

# Technical Whitepaper

**imc Learning Suite**  
**Systemarchitektur und Technologien**

# Technical Whitepaper

imc Learning Suite  
Systemarchitektur und Technologien

Autor(en): Christoph Gast  
Datum: 28.01.2022

Dokument	Beschreibung
Version	14.11.2
Status (Entwurf / Überarbeitung/ Finalisierung)	Finalisierung
Kontaktperson(en)	Christoph Gast, Dr. Peter Zönnchen

Historie	Status	Wer
20.11.2017	Entwurf	Christoph Gast
02.10.2020	Überarbeitung	Christoph Gast
29.11.2021	Überarbeitung	Eric Andre
28.01.2022	Finalisierung	Dr. Peter Zönnchen

# Inhalt

---

<b>1</b>	<b>Einleitung und weitere Informationen</b>	<b>4</b>
<b>2</b>	<b>Die Systemarchitektur</b>	<b>5</b>
<b>3</b>	<b>Netzwerk-Integration und Skalierbarkeit</b>	<b>7</b>
<b>4</b>	<b>Software-Integration und Interoperabilität</b>	<b>10</b>
4.1	Universal API	10
4.2	Benutzerschnittstellen	11
4.2.1	Benutzerimport Mechanismen	11
4.2.2	Authentifizierungsmechanismen	11
4.3	Content-Schnittstellen	12
4.3.1	Die AICC-Schnittstelle	12
4.3.2	Die SCORM-Schnittstelle	12
4.3.3	Die QTI-Schnittstelle	12
4.3.4	Die LTI-Schnittstelle	12

# 1 Einleitung und weitere Informationen

---

Dieses Dokument beschreibt die Architektur sowie die verwendeten Technologien und diskutiert mögliche Betriebsszenarien. Auf fachliche Gesichtspunkte und Funktionalitäten wird dagegen nicht eingegangen. Das Papier richtet sich an Entscheidungsträger und IT-Fachleute, die mit der Einführung eines Learning Management Systems in ihrem Unternehmen betraut sind.

Weitere Dokumentationen, die vertiefende Informationen zu den in diesem Dokument angesprochenen Themenfeldern liefern sind:

- Systemvoraussetzungen
- Dokumentation Universal API (erhältlich auf Anfrage)
- Configuration Documentation (erhältlich auf Anfrage)
- Security Whitepaper.

## 2 Die Systemarchitektur

Die Lernplattform ist eine Web-Anwendung, die serverseitig Java Servlets und JavaServer Pages nutzt. Der Endanwender nutzt einen Browser und kommuniziert damit mit dem System über HTTP bzw. HTTPS.

Über einen Webserver können statische Inhalte direkt ausgeliefert werden. Anfragen nach dynamischen Inhalte (JSPs/Servlets) werden an die Servlet Engine weitergeleitet. Zur Beantwortung dieser Anfragen führt die Servlet Engine die in Java codierte Programmlogik der Lernplattform aus. Um Zugriffe auf persistente Daten zu erhalten ist sie per JDBC an ein Datenbanksystem angebunden.

