Barcelona

The Barcelona release represents the first major milestone on EdgeX Foundry's technical roadmap. Key features for the Barcelona release include:

- Stabilization of key APIs
- Better code quality, fit and finish
- More than double the test coverage across EdgeX microservices
- Addition of reference Device Services supporting BACNet, Modbus, Bluetooth Low Energy (BLE), MQTT, SNMP and Fischertechnik
- Extra "north side" Export Service interfaces that provide connectors to Azure IoT Suite and Google IoT Core as well as support for connections via MQTTS and HTTPS
- Progress towards the definition of unified APIs for security and system management

Release Information

- The themes and major objectives for the first 6 months of the project and this first community release were to:
  - Align the community behind the key tenets for the development of an open, platform independent, technology agnostic platform for the IoT/edge
  - Build community understanding of the platform
  - Obtain community agreement on the architecture
  - Obtain community consensus on the current microservices and APIs that facilitate interoperability
  - Stabilize the platform codebase
- Significant achievements that have been made in the evolution of the code base include:
  - Stabilization of key APIs
  - Improvement in code quality, fit and finish
  - More than double the test coverage across EdgeX microservices
  - Addition of reference Device Services supporting BACNet, Modbus, Bluetooth Low Energy (BLE), MQTT, SNMP and Fischertechnik (demonstrating simple serial device connectivity)
  - Additional "north side" interfaces in the Export Services, namely providing connectors to Azure IoT Suite and Google IoT Core as well as support for connections via MQTTS and HTTPS
  - Demonstration of the longer-term EdgeX strategy for improving performance and startup times and reducing the overall footprint via Go Lang-based implementations of EdgeX microservices. This work will be further showcased in a preview release targeted for January 2018.
  - Progress made towards the definition of unified APIs for security and system management
- Key progress has been made from an operational and process perspective - important for the establishment of a new open source effort and allowing the project to grow more quickly going forward and serve as a product-quality foundation for commercialization. Achievements in this area include:
  - Adoption of a project charter and established governance
  - Establishment of a Technical Steering Committee (TSC) and active project Working Groups (Core and Supporting Services, Applications, Device Services and SDK, QA and Test, DevOps, Security and System Management) to steer and drive the project with contributions from a growing number of companies
  - Development of governance rules, processes, procedures and guidelines to support the project's operations
  - Automation of the build/test and continuous integration processes and infrastructure to support the evolution of EdgeX in an agile and efficient manner
  - Identification and assignment of project Maintainers and Committers across key functional areas to help support the evolution of the code base in a controlled way
  - Establishment of the EdgeX Foundry website (www.edgexfoundry.org) which acts as the central hub for community activity – providing access to the project wiki (consisting of more than 150 pages of project documentation), collaboration platform (RocketChat), mailing lists and other community resources (e.g. Tech Talks, documentation, contribution guidelines etc.)
  - Development of a long term EdgeX roadmap with planned bi-annual releases to evolve EdgeX so that it can support the requirements for business-critical IIoT applications. The next upcoming releases include a Go Lang-based microservices preview in January 2018 followed by the second major release dubbed 'California' in late Spring 2018.
- Within six months of its founding, the EdgeX project ecosystem is vibrant and growing as demonstrated by several factors:
  - The project has gained significant momentum with more than 60 member organizations with Samsung being the most recent addition at the Platinum tier
  - The project has developed an active community with more than 150 people from around the world joining EdgeX Foundry face-to-face meetings over the summer to align on project goals, develop working groups and discuss next steps for the project
  - Individual contributor interest has started to climb with over 350 subscribers to the email lists and RocketChat channel
  - Member companies are already starting to provide commercial products and services based on EdgeX (e.g. IOTech, Mocana, others) helping to expand the EdgeX ecosystem of solutions that can address the needs of the IIoT market

Release Dates

Freeze Date: Oct 17, 2017

Release Date: Oct 20, 2017

Release Notes

Notable Additions or Changes

- The ability to export EdgeX data to Google IoT Core was added to the export services.
- The ability to export EdgeX data to an MQTTS endpoint was added to the export services.
- The ability to export EdgeX data to MQTT topics by device id was added to the export services.

- The addition of 6 new device services for common device/sensor protocols: Modbus, BACNet, SNMP, MQTT, Bluetooth Low Energy (BLE), serial communications (as demonstrated by the Fischertechnik device service).
- Removal of Dell Fuse code that was incomplete at the time of the original contribution (example: device manager)
- Interfaces were developed for the micro service controllers, services, data repositories, etc. to allow for additional/alternate implementations in the future.
- Google Style and additional code clean up was applied to most of the code base to improve the quality, readability and consistency of the code base.
- Unit and integration tests across the micro services were doubled (from more than 1200 to more than 2400).
- Dockerfiles for ARM64 container creation were added to each micro service.
- Blackbox tests to check the micro service public APIs for the core services (Core Data, Core Metadata, and Core Command)

Significant Bug Fixes

- RAML documentation matches micro service APIs
- Addressed some inconsistencies in APIs - example returning null when device profile is not found versus HTTP 404 error as with other GETs)
- The log file path configuration in micro services is now platform agnostic - before it was OS (Windows) specific
- Micro service error responses caused stack traces to be returned as part of the response. This was changed so that by default, only the error status code (ex: 404), and error message are returned. A configuration option (includeStackTrace) will allow stack traces to be displayed when desired (ex: in development).

Micro service and shared library release notes

More details about notable changes and bug fixes are provided in the release notes of each micro service or library.

- Core Domain (Library)
- Core Test (Library)
- Core Exception (Library)
- Core Metadata
- Core Metadata Client (Library)
- Core Data
- Core Data Client (Library)
- Core Command
- Core Command Client (Library)
- Core Config Seed
- Core Config Watcher
- Support Domain (Library)
- Support Logging
- Support Logging Client (Library)
- Support Notifications
- Support Notifications Client (Library)
- Support Scheduler
- Support Rules Engine
- Export Domain (Library)
- Export Test (Library)
- Export Client
- Export Distro
- Device Service SDK
- Device Service SDK Tools
- Device Service Modbus
- Device Service BACNet
- Device Service Bluetooth
- Device Service SNMP
- Device Service Fischertechnik
- Device Service Virtual
- Device Service MQTT

Release Docker Compose

Version 0.2.0 Release Docker Compose file: https://github.com/edgexfoundry/developer-scripts/blob/master/compose-files/docker-compose-barcelona-0.2.0.yml

Version 0.2.1 Release Docker Compose file: https://github.com/edgexfoundry/developer-scripts/blob/master/compose-files/docker-compose-barcelona-0.2.1.yml