

Integrationskonzepte für die HP Quality Center Plattform

Thomas Jähmig



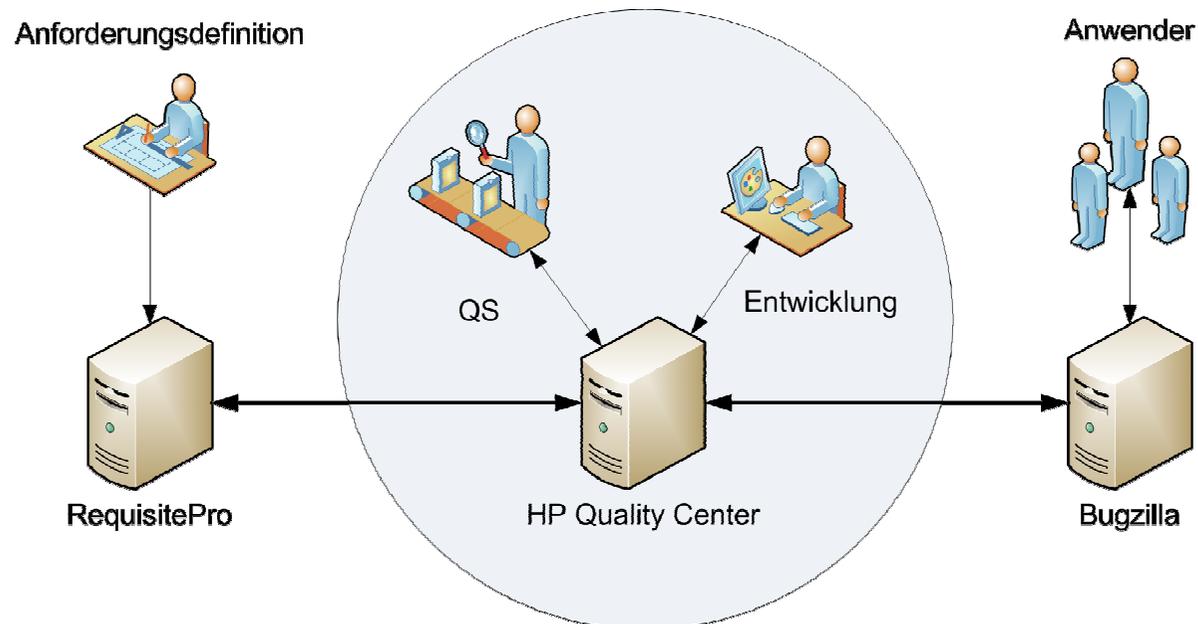
QUALITÄTSMANAGEMENT · EXPERTEN-SERVICE · HP SOFTWARE · IT-SERVICE · SCHULUNGEN

Vivit 2009

- Einführung
- HP QualityCenter Synchronizer Plattform
- Implementierung eigener Adapter
- Alternativen
 - Excel Import/Export
 - Open Test Architecture API

