

Automating smart meters testing

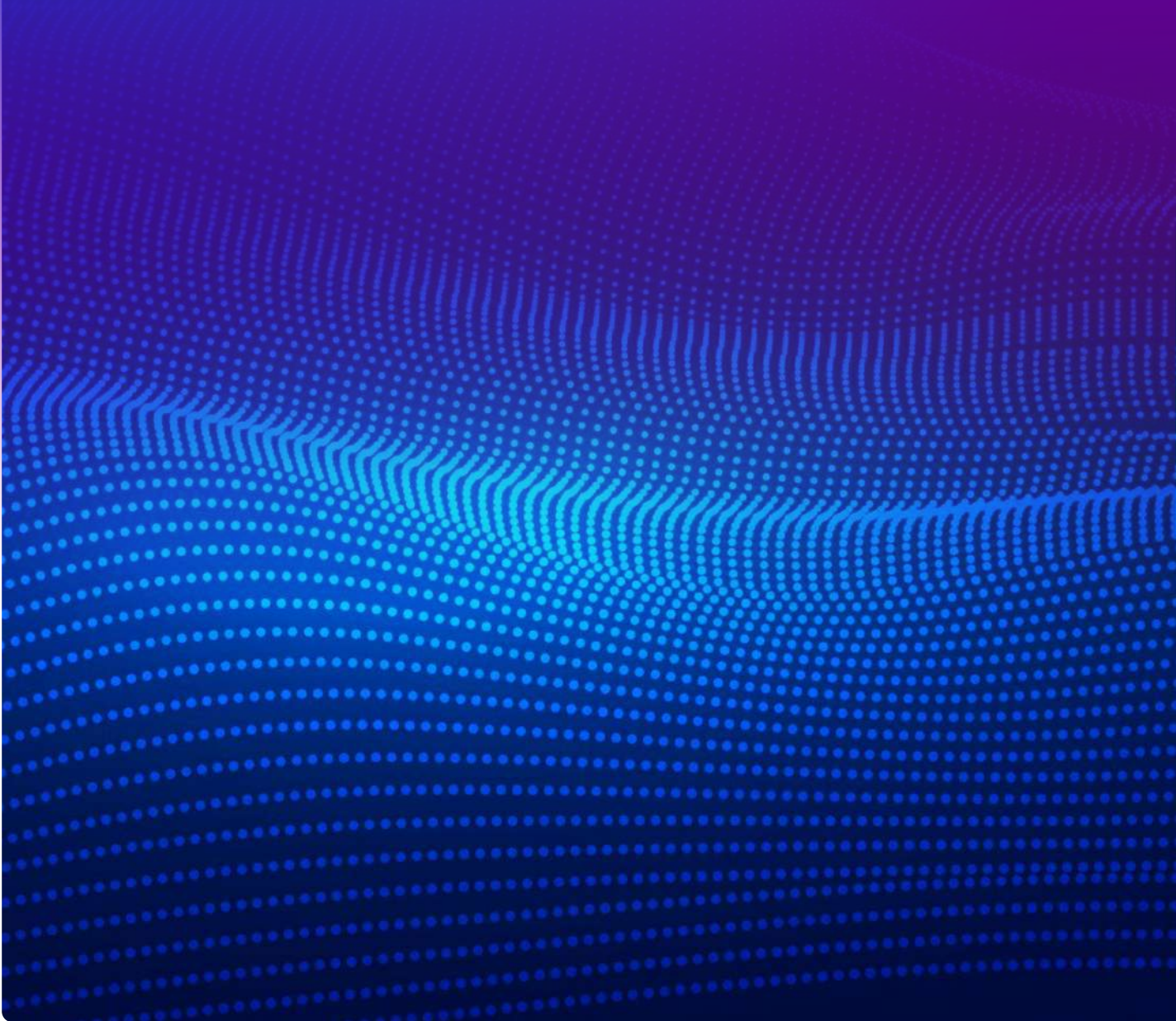


Table of contents

Abbreviations	3
Introduction	3
Business challenges	4
Problem statement	5
Solution	5
Benefits	8
Conclusion	9
References	9
Author information	10