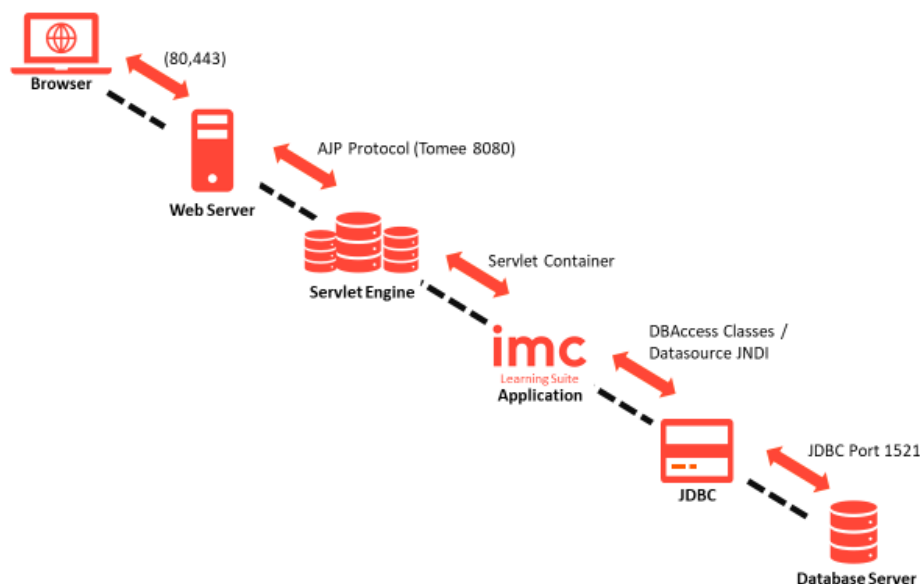


Abb. 2.1: Basis Architektur

Die Lernplattform besteht aus mehreren Komponenten, die unterschiedliche Aufgaben erfüllen:

- ILS stellt eine umfangreiche REST-Schnittstelle bereit, deren Umfang in Kapitel 4.1 genauer beschrieben wird. Sie bietet eine Weboberfläche für die vielfältigen Möglichkeiten der Administration und Konfiguration der Lernplattform. ILS sorgt für Persistenz von Lerndaten durch Verwendung einer relationalen Datenbank bzw. durch Schreiben von Lerninhalten in ein Dateisystem.
- ILP bietet eine moderne und mit Mobilgeräten kompatible Web-Oberfläche für Lerner und weitere Zielgruppen. ILP speichert selbst keine Daten, sondern kommuniziert per REST mit ILS.

- Apache Solr stellt einen schnellen Suchindex für Kataloginhalte bereit. ILS exportiert Metadaten von Lerninhalten regelmäßig nach Apache Solr. ILP verwendet Apache Solr um Kataloginhalte anzuzeigen und eine schnelle Suche innerhalb des Katalogs zu ermöglichen.
- Gamification
  - IGS (imc Gamification System) stellt eine REST-Schnittstelle bereit um Erfahrungspfade, „Badges“ und Regeln zu verwalten. IGS wertet zusätzlich Benutzeraktivitäten gegen diese Regeln aus.
  - PRS (Profile Record Store) enthält aggregierte Informationen welche Punkte auf Erfahrungspfaden und welche „Badges“ Lerner bzw. Gruppen erzielt haben.
  - LRS (Learning Record Store) ist ein Datenspeicher für alle Aktionen, die ein Lerner ausführt, bzw. deren Folgezustände, die sich durch Regelauswertung ergeben.

