

IBM Cúram Social Program Management  
Version 6.0.5

*Cúram Solution Architecture*



**Hinweis**

Vor Verwendung dieser Informationen und des darin beschriebenen Produkts sollten die Informationen in „Bemerkungen“ auf Seite 21 gelesen werden.

**Überarbeitung: März 2014**

Diese Ausgabe bezieht sich auf IBM Cúram Social Program Management v6.0.5 und alle nachfolgenden Releases, sofern nicht anderweitig in neuen Ausgaben angegeben.

Licensed Materials - Property of IBM.

© Copyright IBM Corporation 2012, 2014.

© Cúram Software Limited. 2011. Alle Rechte vorbehalten.

---

# Inhaltsverzeichnis

**Abbildungsverzeichnis . . . . . v**

**Tabellen . . . . . vii**

## Überblick über die Architektur von IBM

**Cúram Social Program Management . . . 1**

Einführung . . . . . 1

Geschäftsarchitektur . . . . . 1

Die IBM Cúram-Anwendungsentwicklungsumgebung . . . . . 3

Entwicklungsumgebungsziele. . . . . 3

Entwicklungsumgebung verwenden . . . . . 4

Modellierung mit Cúram . . . . . 4

Implementierung . . . . . 5

Erstellung . . . . . 5

Bereitstellung . . . . . 5

Laufzeitarchitektur . . . . . 6

Die Beziehung zur Java EE-Architektur . . . . . 6

Zusammenfassung der verwendeten Java-Technologien. . . . . 7

Java EE-Entwurfsmuster . . . . . 8

Laufzeitarchitektur auf Darstellungsebene . . . . . 8

Das Struts-Framework . . . . . 9

Laufzeitarchitektur auf Geschäftslogikebene . . . . . 10

Remote-Schnittstellenebene . . . . . 10

Geschäftsobjektebene . . . . . 11

Fassadenebene . . . . . 11

Serviceebene . . . . . 12

Factory-Methoden . . . . . 12

Connectors . . . . . 12

Entwicklungsarchitektur . . . . . 12

Serverentwicklungsumgebung . . . . . 13

Anwendungsmodell . . . . . 13

Anwendungsmodelldomänen . . . . . 13

Anwendungsmodellentitäten . . . . . 14

Anwendungsmodellprozesse . . . . . 14

Server-Code-Generierung. . . . . 14

Cliententwicklungsumgebung . . . . . 14

Entwicklung der Benutzerschnittstelle . . . . . 15

Client-Code-Generierung . . . . . 16

Geschäftsinfrastruktur . . . . . 16

Entwicklungsumgebung für Regeln . . . . . 16

Verarbeitung der Anspruchs- und Leistungsbe-  
rechtigung. . . . . 17

XML und Drucken . . . . . 17

Workflow . . . . . 18

**Bemerkungen. . . . . 21**

Hinweise zur Datenschutzrichtlinie . . . . . 23

Informationen zu Programmierschnittstellen . . . . . 24

Marken. . . . . 24



---

## Abbildungsverzeichnis

1. Cúram-Geschäftsmodell . . . . .	2	6. Geschäftsobjektebene . . . . .	11
2. Aufrufoptionen . . . . .	6	7. Server-Code-Generierung . . . . .	14
3. Client-Laufzeit-Architektur . . . . .	9	8. Beispielbenutzerschnittstelle in IBM Cúram	16
4. Client-Laufzeit-Architektur (Ebene 2) . . . . .	9	9. Client-Code-Generierung . . . . .	16
5. Server-Laufzeit-Architektur . . . . .	10	10. XML und PDF-Druck . . . . .	18



---

## Tabellen





---

# Überblick über die Architektur von IBM Cúram Social Program Management

In diesem Handbuch erhalten Sie eine Übersicht über die Lösungsarchitektur von IBM Cúram Social Program Management aus Geschäfts-, Entwicklungs- und Implementierungssicht. IBM Cúram Social Program Management enthält eine Umgebung für Organisationen, damit diese ihre eigenen Managementanwendungen für den sozialen Bereich entwickeln oder die von IBM bereitgestellte Lösung erweitern können.