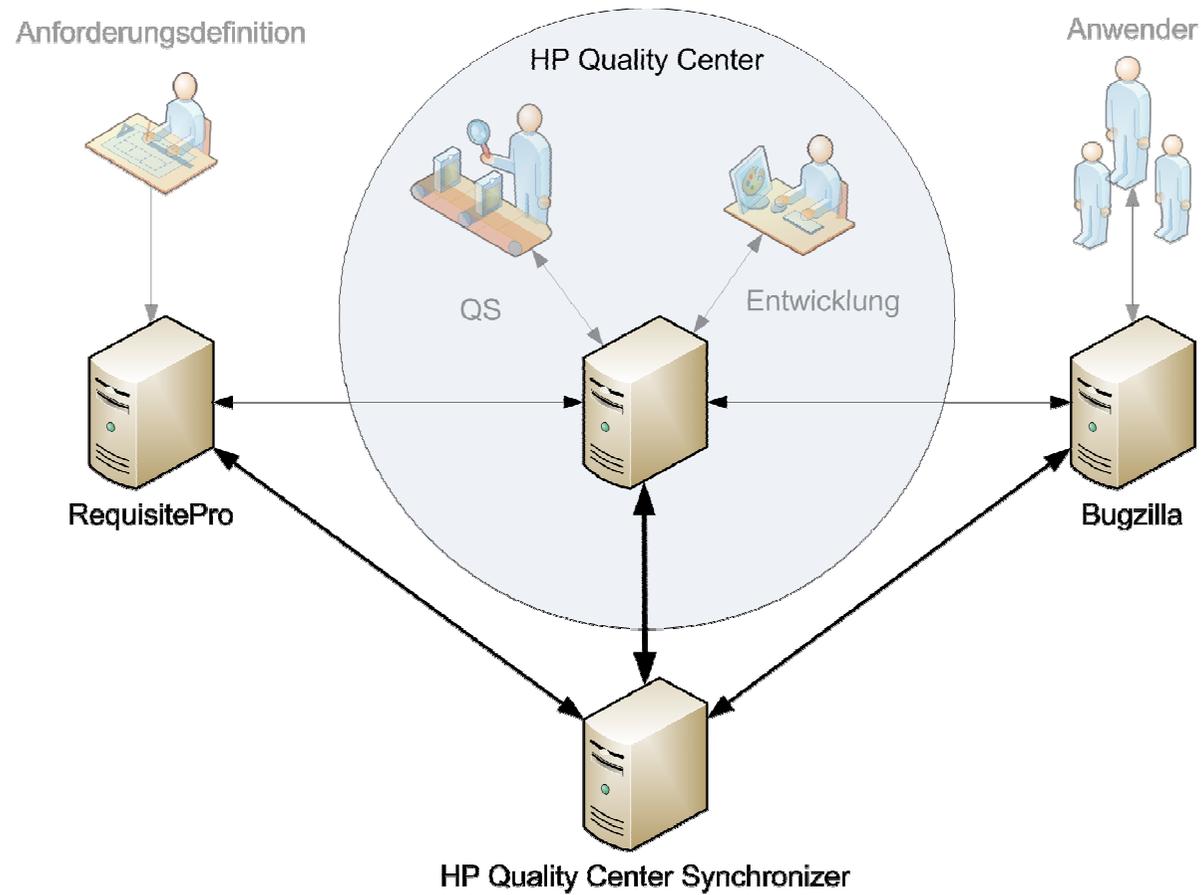
Einführung – ein Beispielszenario

- Historisch gewachsene Tool Landschaft



- andere Requirements-Management Plattformen / Getrennte Defektverwaltungen

HP QC Synchronizer



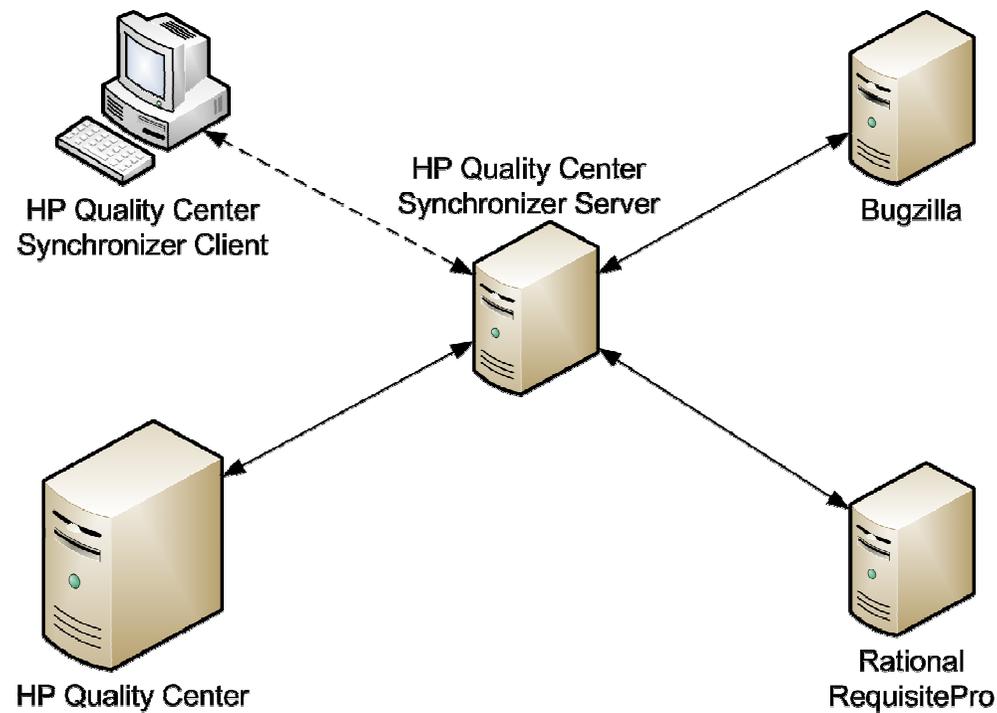
- Plattform: HP Quality Center Synchronizer
- <http://update.external.hp.com/qualitycenter/qc100/sync/qcsynchronizer/index.html>
- Out-of-the-Box Synchronisierung mit:
 - Rational ClearQuest (2003.06.15, 2003.03.16, 7.0, 7.0.1, 7.1)
 - Rational RequisitePro (2003.06.15, 2003.03.16, 7.0, 7.0.1, 7.1)
 - HP Quality Center 10.0 Premier Edition (Defects)
 - HP Service Manager (7.01) / HP Service Center (6.2.2)
- Synchronizer Adapter SDK ermöglicht Entwicklung weiterer Adapter
 - Erhältlich nach Abschluss eines HP Customer Agreements
 - E-Mail an QCSyncAdapterSDK@hp.com:
 - Firmenname, Angaben zum Third-party Produkt, „Internal Use“