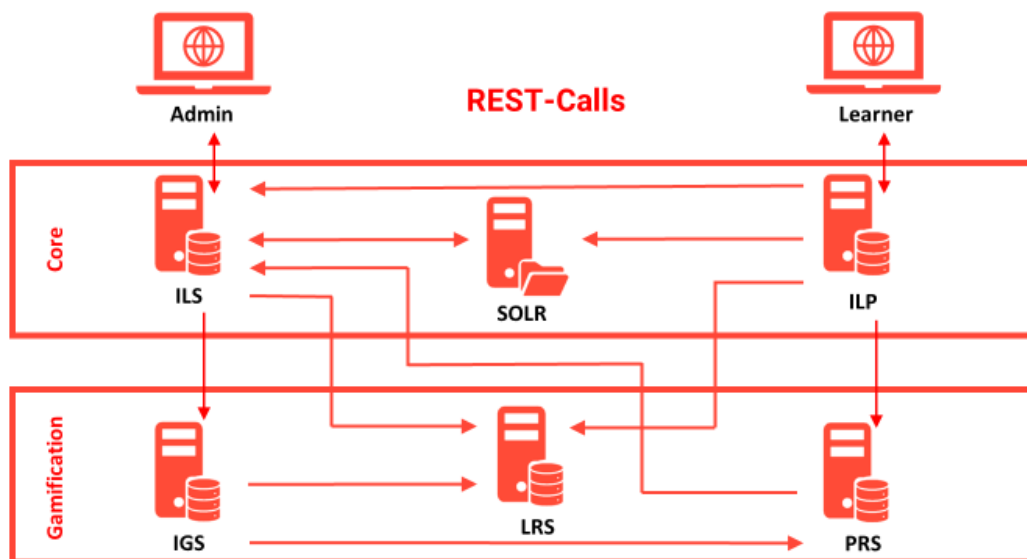


Abb. 2.2: REST Kommunikation der Komponenten innerhalb der Lernplattform

### 3 Netzwerk-Integration und Skalierbarkeit

Die in Kapitel [2 Die Systemarchitektur](#) aufgezeigte Systemarchitektur integriert sich einfach in eine vorhandene IT-Umgebung. Die beschriebenen Komponenten können durch Firewalls getrennt werden und benutzen die aufgezeigten Kommunikationsprotokolle. Für ein einfaches Betriebsszenario kann die Plattform ohne zusätzliche Systemkomponenten in ein vorhandenes Netzwerk eingebunden werden. Der Webserver wird dazu mit der Servlet Engine zusammen auf einer Serverinstanz betrieben. Der Webserver verarbeitet dabei die statischen Dateien, die Servlet-Engine die dynamischen Inhalte. Das Streaming von Inhalten kann durch einen zusätzlichen Streaming-Server (Third Party Produkt) erreicht werden. [Abb. 3.1](#) zeigt ein einfaches Betriebsszenario (ohne Streaming-Server), wie es für Testsysteme genutzt werden kann. Die Betriebssystemumgebung kann, wie bei allen Szenarien virtualisiert genutzt werden.

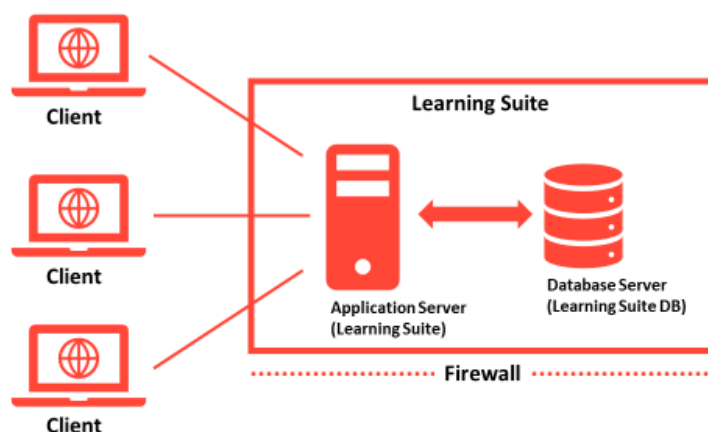


Abb. 3.1: Einfaches Betriebsszenario

Eine Lastverteilung und eine erhöhte Ausfallsicherheit kann durch einen verteilten Betrieb in einem Cluster erreicht werden. Die Plattform erlaubt das Clustern von Applikationsserver zur Lastverteilung. Eine Lastverteilung auf Datenbankseite kann durch die Datenbanksoftware erreicht werden. Durch die Verteilung der Applikationslogik auf mehrere Rechner wird es notwendig den Content auf einen Content-Server auszulagern. Die Rolle des Content-Servers kann durch einen Applikationsserver übernommen oder auf einen eigenen Server ausgelagert werden. [Abb. 3.2](#) zeigt den ausfallsicheren Betrieb mit einer Lastverteilung auf Applikationsserverseite ohne eigenen Content-Server.

