



COMMUNITY MODELLING WITH MESH AND UPDATES ON THE SASKATCHEWAN BASIN SETUP

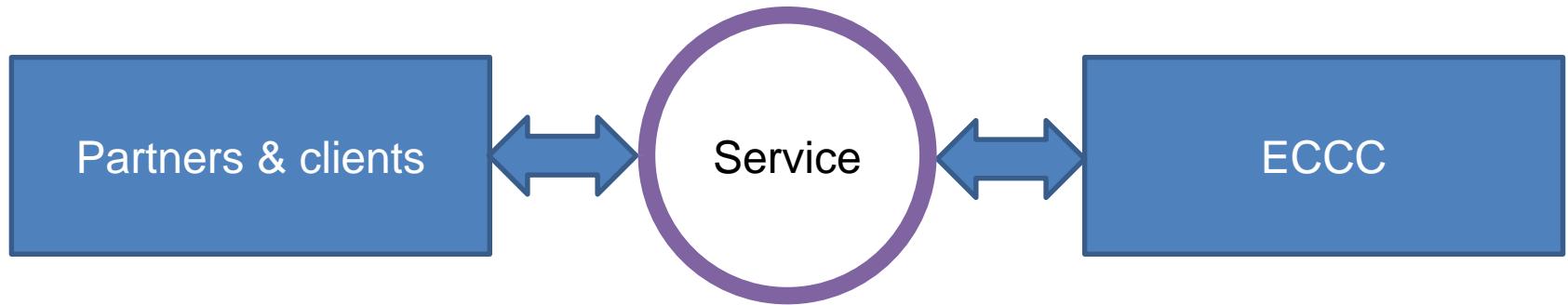
- The Community
- "MESH" model framework
- The SK basin setup

