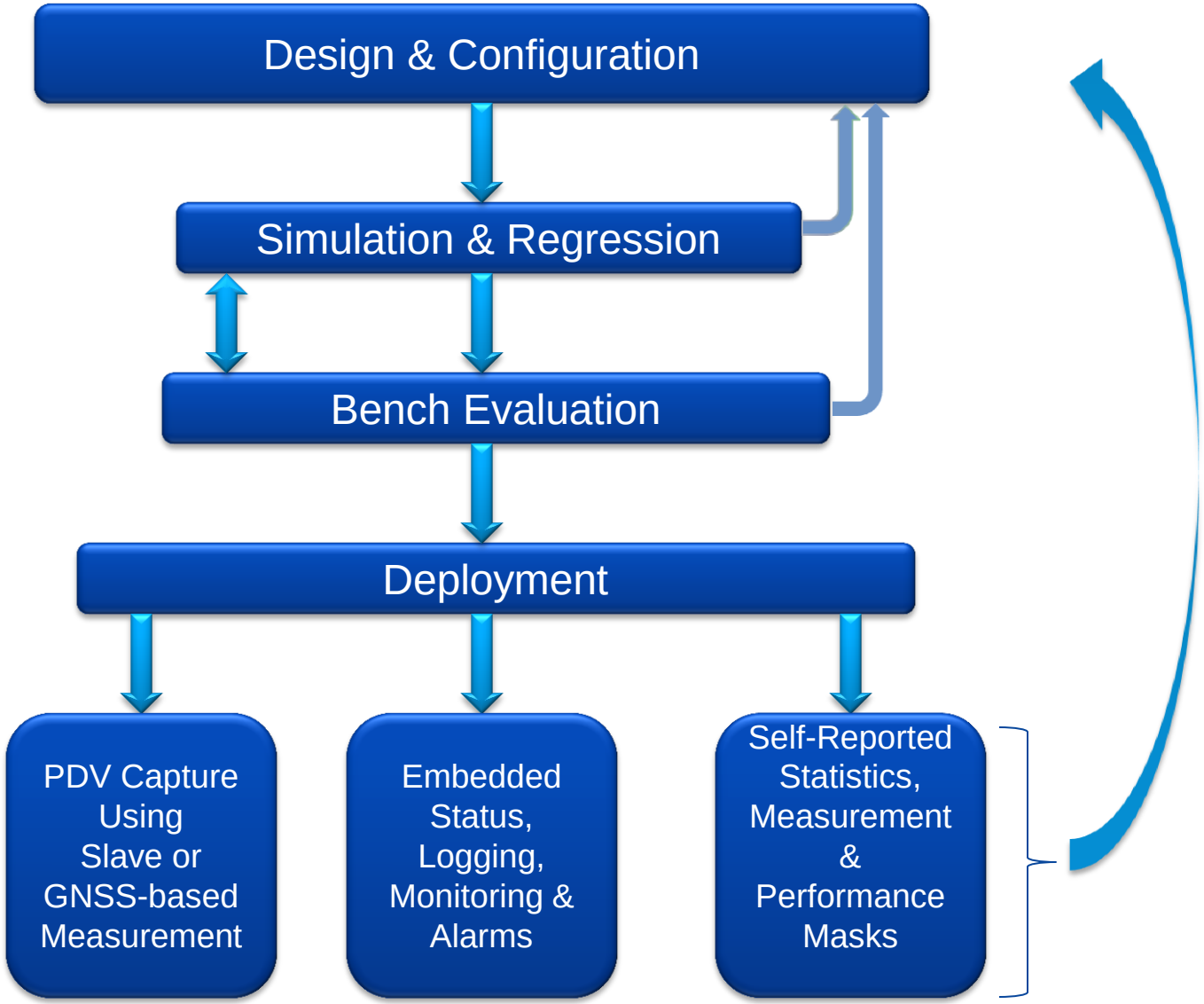
# Synchronization Validation through Automation, Replay, Modeling & Simulation

Tariq Haddad, *Sr. Algorithm Engineer*
Microsemi Corp
+1-613-270-7228
Tariq.haddad@microsemi.com

