Report P5C Flood Data Services

Prepared for Project 0-7095-01 Flood Assessment System for TxDOT (FAST)

By Andy Carter and David Maidment
Center for Water and the Environment, University of Texas at Austin

Attila Bibok and Matt Ables
KISTERS North America, Inc.

31 July 2026

1. Introduction

Task 5 focuses on building and running the Flood Data Services that power the FAST map layers. These services produce three key outputs: (1) roads affected by forecasted flooding (2) bridges at risk during those conditions, and (3) forecast flood inundation extent. These are the three statewide operational product types. The development environment also demonstrates low-water-crossing status for a limited sample in the Austin district. The system operates automatically by pulling in federal flood forecasts and converting them into map-ready data. To support this, the team designed and deployed three connected components:

(a) databases preloaded with flood extents and impacted assets

(b) backend services that turn forecasted flows into flood warning layers using these databases

(c) web-based viewers that allow users to explore the results in real time.

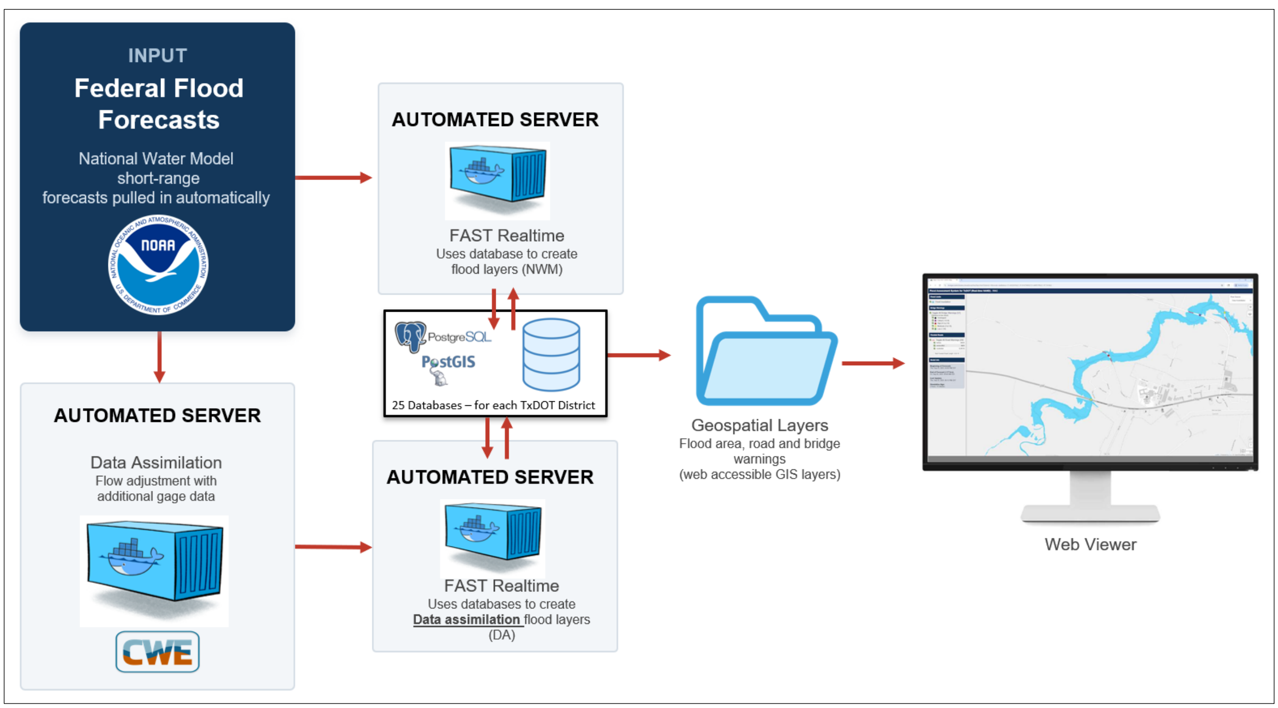


Figure 1. FAST flood data services overview

2. Database

The FAST system's data infrastructure is built using PostgreSQL, an open-source relational database, enhanced with PostGIS to support spatial queries and geographic data types. This section describes the PostgreSQL 17.4 database that supports the FAST flood forecasting system. The database contains twenty (20) tables. Thirteen (13) of these include map information (geometry). Because much of the hydraulic flood mapping work is pre-computed, the FAST system can respond quickly during a forecast by finding and displaying the correct information instead of running new flood simulations every time.

The database is organized into four main groups. The first group contains permanent (static) reference information, including roadway locations and bridge data. The second group stores flood maps that have already been created for each stream reach at multiple flow levels. The third group contains connection tables that link flood model information to roads and bridges and identify the flow level where flooding begins. The final group contains the products created during each forecast cycle, including flood maps, flooded road segments, and bridge warning layers , and low-water-crossing status points demonstrated with the limited Austin sample.

A key design choice in FAST is that it uses stored flood results instead of running a new flood model during each forecast. The system receives predicted stream flows, finds the expected peak flow for each location, and then selects the matching flood area from the database. This approach allows forecasts to be processed quickly because the most time-consuming flood calculations have already been completed.

A FAST database was created for each of the twenty-five (25) TxDOT districts and populated with the permanent (static) data for that district. Creating separate databases for each district allows flood mapping and data management to be handled at a regional level.

PostgreSQL database dumps for each TxDOT district are publicly available at:

https://web.corral.tacc.utexas.edu/nfiedata/acarter/roadflood-db-20250702/

Together, the complete set of FAST databases for all twenty-five TxDOT districts represents approximately 63 GB of data.

Figure 2. FAST database technology stack

2.1 Database Environment

The database runs on PostgreSQL 17.4 with the PostGIS extension installed. All twenty tables and their supporting index objects live in the public schema. A one-letter prefix separates spatial from tabular content, and a two- or three-letter suffix on spatial tables declares the geometry class.

Table 1. Object naming convention

Element	Meaning	Examples
s_ prefix	Spatial table; carries at least one geometry column. Thirteen tables.	s_road_segment_ln, s_bridge_pnt
t_ prefix	Tabular table; no geometry. Seven tables.	t_flow_forecast, t_nextgen_to_nwm
_pnt suffix	Point geometry	s_bridge_pnt, s_bridge_warning_pnt, s_lwc_pnt
_ln suffix	Line geometry	s_road_segment_ln, s_flood_road_trim_ln
_ar suffix	Area (polygon) geometry	s_flood_inundation_ar, s_flood_merge_ar
idx_ prefix	Index. Names generally follow idx_<table>_<column>.	idx_s_flood_road_trim_ln_tile_id

Every typed geometry column in the schema uses EPSG:4326 geographic coordinates on the WGS 84 datum. A distinction to hold in mind with this schema is how long a row is expected to live. Seven tables are persistent (static) datasets. The other thirteen tables are computed as each new flow forecast from the National Water Model is made available (dynamic).

Table 2. Lifecycle classification of the twenty tables

Lifecycle	Tables	Refresh trigger
Static	s_road_segment_ln, s_bridge_pnt	New road linework and/or new bridge points
Static	s_flood_inundation_ar	New flood inundation with hydraulic modeling
Static	t_nextgen_to_nwm	Change to hydrofabric
Static	t_road_flood_trigger, t_bridge_rating_curve	Change to hydrofabric, new terrain, road elevation model, road lines or new hydraulic modeling
Static	s_lwc_static_pnt	New or revised crossing geometry, attributes, hydrologic linkage, or flow thresholds
Dynamic — forecast state	t_current_forecast, t_flow_forecast, t_flow_per_nextgen, t_max_flow_per_nextgen,	Every National Weather Service forecast cycle

Lifecycle	Tables	Refresh trigger
Dynamic — products	s_selected_flood_ar, s_flood_merge_ar, s_flood_merge_by_tile_ar, s_flood_road_ln, s_flood_road_ln_tile, s_flood_road_trim_ln, s_bridge_warning_pnt, s_flood_grid_ar	Every National Weather Service forecast cycle
Dynamic — products	s_lwc_pnt	Every forecast cycle

2.2 Database Tables

Table 2. Entity inventory

Table	Group	Geometry	Role
s_road_segment_ln	Reference	Line	Roadway centerline inventory
s_bridge_pnt	Reference	Point	Bridge inventory
s_lwc_static_pnt	Reference	Point	Low-water-crossing locations and flow thresholds
s_flood_inundation_ar	Library	Area	Precomputed flood extent per reach per discharge
t_nextgen_to_nwm	Linkage	—	Hydrofabric to National Water Model crosswalk
t_road_flood_trigger	Linkage	—	Discharge at which each roadway segment floods
t_bridge_rating_curve	Linkage	—	Discharge to stage relationship at each structure
t_current_forecast	Forecast	—	Pointer to the published model run
t_flow_forecast	Forecast	—	Ingested forecast, eighteen hours pivoted to columns
t_flow_per_nextgen	Forecast	—	Forecast restated on the hydrofabric, as an array
t_max_flow_per_nextgen	Forecast	—	Peak discharge and peak hour per reach
s_flood_grid_ar	Product	Area	Tiling of flooding area per forecast
s_selected_flood_ar	Product	Area	Library extents selected for this run
s_flood_merge_ar	Product	Area	Dissolved flood surface attributed to tiles
s_flood_merge_by_tile_ar	Product	Area	Tile-clipped flood surface for delivery
s_flood_road_ln	Product	Line	Roadway segments whose threshold is exceeded
s_flood_road_ln_tile	Product	Line	Impacted roads split along the tile mesh
s_flood_road_trim_ln	Product	Line	Impacted roads trimmed to the wet extent
s_bridge_warning_pnt	Product	Point	Per-structure warning record for the run
s_lwc_pnt	Product	Point	Per-crossing forecast status and overtopping result

A data dictionary describing each of the twenty tables is provided in Appendix A.

Seven main IDs (identifiers) hold the database together. If you understand what each ID represents you can understand how the tables connect and how to read most queries.

There are two different stream networks used in this database. The first is called the “nextgen_id.” This comes from the Texas Division of Emergency Management’s Pin2Flood project. Each stream in that system has its own unique nextgen_id, and the flood maps in this database are built to match those streams. The second stream network comes from the National Water Model (NWM). In that system, each stream has a unique “feature_id.” The National Weather Service uses these feature_id values when it creates and shares streamflow forecasts. In short, nextgen_id is used for the stored flood maps, while feature_id is used for the live flow forecasts.

Table 3. Identifier domains and their type consistency

Identifier	Identifies
nextgen_id	A hydrofabric flowline. The unit on which flood inundation is mapped.
feature_id	A National Water Model channel id… how flow forecasts are provided by the. National Weather Service
road_id	A roadway segment
uuid_bridge	A bridge structure
tile_id / id	A tile of a segmented merged polygon
model_run_time	A forecast cycle
lwc_id	A low-water crossing

2.3 Database Join Paths

The table below shows how the different tables in the database connect to each other and how many records are linked between them. This is called “cardinality,” which just means how many matches exist between two tables. For example, “1 : N” means one record in the first table can connect to many records in the second table. “1 : 1” means one record connects to exactly one record. “N : 1” means many records connect back to one.

These connections show how data moves through the system from streamflow forecasts, to flood maps, to flooded roads, to bridge warnings and low-water-crossing status points.