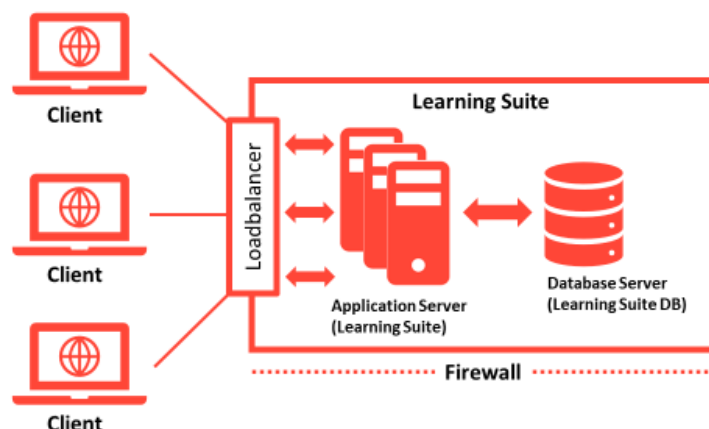


Abb. 3.2: Einfaches Betriebsszenario

Eine erhöhte Ausfallsicherheit auf der Datenbank kann durch einen Failover-Cluster erreicht werden.

Um auch Standorte mit schlechter Anbindung an den Hauptserver mit Inhalten versorgen zu können, kann der Content vom System getrennt werden. Der so entstehende Content-Server wird an den jeweiligen Standorten installiert und mit einer Cache-Funktion versehen. Die Unterstützung der SCORM-Schnittstelle kann durch einen Reverse-Proxy auf dem Content-Server gewährleistet werden.

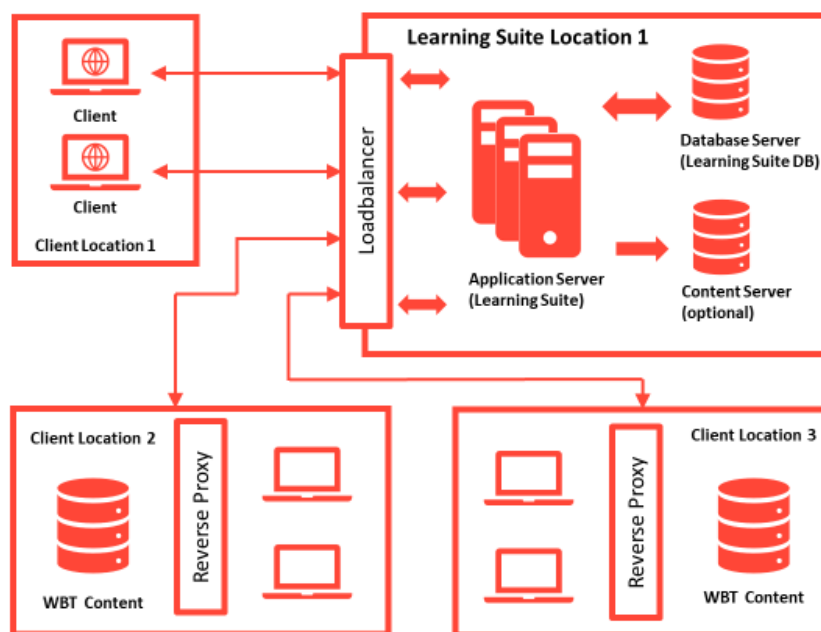


Abb. 3.3: Verteilte Content-Server



Anforderungen wie große Benutzerszenarien im Millionenbereich, ein weltweiter Einsatz der Plattform, ständig veränderten Lastszenarien oder Outsourcing der Infrastruktur begegnet die Plattform mit der Einsatzfähigkeit in der Microsoft Azure Cloud. Die einfache Aufteilung auf die verschiedenen Funktionsgruppen erlaubt die Nutzung der angebotenen Azure Services als Systemkomponenten der Plattform. Die Erweiterung um die Programmlogik zum Deployment, Controlling und Management der Plattform in der Azure Cloud resultiert in Fähigkeiten wie lineare Skalierung, Bereitstellung in weltweiten Datencentern und sichere Integrationen von diesen Services in die eigene IT-Infrastruktur. [Abb. 3.4](#) zeigt die Nutzung der Microsoft Azure Komponenten für den Betrieb der Plattform in der Azure Cloud.