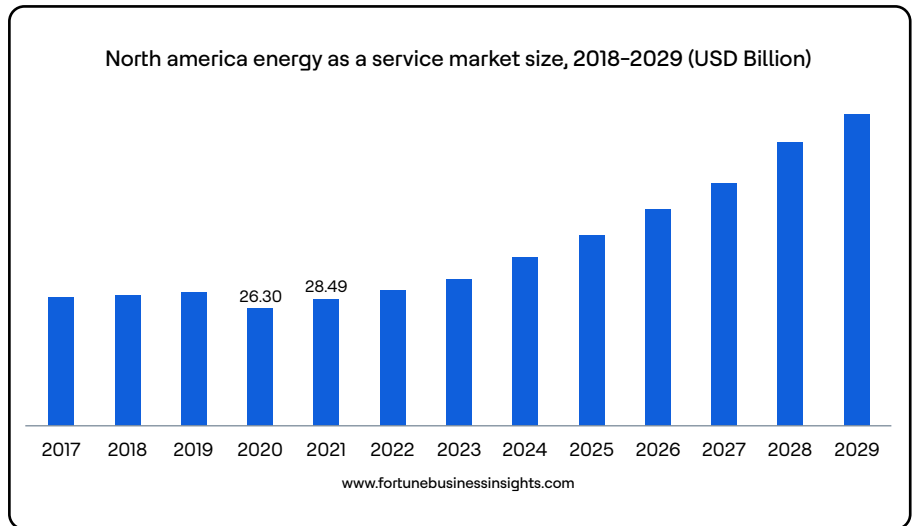
Abbreviations

E&U	Energy and Utility
AI	Artificial Intelligence
ML	Machine Learning
BDD	Behaviour driven development
OCR	Optical Character Recognition
ATSG	Automated Test Script Generation (ATSG)
TMS	Test Management System
DMS	Defect Management System
SR	Service Request
PPMID	Prepayment Meter Interface Device
SMETS	Smart Metering Equipment Technical Specifications
SMETS1	First-generation smart gas/electricity meters
SMETS2	Second-generation smart gas/electricity meters

Introduction

The energy and utility business is experiencing a significant growth, projected at CAGR of 11% through the year 2029, according to [fortunebusinessinsight.com](https://www.fortunebusinessinsight.com). Additionally, the smart meter business is expected to grow at a CAGR of 8% during the same period. This optimistic forecast underscored the potential for tremendous progress in the services industry, providing support and services to overcome several challenges and enhance operational efficiencies, thereby facilitating the projected growth.

In this paper, we examine solutions for testing smart meters through the lens of HCLTech's testing solution MagnusIP, which optimizes efforts for testing metering applications and making them more robust.



source: www.fortunebusinessinsights.com

The above graph represents how North American energy as a service is expected to grow in the years to come. Other important regions like Europe APAC will witness this growth in a similar fashion. To address the challenges faced in the sector it is imperative for the services industry to build capabilities around this.

Business challenges

To achieve the projected growth described in the previous section, the industry must overcome several challenges, including:

- The rapid expansion of smart metering devices resulting in increased amount of data generation making manual validation inefficient and time-consuming
- Operations sometimes require physical buttons inputs on meters, which might not be possible through prominently used test automation solutions available in the market
- Lack of mechanism to access device logs for diagnostics and corrective actions
- High testing cost due to the need for regression testing of the metering system at regular intervals which makes it impossible to manually test the devices due to the high count and installation in the field across geographical locations
- Ensuring compliance with organizations policies regarding data access
- Dealing with complex deployment topologies encountered involving various types of meters and network connectivity options
- Continuous firmware updates to support new meter features or evolving industry standards

Problem statement

The testing challenges faced by QA testers hinder the expansion of test automation footprints. Manual test operations is time-consuming, costly and inefficient. Thus a smart and comprehensive test automation solution is essential, meeting the following requirements:

- Ability to scale accommodating a growing number of smart meter devices
- Ability to perform physical operations through the metering devices physical buttons
- Orchestration of test scripts for the smart metering device regression testing
- Comprehensive user management is customizable tailored to product requirements
- The solution should be modular for better adaptability to the complex deployment topologies minimizing the automation maintenance cost
- Support loose coupling between the meter firmware and the automation scripts to support new meter make and versions

Solution

Magnus, the HCLTech's test automation framework fills the gap left by market solutions for smart meters testing by providing an end-to-end perspective and robust support for newer versions. It excels in functional regression testing for web, mobile and device testing with seamless integration capabilities with third-party tools like ALM and defects management systems.

Below is an overview of how Magnus is utilized for smart meter device testing, beginning with the diagram below, which depicts the typical setup of the Magnus framework for smart meter testing.

