

WHITE PAPER

How Do You Test Smart Meters (AMI Meters) Differently?

A functional test plan for AMI meters: from accuracy to firmware, communications, and events

Executive Summary

An AMI meter has to be accurate, and it also has to report interval data, accept new rate programs over the air, take firmware updates, disconnect on command, and flag tampering and outages. Utilities still test AMI meters for accuracy under ANSI C12.1, exactly as they would any other meter. The difference is that accuracy becomes one campaign inside a larger functional test plan.

A good functional test plan is built backward from the goal, a meter the utility can trust in the field:

1. **Define objectives.** Map each test to something the utility needs the meter to do.
2. **Standardize the bench.** Pin down every piece of equipment so results repeat.
3. **Build test cases.** Use one consistent structure, and group cases into campaigns.
4. **Write acceptance criteria.** Make every pass or fail exact and measurable.
5. **Automate.** Let the bench run the steps and check the results.
6. **Execute and report.** Schedule for coverage and keep a defensible record.

1. Accuracy Testing vs. Functional Testing

Accuracy testing asks one question: does the meter measure energy correctly? **Functional testing** asks a bigger one: does the meter do everything the utility needs it to do, correctly, every time, and under the conditions it will actually face?

A modern meter is a networked computer that bills customers, reports events, and acts on remote commands. Every one of those behaviors can fail, and a functional defect can over- or under-charge every customer on a rate program. Finding it on the test bench costs almost nothing compared with finding it after deployment. A written, repeatable plan also lets the utility, the meter manufacturer, and a third party reproduce the same result.

2. What Stays the Same: Accuracy Testing

An AMI meter is held to the same ANSI C12.1 accuracy classes as any solid-state meter. A standards-based accuracy campaign covers:

- No-load (creep) and starting load
- A load-performance curve across the meter's current range
- A power-factor sweep: unity, lagging, and leading
- Variation of voltage and frequency
- Element tests on polyphase meters

Accuracy is read locally, through the optical port, so the communications network stays out of the loop: the test checks the metrology, not the radio. The test equipment still needs to be at least four times more accurate than the meter. Our white paper, [How Accurate Does Meter Test Equipment Need to Be?](#), covers that ratio.

Automated test boards such as the [WECO 4050X, 4150X, and 4330X](#) run this campaign for single-phase and polyphase meters, with a removable RADIAN reference standard at $\pm 0.04\%$, $\pm 0.02\%$, or $\pm 0.01\%$. The [WECO 2051X](#) covers single-position testing.

One point is easy to miss: an AMI meter reports its own readings, but those readings come from the meter's own metrology. Only an independent test against a traceable reference standard shows whether they are right.

3. Start from Utility Use Cases

Every test should trace back to something the utility needs the meter to do. A typical set of use cases:

- Remote meter reading of registers and interval data
- Remote configuration, such as new time-of-use programs
- Remote firmware upgrades
- Health and status events
- Theft and tamper detection
- Remote connect and disconnect
- Outage detection and reporting
- Power quality reporting
- Local optical reads that agree with the display

Each use case becomes an observable behavior. For example, "remote disconnect" becomes: on a disconnect command, the meter opens the service switch and reports disconnected status on both the network read and the display. That observability is what makes it testable.

4. Standardize the Test Bench

Repeatability is the whole game. If the equipment isn't pinned down by make, model, firmware version, and configuration, two runs of the same test can disagree, and nobody will know why. A standard bench includes:

- The meter under test and its communication module, at a defined form and firmware version
- A test board that generates voltage, current, and phase angle, with a reference standard for accuracy
- A head-end system, or an emulator of one, for remote reads, configuration, and firmware
- A control PC and test software, such as [WATT-Net](#)
- Fixtures for tamper, tilt, and magnet tests
- An optical probe for local reads

Define the common equipment and the common starting state once, and reference them in every test case. The starting state is typically: safety procedures followed, meter seated, optical probe attached, equipment powered and communicating, lab temperature and humidity in range, and nominal voltage applied with no load.