PPWB – 50 Years
Edmonton, AB
November 27, 2019

Dan Princz, Anthony Liu,
Fuad Yassin, Bruce Davison



THE COMMUNITY

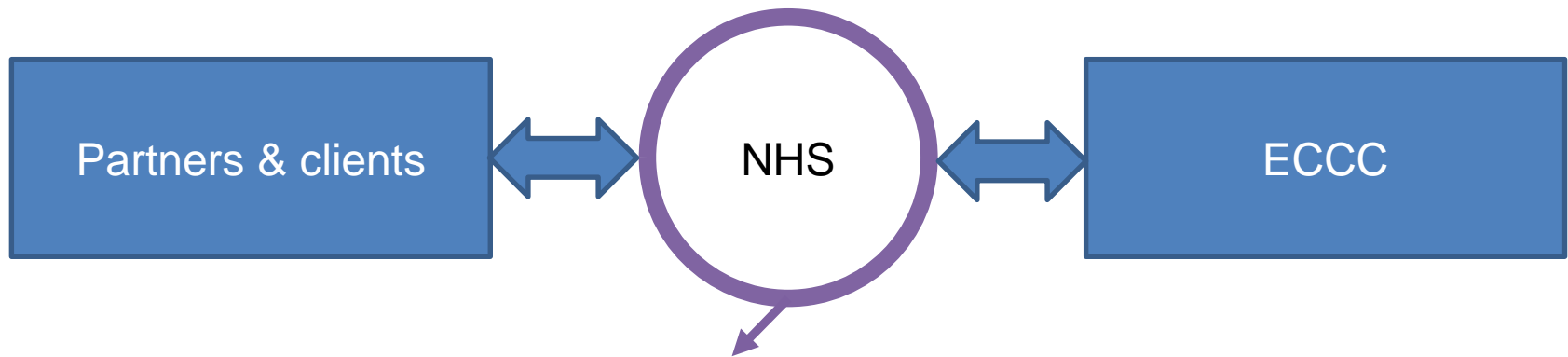


- Global Water Futures
- Other Universities
- P/Ts (in progress)
- PPWB COFF

National
Hydrological
Service

- CCMEP (Montreal)
 - CCCma (Victoria)
 - HAL (Edmonton)
-

NHS SERVICE - LIAISING

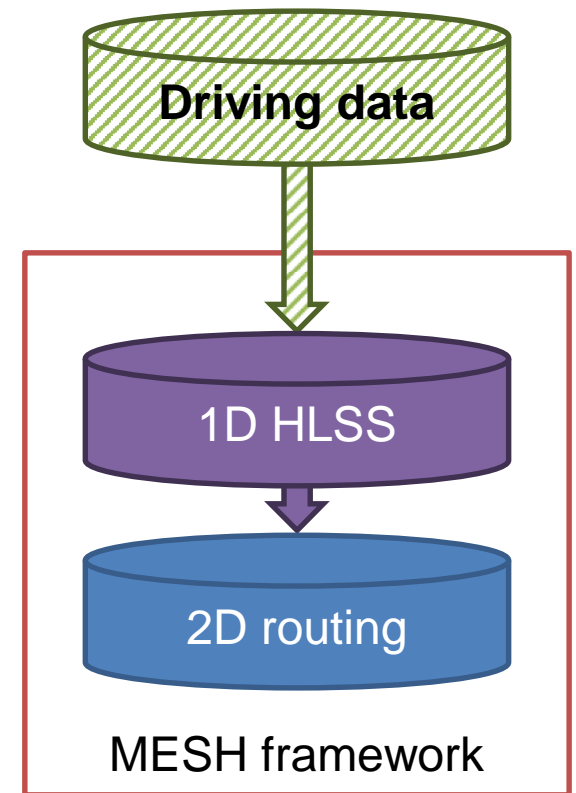


- Facilitates coordination between groups
- Builds relationships
 - In particular, trust with respective groups when implementing and testing experimental codes (e.g., research implemented in community codes yet to be published; internal experimental codes yet to be formally assessed)
- Has capacity to provide general maintenance and/or servicing of community-facing models/framework
- Oversees the migration of codes (i.e., knowledge transfer) between experimental branches, and between community-facing and internal-ECCC-facing versions of codes

MESH MODEL FRAMEWORK

MESH: "Modélisation Environnementale communautaire – Surface Hydrology"

- Flexible, accessible & portable modelling framework
- Most applications:
 - 1D Hydrology-Land Surface Scheme (HLSS) + 2D distributed hydrological routing
 - Optionally: Glacier dynamics; deep soil profile (e.g., permafrost in changing climate); reservoir routing (via simple or complex parameterization, zone-based managed reservoir model, or using volume-elevation-discharge tables); district-based irrigation and diversions; cold regions hydrology, including snow distribution (spatial); variable runoff contribution for prairie pothole areas (statistical parameterization)



MESH MODEL FRAMEWORK

- Community practice for model development
 - Relatively lightweight
 - Decoupled from internal ECCC infrastructure
 - Compatible with open-source compilers (GCC)
 - Scalable to run on laptops up to computing clusters
 - Compatible with Linux, Windows, and MacOS (multi-platform)
 - Many applications
 - Academia (often in coordination with ECCC): Hindcasting, process refinement/development, future climate scenarios
 - Provinces & territories: Undergoing assessment, largely via collaboration with academic partners
 - ECCC – CMC: process refinement/development; NHS: Hindcasting, regional future climate scenarios
-