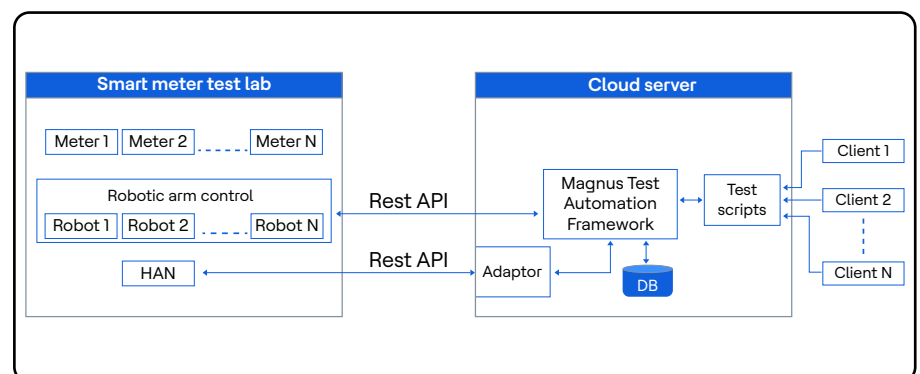


Figure 1 Deployment setup for smart meter testing solution

Magnus serves as the central orchestrator in the overall solution consisting various utilities that cater to different functions to meet the test requirements. It can be hosted over public/private cloud or on-premises.

- Test console: Provides a UI to the users/testers for creating, updating and deleting test cases, groups and suites. Testers can schedule and run a test execution cycle, modifying the settings for the test environment and report directories and test data.
 - Magnus supports scriptless testing through the Automated Test Script Generation (ATSG) engine, enabling the use of pre-tested keywords for portability. Custom keywords can be added to enhance testing capabilities without script modification
 - Magnus communicates with the adaptor robotic devices through rest API calls.
 - Magnus supports the auto-logging of testing defects into the defect management tool. It supports integration with any third-party defect management tools.
 - Magnus supports parallel testing with up to 15 concurrent users
- Adapter: Creates and sends SR based on the test requirements and receives responses from the smart meter test environment. It communicates test information and relevant data to the adapter.
- Robotic arm control: Controls physical robotic hardware installed on device cabinets to perform actions and capture images. Each device set or cabinet is installed with a robotic mechanism controlled by robotic arm control software through the Magnus Test Automation Framework.
- Logging: Sniffer devices and debug interfaces capture ZigBee and debug logs, respectively, for each test. Logging systems are installed in meter cabinets for this purpose.
- Meter devices: Physical smart devices like meters and Prepayment Meter Interface Devices (PPMID) are utilized for testing compliant with either SMETS1- or SMETS2 standards as per the requirement.

Magnus Test Automation end-to-end solution for the smart meter industry:

Magnus Test Automation end-to-end solution for the smart meter industry:

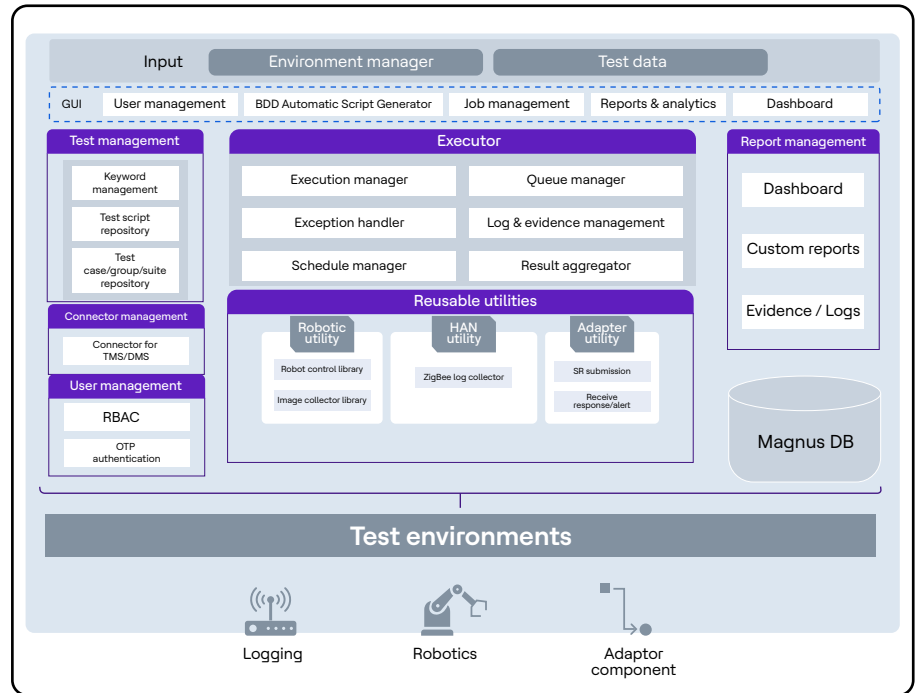


Figure 2 Magnus block diagram (Typical Smart Meter Setup)