# Introduction

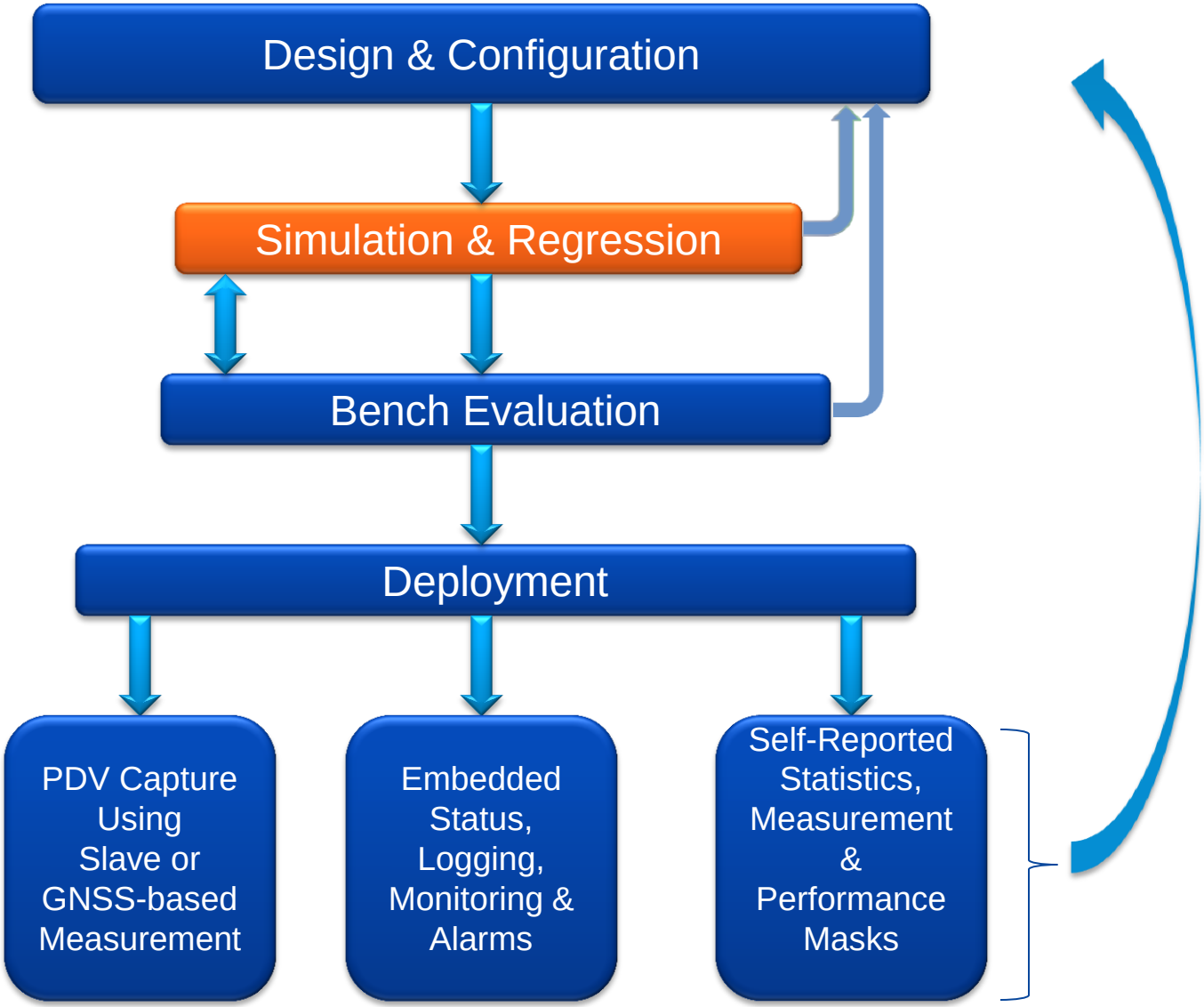
- Continuous Development & Deployment Flow
  - Design & Configuration.
  - Simulation & Regression.
  - Bench Testing.
  - Deployment
    - PDV Captures.
    - Monitoring & Alarms.
    - Metrics & Performance Self-Reporting.
- Simulation
  - Input Choices (Models vs. Replay).
  - Implementation Considerations.
  - Hybrid and BC reference chains.
- Simulation Validation vs. Bench Validation
- Summary

# Sync Development & Deployment Flow



# Sync Development & Deployment Flow

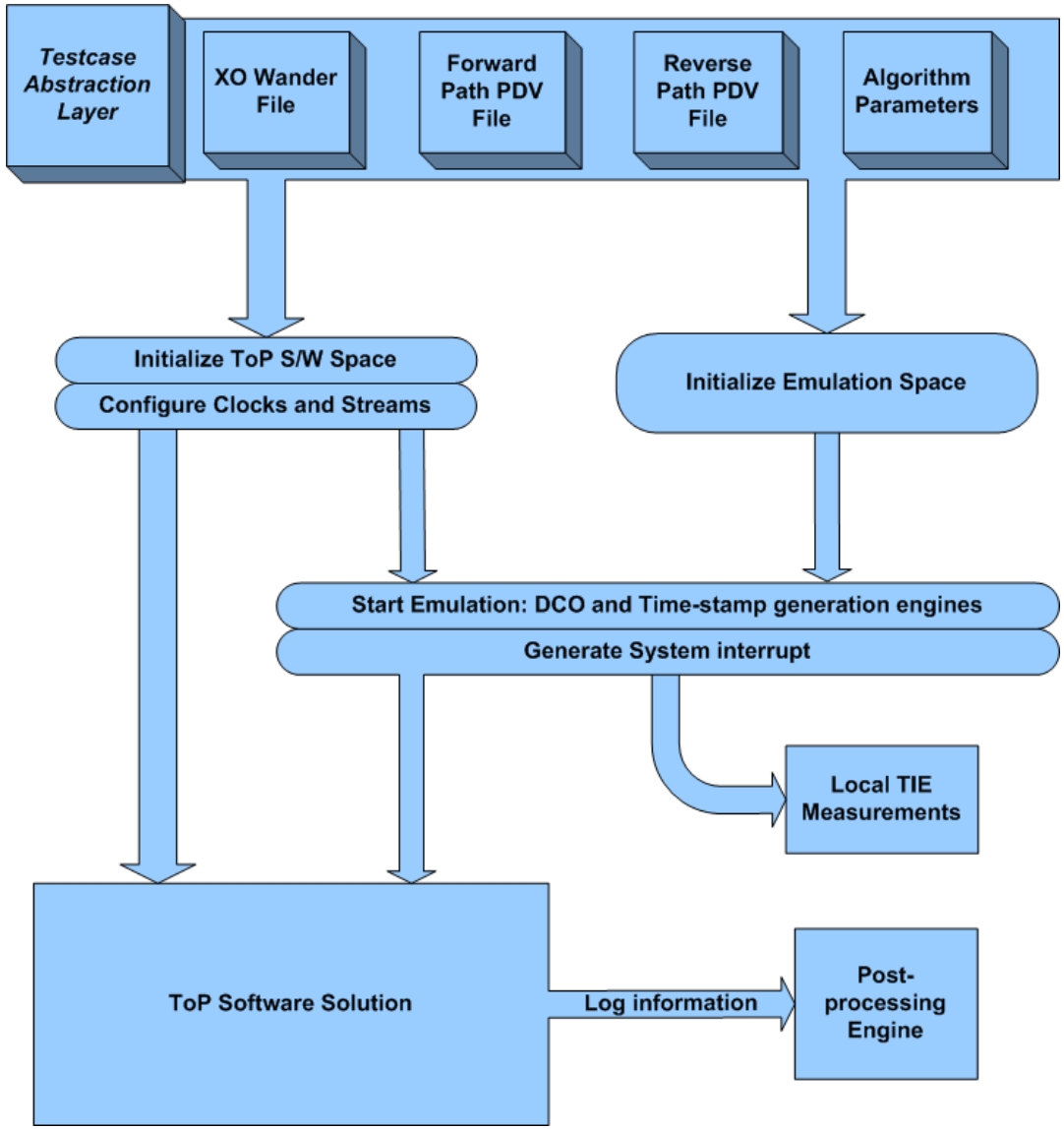
- Simulate real customer software in a controlled system emulation environment.
- Use this environment to run batches of standard testcases.



