, XobsXOSN.txt (N=0-9), XobsXOMN.txt (N=0-9), Xobs.txt
External tests	Test made during reading of indata files
Generic	modeloption, snow albedo, water holding capacity, regions, soil temperature
Generic – info.txt	dates and period, duplicate variables, significant figures
Generic - geodata	subid, routing, area, sum of area fractions, slope, river length, lake depth, abstraction
Generic - geoclass	landuse, soiltype, specialclass, soildepth
Generic - lakedata	area, lakebasin main routing
Generic - damdata	regvol, inflow, production flow
Generic - mgmtdata	Subids defined
Generic - outregion	outregid
Water – Processes above ground - Evaporation	petmodel, actual soil evapotranspiration
Water – Processes above ground – Atmospheric deposition	atmospheric deposition
Water – Land routines – Soil water	deepgroundwatermodel, infiltrationmodel, soil temperature model
Water – Land routines – Snow	snowfallmodel, snowmeltmodel, snowevapmodel, snowdensitymodel
Water – Land routines – Glaciers	glacier volume-area, glacierinimodel
Water – Rivers and lakes – Rivers	lakerivericemodel
Water – Rivers and lakes – Lakes	swtemperaturemodel, lakes with two outlets
Water – Rivers and lakes – Floodplains	floodplainmodel
Water management – Constructed wetlands	wetlandmodel (not completed)
Nitrogen and phosphorus on land – Nutrient sources	fertilization, plant residues, rural load
Nitrogen and phosphorus on land – Vegetation and soil surface processes	growthstartmodel, plant uptake, erosionmodel, filtering and delay of erosion
Nitrogen and phosphorus on land – Nutrient soil processes	initial values, denitrification, pool transformations, percolation, phosphorus balance
Nitrogen and phosphorus on land – Nutrient leakage from outer source	soilleakage

Table. All current test processes and their major test cases.

2.1.3 The test output

The result of the test is collected in the test-log file. At the top of the file is given the settings used for the tests. The rest of the information is grouped after test processes. If some test failed for a test process the number of failures within the test process is printed after the other test process information. In the end of the file the total status of the test is printed. The file may also include information (INFO:) and warning (WARN:) output.

The test output can be had with three levels of information. The lowest level only tells the result (passed or failed) for selected test cases of each test process. The middle level will in addition present if each test case under a test process has passed. The third level will include tested values for some test cases. This output in the test-log is different depending on kind of test case; for parameters all values are printed, for input the range is printed.

When a simulation is stopped due to error or failed tests, sometimes the results from external tests are printed an extra time at the end of the test-log file. This is done to be sure the information is not lost if the failure happened before it had been written.