Magnus controls various components and utilities to manage the overall automation solution. It is built on Behavior-driven Development (BDD) framework, enabling efficient test design and optimized test execution of applications and devices. Magnus will interface with adapter and pass the required test information for the SR and SRV creation. The overall testing solution contains the following components:

- **Executor:** Responsible for the scheduling, execution, queueing and handling exceptions during the test execution cycle. It aggregates test results during the execution, accessed by the reports management module to generate the test execution reports. The executor receives test data, environment and scheduling data from the Magnus test console UI.
- **Test management:** Empowers users to create keywords and test scripts through the UI. Utilizing BDD methodology, it automatically generates test scripts from 'test stories' written in English, like Gherkin notation. This module abstracts test cases from underlying test engines and scripting languages, leveraging reusable utilities and keywords developed specifically for the customer. Users can also group the test scripts into test groups and test suites. It also has built-in connectors to connect to different test management tools like HP-ALM, JIRA, TFS, etc., so that the customer can connect to their existing test management tools and use the existing test cases.
- **User management:** Ensures secure access to the Magnus UI through role-based authorizations and two-factor authentications.

- **Reports management:** Extracts data from the report aggregator and presents it through a dashboard for the end users. Manages the relevant logs and evidence for the test execution cycles allowing the users to view and investigate through the generated test reports. Custom reports creation is also supported.
- **Reusable utilities:** Magnus consists of utilities that are used to control important functions during the test execution. Its three main utilities are:
 - **Robotic utility** governs the robotics across all cabinets through a robotic server installed in the smart meter test lab. It dispatched necessary test information, including device make model and button-pressing sequences. It also handles the images captured and sent from each of the robots to process the OCR and extract details.
 - **HAN utility** oversees the collection of ZigBee traffic among HAN devices using a dedicated sniffer. A server-based HAN Engine employs its Zigbee Log Collector Library to initiate log capture and retrieval functionalities based on requests received from the Magnus Test Automation Framework.
 - **Adapter utility** handles connectivity and the SR queues with the adapter transferring all the test data and details from the framework to the adapter to create the required SRs and SRVs. It receives the response and alert data from the adapter to be passed to the Magnus for validation and processing of test results.
- **Magnus DB:** The application database stores data relevant to Magnus operation and test data associated with each test case and run.

Benefits

Achieving the test automation through the Magnus framework has provided the following benefits to the smart meter businesses:

- No limitation on the device count since the solution can scale up the test operations for any number of devices based on its mocking capabilities
- Integration with the robot allows remote operation of the meter's physical buttons
- Development of a custom keywords library that helps in the re-usability of functions and quick test script generation
- Automated script generation for controlling the meter device, robotic device and logger
- Reduce the overall testing effort by 10%

- 40% reduction in the regression test time
- Over 25% effort reduction was achieved in test design over traditional tools

Conclusion

Based on the complexities and challenges encountered by the smart metering industry, an industry-specific end-to-end automation framework is greatly needed and HCLTech's Magnus is the right solution.

References

<https://www.fortunebusinessinsights.com>

Author information



Abhishek Tomar

Abhishek Tomar works as Deputy General Manager with HCLTech. He is leading the PTS Solutions CoE Professional Services team and has vast experience in software development and testing for embedded systems in the Consumer Electronics domain. His interest lies in exploring machine learning methods to solve industry problems.



Subramanyam Palivela

Subramanyam Palivela has a master's in technology from NIT Warangal. He is currently working as a Solutions Architect with the product testing team and has developed automation solutions using AI/ML and deep learning for web, desktop, mobile and microservices-based applications. He has also supported and developed automation solutions for several Fortune 500 customers.

HCLTech | Supercharging Progress™

HCLTech is a global technology company, home to 222,000+ people across 60 countries, delivering industry-leading capabilities centered around Digital, Engineering and Cloud powered by a broad portfolio of technology services and software. The company generated consolidated revenues of \$12.3 billion over the 12 months ended December 2022. To learn how we can supercharge progress for you, visit hcltech.com.

hcltech.com