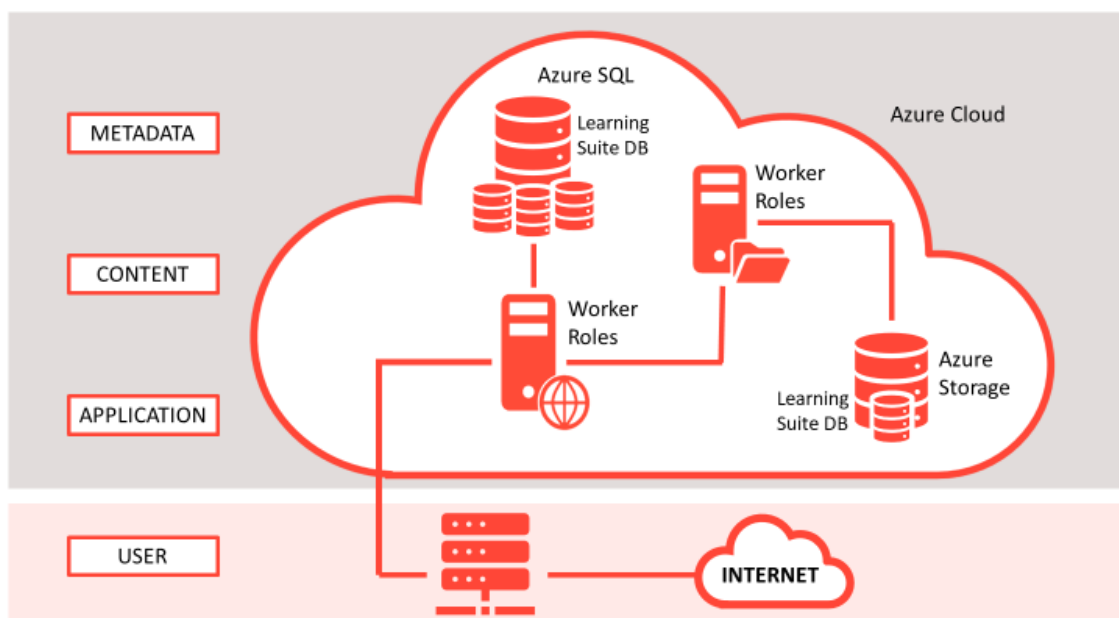


Abb. 3.4: Einfaches Azure Deployment

## 4 Software-Integration und Interoperabilität

---

### 4.1 Universal API

Die imc Learning und Talent Suite bietet eine REST-Schnittstelle für folgende Inhalte an:

- Personen (Export): Profildaten des angemeldeten Benutzers
- News (Export), die einem als sichtbar in mobilen Apps gekennzeichneten News-Panel zugeordnet sind
- Kurse (Export)
- News
  - Lehrplan und Bibliothek
  - Tests, Fragen und Feedbacks
  - Buchen und Starten von Kursen
  - Rating (inkl. Abgabe für Lerner)
  - Kursraumkonfiguration
  - Voraussetzungen
  - Kursvorlagen
    - Vormerkungen
- Lerninhalte ausgewählter Inhaltstypen (Export)
- Katalog (Export)
- Termine (Export)
- SCORM-Daten (Import/Export)
- Suchen (Export): Ausgewählte einfache und aggregierte Suchen für Lerner, Tutoren, HR und Vorgesetzten.
- Portalseiten (Export)
- Gruppen (Export)
- Programme (Export)
- Bookshelf (inkl. Änderungsmöglichkeit für Lerner)

Einsatzgebiet:

- Mobile Apps
- Kundenindividuellen Oberflächen
- Portalsysteme

Siehe auch: Technische Spezifikation der Universal Schnittstelle

## 4.2 Benutzerschnittstellen

### 4.2.1 Benutzerimport Mechanismen

Das System unterstützt die Profildatenquellen

- CSV
- LDAP
- Shibboleth
- LOCAL (Selbstregistrierung der Benutzer am System).