# Sync Development & Deployment Flow: Simulation

- Benefits of simulation
  - Speed.  Runs at the packet rate by using an inter-even predictive approach, thus achieving phenomenal simulation speeds. A typical case can run a 1 hour lab test scenario in ~1 second.
  - Repeatability.  Control external variables such as PDV & oscillator.
- Servo choices
  - Mathematical model.
  - Real implementation code.
- PDV choices
  - Mathematical/Standard models.
  - Replay of captured data.
- Oscillator choices
  - Mathematical model (including ageing, temperature, etc.)
  - Replay of captured data.
- Configuration choices
  - Default parameters.
  - Customization of parameters to identify optimal performance.

# Sync Development & Deployment Flow: Simulation



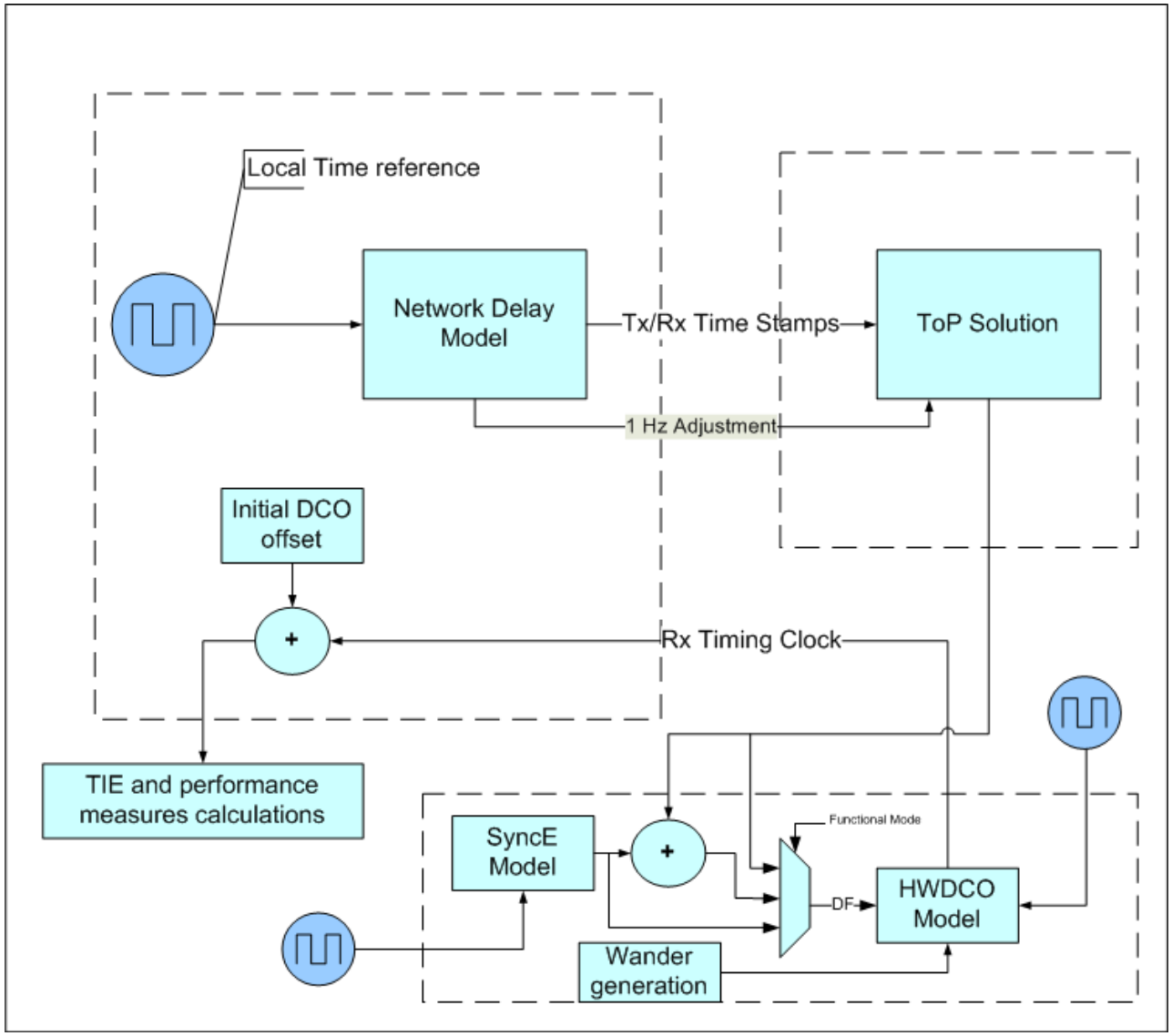
# Sync Development & Deployment Flow: Simulation Environment Abstraction Layer

- Client oscillator file, sampled at the forward path packet rate.
  - Real capture of XO sampled wander variations, which is then added to the system master clock frequency.
  - Other temperature effect variations is modeled and added to variations caused by the local XO.
  - Simple sinusoidal and ramp XO models are used for special test cases like BW measurements.
- The forward and reverse PDV files are either G.8261 standard-specific or a replay test-specific captures.
- Uniform and non-uniform Tx/Rx-scheduling file. (optional)
- Programmable initial time and frequency offset between master and client nodes.
- Having a library of network/case-specific files allows for running batches of 100s of test cases for regression testing in record times (A few hours testcase can run in seconds).