MESH MODEL FRAMEWORK

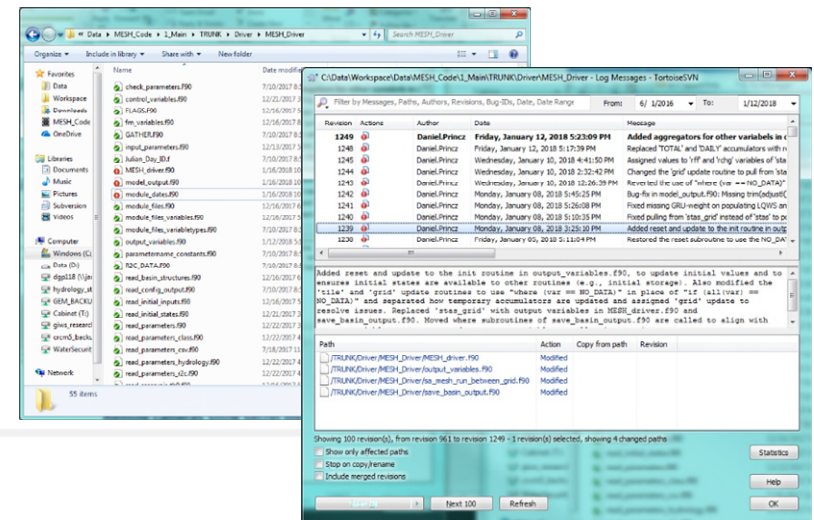
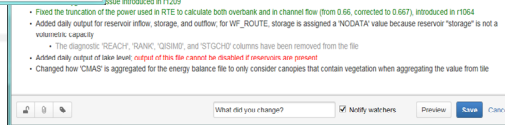
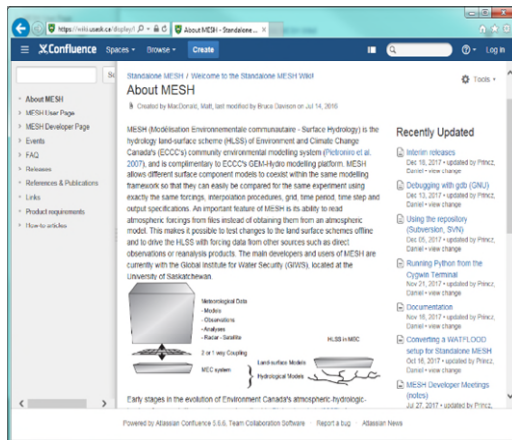
Community accessible development and support tools

Wiki

- Presently hosted by University of Saskatchewan
- Open-access for viewing
- Controlled access to contribute and for experimental developments

Code and script repository

- Open-access to download
- Controlled access to contribute and for experimental developments



MESH MODEL FRAMEWORK

Active collaborators



UNIVERSITY OF SASKATCHEWAN
Global Institute for
Water Security
USASK.CA/WATER



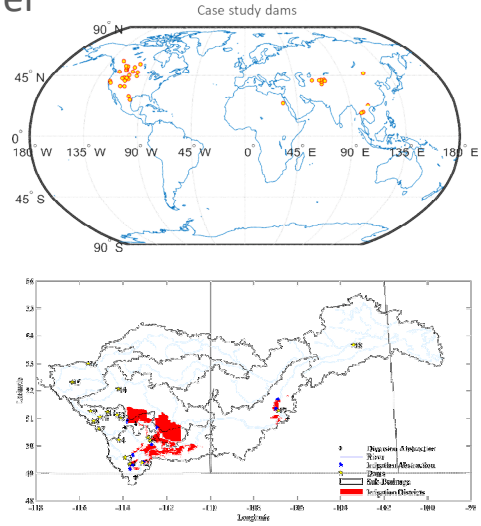
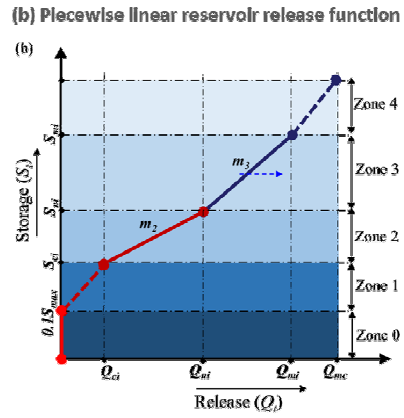
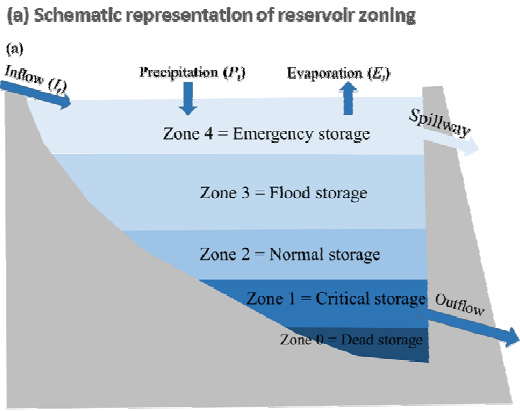
Environment and
Climate Change Canada
Environnement et
Changement climatique Canada