Felder aus einer CSV-Datei bzw. die LDAP-Attribute können den Plattform-Attributen zugeordnet werden. Bei diesem Vorgang werden die Zugriffsberechtigungen gesetzt, die Lizenzrechte geprüft und die Mandanten- und Gruppenzuordnung durchgeführt.

Werden Personendaten führend in einem externen System gepflegt, ist das identifizierende Profilattribut nicht zwingend der Anmeldenamen (Login). Deshalb muss für jede externe Profildatenquelle ein zusätzliches Personenprofilattribut gepflegt werden: der so genannte externe Kennzeichner (External Identifier). Die Identifizierung einer Person im externen System erfolgt über dieses Attribut.

### 4.2.2 Authentifizierungsmechanismen

Zur Benutzerauthentifizierung werden folgenden Verfahren systemseitig unterstützt:

- Lokale Anmeldung (System-Datenbank wird verwendet)
- Anmeldung über einen angeschlossenen LDAP-Server
- Single Sign-On (remoteUser des HTTP-Request-Headers wird verwendet)
- Shibboleth Version 1.3
- SAML2

Werden parallel mehrere Authentifizierungsmechanismen verwendet, erfolgen die Anmeldeversuche in einer Authentifizierungssequenz in der Reihenfolge, die in der Systemkonfiguration eingestellt wurde.

Nach einer definierten Menge erfolgloser Anmeldeversuche erfolgt die Sperrung des Benutzers, die nach einer konfigurierten Zeitspanne automatisch oder manuell freigegeben wird.

## 4.3 Content-Schnittstellen

### 4.3.1 Die AICC-Schnittstelle

Das AICC (Aviation Industry CBT Committee) hat einen – inzwischen älteren – Standard für Lerninhalte, der von einigen Autorenwerkzeugen unterstützt wird, entwickelt. Durch die Einhaltung des Standards wird die Integration in ein LMS, das diesen Standard ebenfalls unterstützen muss, möglich. Die Lerninhalte können dann standardisiert in das LMS importiert werden und mittels des LMS abgespielt werden. Die Lernplattform unterstützt Contents, die gemäß des AICC Standards für webbasierte Contents über HACP (HTTP AICC CMI Protocol) kommunizieren.

### 4.3.2 Die SCORM-Schnittstelle

SCORM (Sharable Content Object Reference Model) ist ein Standard für webbasierte Lerninhalte, der von vielen Autorenwerkzeugen unterstützt wird. Durch die Einhaltung des Standards wird die Integration in ein LMS, das diesen Standard ebenfalls unterstützen muss, möglich. Die Lerninhalte können dann standardisiert in das LMS importiert werden und mittels des LMS abgespielt werden. Die Lernplattform unterstützt Contents, die gemäß des SCORM 1.2 (Level 3) Standards oder des SCORM 2004 (3rd Edition) Standards erstellt wurden.

### 4.3.3 Die QTI-Schnittstelle

IMS QTI ist ein Standard für Online-Test-Materialien. Die QTI-Schnittstelle bietet die Möglichkeit imc Learning und Talent Suite konforme Testinhalte sowohl zu importieren als zu exportieren. Die Schnittstelle unterstützt ausschließlich Elemente vom Typ Item.

### 4.3.4 Die LTI-Schnittstelle

LTI steht für „Learning Tools Interoperability“ und ist ein Standard der vom IMS Global Learning Consortium spezifiziert wurde (<https://www.imsglobal.org/activity/learning-tools-interoperability>). LTI ermöglicht das Einbinden externer Lernmaterialien, sogenannter „Tools“, in MC Learning Suite und imc Learning Portal.

Die Bezeichnung „Lernmaterial“ wird dabei deutlich weiter gefasst als bei den Standards SCORM und AICC. Ein Tool kann ein elektronisches Buch (e-book), ein Quiz oder Test, ein Video, ein WBT oder letztendlich eine beliebige Webseite sein. Das Tool selbst wird nicht importiert, sondern von einem Tool Provider bereitgestellt. ILS/ILP fungieren als Tool Consumer die das Tool referenzieren und bei Bedarf aufrufen.