# Sync Development & Deployment Flow: Simulation Hybrid (SyncE+1588) Modeling

- A SyncE programmable SPLL model is used to emulate the electric mode.
- The default SPLL resolution is NTP (2.3283064e-10 sec), with input 10 GHz clock. (High resolution)
- To solve multi-rate IEEE1588 and SyncE operation, the simulator system implements a hardware DCO model and an identical shadow SyncE DCO model. At the same time instance, both DCOs should have exact same values.

# Sync Development & Deployment Flow: Simulation Functional Diagram

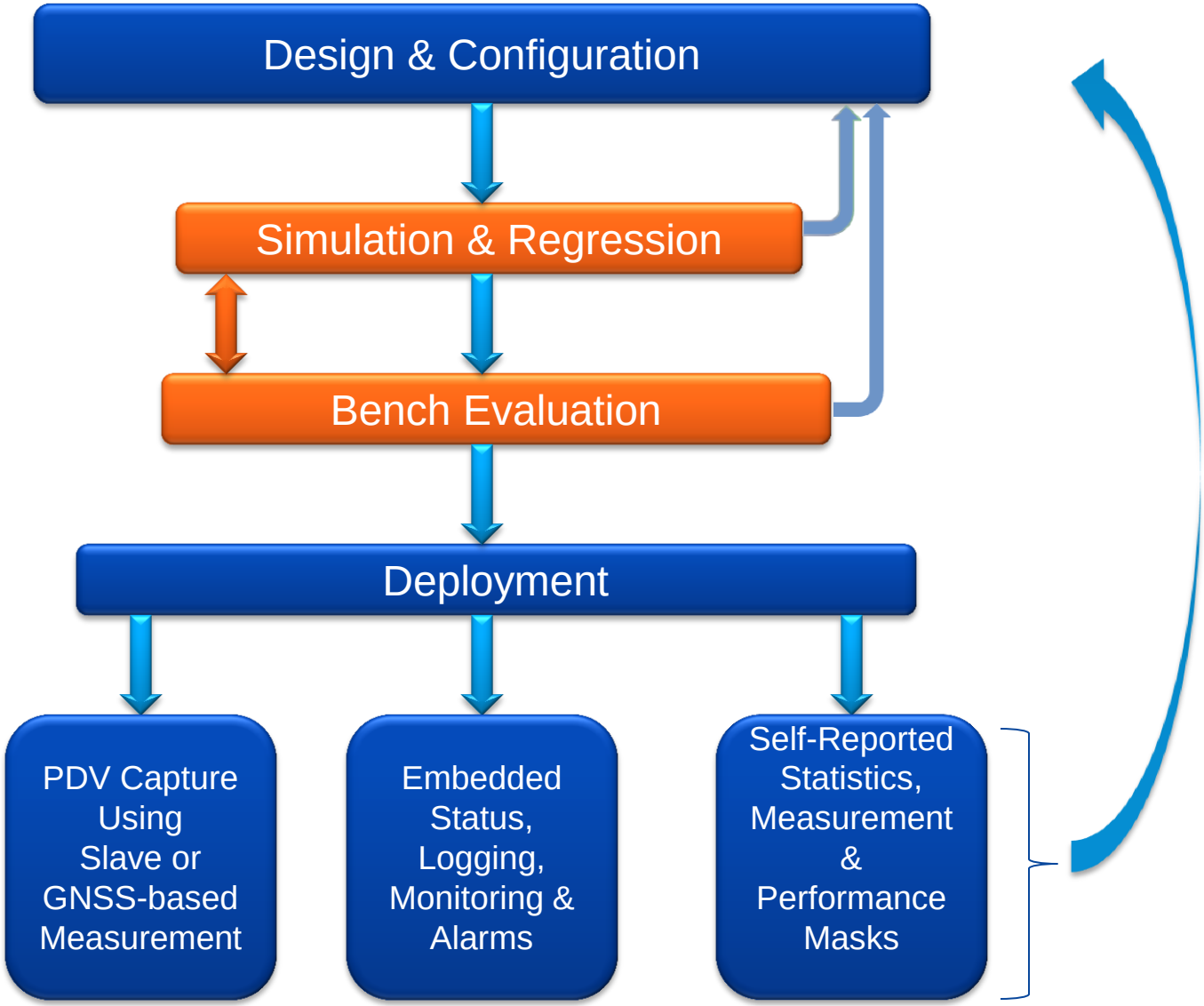


# Sync Development & Deployment Flow: Simulation BC Reference Chain Model

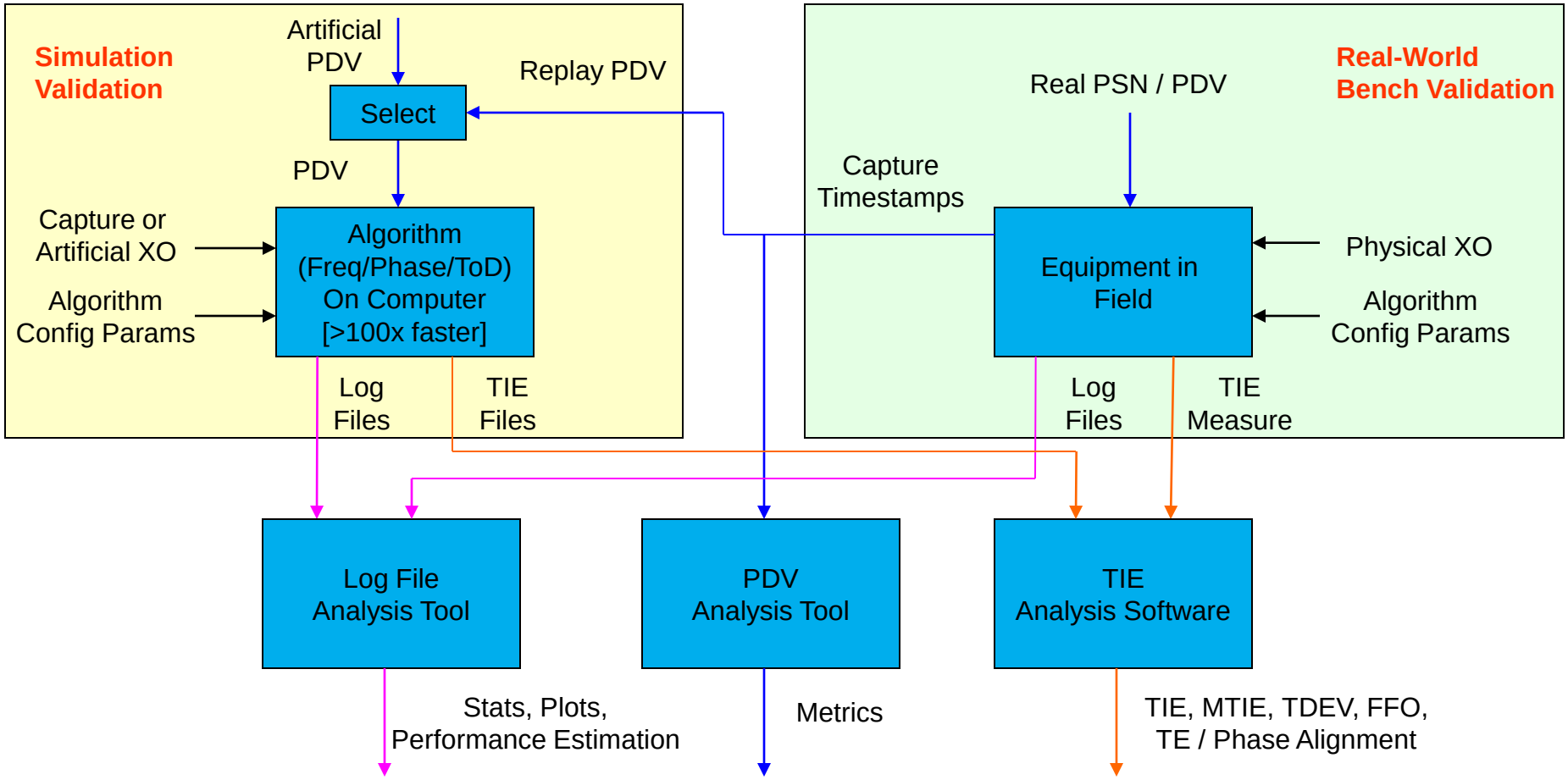
- An N-node BC reference chain was modeled in the simulator.
- The variations generated at node n-1 is used as to replace the master node xo in the classical one node scenario. Note that node n will act as a LPF to n-1 variations.
- Processing is done sequentially over the time window. Finish and buffer the variations from node n-1, then use that to process node n.
- The same simple PDV file can be used in this case with random initial offsets to provide the required de-correlation.
- Similarly, the same xo wander file can be used with random offsets for multi-node processing.
- Different performance measures with respect to the initial master node are provided.

# Sync Development & Deployment Flow

- After initial simulator testing, bench evaluation ensures the product is ready for field deployment.