Table 4. Join paths between entities

From (Table)	To (Table)	Join predicate	Cardinality
s_road_segment_ln	t_road_flood_trigger	road_id	1 : N
t_nextgen_to_nwm	t_road_flood_trigger	nextgen_id	1 : N
s_bridge_pnt	t_bridge_rating_curve	uuid_bridge	1 : N
t_nextgen_to_nwm	t_flow_forecast	feature_id	1 : N
t_flow_forecast	t_flow_per_nextgen	feature_id, model_run_time	N : 1
t_flow_per_nextgen	t_max_flow_per_nextgen	nextgen_id, model_run_time	1 : 1
t_max_flow_per_nextgen	s_flood_inundation_ar	nextgen_id, with flow ≤ max_flow	1 : N, one selected
t_max_flow_per_nextgen	s_selected_flood_ar	nextgen_id	1 : 1
s_flood_grid_ar	s_flood_merge_ar	id = tile_id	1 : N
s_flood_merge_ar	s_flood_merge_by_tile_ar	tile_id	1 : N
t_road_flood_trigger	s_flood_road_ln	road_id, nextgen_id	1 : 1 when triggered
s_flood_road_ln	s_flood_road_ln_tile	road_id	1 : N, one row per tile crossed
s_flood_road_ln_tile	s_flood_road_trim_ln	tile_id, road_id	1 : N after trimming
s_bridge_pnt	s_bridge_warning_pnt	uuid_bridge	1 : 1 per workflow
t_flow_per_nextgen	s_lwc_static_pnt	feature_id	N : N before aggregation by feature
s_lwc_static_pnt	s_lwc_pnt	lwc_id	1 : 1 per workflow

2.4 Static Layers

Before FAST can provide real-time flood information, a collection of permanent geospatial and tabular datasets must first be created and loaded into the database. These static datasets provide the foundation needed to quickly convert forecasted stream conditions into mapped flood impacts during an event. The preparation process includes:

(1) creating flood inundation areas from terrain data, stream networks, and established flow thresholds

(2) identifying roads and bridges, and low-water crossings that may be affected by those flood area

(3) adding information such as roadway characteristics, functional classification, and bridge elevations, and low-water-crossing flow thresholds.

Together, these datasets allow FAST to rapidly determine which transportation assets are at risk when new flood forecasts become available.

2.4.1 Flood Inundation Maps (s_flood_inundation_ar)

The flood inundation polygons used by FAST originated from work completed for the Texas Division of Emergency Management (TDEM) through the Pin2Flood project. These datasets represent predicted flood extents for different streamflow levels along stream reaches throughout Texas. The flood extents were developed using the Height Above Nearest Drainage (HAND) approach applied to 3-meter terrain data. Multiple flood levels were generated at approximately 1-foot elevation intervals. To prepare these datasets for use in FAST, the original flood polygons were updated using the road elevation model to remove areas where roads would not actually be affected. This refinement allows FAST to identify only roadway locations where flooding is expected to occur because water levels reach or exceed the roadway elevation.

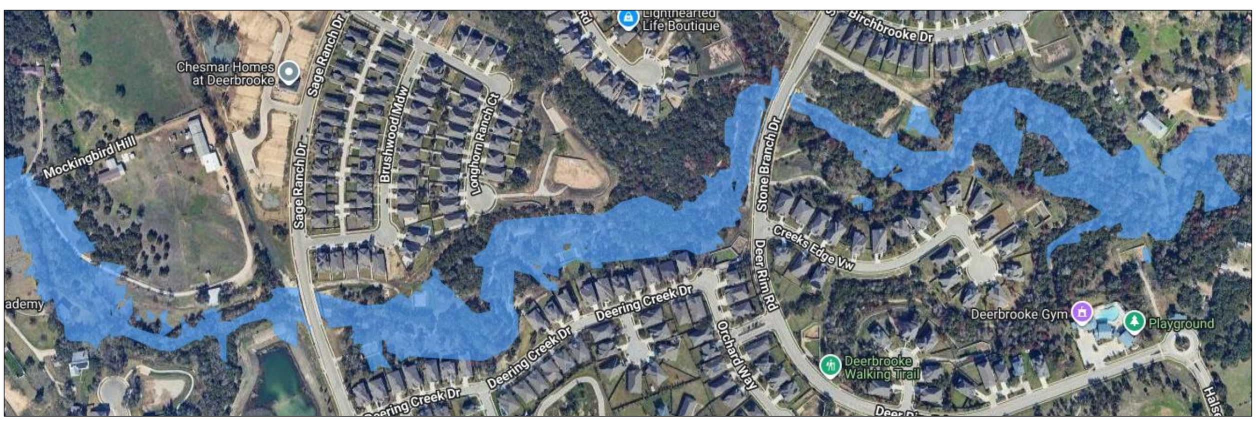


Figure 3. Sample flood inundation with road elevation masking

2.4.2 Flooded Roads (s_road_segment_ln)

Using OpenStreetMap (OSM) roadway data, the Texas road network was broken into more than 2.6 million individual segments (https://web.corral.tacc.utexas.edu/ras2fim/road_lines.html). Elevations along these roads were determined utilizing the Road Elevation Model (https://web.corral.tacc.utexas.edu/ras2fim/road_elev.html). For each segment, a flood threshold was assigned to represent the streamflow level at which flooding first begins to affect that roadway. These thresholds were determined by comparing each road segment against a library of precomputed flood inundation polygons. By identifying when a segment intersects with a given flood extent, the system can estimate the flow level at which water reaches or overtops that road.

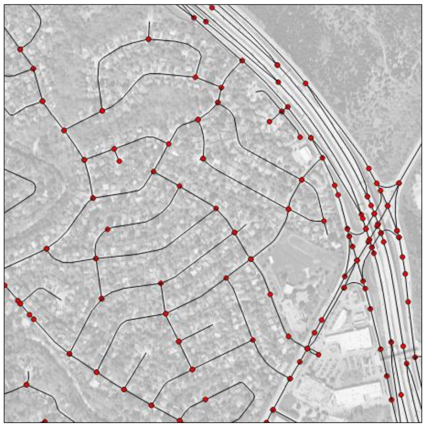


Figure 4. Sample road lines with segmentation

2.4.3 Bridge Warning Points (s_bridge_pnt)

Warning data for points for bridges that cross over a stream identified in the National Water Model (NWM) were orginally extracted using the "TX-Bridge" methods detailed in section 5.2.1 of Technical Memorandum FY24RU1 [TMFY24RU1Project0_7095_01.pdf].
https://caee.webhost.utexas.edu/prof/maidment/FAST/Documents/TMFY24RU1Project0_7095_01.pdf

For this TxDOT project, Dr. Yan Liu of Oak Ridge National Laboratory revised this methodology and performed a GPU assisted bulk extraction of bridge profile lines systematically across the entire state of Texas. His data area located here:
https://web.corral.tacc.utexas.edu/nfiedata/sf3/alpha10/. Through his methods, we believe that the team can provide real-time flood forecasts at a little more than 14,000 bridges across the state. To qualify for forecast, the bridge must:

(1) Cross a river where a flood inundation and flow forecast are available.
(2) LiDAR extraction have a minimum conveyance area under the bridge
(3) Have a bridge profile that does not exceed a grade of 11%
(4) Have two (and only two) abutment points representing the beginning and end of a bridge

To document which bridges meet these qualifications, the team built an interactive website that shows the statewide results of Dr. Liu's extracted data. That viewer is located at: https://roadflood.com/bridge_accounting/snbi_fgb_viewer.html

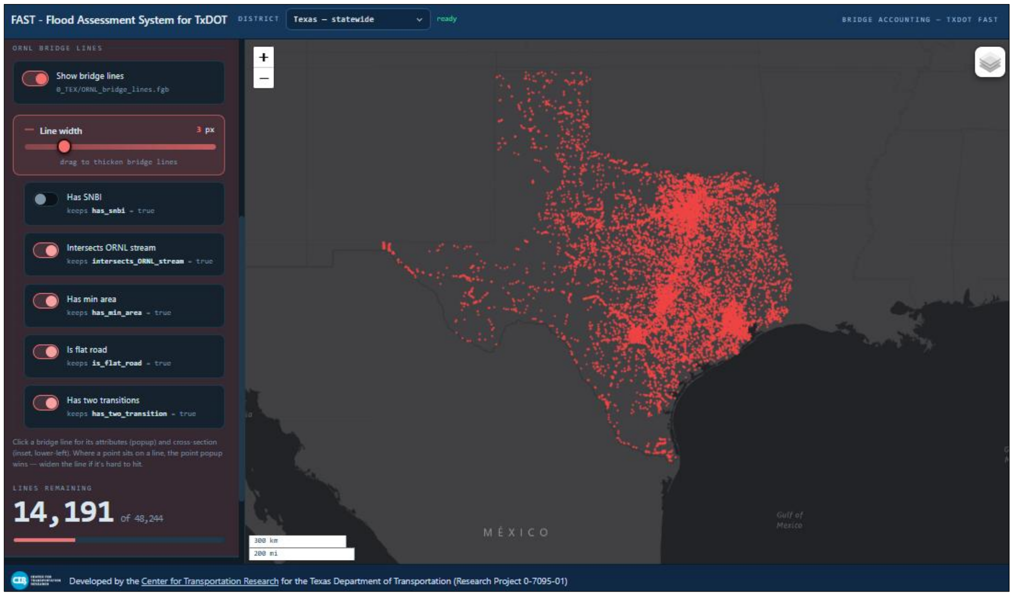


Figure 5. FAST Bridge accounting of ORNL extracted bridge lines

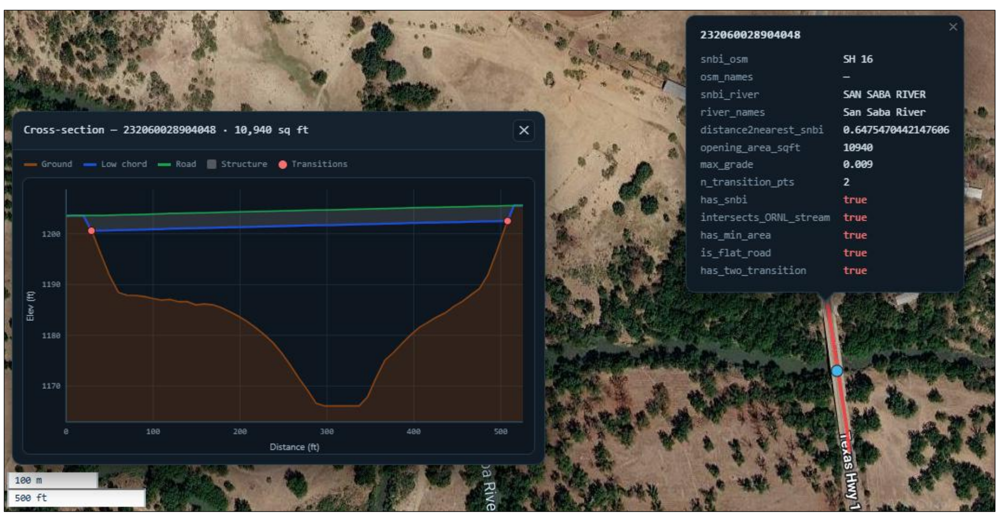


Figure 6. FAST Bridge accounting – example of SH16 over San Saba River

2.4.4 Low-Water Crossings (s_lwc_static_pnt)

The low-water-crossing reference layer stores the point location and identifying information for each modeled crossing. Each crossing is linked to a National Water Model feature using feature_id. The table also stores the discharge thresholds associated with overtopping and with approximately 0.5, 2, and 5 feet of water depth.

The principal fields are lwc_id, hydro_id, model_id, feature_id, name, osm_id, fclass, q_overtopped, q_0_5_ft, q_2_ft, q_5_ft, and point geometry in EPSG:4326. These values are persistent reference data and change only when the crossing inventory, hydrologic linkage, or hydraulic thresholds are revised.