Before testing, meters are warmed up and given a first functional check. The [WECO 9000](#) warm-up board provides safe functional validation of AMI meters before accuracy testing, and the [WECO 9200](#) warm-up load board brings meters to a stable operating state, with variable voltage and current and an isolated phantom load for each socket.

5. Structure Every Test Case the Same Way

Every test case follows the same six-part structure. Consistency makes cases easy to write, review, and automate:

Part	What it contains
Objective	One sentence: the behavior being validated
Applicable forms	Which meter forms the case runs on
Equipment	"Common" plus anything special
Initial conditions	The known starting state
Steps	Numbered, exact actions: set, apply, wait, read
Acceptance criteria	Exact expected values, events, and tolerances

Write steps as deterministic actions: set the clock to a specific time, apply a defined voltage, current, and power factor, wait a defined interval, then read a defined value. Because the inputs are exact, the expected results can be calculated in advance, which is what makes a clean pass or fail possible.

Worked test case: remote reconfiguration

Part	Example
Objective	Confirm a new program can be pushed over the air, the meter's behavior changes, and an invalid program is rejected
Applicable forms	All supported forms; full case on the representative residential and commercial forms
Equipment	Common, plus three program files: the baseline, an invalid program (missing ID), and a modified program (new interval length)
Initial conditions	Common; baseline program loaded
Steps	1. Read the program ID and interval length. 2. Apply a defined load; read registers and intervals. 3. Push the invalid program. 4. Read the status. 5. Push the modified program. 6. Apply load; read interval data. 7. Restore the baseline program.
Acceptance criteria	Baseline matches the expected program; the invalid program is rejected, an error flag is set, and the program is unchanged; the modified program logs a reprogrammed event and the new interval length takes effect; the baseline is restored

The most valuable step is the negative test in the middle. A meter that silently accepts a bad rate program is a billing incident waiting to happen.

6. Organize Test Cases into Campaigns

Group related cases into campaigns, one per capability, so each can be run, scheduled, and reported on its own. The most important campaigns:

- **Register and interval data.** Apply exactly known loads for exact times, then read every register and interval back through both the optical port and the network. Functional checks use a generous tolerance (around $\pm 10\%$) to absorb bench and timing error; the tight ANSI limits live in the accuracy campaign.
- **Interval edge cases.** Outages that start mid-interval or span a whole interval, clock changes, and daylight saving time transitions. Each interval must carry the right value and the right quality flag (outage, skip, time change, or partial), because billing systems use those flags to decide whether to trust or estimate the data.
- **Remote reconfiguration.** A valid program changes the meter's behavior; an invalid one is rejected.
- **Firmware upgrade.** Test a valid image, a file with a bad signature, an incomplete file, and a power outage in the middle of the upgrade. The meter should reject bad files, recover cleanly, and change its version only on a valid, complete image.
- **Communications.** Read the same data locally and remotely and require the two to agree value by value. The data lives in standard ANSI C12.19 tables. It is read locally through the optical port under ANSI C12.18, and remotely over the network, often using ANSI C12.22. Any disagreement points to a protocol or mapping defect.

- **Events and status.** Tamper tests use a magnet, a tilt fixture, and an inverted meter. Remote disconnect must change the service status on both the network read and the display. Outage tests check counters and time recovery; power quality tests check that sags, swells, and harmonics trigger the right events at the right thresholds. The pass criterion is an exact event or status bit, not "something happened."
- **Accuracy and power factor.** The standards-based campaign described in section 2.

Across campaigns, cover four kinds of behavior: normal operation, boundary and edge cases, faults and abuse, and invalid input. Teams routinely test the normal path and forget the last one.

7. Write Acceptance Criteria That Decide Themselves

A good acceptance criterion is:

- **Measurable:** a specific field, event, or status bit
- **Deterministic:** one, and only one, correct outcome
- **Tolerance-bounded:** numeric values state an explicit $\pm$ range
- **Machine-checkable:** a script can decide pass or fail with no human
- **Traceable:** it ties back to an objective and a use case

Tolerances exist because every reading carries error from the test board's load generation and stabilization, the meter's accuracy class, and timing between the scripts and the equipment. If a person has to interpret the result, the criterion isn't ready to automate.