# Sync Development & Deployment Flow: Simulation vs. Bench



# Sync Development & Deployment Flow: Bench vs. Simulator Evaluation

## Simulator Validation

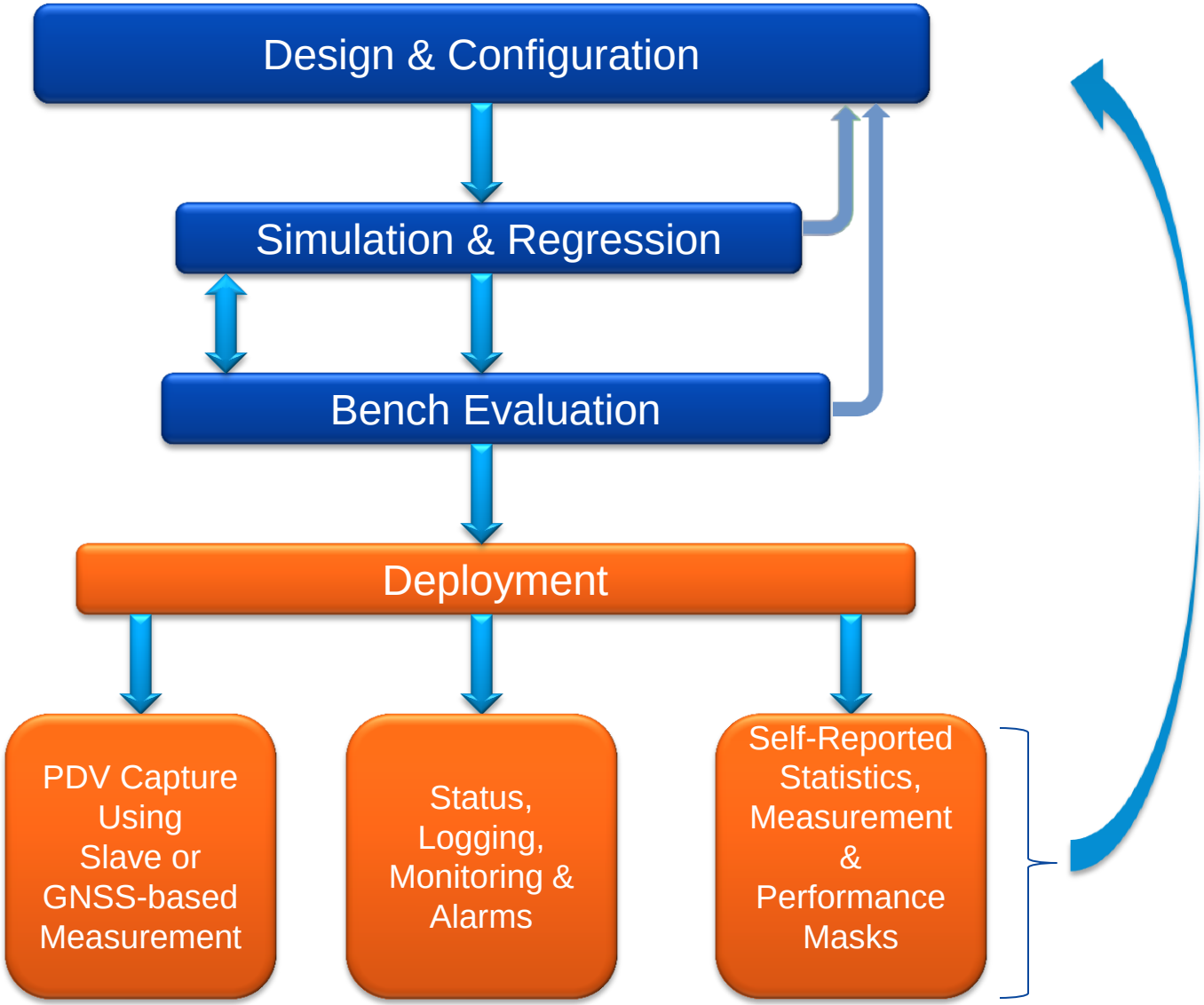
- Fast (1 hour real time in 1 second)
- Portable (can run anywhere without slave equipment)
- Fast batch modes
- Predictable, you can repeat the same run and reproduce errors.
- Essential and convenient tool for algorithm development for easy debugging

## Bench Validation

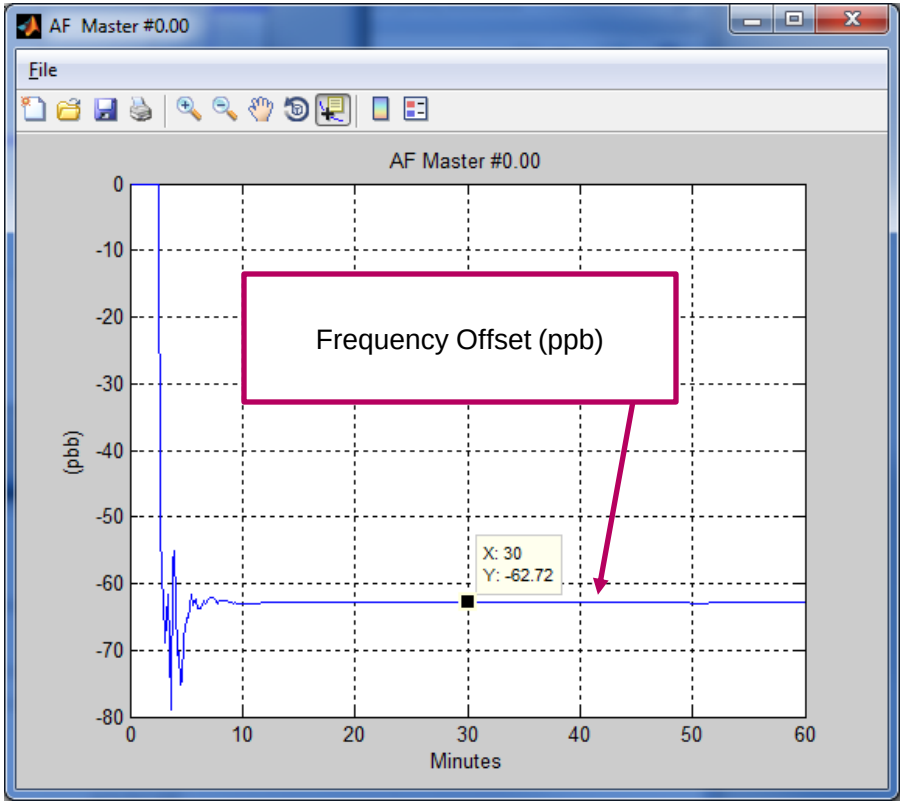
- Real Time
- Obviously not portable.
- Can be setup for batch testing, whoever, very time consuming.
- Non-stationary. Several environmental variables.
- Almost impossible to debug causes of algorithmic and implementation bugs.