The deployed schema supports LWC records in every district database, but the reference inventory is not fully populated. The current data are a limited demonstration sample in the Austin district and should not be interpreted as complete Austin or statewide LWC coverage.

2.5. Dynamic Layers

2.5.1 Hydrologic Linkage

Three tables carry the intelligence that connects hydrology to transportation.

T_NEXTGEN_TO_NWM is a crosswalk between hydrofabric "Pin2Flood" reach identifiers and National Water Model feature identifiers.

T_ROAD_FLOOD_TRIGGER is how FAST quickly determines which roads are predicted to be flooded. For each roadway segment it stores the controlling reach and the discharge at which that segment first floods. That number is the product of a spatial analysis performed from the road elevation model, road segments and inundation library. Having it precomputed means the forecast can determine which roads flood by comparing two integers rather than by intersecting a line layer against a polygon layer.

T_BRIDGE_RATING_CURVE plays the same role for structures, converting discharge to stage so that forecast flow can be compared against the low chord elevation.

2.5.2 Forecast State

Four tables hold the state of a forecast cycle.

T_FLOW_FORECAST is the landing table, one row per National Water Model feature, with the forecast horizon pivoted into eighteen columns named flow_t00 through flow_t17. The width corresponds to the eighteen-hour horizon of the National Water Model short-range forecast.

T_FLOW_PER_NEXTGEN restates the same forecast on the hydrofabric and, in doing so, replaces the eighteen columns with a single array. This array can be indexed by hour. The table also precomputes max_flow and max_hour.

T_MAX_FLOW_PER_NEXTGEN reports the peak flow and is the value all three impact products read.

T_CURRENT_FORECAST holds a single value of the model run that is currently published.

2.5.3 Operational Products

The flood mapping process in FAST follows three statewide operational paths—inundation, roads, and bridges—and a fourth demonstration path for low-water crossings. S_SELECTED_FLOOD_AR -- S_FLOOD_MERGE_AR -- S_FLOOD_MERGE_BY_TILE_AR

The inundation path starts by selecting the correct flood polygon for each stream reach. These polygons are then merged to remove overlaps between neighboring areas. After that, the merged flood surface is cut into tiles so it can be quickly processed and delivered.

S_FLOOD_ROAD_LN -- S_FLOOD_ROAD_LN_TILE -- S_FLOOD_ROAD_TRIM_LN

The roadway path identifies which road segments are affected by flooding. First, it flags segments where the flow exceeds the flood threshold. Next, those segments are split along the tile grid. Finally, each piece is trimmed to match the flood inundation area. This last step is important because it prevents entire road segments from appearing flooded when only part of them is underwater. It also calculates the flooded length of each segment, which is used to report total roadway impacts.

S_BRIDGE_WARNING_PNT

The bridge path produces a single table that combines bridge information with forecast results. Key values are included are the predicted water level and the remaining clearance between the water and the bridge. The clearance is useful because it shows how close a bridge is to being overtopped, allowing bridges to be categorized by flood risk.

S_LWC_STATIC_PNT -- S_LWC_PNT

For the low-water-crossing path, FAST first finds the maximum forecast flow for each linked feature_id. It copies each crossing from s_lwc_static_pnt into the workflow-specific s_lwc_pnt table, adds max_flow and model_run_time, and sets is_overtopped when the maximum flow reaches or exceeds q_overtopped. The depth thresholds remain in the output so downstream applications can classify the severity of the crossing condition.

2.5. Operational Use in Production

This section provides an overview as to how this database is utilized on a backend server to compute the flood inundation limits, flooded road lines and bridge warning points shown on the front-end FAST user interface and geospatial endpoints.

2.5.1 The Operational Cycle

A FAST update cycle on a back-end server is triggered by the publication of a new National Water Model short-range forecast. These are typically provided by the National Weather Service every hour at https://nomads.ncep.noaa.gov/pub/data/nccf/com/nwm/prod/ . After the system starts, it runs seven main steps. These steps are automated and require no user input. Some steps reduce the data, some select the right values, some work with map data, and the last step publishes the results. The figure below shows these steps.

STEP	PROCESS	TABLE WRITTEN
INPUT	NWM short-range flow forecast published by NOAA (18 hourly time steps)	*external NetCDF*
1	Ingest forecast; subset to NWM reaches in the study area; pivot 18 hourly forecasts steps into columns	**t_flow_forecast**
2	Crosswalk flood inundation polygons with National Water Model stream reaches; Reduce each hydrofabric reach to its forecast peak and the hour at which the peak occurs	**t_flow_per_nextgen** **t_max_flow_per_nextgen**
3	For each reach, select the greatest library stage not exceeding max_flow from s_flood_inundation_ar	**s_selected_flood_ar**
4	Union overlapping reach footprints, then clip the result to the s_flood_grid_ar tile mesh	**s_flood_merge_ar** **s_flood_grid_ar** **s_flood_merge_by_tile_ar**
5	Flag segments where max_flow reaches the stored min_flood_flow; tile, then trim to the wet extent	**s_flood_road_ln** **s_flood_road_ln_tile** **s_flood_road_trim_ln**
6	Convert flow to water-surface elevation via the rating curve; compare to low-chord elevation	**s_bridge_warning_pnt**
7	Store when this flow forecast was made by the National Weather Service	**t_current_forecast**
OUTPUT	Vector layers of flood limits (area), flooded road (lines) Bridge warnings (points)	Output vector files (GIS)

Figure 7. Operational processing chain for a single forecast cycle

This workflow does not run hydraulic models in real time. All heavy computations were completed ahead of time and stored in S_FLOOD_INUNDATION_AR and T_ROAD_FLOOD_TRIGGER. During an update, the system mainly performs simple calculations and map-based (geometric) processing. The arrangement and interaction of the static and dynamic flood layers is shown in Figure 8.

Processing time can increase during large flood events because the geometric steps depend on how much area is flooded. In other words, the system does more work when more land is underwater.

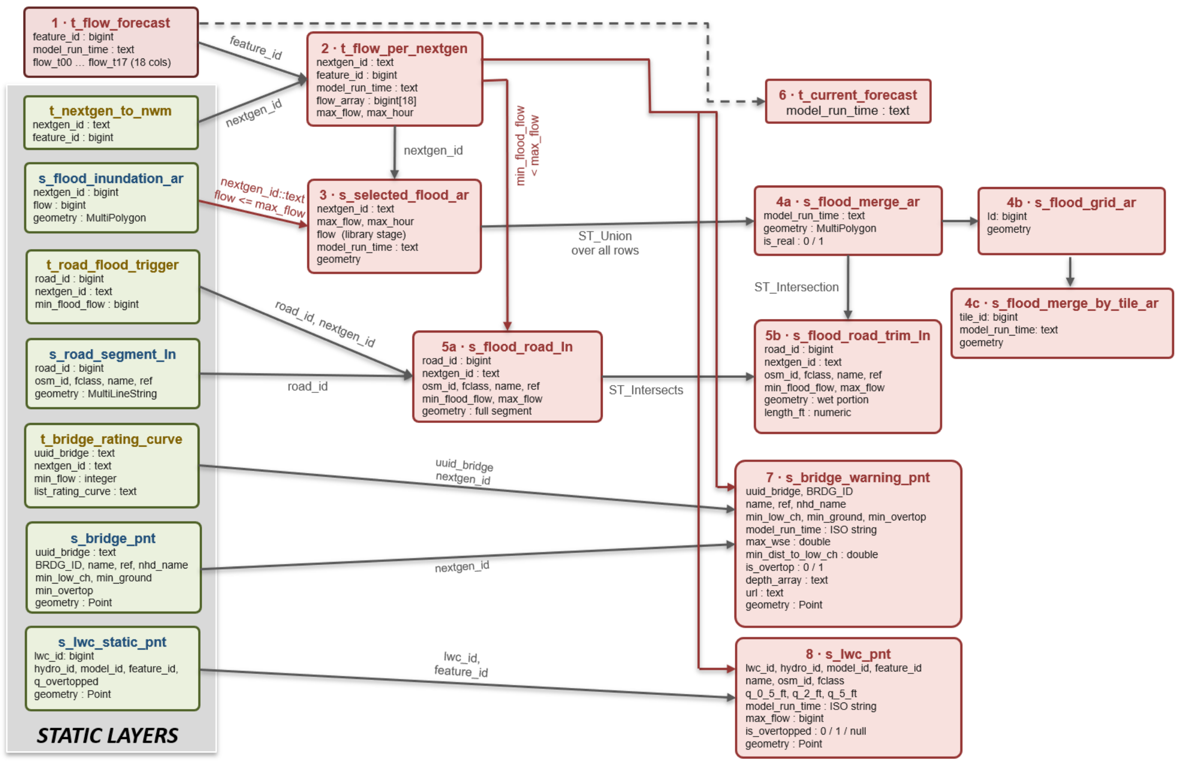


Figure 8. Table generation schematic from static layers

3. Backend Services

The backend is the automated engine room of FAST. It fetches the newest national flood forecast, matches it against a pre-built library of Texas flood scenarios and bridge data, and continuously republishes three Texas-specific flood warning spatial layers:

(1) **roads that are predicted to flood**
(2) **polygons showing how far the water spreads**
(3) **bridges where the water is climbing toward the underside of the span**
(4) low-water crossings where forecast flow reaches or exceeds the modeled crossing thresholds

In the Task 5 architecture, this system is component (b) - the backend services that convert forecasted flows into flood-warning layers. It sits between the reference databases it reads from (component a) and the web viewers that display its results (component c). The whole process is automated and cloud-based, running on Amazon Web Services (AWS) so that it operates continuously without human interaction. The application code lives in the fast_realtime repository and is publicly available on GitHub. https://github.com/andycarter-pe/fast_realtime

3.1 Backend Components

Every run combines a constantly-changing input with a flood library that was prepared in advance. The static reference layers never change, and each new forecast is compared against them to produce fresh dynamic layers that do change.

- **The live forecast.** The backend reaches out to NOAA's public National Water Model (NWM) archive and pulls the latest "short-range" forecast which is a set of 18 hourly predictions of how much water will be flowing in every stream and river reach in the country over the next 18 hours. It automatically detects the most recent forecast and downloads it.
- **The static reference library**. Alongside that live feed sits a PostGIS geospatial database. This is a specialized database built to store maps and shapes. It holds a set of static precomputed flood-related layers organized by stream segment. For each segment these include flood-inundation polygons at many different water levels, road flooding thresholds, bridge overtopping points, and rating curves. Each bridge also carries its minimum low-chord elevation which is the height of the underside of the bridge deck.

3.2 Backend Architecture

The backend is built from a few coordinated AWS services that work together to ingest the forecast, process it against the flood thresholds, and publish the updated map layers.

- **The application server (Amazon EC2).** A Python application runs on an Amazon EC2 instance which is a virtual computer in the cloud. It runs inside a Docker container, which packages the FAST code together with all the programmatic dependencies. This application is the part that continuously watches for new short-range forecasts.
- **The database (Amazon RDS with PostGIS).** When a new forecast is detected, the application triggers the SQL script, which connects to the PostgreSQL/PostGIS database

hosted on Amazon RDS (Amazon's managed database service). This is where the static reference layers live and where the spatial queries and calculations that compare forecasts to thresholds are carried out.

- **The published output (Amazon S3).** The finished dynamic layers are written to Amazon S3 cloud storage in map-ready formats, where the FAST web viewers and other downstream applications read them.