---

## Einführung

Eine Übersicht über die Lösungsarchitektur von IBM Cúram Social Program Management aus der Sicht von Geschäft, Entwicklung und Bereitstellung.

IBM Cúram Social Program Management ist eine umfassende COTS-Lösung (Standardlösung) für Sozialdienste und -behörden. IBM Cúram Social Program Management enthält außerdem eine Entwicklungsumgebung für Organisationen, die ihre eigenen Anwendungen für die Verwaltung von Sozialdiensten und -behörden entwickeln oder die von IBM bereitgestellten Anwendungen erweitern möchten. Diese Architekturbeschreibung deckt auch die Komponenten vorhandener Standardgeschäftsanwendungen ab. Es wird auch die technische Architektur des Frameworks beschrieben, das diese Anwendungen unterstützt und von Entwicklern wiederverwendet werden kann, die diese Anwendungen erweitern möchten.

---

## Geschäftsarchitektur

IBM Cúram Social Program Management wurde speziell für das Gesundheits- und Sozialwesen sowie für Personaldienstleistungs-, Sozialversicherungs- und Militär-/Veteranenorganisationen (zusammenfassend als Sozialdienste und -behörden bezeichnet) entwickelt.

Die IBM Cúram Social Program Management-Lösung besteht aus mehreren Ebenen, wobei jede Ebene bestimmte und eindeutige Geschäftsanforderungen sowie technische Anforderungen von Sozialdiensten und -behörden erfüllt.