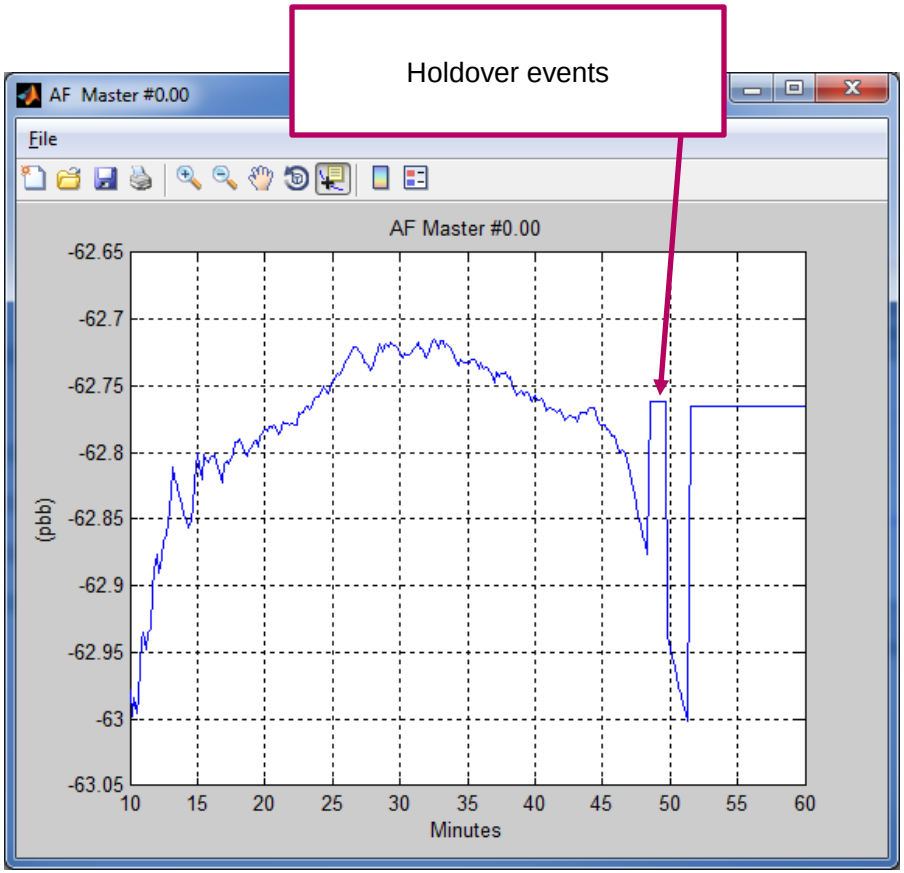
# Sync Development & Deployment Flow

- Deployment
  - Adaptive servo to respond to network characteristics
  - On-going monitoring and self-reporting of performance
  - Data collection for future design improvements
- PDV Capture
  - Self-capture (subject to local oscillator variation)
  - GNSS based capture
- Monitoring
  - Reporting of status
  - Reporting of alarms
  - Logging of detailed information for later review/collection in event of failure or alarm
- Performance Masks
  - Self-reporting of performance against standard metrics (e.g. clusterTDEV, FFO)

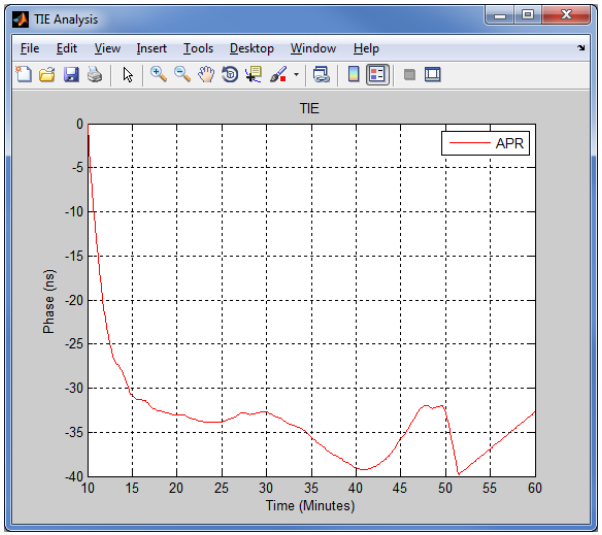


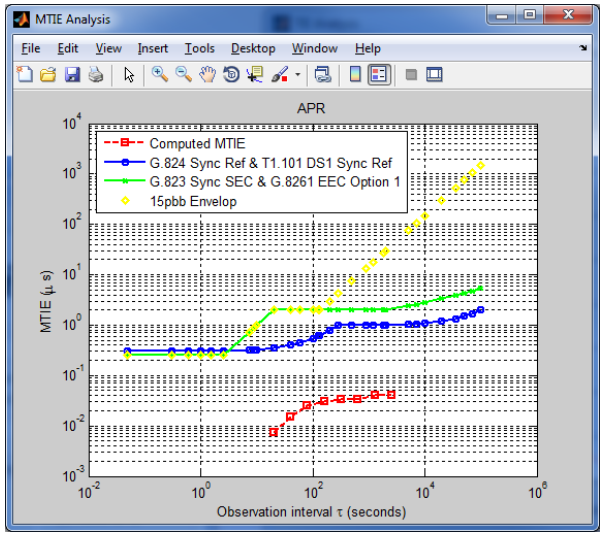
# Sync Development & Deployment Flow: Self-Reported Status & Metrics