3.3 District Based Processing

The size of Texas makes it difficult to compute in a single pass; therefore, the backend works through the state's 25 TxDOT districts one after another. A worker-loop steps through a list of district configuration files (Austin, Houston, Dallas, El Paso) and hands each to the update engine in turn. This keeps every run to a manageable size and means a computing problem in one district does not stall the others.

The system also supports three operating modes from the same codebase:

(1) a real-time mode driven by the live national forecast
(2) a nowcast mode focused on current conditions
(3) and a historic mode that replays past events.

This is the same distinction reflected in the Development environments on the FAST landing web-page.

3.4 Statewide Merged Database and WFS

The TXFull PostGIS database is the statewide publishing database used by GeoServer. PostgreSQL foreign-data-wrapper schemas named foreign_01 through foreign_25 connect TXFull to the 25 district databases. A merge step combines district products into statewide materialized views and records the originating district in source_db.

The statewide materialized views are:

- **mv_flood_merge_tx:** Flood-inundation polygons.
- **mv_flood_road_trim_ln_tx:** Flooded road lines trimmed to the wet extent.
- **mv_bridge_warning_pnt_tx:** Bridge-warning points.
- **mv_lwc_pnt_tx:** Low-water-crossing status points. Current content is limited to the Austin-district demonstration data.

GeoServer can expose read-only WFS virtual layers for four operational workflow sources: da_sr, da_sr_nc, nwm_sr, and nwm_sr_nc. For each source, it supports the same four layer types at statewide and individual-district scope:

- ***_flood:** Forecast flood-inundation polygons.
- ***_roads:** Flooded road lines trimmed to the wet extent.
- ***_bridges:** Bridge-warning points.
- ***_lwc:** Low-water-crossing status points. Current coverage is limited to the Austin demonstration sample.

For example, nwm_sr_tx_flood is the statewide NWM forecast flood layer, da_sr_AUS_lwc is the Data Assimilation low-water-crossing demonstration layer for Austin, and da_sr_WFS_roads is the road layer for the Wichita Falls district. In a database name such as 03_WFS_realtime_hand, WFS means the Wichita Falls district; it does not mean Web Feature Service.

3.5 Supported Output Capabilities

FAST supports four delivery capabilities for flood, road, bridge, and low-water-crossing products:

- **EsriJSON:** JSON structured for ArcGIS clients. It is generated as a separate S3 output when publish_esri_json is enabled.
- **GeoJSON:** Open JSON vector files for web maps, scripts, and general GIS interchange.
- **WFS:** A read-only GeoServer Web Feature Service for querying the published feature layers. WFS is a service interface, not a single file format.
- **FlatGeobuf:** A compact, spatially indexed binary vector file intended for efficient transfer and partial reads of large layers.

GeoJSON and FlatGeobuf are generated by separate output tasks so that production of one format does not delay the other. EsriJSON is independently controlled by output configuration. WFS reads the published PostGIS layers through GeoServer rather than reading the S3 files.

4. Web Viewer

The bridges.txdot.kisters.cloud web application was developed jointly by the KISTERS North American and the University of Texas at Austin under Task 5 for the deployment of operational flood warning geospatial layers for TxDOT.

Since August 2025, the FAST web interface has expanded to provide statewide coverage across Texas and has evolved over the past 11 months. The system delivers hourly updates of predicted flood extents derived from the National Water Model (NWM) short-range forecast. Forecasts are generated independently for each of TxDOT's 25 districts, enabling district-level visualization of flood risk and supporting localized response planning.

The system is published through three environments, each documented in its own section below:

(1) **Production** (stable, operational products for everyday use)
(2) **Development** (experimental and diagnostic tools)
(3) **Training** (guided practice and onboarding)

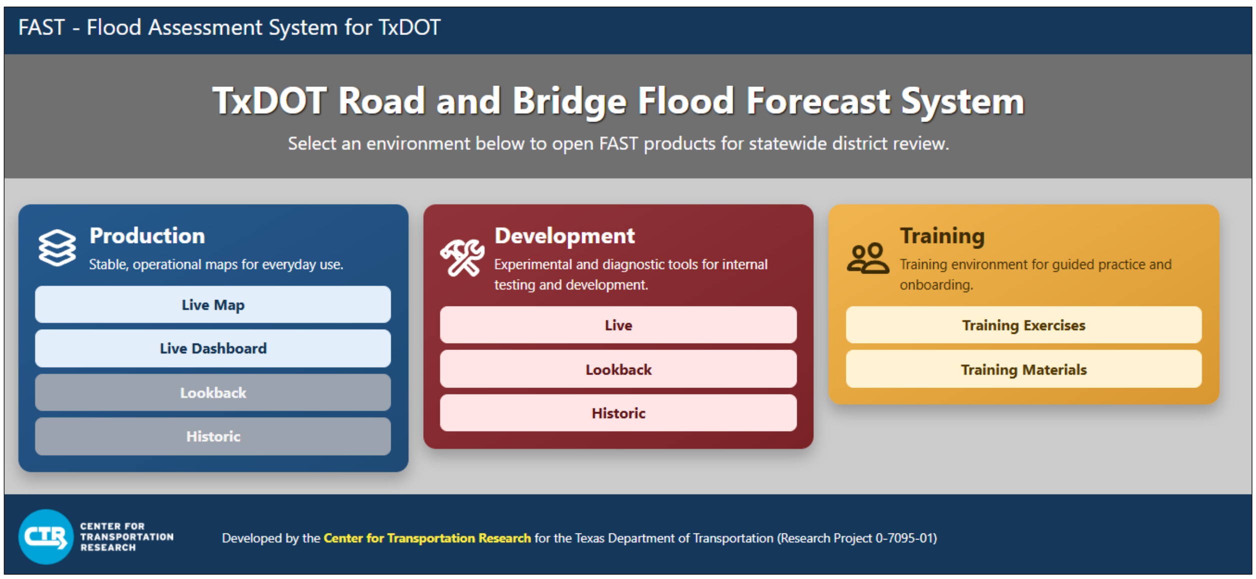


Figure 9. Landing Page bridges.txdot.kisters.cloud

4.1 Production

Production is the day-to-day operational face of FAST and the environment TxDOT district staff and maintenance crews should reach for first during an active weather event. Production exposes two web-based viewers: an interactive **Live Map** and a statewide **Live Dashboard**. Both refresh with each new model run so the picture stays close to real time. A Lookback and Historic function are planned but not yet deployed.

4.1.1 Production Live Map

The Live Map is the core FAST product. It is an interactive, statewide map of predicted flooding for roads and bridges that users can pan, zoom, and explore in real time. The production system allows the users to see the forecasts from two different flow sources.

Table 5. Production Flow Sources

Flow Source	Description
Data Assimilation (DA)	A real-time flow source that blends live observations (such as stream-gage readings) with model output to produce an observation-corrected estimate of current streamflow.
National Water Model (NWM)	National Water Model (NWM) NOAA's national hydrologic model, which simulates and forecasts streamflow for millions of river reaches nationwide and feeds FAST's road and bridge predictions.

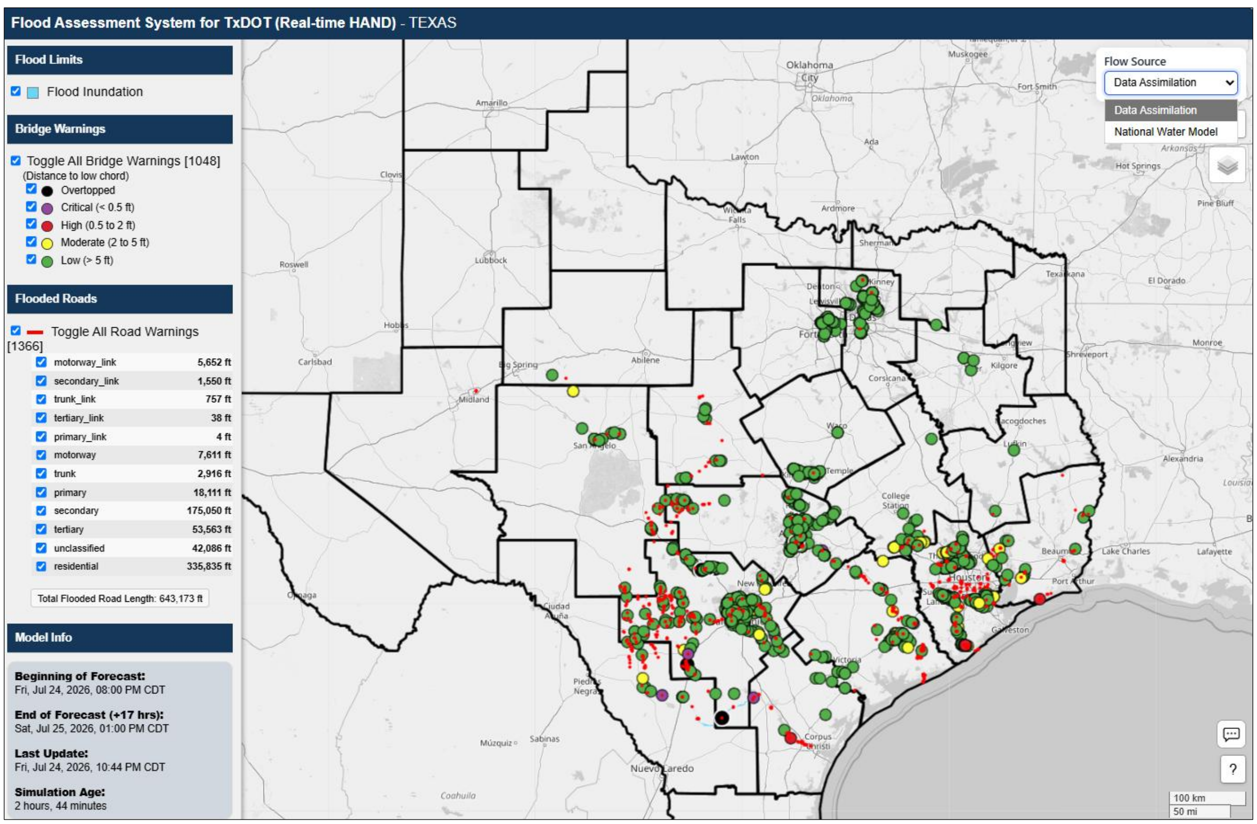
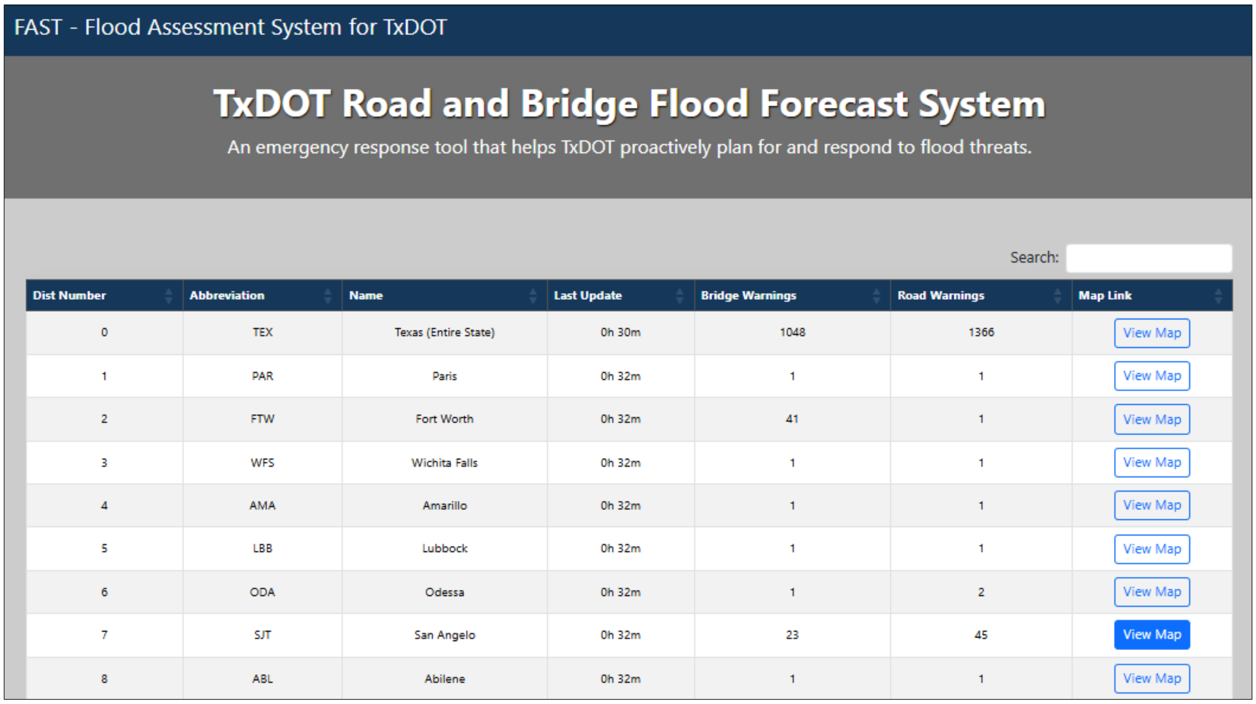