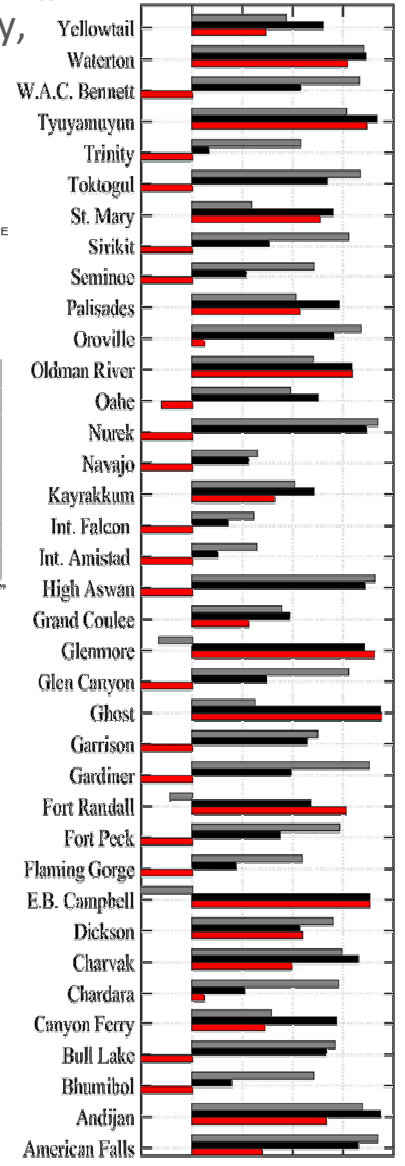
Dynamically Zoned Target Release (DZTR) model

Research → Operations, via coordination with NHS/CMC and research community, and implementation via community-facing MESH model

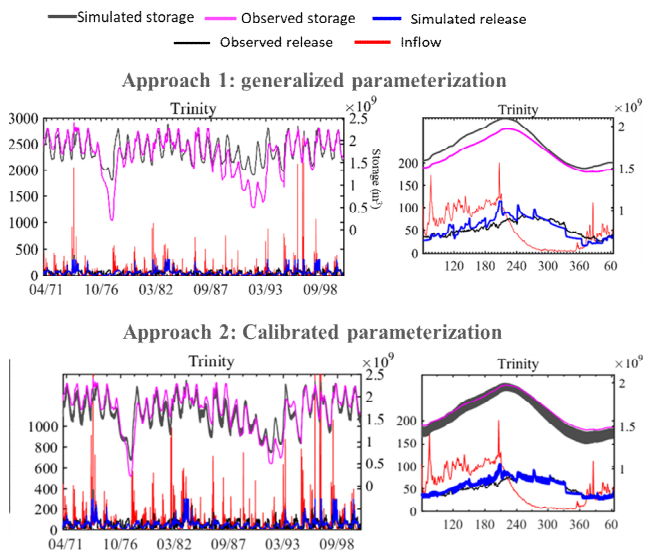
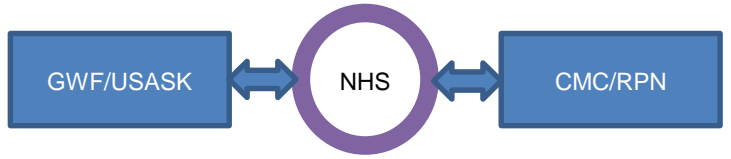
■ No-reservoir assumption ■ DZTR flow ■ DZTR storage



Model has three monthly release parameters Q_{ct} , Q_{nr} , and Q_{ml} and three monthly storage zone parameters S_{ct} , S_{nr} and S_{ml}



- Implemented as standalone model
- Added to MESH, connected to routing code
- Ported to GEM-Hydro (same routing code as in MESH)
 - Undergoing testing
 - May eventually support TB flow forecasting deliverables



NSE

SASKATCHEWAN BASIN SETUP

- ECCCC/SRB model and tools (Anthony Liu)
 - Borrows USASK developed SRB model (Yassin, 2019)
 - Borrows GWF developed forecasting workflow (YT)
 - Uses MESH within a scripting environment
 - Runs decoupled from operational systems in a forecast mode (driven by ECCCC products from DataMart)
 - Runs HRDPS/RDPS/GDPS forecasts; looking to include GEPS
 - Dynamic visualization of model outputs
-