8. Automate and Schedule for Coverage

With the bench fixed and every step and criterion exact, the plan becomes an automated suite. It runs the same way every time, runs many forms and campaigns unattended, re-runs in full on every new firmware build, and produces a record the utility, the manufacturer, and a third party can all reproduce.

Coverage is meter forms multiplied by campaigns, so balance it against time. Run the full suite on a few representative forms and targeted campaigns on the rest. For example:

Meter form	Data	Edge cases	Reconfig	Firmware	Comms	Events	Accuracy & PF
Representative residential (e.g., 2S)	✓	✓	✓	✓	✓	✓	✓
Representative commercial (e.g., 9S)	✓	✓	✓	✓	✓	✓	✓
Other forms (e.g., 12S, 16S, 45S)	✓	✓					✓

Budget for long tests: some outage scenarios leave a meter powered down for many hours. With this approach, a dozen or more forms can be covered in about two weeks rather than months, with every result logged and traced to its criterion.

9. Keep a Defensible Record

Automation produces a lot of data, and the record is what makes the testing defensible. Every result should tie back to the meter, the test case, the bench configuration, and the firmware version it ran on, and it should stay with the meter's history after the meter leaves the shop.

On RADIANT and WECO test systems, [WATT-Net](#) is the software that keeps that record. It runs reusable test sequences on the Test Board, stores every result in one central database, and keeps the history with each meter. Its modules carry the record across the meter's life:

- [Meter Shop Operations](#): acceptance testing of incoming AMI meter shipments using ANSI Z1.4 or ANSI Z1.9 sampling tables, first article verification, and workstation alerts for quarantined or red-tagged meters.
- [Sample and Periodic Testing](#): in-service sample and periodic testing programs once the meters are deployed.
- [Equipment Support](#): return tracking for meters that fail, across manufacturers, so warranty dates aren't missed, and calibration scheduling for the reference standards on the bench.
- [Asset Lifecycle Management](#): tracking from purchase to retirement.

With the Listener integration, test results can also be sent to an ERP or other business system. Our white paper, [WATT-Net as the Meter Asset Management System of Record](#), covers this in more detail.

10. Why Utilities Build a Meter Farm

Some behaviors take days or weeks to show up: firmware effects, head-end integration, long outages, and burn-in. That is why many utilities keep a meter farm: a controlled population of live, communicating meters where changes can be tested before they reach customers.

The [WECO SMART AMI Meter Farm](#) gives a utility row-level remote control of voltage, current, and energy flow. Utilities use it for pre-deployment validation, firmware validation, head-end integration, solar and DER interconnection, EV charging and time-of-use rate validation, and long-duration burn-in. Our case study, [WECO SMART AMI Meter Farm](#), shows how a large community-owned utility on the West Coast uses one.

Job	Tool
Accuracy testing, single-phase and polyphase	WECO 4050X, 4150X, and 4330X
Single-position accuracy testing	WECO 2051X
AMI functional checks before accuracy testing	WECO 9000 warm-up board
Stable warm-up with phantom load per socket	WECO 9200 warm-up load board
Firmware, head-end, DER, and long-duration testing	WECO SMART AMI Meter Farm
Test sequences, results, acceptance sampling, and return tracking	WATT-Net software

11. Conclusion

Testing an AMI meter starts with the same accuracy test as any other meter, then asks the larger question: does it do everything the utility needs, every time, under real conditions? A plan that starts from use cases, fixes the bench, writes every test case the same way, and makes every criterion exact can be automated, repeated, recorded, and defended, on every form and every firmware build.

About Radian Research

Radian Research, Inc., headquartered in Lafayette, Indiana, is a 100% employee-owned company that has manufactured precision electricity measurement and test equipment for electric utilities, meter shops, and calibration laboratories for more than 35 years. Over 300 million meters produced in the past 30 years are referenced to a RADIANT standard. RADIANT's instruments are traceable to NIST through the company's own ISO/IEC 17025-accredited metrology laboratory.