Figure 10. Production FAST Live Map

4.1.2 Production Live Dashboard

The Live Dashboard is a statewide roll-up and serves as a method to quickly view the numerical impacts of flooding. Rather than a map, it presents a sortable table with one row per district, so a statewide reviewer can see at a glance which districts are under the most pressure and jump straight into the relevant map.



Dist Number	Abbreviation	Name	Last Update	Bridge Warnings	Road Warnings	Map Link
0	TEX	Texas (Entire State)	0h 30m	1048	1366	View Map
1	PAR	Paris	0h 32m	1	1	View Map
2	FTW	Fort Worth	0h 32m	41	1	View Map
3	WFS	Wichita Falls	0h 32m	1	1	View Map
4	AMA	Amarillo	0h 32m	1	1	View Map
5	LBB	Lubbock	0h 32m	1	1	View Map
6	ODA	Odessa	0h 32m	1	2	View Map
7	SJT	San Angelo	0h 32m	23	45	View Map
8	ABL	Abilene	0h 32m	1	1	View Map

Figure 11. Production FAST Live Dashboard

4.2 Development

The development environment was created by the research team to create and test experimental tools inside the mapping environment prior to deploying into the live production system. This includes items like where new flow sources, symbology, and interface. Its data and controls may change without notice and should not be relied on operationally. Development offers three viewers: an expanded **Live Map**, a **Lookback** review mode, and a **Historic** replay mode.

4.2.1 Development Live Map

The Development Live Map carries the same core panels as Production (Flood Limits, Bridge Warnings, Flooded Roads, Model Info, Feedback) but adds a richer set of flow options and several experimental capabilities.

Table 6.Development Flow Sources

Flow Source	Description
Forecast — Data Assimilation	The worst (highest) predicted flood condition across the next 18 hours, driven by the observation-corrected Data Assimilation flow source. Built from adjusting the NWM with real-time, gage-informed streamflow.
Forecast — National Water Model	The worst (highest) predicted flood condition across the next 18 hours, driven by NOAA's National Water Model. The peak threat expected in the forecast window from the national model.
Nowcast — Data Assimilation	An estimate of present conditions (rather than a future forecast) from the Data Assimilation flow source. Useful for confirming what is happening right now against live observations.
Nowcast — National Water Model	An estimate of present conditions from the National Water Model.
NWM Lagged Ensemble	An ensemble built from several successive National Water Model forecast cycles offset ("lagged") in time. (Time lagged ensembles). Averaging these staggered runs reduces reliance on any single forecast and smooths variability, providing a more stable estimate of future conditions and uncertainty.

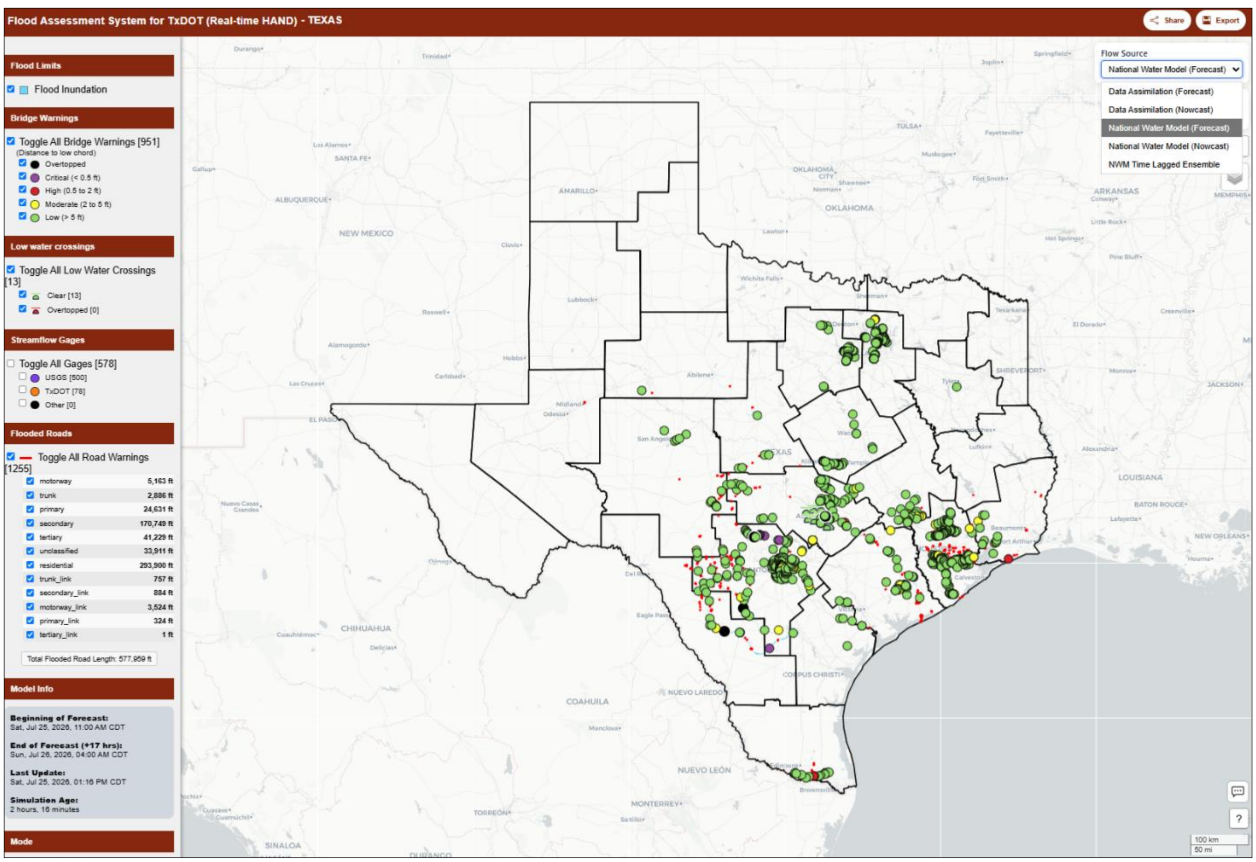


Figure 12. Development FAST Live Map

Here are some of the features being tested in the development environment:

- **<u>Select District:</u>** The Development Live Map adds the ability to select a district, filtering the map and its warnings to a single TxDOT district so reviewers can focus on their own area of responsibility rather than the full statewide extent.

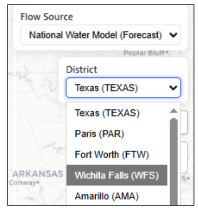


Figure 13. Development FAST Map – Select District

- **<u>Gages with DA vs. NWM comparison:</u>** Stream-gage locations can be displayed to compare the Data Assimilation flow against the National Water Model side by side. This DA-vs-NWM comparison helps analysts see where the observation-corrected estimate diverges from the national model and judge which to trust at a given location.

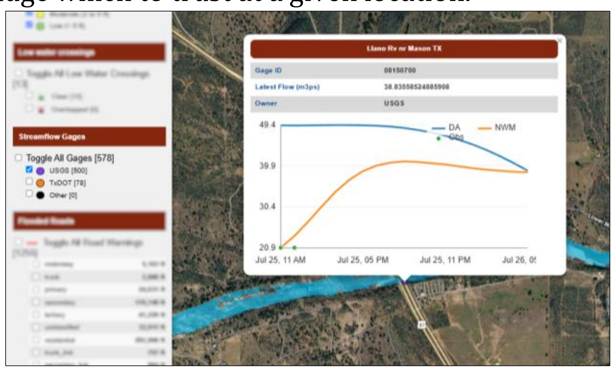


Figure 14. Development FAST Map – Gage Data

- **Low-water-crossings**: The development environment demonstrates forecast status for a limited sample of modeled low-water crossings in the Austin district. FAST stores crossing geometry and flow thresholds in s_lwc_static_pnt and writes workflow-specific values, including forecast maximum flow and overtopping status, to s_lwc_pnt. The current sample is not a complete Austin or statewide LWC inventory. Below is the demonstration of how low water crossings might be shown in future iterations of FAST if TxDOT elects to deploy staff to acquire the culvert data needed to simulate using the HY-8 culvert modeling method (detailed in another report)..

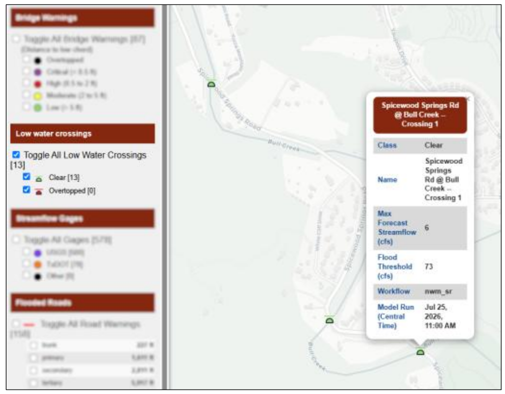


Figure 15. Development FAST Map – Low Water Crossings

- **<u>Bridge cross section pop-up:</u>** The production environment had a link to another webpage that would show the interactive bridge cross section. For the development environment, the team experimented with embedding that cross section directly into the pop-up for a bridge warning.