- Link → Verbindung zwischen zwei Endpunkten
 - erster Endpunkt ist immer HP Quality Center
 - zweiter Endpunkt wird durch einen entsprechenden Adapter angesprochen
- Grundlegende Endpunktkonfiguration enthält Verbindungs- und Zugangsdaten
 - ggf. weitere Parameter, z.B. zur Selektion eines bestimmten Projektes
- Begrenzung der Menge an zu synchronisierenden Objekten anhand von im Quality Center definierten Filtern (Favoriten)
 - z.B. nur Defects ab Priorität „High“

- Events (Creation, Update, Deletion)
 - Konfiguration uni-/bidirektionaler Synchronisierung in Abh. des Events
- Verknüpfen von Feldern zwischen den Endpunkten (Field Mapping)
 - Feld Typen: String, Date, Number, User-List, Single/Multi Value List, Attachment
 - uni-/bidirektionale Verknüpfung
 - Typumwandlung nur in Ausnahmen möglich
 - z.B. Date → String (unidirektional)
 - Werte von Listen müssen zusätzlich verknüpft werden

- Synchronisierungstypen:
 - Incremental Synchronization
 - Defects/Requirements erstellen
 - Änderungen überführen
 - Full Synchronization
 - Defects/Requirements erstellen
 - Änderungen überführen
 - Defects/Requirements löschen
- Auslösen der Synchronisierung
 - Manuell im Client
 - Automatisiert

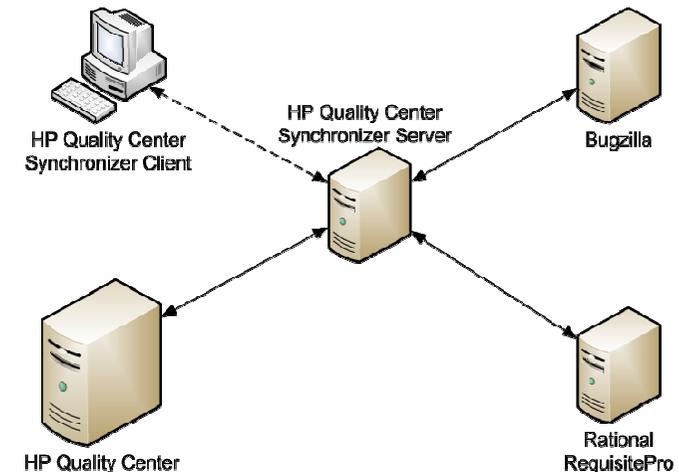
Systemarchitektur – Beispiel I



Systemarchitektur – Beispiel I



- Anforderungen:
 - Requirements bidirektional synchronisieren,
 - Info's aus RequisitePro haben Vorrang
 - Defects
 - Neu: Bugzilla → HP QC
 - Änderungen: Bugzilla ↔ HP QC