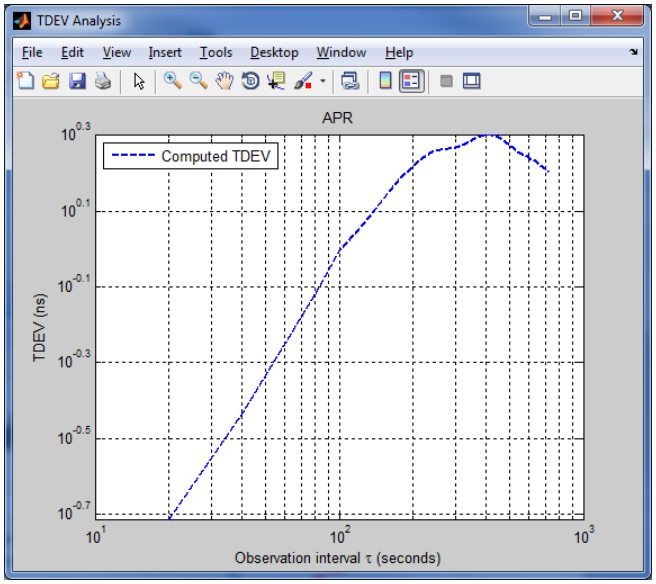


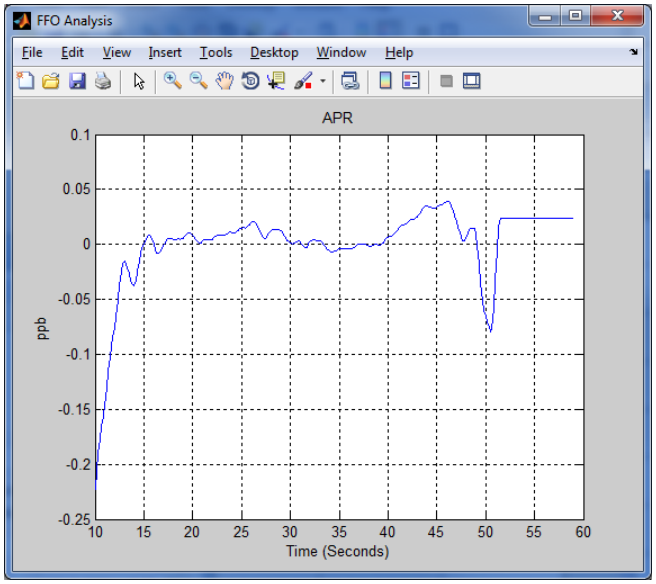



# Sync Development & Deployment Flow: Self-Reported Status & Metrics



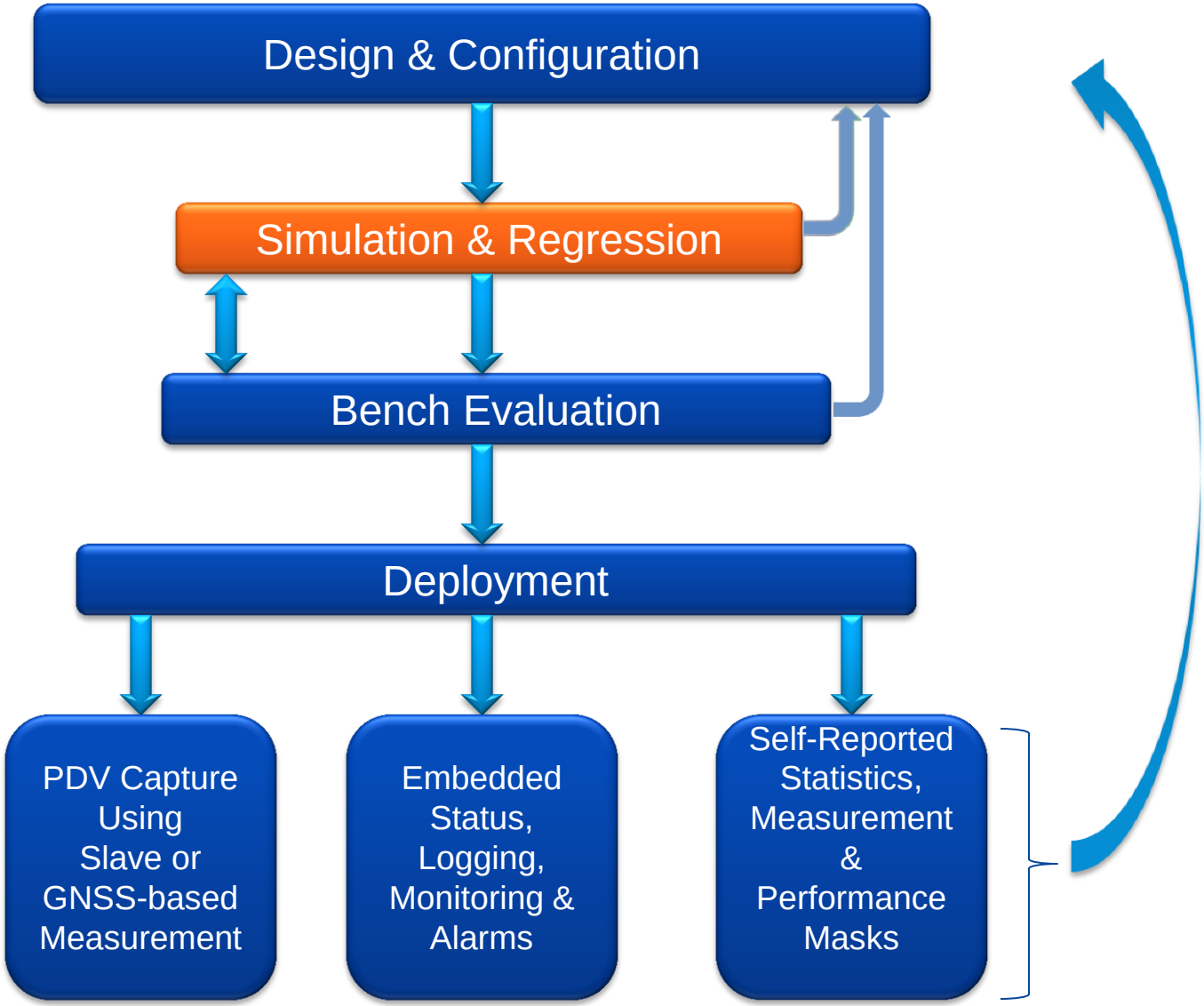



# Sync Development & Deployment Flow: Summary

- Research & Design is only the first step for successful IEEE1588 synchronization
- Simulation & regression is critical to debugging current generation and continuous improvement in next generation of servo design
- Field deployments can capture and feedback a lot of useful information for incorporation into next generation design



Microsemi

# Thank-you