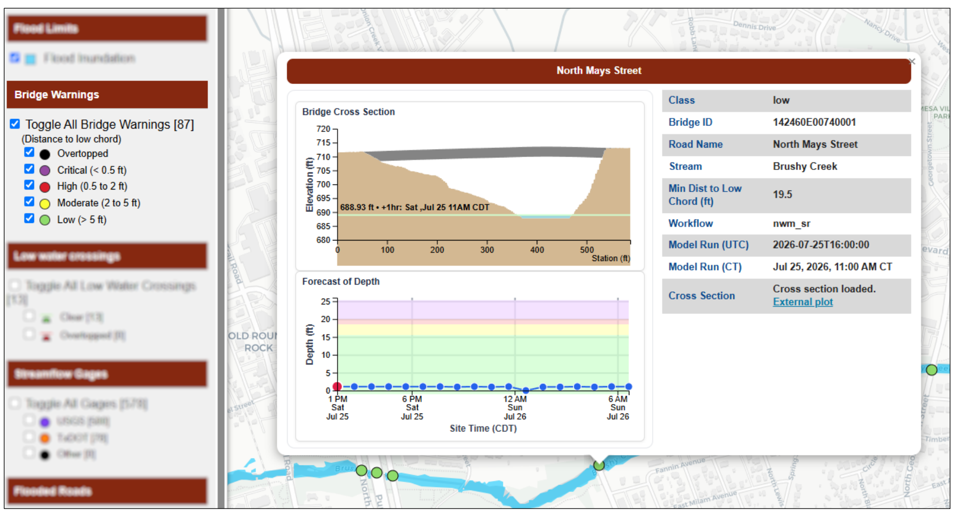


Figure 16. Development FAST Map – Bridge cross section embedment

4.2.2 Development Lookback Map

Lookback is a review mode intended to show the nowcast of the last 24 hours. Using an hourly time slider, users can scrub backward through each of the past 24 hours to watch how nowcast conditions (including the road and bridge warnings) appeared, intensified, or cleared leading up to the present. It answers "what has been happening over the last day?" rather than showing only the latest single snapshot.

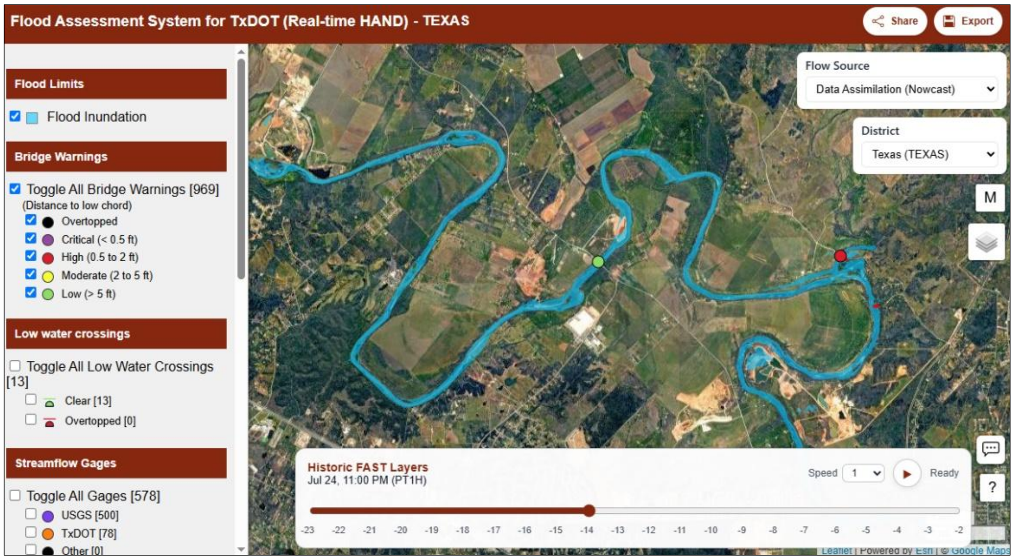


Figure 17. Development FAST Lookback Map

4.2.3 Development Historic Map

Historic extends the same idea to the past: it is intended to let users see the nowcast for a date range of their choosing. By selecting start and end dates, a user can replay how FAST portrayed conditions during a specific past flood event. This tool is intended for after-action review, case studies, and validating the system against known outcomes. Is still requires substantial development.

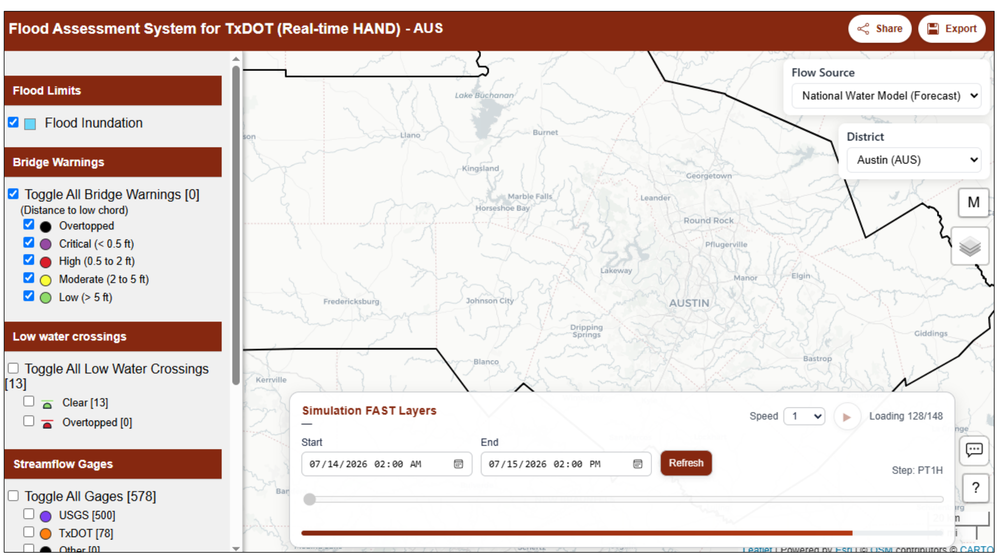


Figure 18. Development FAST Historic Map

4.3 Training

The Training environment is where new users learn how to use FAST before interacting with live operations. It pairs learning-by-doing against real historic events with a library of reference media, giving each district an onboarding path before staff work in Production.

4.3.1 Training Exercises

Titled "TxDOT Historic Flood Training," this tool lets trainees review historical FAST mapping outputs for three separate events. Opening an event replays the map exactly as it would appear during a live incident, so users can practice interpreting warnings against known outcomes. Each event is listed with Date, Forecast Time (UTC), a Map Link, and a File Link to the underlying output.

4.3.2 Training Materials

A companion library of self-serve learning resources is provided. This includes a video (~ 5 minutes), a one page quick start guide and a PowerPoint presentation for introducing new users to the FAST Production interface.

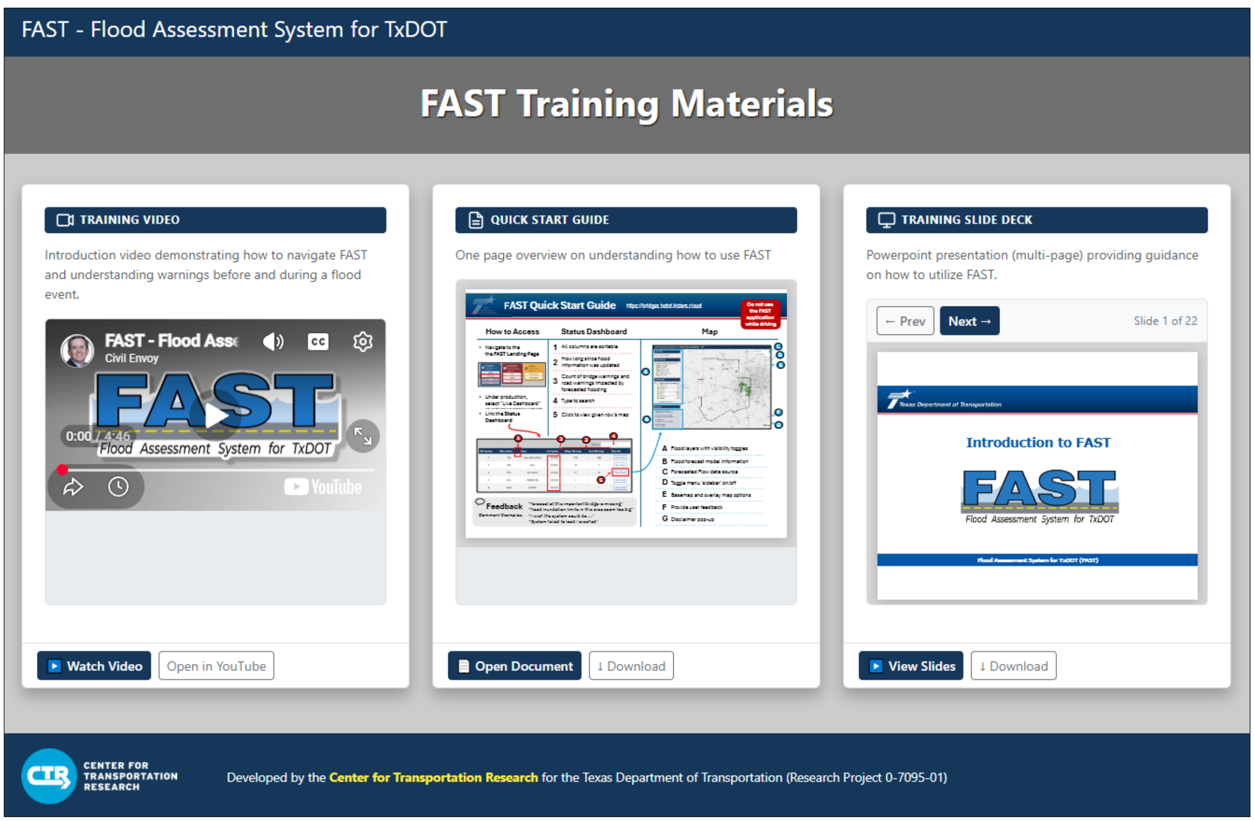


Figure 19. FAST Training Materials

Appendix A. FAST Database Data Dictionary

This appendix documents every column of all twenty tables in the FAST PostgreSQL database.

A.1 Static Reference Layers

s_road_segment_ln

Group: Static reference. **Geometry:** MultiLineString, EPSG:4326. **Lifecycle:** Persistent Roadway centerline inventory for the study area. Every road-related product traces back to a row in this table. Statewide data are available at: https://web.corral.tacc.utexas.edu/ras2fim/road_lines.html

Table 7. Columns of s_road_segment_ln

Column	Type	Description
road_id	bigint	Internal roadway segment identifier. Functions as the business key joining this layer to t_road_flood_trigger and to all s_flood_road_* products. Not declared as a primary key.
osm_id	text	Source identifier carried over from OpenStreetMap, retained for provenance and for re-derivation when the base network is refreshed.
fclass	text	Functional class of the segment (motorway, trunk, primary, residential, and similar). Used to prioritize which impacted segments are surfaced to operators.
name	text	Common roadway name.
ref	text	Route designation (for example, an interstate or state highway number).
geometry	geometry(MultiLineString,4326)	Segment centerline geometry. Indexed with GiST.

s_bridge_pnt

Group: Static reference. **Geometry:** Point, EPSG:4326. **Lifecycle:** Persistent.

Bridge inventory elevations required to evaluate overtopping and distance to low chord. This is the reference side of the bridge warning product.

Table 8. Columns of s_bridge_pnt

Column	Type	Description
uuid_bridge	text	Stable surrogate identifier for the structure. Used as the join key to t_bridge_rating_curve and s_bridge_warning_pnt.
BRDG_ID	text	National bridge incentory reference number. Agency bridge identifier. Declared with double quotes in the DDL, so the mixed-case spelling is preserved and every reference to it must also be quoted.
min_low_ch	double precision	Minimum low-chord elevation. The controlling threshold: when forecast water-surface elevation approaches this value, clearance is lost.
min_ground	double precision	Minimum ground elevation
min_overtop	double precision	Minimum elevation at which the deck is overtopped.
name	text	Road name (from OpenStreetMap)
ref	text	Reference name (from OpenStreetMap)
nhd_name	text	Name of the watercourse crossed, from the National Hydrography Dataset.
geometry	Geometry (Point,4326)	Structure location. Indexed with GiST.