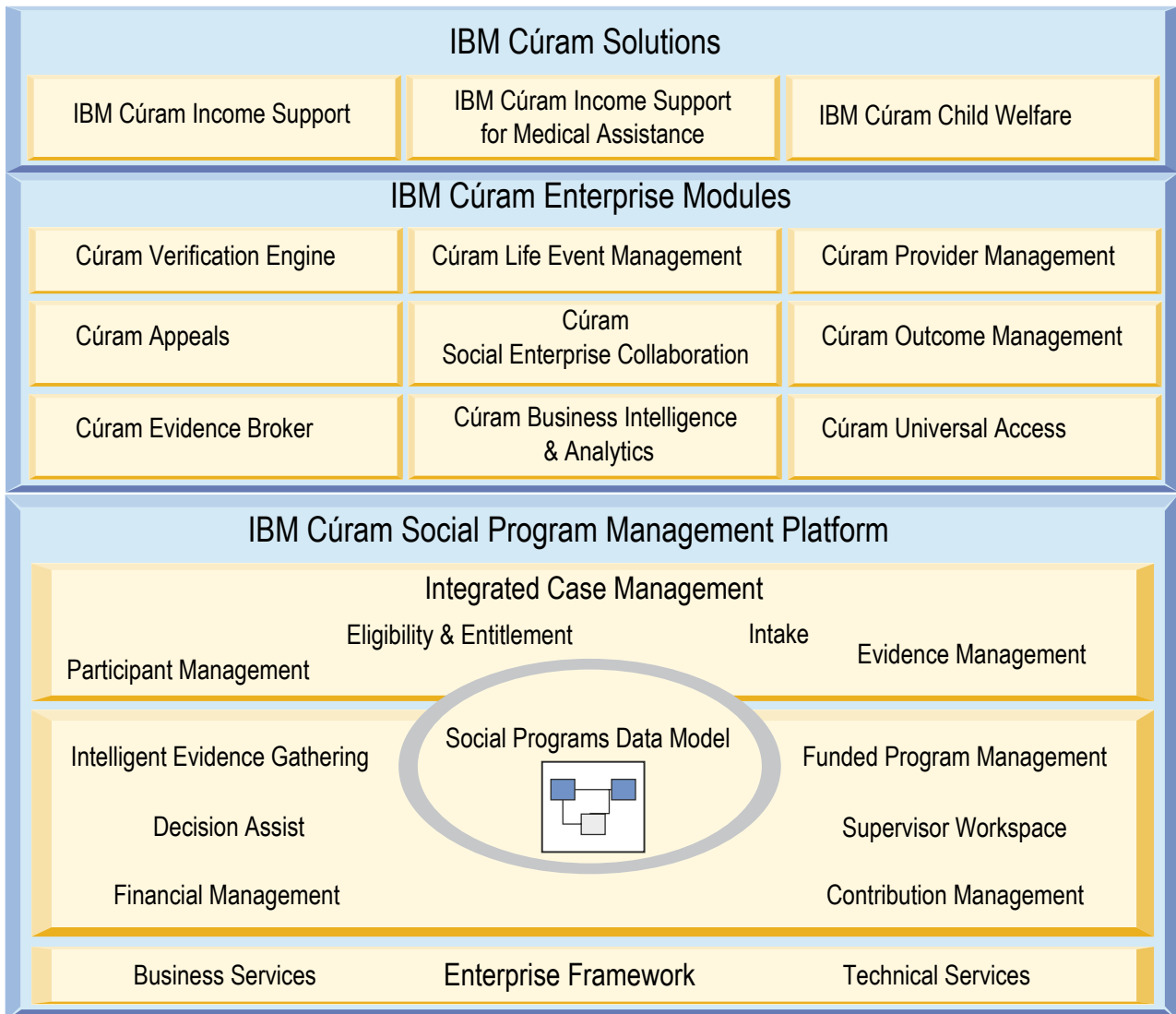


Abbildung 1. Cúram-Geschäftsmodell

Da IBM Cúram Social Program Management die Bereitstellungsmodelle sowohl der Beveridge- als auch der Bismarck-Programme unterstützt, bietet IBM Cúram Social Program Management die erforderliche Grundlage, um die Leistungen und Services zur Unterstützung von bedarfs- und beitragsorientierten Programmen zu verwalten. Die Geschäftsinfrastruktur ist so ausgelegt, dass eine Unterstützung vieler allgemeiner Prozesse bereitgestellt wird, die in der gesamten Organisation vorhanden sind, einschließlich Sicherheits-, Workflow- und Integrationstools.

Die Serviceebene umfasst die allgemeinen Services, die die Service- und Leistungsbereitstellungsprozesse von Sozialdiensten und -behörden unterstützen. Zu diesen Services gehören das branchenführende integrierte Fallmanagement, die integrierte Verwaltung der Anspruchs- und Leistungsberechtigung sowie Teilnehmermanagement, Finanzmanagement, Beitragsverwaltung und Angabenmanagement.

In allen Sozialdiensten und -behörden definieren die SEM-Prozesse geschäftskritische Aktivitäten, die bei der Bereitstellung von Services und Leistungen durchgeführt werden. Diese allgemeinen Prozesse umfassen: Triage, Anliegen, Feststellung,

Planung, Bereitstellung und Bewertung. Jeder Prozess wird durch bereitgestellte Geschäftsprozesse anhand globaler Best Practices unterstützt.

Mit einer Reihe von Anwendungsmodulen werden die allgemeinen Geschäftsfunktionen sowie globale Regeln und Angaben unterstützt, die für die Programmbereitstellung erforderlich sind. IBM Cúram Income Support unterstützt zum Beispiel die Erfassung, Verwaltung, Anwendung und Verarbeitung von Geschäftsregeln und -angaben in Bezug auf Programme für Einkommensbeihilfen weltweit.

---

## Die IBM Cúram-Anwendungsentwicklungsumgebung

IBM Cúram stellt eine Umgebung für die Erstellung von Java EE-konformen Anwendungen für Sozialdienste und -behörden bereit.

Diese Umgebung umfasst die folgenden Features:

- Entwicklungsunterstützungen, um die Erstellung von Java EE-konformen Anwendungen der N-Ebene zu vereinfachen.
- Eine Geschäftsinfrastruktur auf hoher Ebene, die von den meisten auf Unternehmen abgestimmten Systemen benötigt werden, insbesondere für Sozialdienste und -behörden. Die Infrastruktur wird vor allen Dingen bereitgestellt, um das Erfassen von Regeln für die Anspruchsberechtigung und Leistungshöhe von Produkten oder Programmen zu ermöglichen und um diese Regeln in den Cúram-Anwendungen auszuführen.
- Ein UML-Anwendungsmodell für diese Branchen.
- Standardsoftwarekomponenten, die auf dem Anwendungsmodell basieren.
- Eine vordefinierte Referenzanwendung, die mit diesen Komponenten erstellt wurde.