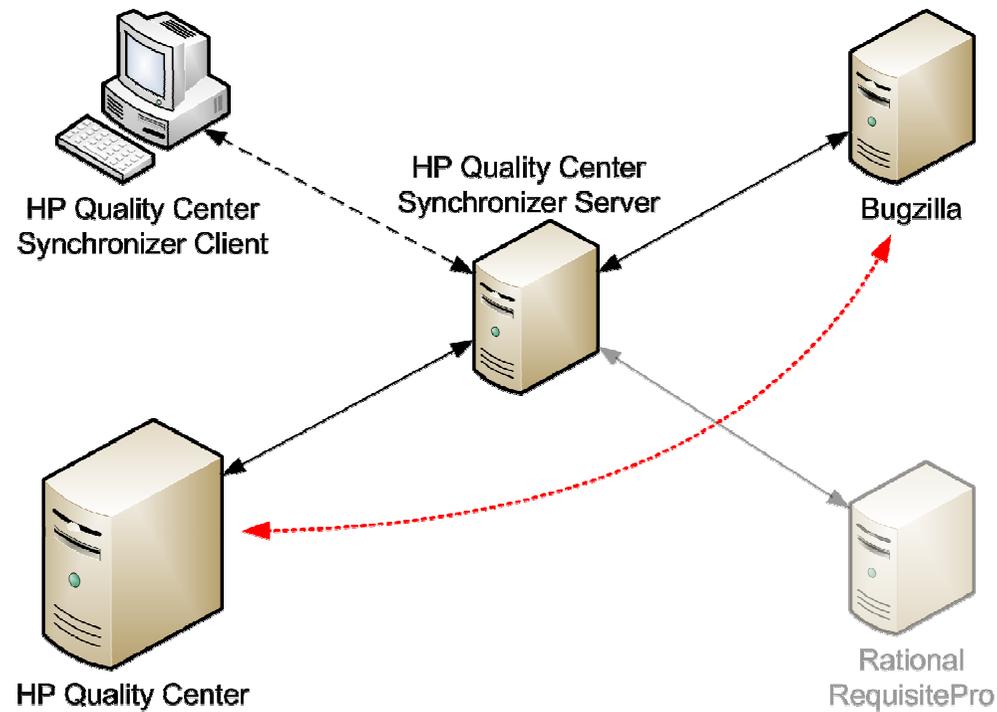
- Umsetzung:
 - 2 Links erstellen und konfigurieren
 - RequisitePro ↔ HP QC (Out-of-the-Box)
 - Bugzilla ↔ HP QC (durch eigenen Adapter)
 - Konfiguration testen
 - Automatisierte Synchronisierung konfigurieren

- Voraussetzungen
 - Schnittstelle zur gewünschten Plattform
 - SOAP, XML-RPC, REST, ...
 - API (z.B. Java)
 - COM
 - Adapter SDK

- Mögliche Probleme:
 - Große strukturelle Unterschiede zwischen QC und der entsprechenden Plattform
 - Einige Funktionen (z.B. Löschen von Objekten) werden nicht unterstützt
 - Unvollständige Schnittstelle
 - Open Source Software: → Erweiterung der Schnittstelle
 - Closed Source Software: → Anfrage beim Hersteller
 - unterschiedlicher Umfang von Wertetabellen
 - z.B. Priorität bei Defects → gemeinsame Werte schaffen
- ➔ Plattform-Integration / Entwicklung des Adapters bedarf genauer Planung!!!

- Grundlegendes Vorgehen
 - *adapter.xml* (enthält Angaben zur Factory Klasse; Einstiegspunkt)
 - Referenz zu *adapter-spi.jar*
 - Interfaces: *AdapterFactory*, *Adapter*, *AdapterConnection*
 - Defects: *DefectManager*, *DefectTypeRecord*
 - Java Archiv (JAR) erzeugen und deployen
- Schnittstelle: XML-RPC
 - Implementierung der Schnittstelle noch sehr unvollständig
 - Erweiterung der Schnittstelle um folgende Funktionen:
 - ID's aller Bugs eines Produkts zurückgeben,
 - Update von Bugs,
 - Liste an Komponenten eines Products zurückgeben.

