

Sprint 1 – WK49-WK50

- Prerequisites:
 - MDM4 API documentation
 - MDM4 business object model
 - Eclipse tooling definitions (maven as build)
- Common Goals:
 - Define MDM API as a valid component and its position
 - MDM API is the Location of the business logic
 - Review and evaluate the MDM API features and decide on their integration into MDM 5
 - Define new features necessary in the API
 - Define compatibility of MDM4 and MDM5 application model
 - Define locking of openMDM® 5 application model version and API version
- EPOS
 - Obtain features of the openMDM® 4 API
- CANOO
 - Define Main Goals for the Architecture and publish them for further maintenance in the MDM@WEB Project
 - Define interface spec language
- Results:
 - Initial Feature set of MDM API 5
 - Initial Version of the architecture picture, showing API and its position with dependent components
 - Resolved and clarified and documented, how MDM and ODS relate to each other
 - Proposal if MDM API should be handled as a separate project or not

Sprint 1 -> 2 Bridge (20150108)

Place: Audi Ingolstadt

Date: 2015-01-08

Participants: Hans-Dirk Walther, Andres Almiray, Reingard Pirthauer, Stefan Beese, Christian Rechner, Sven Wittig

- Requirements will be documented until further decision on community level in Jira. The Jira instance will be provided by CANOO AG without additional costs. CANOO provides access for EPOS and Audi until 20150116. When a community solution is available, the requirements will be migrated.
- The catalogue of baseline documents will be stored in a cloud storage provided by CANOO AG. The catalogue will be maintained by CANOO AG and must be referenced within the new generated architecture documentation.
- All requirements have to be documented with the corresponding stakeholders. Architecture decisions have to be taken and documented with reference to the corresponding requirements.
- Text documentation will be provided in DITA until further decision. If necessary the doc sources will be migrated afterwards. The DITA tooling can be obtained from Audi on special request.
- All documents generated by this project must be published under the EPL on the eclipse sites from now on. Until further decision the generated documents will be published. The document sources will be migrated if necessary and published when documentation methodology and tooling for the community are defined.
- A listing of openMDM® components published with the last integration client will be provided by Audi
- The API / business layer project will be placed within a new eclipse project „openMDM® 5 business layer“ driven by Audi.
- For the openMDM® API interface definition UML will be used (EPOS will work with enterprise architect)

Sprint 2 – WK02-WK04

- Prerequisites:
 - Agreement on the features for the MDM API 5 from Sprint 1
 - Initial architecture picture
 - Decision of MDM API will be handles as a separate project
 - Decision on interface spec language
- Common Goals:
 - Agreement on the positioning of the MDM API within the project landscape
 - If decided as a separate project: infrastructure available
- EPOS
 - first draft of the MDM API definition
 - First draft of the API test cases
 - Drive decision on API protocol with the architecture committee
- CANOO
 - Definition of a component as a part of the openMDM toolkit which can be subject to
 - Testing
 - Commercial order placement
 - System configuration by a systemn configurator
 - Possibly Deployment units
 - Picture of the runtime communication of components as decision criteria for the architecture committee on the protocol for the openMDM® API

Sprint 3 – WK05-WK07

- Prerequisites:
 - (from now all published in openMDM® web site)
 - API definition draft
 - API Test Cases draft
 - Component definition draft
 - repository of prerequisite documents considered
- Common Goals:
 - Definition of the several aspects of a component (processes, use cases and possibly resources)
 - complete architecture overview and documentation structure
- EPOS
 - detailed API definition
 - detailed API test cases
- CANOO
 - feature list (high level requirements) / architecture goals consolidated (ready for decision by the architecture committee)
 - component definition refined
 - delivery conditions
 - testing issues
 - development / acceptance criteria
 - architecture documentation draft

Deliverables: Architecture

- Liefergegenstände im Rahmen des openMDM® architecture service
- Liefergegenstand ist eine komplette Architekturdokumentation für den openMDM® Systembaukasten in der Version 5. Die bisherigen Ergebnisse der openMDM® TAG (technische Arbeitsgruppe) sollen redaktionell zusammengeführt und inhaltlich vervollständigt werden.
- Initiale Dokumentation der Architekturziele
- Architekturbild für openMDM® Systeme
- Schichtenmodell
- Entwicklungs- und Laufzeitsicht
- Prozessmodell, Use Case Modell
- Geschäftsobjektmodell
- High level bzw. framework- relevante Schnittstellen (intern, extern)
- Prüfkriterien für die architecture compliance von openMDM® Komponenten
- Prüfkriterien für die architecture compliance von openMDM® Systemen
- Dokumentation für die Entwicklung relevanter Randbedingungen
- Alle Bilder und Modelle sind mit detaillierten Erläuterungen zu versehen. Designentscheidungen sind explizit zu dokumentieren und mit Architekturzielen zu verknüpfen. Besonderer Wert ist auf eine geeignete Strukturierung und die Nachvollziehbarkeit und Verständlichkeit sowohl für openMDM®- Neueinsteiger als auch für Personen mit Detailkenntnis zu legen.
- Die Architekturdokumentation soll je nach Bedarf mehrere Schleifen eines dokumentierten Reviewprozesses bei allen Mitgliedern des openMDM® eclipse working group architecture committees durchlaufen:
 - Veröffentlichung zum Review und Benachrichtigung aller Mitglieder
 - Review und Rückmeldung der Mitglieder innerhalb einer sinnvollen Zeitspanne
 - Offene Punkte
 - Punkte mit abweichendem Inhalt
 - Verständlichkeit für Experten
 - Verständlichkeit für Außenstehende
 - Einarbeitung der Rückmeldung. Fehlende Rückmeldung gilt als Abnahme / Zustimmung
- Eine Abnahme durch das architecture committee der openMDM® eclipse working group ist erforderlich. Sie ist Voraussetzung für die Abnahme durch den Auftraggeber.

Deliverables: business layer

- openMDM 5 business object layer: Konzept, Dokumentation und Implementierung
- Technische Projektleitung des eclipse Projekts
- Erstellen des eclipse project Steckbriefs
- Initiale Dokumentation der Anforderungen
- Use Case Modellierung
- Geschäftsobjektmodellierung
- Definition, Entwicklung und Dokumentation der OSGi Basis- Services
- Definition, Entwicklung und Dokumentation des modularisierten openMDM5 ASAM ODS Applikationsmodells
- Definition, Entwicklung und Dokumentation der modularisierten openMDM Geschäftsobjekt-Schnittstelle (MDM API v5)
- Migration der existierenden openMDM 4 MDM API Implementierung inkl. zugehöriger Bibliotheken auf die neue MDM 5 Geschäftsobjekt-Schnittstelle
- Portierung / Bereitstellung der core business interfaces - keine Implementierung. Enthält jeweils nur eine der folgenden Schnittstellen: worker, scheduler, entry (Datenstruktur generischer Objekte), properties
- Portierung, Dokumentation und Lieferung des openMDM 4 Integrationservers (ohne Benutzeroberfläche)