---

s_lwc_static_pnt

Group: Static reference. **Geometry**: Point, EPSG:4326. **Lifecycle**: Persistent.

Persistent point inventory and hydraulic thresholds for low-water crossings.

Table 10. Columns of s_lwc_static_pnt

Column	Type	Description
lwc_id	bigint	low-water-crossing identifier and primary key
hydro_id	bigint	The modeled discharge the polygon corresponds to. Together with nextgen_id this forms the lookup key, supported by a descending composite index.
model_id	bigint	Model identifier
feature_id	bigint	National Water Model feature supplying forecast flow
name	text	Low water crossing name

Column	Type	Description
osm_id	text	OpenStreetMap identifier
fclass	text	Road functional class based on OpenStreetMap
q_overtopped	bigint	Discharge at which the crossing is overtopped
q_0_5_ft	bigint	Discharges associated with approximately 0.5 feet of water depth
q_2_ft	bigint	Discharges associated with approximately 2 feet of water depth
q_5_ft	bigint	Discharges associated with approximately 5 feet of water depth
geometry	Geometry (Point,4326)	Low water crossing location

s_flood_inundation_ar

Group: Inundation library. **Geometry:** MultiPolygon, EPSG:4326. **Lifecycle:** Persistent.

Precomputed flood extents. For each hydrofabric reach the library stores one polygon per modeled discharge.

Table 9. Columns of s_flood_inundation_ar

Column	Type	Description
nextgen_id	bigint	Hydrofabric reach identifier. Declared as bigint here but as text in every other table that carries the same concept; see Section 7.2.
flow	bigint	The modeled discharge the polygon corresponds to. Together with nextgen_id this forms the lookup key, supported by a descending composite index.
geometry	geometry(MultiPolygon,4326)	Inundation extent for that reach at that discharge. Indexed with GiST.

A.3 Hydrologic Linkage

t_nextgen_to_nwm

Group: Hydrologic linkage. **Geometry:** none. **Lifecycle:** Persistent.

Crosswalk between the hydrofabric reach identifiers used throughout the schema and the National Water Model feature identifiers used by the incoming forecast.

Table** 10. **Columns of t_nextgen_to_nwm

Column	Type	Description
nextgen_id	text	Hydrofabric reach identifier.
feature_id	bigint	National Water Model channel-routing feature identifier.

---

t_road_flood_trigger

Group: Hydrologic linkage. **Geometry:** none. **Lifecycle:** Persistent.

Precomputed flooding threshold for each roadway segment. This is the table that converts a discharge forecast into a roadway impact decision.

Table** 13. **Columns of t_road_flood_trigger

Column	Type	Description
road_id	bigint	Roadway segment, referencing s_road_segment_ln.
nextgen_id	text	Hydrofabric reach whose discharge controls flooding at this segment.
feature_id	bigint	The corresponding National Water Model feature.
min_flood_flow	bigint	Lowest discharge on the controlling reach at which the segment is inundated. A segment is flagged when the forecast peak reaches or exceeds this value.

---

t_bridge_rating_curve

Group: Hydrologic linkage. **Geometry:** none. **Lifecycle:** Persistent.

Discharge-to-stage relationship at each bridge, allowing forecast flow to be converted into a water-surface elevation that can be compared against the low chord.

Table 14. Columns of t_bridge_rating_curve

Column	Type	Description
uuid_bridge	text	Structure, referencing s_bridge_pnt.
nextgen_id	text	Hydrofabric reach supplying discharge at the structure.
min_flow	integer	Lowest discharge represented by the curve. Below this value the structure is not evaluated.
list_rating_curve	text	The rating curve itself, held as a serialized list rather than as structured rows or a numeric array.

A.4 Forecast State

t_current_forecast

Group: Forecast state. **Geometry:** none. **Lifecycle:** Per run.

Single-value pointer naming the model run that is currently published.

Table 11. Columns of t_current_forecast

Column	Type	Description
model_run_time	text	Identifier of the published model run. Held as text rather than a timestamp.

t_flow_forecast

Group: Forecast state. **Geometry:** none. **Lifecycle:** Per run.

Landing table for the incoming discharge forecast, one row per National Water Model feature, with the forecast horizon into eighteen columns.

Table 16. Columns of t_flow_forecast

Column	Type	Description
feature_id	bigint	National Water Model feature identifier.
model_run_time	text	Model run the row belongs to.
flow_t00 … flow_t17	bigint (18 columns)	Forecast discharge at each hour of the horizon, hour zero through hour seventeen. The eighteen-column width corresponds to the National Water Model short-range forecast horizon.

---

t_flow_per_nextgen

Group: Forecast state. **Geometry:** none. **Lifecycle:** Per run.

The forecast restated on the hydrofabric, with the wide hourly columns collapsed into an ordered array and the peak extracted.

Table 12. Columns of t_flow_per_nextgen

Column	Type	Description
nextgen_id	text	Hydrofabric reach.
feature_id	bigint	Source National Water Model feature, retained for traceability.
model_run_time	text	Model run the row belongs to.
flow_array	bigint[]	The eighteen hourly values as a Postgres array, preserving the full hydrograph for charting and for hour-specific queries.
max_flow	bigint	Peak discharge across the horizon.
max_hour	integer	Zero-based hour at which the peak occurs.

---

t_max_flow_per_nextgen

Group: Forecast state. **Geometry:** none. **Lifecycle:** Per run.

Peak-only summary per reach.

Table 13. Columns of t_max_flow_per_nextgen

Column	Type	Description
nextgen_id	text	Hydrofabric reach.
model_run_time	timestamp without time zone	Model run the row belongs to. This is the only model_run_time column in the schema declared as a timestamp; all others are text. See Section 7.2.
max_flow	integer	Peak discharge across the horizon. Declared as integer here but as bigint in t_flow_per_nextgen.
max_hour	integer	Hour of the peak.

A.5 Operational Products

s_selected_flood_ar

Group: Operational product. **Geometry:** MultiPolygon, EPSG:4326. **Lifecycle:** Per run.

The flood inundation polygon chosen for this run for each stream reach matching the nearest forecast peak flow.

Table 14. Columns of s_selected_flood_ar

Column	Type	Description
nextgen_id	text	Hydrofabric reach.
max_flow	bigint	Forecast peak discharge that drove the selection.
model_run_time	text	Model run the row belongs to.
max_hour	integer	Hour at which the peak occurs
flow	bigint	The library discharge actually selected
geometry	geometry(MultiPolygon,4326)	The selected inundation extent.

s_flood_grid_ar

Group: Static reference. **Geometry:** geometry **Lifecycle:** Per run.

Regular tiles (rectangles) to the limits of the merged flood inundation polygon in from the s_selected_flood_ar.

Table 20. Columns of s_flood_grid_ar

Column	Type	Description
id	bigint	Tile identifier and the only declared primary key in the schema. Appears as tile_id in the derived tables.
geom	geometry	Tile footprint.

s_flood_merge_ar

Group: Operational product. **Geometry:** geometry **Lifecycle:** Per run.

Reach-level extents dissolved into a single continuous flood polygon.

Table 21. Columns of s_flood_merge_ar

Column	Type	Description
tile_id	bigint	Tile from s_flood_grid_ar that the geometry belongs to.
geometry	geometry	Dissolved flood extent. No spatial index is declared on this table.
model_run_time	text	Model run the row belongs to.

s_flood_merge_by_tile_ar

Group: Operational product. **Geometry:** geometry (untyped). **Lifecycle:** Per run.

Tile-clipped delivery form of the merged flood inundation polygon

Table 22. Columns of s_flood_merge_by_tile_ar

Column	Type	Description
tile_id	bigint	Tile identifier.
model_run_time	text	Model run the row belongs to.
geometry	geometry	Flood inundation polygon limits by rectangular tile
is_real	integer	Flag to indicate when actual flood polygons exist within tile

s_flood_road_ln

Group: Operational product. **Geometry:** MultiLineString **Lifecycle:** Per run.

Roadway segments whose flooding threshold is reached by this forecast, carrying full segment geometry and the descriptive attributes needed to render and label them.

Table 15. Columns of s_flood_road_ln

Column	Type	Description
road_id	bigint	Roadway segment, referencing s_road_segment_ln.
nextgen_id	text	Controlling hydrofabric reach.
osm_id	text	OpenStreetMap identifier
fclass	text	Functional class (type of road as defined by OpenStreetMap)
name	text	Roadway name per OpenStreetMap
ref	text	Route designation per OpenStreetMap
min_flood_flow	bigint	Minimum flow at which this road segment experiences flooding from this ‘nextgen_id’ stream
max_flow	bigint	Forecast peak discharge on the ‘nextgen_id’ stream reach.
model_run_time	text	Model run the row belongs to.
geometry	geometry(MultiLineString,4326)	Full segment geometry.

s_flood_road_ln_tile

Group: Operational product. **Geometry:** geometry (untyped). **Lifecycle:** Per run.

The impacted-roads layer split along the tile mesh so it can be served and refreshed tile by tile.

Table 24. Columns of s_flood_road_ln_tile

Column	Type	Description
tile_id	bigint	Tile identifier.
road_id	bigint	Roadway segment.
nextgen_id	text	Controlling hydrofabric reach.
osm_id	text	OpenStreetMap identifier
fclass	text	OpenStreetMap class of road
name	text	Roadway name per OpenStreetMap
ref	text	Route designation per OpenStreetMap
min_flood_flow	bigint	Minimum flow at which this road segment experiences flooding from this ‘nextgen_id’ stream
max_flow	bigint	Forecast peak discharge on the ‘nextgen_id’ stream reach.
model_run_time	text	Model run the row belongs to.
geometry	geometry	Segment geometry clipped to the tile

s_flood_road_trim_ln

Group: Operational product. **Geometry:** geometry **Lifecycle:** Per run.

The tiled roads layer trimmed to the wet extent of the flood inundation area, so that a rendered line covers only the portion of the segment under water and its length can be reported.

Table 25. Columns of s_flood_road_trim_ln

Column	Type	Description
tile_id	bigint	Tile identifier.
road_id	bigint	Roadway segment.
nextgen_id	text	Controlling hydrofabric reach.
osm_id	text	OpenStreetMap identifier
fclass	text	OpenStreetMap class of road
name	text	Roadway name per OpenStreetMap
ref	text	Route designation per OpenStreetMap
min_flood_flow	bigint	Minimum flow at which this road segment experiences flooding from this 'nextgen_id' stream
max_flow	bigint	Forecast peak discharge on the 'nextgen_id' stream reach.
model_run_time	text	Model run the row belongs to.
length_ft	double precision	Length of the trimmed portion in feet.
geometry	geometry	Trimmed geometry.

s_bridge_warning_pnt

Group: Operational product. **Geometry:** Point, EPSG:4326. **Lifecycle:** Per run.

Per-structure warning record for the current run. Structurally this is s_bridge_pnt with six forecast columns appended, so the warning layer can be served without joining back to the inventory.

Table 26. Columns of s_bridge_warning_pnt

Column	Type	Description
uuid_bridge	text	Unique bridge identifier
BRDG_ID	text	SBNI bridge identification number
min_low_ch	double precision	Minimum elevation of the low chord of the bridge deck
min_ground	double precision	Minimum ground elevation on bridge cross section
min_overtop	double precision	Elevation where water overtops bridge
name	text	Roadway name per OpenStreetMap
ref	text	Route designation per OpenStreetMap
nhd_name	text	Stream named from National Hydrography Dataset (NHD)
model_run_time	text	Model run the row belongs to.
max_wse	double precision	Peak forecast water-surface elevation, obtained by evaluating the rating curve at the forecast peak discharge.
min_dist_to_low_ch	double precision	Smallest clearance between forecast water surface and minimum low chord
is_overtop	bigint	Flag indicating that the forecast water surface reaches the overtopping elevation.
depth_array	text	Hourly depth or clearance series over the forecast horizon, serialized into a text column rather than stored as an array.
url	text	Link to a per-structure detail view
geometry	geometry(Point,4326)	Structure location.

s_lwc_pnt

Group: Operational product. **Geometry**: Point, EPSG:4326. **Lifecycle**: Per run.