Demonstration – Fallbeispiel I



- Einsatz des HP QC Synchronizer empfiehlt sich bei:
 - Zeitgesteuerte Synchronisierung
 - Nutzung einer bereits unterstützten Plattform (HP QC 10 (Defekte), RequisitePro und ClearCase)
 - nur Synchronisierung von Feldern (also keine Validierung, Transformation, Generierung von Strukturen, Referenzierung zu anderen Objekten)
 - Plattform bietet Möglichkeiten zum Auslesen des aktuellen Datenbestandes
 - Häufige Synchronisierung gefordert

- Einführung
- HP QualityCenter Synchronizer Plattform
- Implementierung eigener Adapter
- Alternativen
 - Excel Import/Export
 - Open Test Architecture API

- HP QC Synchronizer für bestimmte Anforderungen möglicherweise ungeeignet
 - Keine entsprechende Schnittstelle im Third-Party Produkt
 - Entwicklung eines Adapters zu aufwendig
 - nur einfacher Import von Daten in HP Quality Center nötig
 - aufwändigere Logik zur Synchronisierung notwendig
 - Verknüpfungen zu anderen Objekten (z.B. Tests) sollen ebenfalls synchronisiert werden
- Excel Import / Export
- Open Test Architecture API

- Einführung
- HP QualityCenter Synchronizer Plattform
- Implementierung eigener Adapter
- Alternativen
 - Excel Import/Export
 - Open Test Architecture API

Import

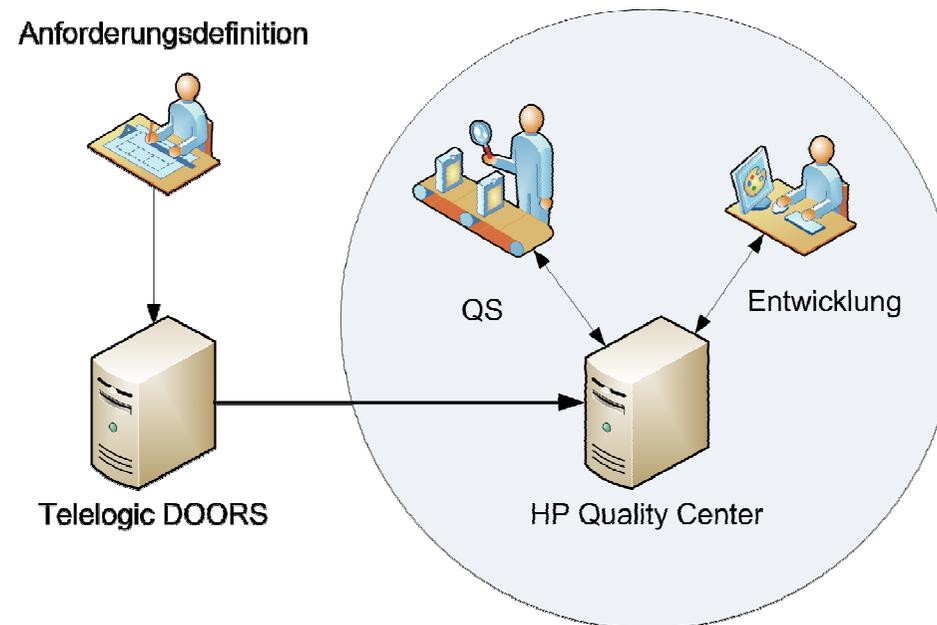
- assistentengestützter Import von Excel Daten in das HP Quality Center
- möglich für Requirements, Tests und Defects
- Anpassungen durch Erweiterung des VBA Makros möglich

Export

- Excel Report Generator im HP Quality Center
- Definition durch SQL Query
- Post-Processing durch VBA Makro

Fallbeispiel II – Excel Import

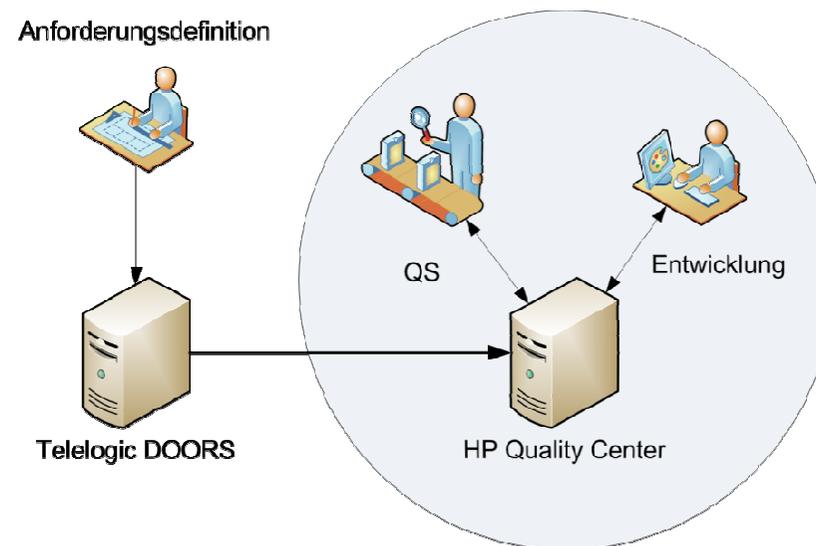
Excel Import



Fallbeispiel II – Excel Import

Anforderungen:

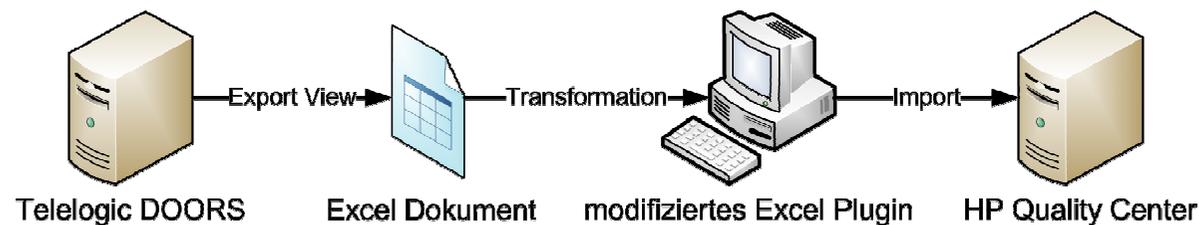
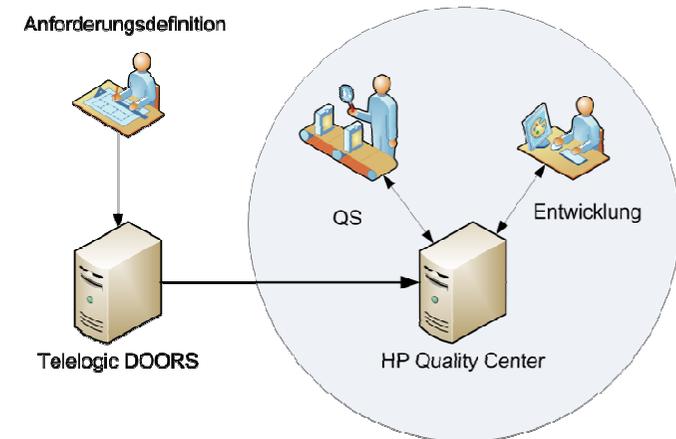
- Requirements werden mit Telelogic DOORS verwaltet
- unidirektionaler Import in HP Quality Center / Update durch neuen Import
- Automatisches Anlegen der Teststruktur



Fallbeispiel II – Excel Import

Lösung:

- Export der Anforderungen als Excel Tabelle
- Anpassung des MS Excel AddIn
 - Transformation der Daten vor dem Import,
 - Referenzierung nicht anhand von Namen,
 - sondern anhand der ID in DOORS,
 - automatisiertes Anlegen der Teststrukturen.

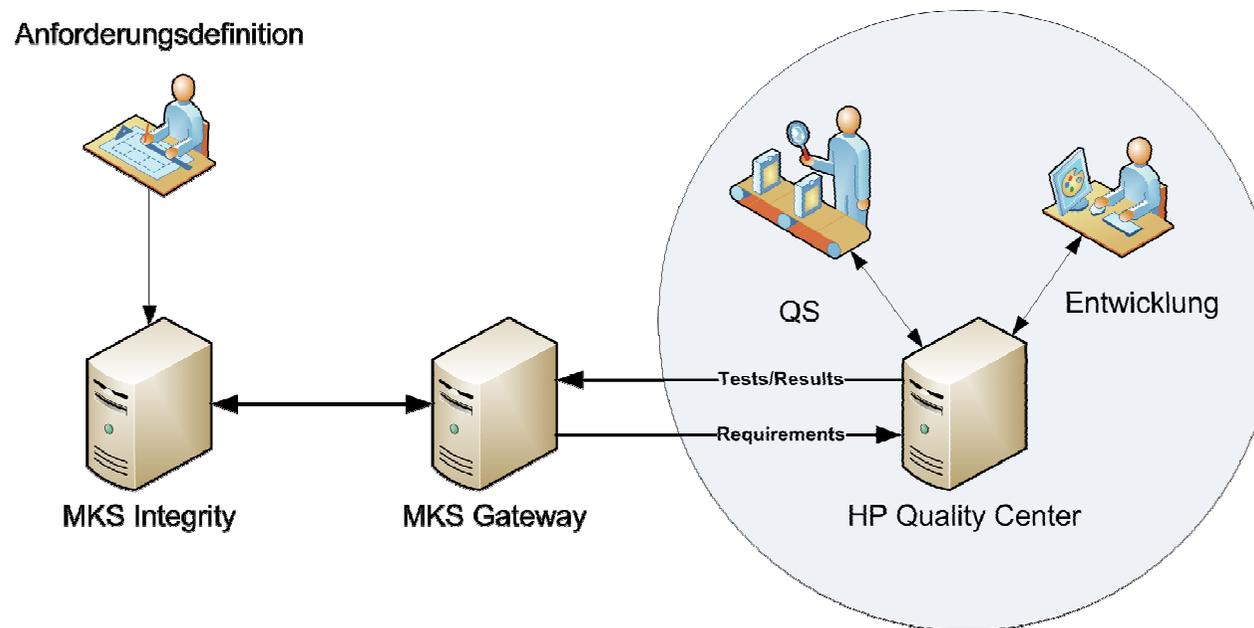


- Einführung
- HP QualityCenter Synchronizer Plattform
- Implementierung eigener Adapter
- Alternativen
 - Excel Import/Export
 - Open Test Architecture API

- OTA API ist DAS Framework hinter den meisten HP Quality Center Erweiterungen / Add-Ins
 - QC internes Workflow Scripting
 - Excel/Word AddIn
- Zugriff auf ALLE im HP Quality Center hinterlegten Informationen
- Das Mittel der Wahl bei besonderen Anforderungen an die Integration
- Auch zur Lösung anderer Probleme/Anforderungen nützlich
 - z.B. besser strukturierte und aufgearbeitete Reports

Fallbeispiel III – OTA API

- Requirements Import, Tests und Test Results Export

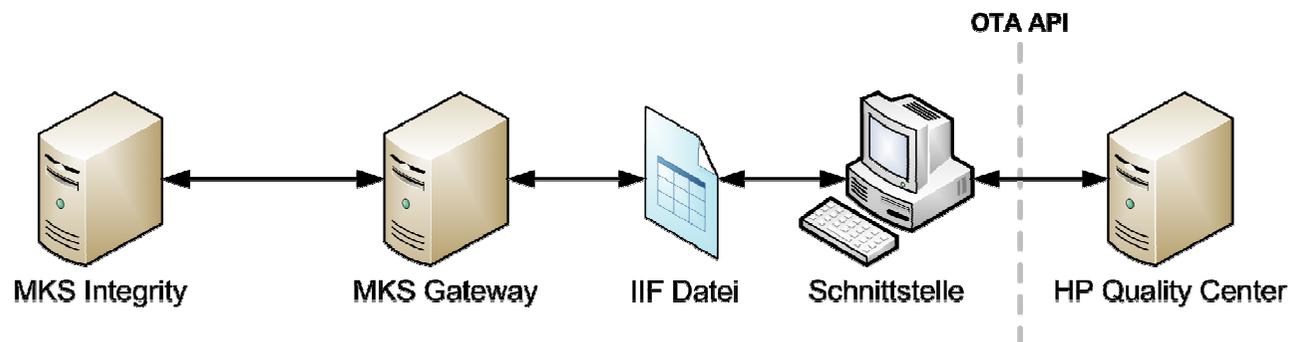


- Datenaustausch über das Dateiformat IIF (Item Interchange Format)

Fallbeispiel III – OTA API



- Schaffung einer IIF Schnittstelle an HP Quality Center
 - Auslesen und Erzeugen von Objekten in HP QC über die OTA API
 - Import:
 - Requirements aus IIF Datei → Import über OTA API
 - Export:
 - Requirements/Tests/Results als Elemente der IIF Datei
 - Verknüpfungen zwischen den Elementen





Vielen Dank für Ihre Aufmerksamkeit!

Thomas Jähmig
tjaehmig@proficom-ag.de
0162/7216690

profi.com AG
business solutions

Stresemannplatz 3
01309 Dresden
Telefon: +49 (351)440080
Fax: +49 (351)4400818

www.proficom-ag.de

Potsdamer Platz 11
10785 Berlin
Telefon: +49 (30)25894058
Fax: +49 (30)25894100