Workflow-specific low-water-crossing status output.

Table 16. **Columns of s_lwc_pnt**

Column	Type	Description
lwc_id	bigint	Low-water-crossing identifier and primary key
hydro_id	bigint	Source hydrologic identifier of the streamline

Column	Type	Description
model_id	bigint	Model identifier
feature_id	bigint	National Water Model feature supplying forecast flow
name	text	Low water crossing name
osm_id	text	OpenStreetMap identifier
fclass	text	Road functional class based on OpenStreetMap
q_0_5_ft	bigint	Discharges associated with approximately 0.5 feet of water depth
q_2_ft	bigint	Discharges associated with approximately 2 feet of water depth
q_5_ft	bigint	Discharges associated with approximately 5 feet of water depth
is_overtopped	bigint	1 when max_flow reaches q_overtopped, 0 otherwise, and null when no threshold exists.
max_flow	bigint	Maximum forecast flow for the linked feature and workflow
model_run_time	text	Model run the row belongs to.
geometry	Geometry (Point,4326)	Low water crossing location

Appendix B. SQL for the Operational Cycle

Once a new flow forecast is received, this is a typical SQL query needed to create the real time flood map products for FAST.

```
-- SETTING A TIMEOUT FOR HEAVY QUERRIES
-- revised 2025.06.10
SET statement_timeout TO '3min';

-- Acquire an advisory lock to prevent concurrent executions
SELECT pg_advisory_lock(20250628);

-- ITEM #0
-- Establish flows per stream

DROP TABLE IF EXISTS t_flow_per_nextgen;

CREATE TABLE t_flow_per_nextgen AS
WITH unique_ids AS (
    SELECT DISTINCT nextgen_id::text AS nextgen_id
    FROM s_flood_inundation_ar
),
crosswalked AS (
    SELECT u.nextgen_id, x.feature_id
    FROM unique_ids u
    JOIN t_nextgen_to_nwm x ON u.nextgen_id = x.nextgen_id
),
flows_with_array AS (
    SELECT
        c.nextgen_id,
        f.feature_id,
        f.model_run_time,
        ARRAY[
            flow_t00, flow_t01, flow_t02, flow_t03, flow_t04, flow_t05,
            flow_t06, flow_t07, flow_t08, flow_t09, flow_t10, flow_t11,
            flow_t12, flow_t13, flow_t14, flow_t15, flow_t16, flow_t17
        ] AS flow_array
    FROM crosswalked c
    JOIN t_flow_forecast f ON c.feature_id = f.feature_id
)
SELECT
    nextgen_id,
    feature_id,
    model_run_time,
    flow_array,
    (
        SELECT MAX(val) FROM unnest(flow_array) AS val
    ) AS max_flow,
    (
        SELECT i - 1
        FROM generate_subscripts(flow_array, 1) AS i
        WHERE flow_array[i] = (
            SELECT MAX(val) FROM unnest(flow_array) AS val
        )
        LIMIT 1
    ) AS max_hour
FROM flows_with_array;

CREATE INDEX idx_t_flow_per_nextgen_nextgen_id ON t_flow_per_nextgen(nextgen_id);


-- ITEM #1
-- select the appropriate flood area polygons
DROP TABLE IF EXISTS s_selected_flood_ar;

-- Composite index on nextgen_id and flow DESC helps with both filtering and ordering
CREATE INDEX IF NOT EXISTS idx_flood_inundation_ar_nextgen_flow
ON s_flood_inundation_ar(nextgen_id, flow DESC);

CREATE TABLE s_selected_flood_ar AS
SELECT DISTINCT ON (t.nextgen_id)
    t.nextgen_id,
    t.max_flow,
    t.model_run_time,
    t.max_hour,
```

```
    s.flow,
    s.geometry
FROM t_flow_per_nextgen t
JOIN s_flood_inundation_ar s
  ON s.nextgen_id::text = t.nextgen_id
 WHERE s.flow <= t.max_flow
ORDER BY t.nextgen_id, s.flow DESC;

-- ITEM #2
-- select the appropriate flooded road lines
DROP TABLE IF EXISTS s_flood_road_ln;

CREATE TABLE s_flood_road_ln AS
WITH below_trigger_roads AS (
    SELECT
        ft.road_id,
        ft.nextgen_id,
        ft.min_flood_flow,
        mf.max_flow,
        mf.model_run_time
    FROM
        t_road_flood_trigger ft
    INNER JOIN
        t_flow_per_nextgen mf
        ON ft.nextgen_id = mf.nextgen_id
    WHERE
        ft.min_flood_flow < mf.max_flow
),
joined_roads AS (
    SELECT
        s.geometry,
        s.osm_id,
        s.fclass,
        s.name,
        s.ref,
        s.road_id,
        btr.nextgen_id,
        btr.min_flood_flow,
        btr.max_flow,
        btr.model_run_time
    FROM
        below_trigger_roads btr
    JOIN
        s_road_segment_ln s
        ON btr.road_id = s.road_id
),
deduped_by_attributes AS (
    SELECT DISTINCT ON (
        osm_id, fclass, name, ref, road_id, nextgen_id,
        min_flood_flow, max_flow, model_run_time
    )
    geometry,
    osm_id,
    fclass,
    name,
    ref,
    road_id,
    nextgen_id,
    min_flood_flow,
    max_flow,
    model_run_time
    FROM joined_roads
    ORDER BY
        osm_id, fclass, name, ref, road_id, nextgen_id,
        min_flood_flow, max_flow, model_run_time
),
deduped_by_geometry AS (
    SELECT DISTINCT ON (geometry)
        geometry,
        osm_id,
        fclass,
        name,
        ref,
        road_id,
        nextgen_id,
        min_flood_flow,
        max_flow,
        model_run_time
    FROM deduped_by_attributes
```

```
    ORDER BY geometry
)
SELECT * FROM deduped_by_geometry;

-- ITEM #3
-- create a grid of tiles over the s_selected_flood_ar

CREATE TABLE s_flood_grid_ar AS
WITH ext AS (
    SELECT ST_SetSRID(ST_Extent(geometry)::box2d, 4326) AS geom_extent
    FROM s_flood_merge_ar
),
grid AS (
    SELECT ST_Collect(sg.geom) AS geom_collection
    FROM ext, ST_SquareGrid(0.25, ext.geom_extent) AS sg(geom)
),
dumped AS (
    SELECT (ST_Dump(geom_collection)).geom
    FROM grid
)
SELECT
    row_number() OVER () AS id,
    dumped.geom
FROM dumped;

ALTER TABLE s_flood_grid_ar ADD PRIMARY KEY (id);

CREATE INDEX idx_s_flood_grid_ar_geom ON s_flood_grid_ar USING GIST (geom);

-- ITEM #4
-- merge flood polygons by tiles
-- heavy calculation
DROP TABLE IF EXISTS s_flood_merge_by_tile_ar;

-- Create merged flood polygons by tile
CREATE TABLE s_flood_merge_by_tile_ar AS
SELECT
    g.id AS tile_id,
    MIN(sfa.model_run_time) AS model_run_time,
    ST_Multi(ST_Union(sfa.geometry)) AS geometry,
    1::integer AS is_real
FROM
    s_flood_grid_ar g
JOIN
    s_selected_flood_ar sfa
ON
    ST_Intersects(g.geom, sfa.geometry)
GROUP BY
    g.id;

-- ITEM #5
-- Intersect and index s_flood_road_ln by tile_id from s_flood_grid_ar
DROP TABLE IF EXISTS s_flood_road_ln_tile;

CREATE TABLE s_flood_road_ln_tile AS
SELECT
    g.id AS tile_id,
    r.osm_id,
    r.fclass,
    r.name,
    r.ref,
    r.road_id,
    r.nextgen_id,
    r.min_flood_flow,
    r.max_flow,
    r.model_run_time,
    ST_Intersection(r.geometry, g.geom) AS geometry
FROM
    s_flood_grid_ar g
JOIN
    s_flood_road_ln r
ON
    ST_Intersects(r.geometry, g.geom);

CREATE INDEX idx_s_flood_road_ln_tile_geom ON s_flood_road_ln_tile USING GIST (geometry);
CREATE INDEX idx_s_flood_road_ln_tile_tile_id ON s_flood_road_ln_tile (tile_id);

-- ITEM #6
-- Create the clipped flooded road lines per tile
```

```
DROP TABLE IF EXISTS s_flood_road_trim_ln;

CREATE TABLE s_flood_road_trim_ln AS
SELECT
    r.tile_id,
    r.osm_id,
    r.fclass,
    r.name,
    r.ref,
    r.road_id,
    r.nextgen_id,
    r.min_flood_flow,
    r.max_flow,
    r.model_run_time,
    ST_Intersection(r.geometry, f.geometry) AS geometry,
    ROUND(
        ST_Length(
            ST_Transform(
                ST_Intersection(r.geometry, f.geometry), 3857
            )
        ) * 3.28084
    ) AS length_ft
FROM
    s_flood_road_ln_tile r
JOIN
    s_flood_merge_by_tile_ar f
    ON r.tile_id = f.tile_id
WHERE
    ST_Intersects(r.geometry, f.geometry);

CREATE INDEX idx_s_flood_road_trim_ln_geom ON s_flood_road_trim_ln USING GIST (geometry);
CREATE INDEX idx_s_flood_road_trim_ln_tile_id ON s_flood_road_trim_ln (tile_id);


-- ITEM #7
-- Clip the flood inundation to each tile's limit
DROP TABLE IF EXISTS s_flood_merge_ar;

CREATE TABLE s_flood_merge_ar AS
SELECT
    f.tile_id,
    ST_Intersection(f.geometry, g.geom) AS geometry
FROM
    s_flood_merge_by_tile_ar f
JOIN
    s_flood_grid_ar g ON f.tile_id = g.id
WHERE
    ST_Intersects(f.geometry, g.geom);

-- Assign SRID if missing (only needed if ST_SRID returns 0)
UPDATE s_flood_road_ln SET geometry = ST_SetSRID(geometry, 4326) WHERE ST_SRID(geometry) = 0;
UPDATE s_flood_merge_ar SET geometry = ST_SetSRID(geometry, 4326) WHERE ST_SRID(geometry) = 0;

-- Create the new table with a single row containing the first model_run_time value from t_flow_forecast
DROP TABLE IF EXISTS t_current_forecast;

CREATE TABLE t_current_forecast AS
SELECT model_run_time
FROM t_flow_forecast
LIMIT 1;

-- Need to add col to s_flood_merge_ar
ALTER TABLE s_flood_merge_ar
ADD COLUMN model_run_time TEXT;

-- Insert one row if s_flood_merge_ar is empty
INSERT INTO s_flood_merge_ar (model_run_time) SELECT NULL WHERE NOT EXISTS (SELECT 1 FROM s_flood_merge_ar);

UPDATE s_flood_merge_ar
SET model_run_time = (SELECT model_run_time FROM t_current_forecast LIMIT 1);

-- Release the advisory lock manually (optional, as it auto-releases at session end)
SELECT pg_advisory_unlock(20250628);
```