## Entwicklungsumgebungsziele

Eine Zusammenfassung der Entwicklungsumgebungsziele.

### **Fokus auf dem Geschäftsproblem:**

IBM Cúram reduziert die nicht geschäftsspezifische Funktionalität, die entwickelt werden muss. Da Entwickler sich nicht mehr so sehr auf die schwierige und fehleranfällige Client/Server-Entwicklung konzentrieren müssen, können sie sich mehr den Aktivitäten im direkten Zusammenhang mit der Geschäftslösung widmen.

### **Modellbasierte Entwicklung:**

Der Ausgangspunkt der IBM Cúram-Softwareentwicklung ist ein plattformunabhängiges Modell, das mit UML (Unified Modeling Language) entwickelt wird.

### **Codegenerierung:**

Viele Teile einer Anwendung sind Formeln und können daher durch Stereotypen und Muster im Anwendungsentwurf genau ausgedrückt werden. Dank eines modellorientierten Ansatzes kann mit den Werkzeugen, die mit der Umgebung bereitgestellt werden, viel mehr Code generiert werden.

### **Vermeidung von Plattforabhängigkeiten:**

Die IBM Cúram-Architektur "packt" bestimmte Plattforabhängigkeiten in generierten Code und in Infrastrukturkomponenten. Dadurch müssen sie zu einem späteren Zeitpunkt nicht mehr geändert werden und die Portierbarkeit der Architektur wird maximiert.

### **Verwendung von anerkannten Architekturmustern:**

Die IBM Cúram-Architektur verwendet viele Muster, beispielsweise Facto-

ry- und Proxy-Mechanismen, um die Wartungsfreundlichkeit, Leistung und Portierbarkeit der Anwendung zu erhöhen.

#### **Vereinfachte Entwicklung der Benutzerschnittstelle:**

Das Entwickeln von qualitativ hochwertigen Benutzerschnittstellen kann sehr aufwendig sein. IBM Cúram generiert Benutzerschnittstellen anhand einfacher plattformunabhängiger Definitionen, sodass die Produktivität der Entwickler enorm erhöht wird.

#### **Anwendungen nach Branchenstandard:**

IBM Cúram vereinfacht die Anwendungsentwicklung und erstellt stabile Client/Server-Anwendungen auf der Grundlage von standardisierten Technologien. Die wichtigsten Ziele der IBM Cúram-Anwendungsentwicklung sind Laufzeitleistung, plattformübergreifende Implementierung sowie qualitativ hochwertige Benutzerschnittstellen.

## **Entwicklungsumgebung verwenden**

Die IBM Cúram Social Program Management-Umgebung verwendet in der Regel den folgenden Ansatz bei der Entwicklung, der durch die folgenden verknüpften Zyklen dargestellt wird:

- Modellierung
- Implementierung
- Erstellung
- Bereitstellung

## **Modellierung mit Cúram**

Der Ausgangspunkt für eine Entwicklung ist ein plattformunabhängiges Anwendungsmodell. IBM Cúram Social Program Management-Anwendungen folgen einer serviceorientierten Architektur. Die von der Anwendung bereitgestellten Services werden als UML-Schnittstellen im Modell definiert.

Services auf einer niedrigeren Ebene werden ebenfalls hier definiert: Es wird eine Anwendung entwickelt, die eine Ebenenmethode verwendet. Alle Schnittstellen im Modell werden als „Geschäftsobjekte“ bezeichnet. Wenn zwischen Services unterschieden werden soll, die von der Anwendung nur intern verarbeitet werden bzw. die für externe Anwendungen und Benutzerschnittstellen sichtbar sind, werden letztere „Fassaden“ genannt. Fassaden definieren die externe Ansicht einer IBM Cúram Social Program Management-Anwendung. Interne Services werden durch eine Kombination aus „Geschäftsprozessobjekten“ und „Geschäftsentitätsobjekten“ bereitgestellt. Entitätsobjekte definieren die "Dinge", die von der Anwendung modelliert werden. Entitätsobjekte unterstützen auch Datenzugriffsoperationen, um Entitätsinstanzen persistent zu speichern und abzurufen.

Es ist wichtig, dass Sie daran denken, dass das Anwendungsmodell plattformunabhängig ist. Im Modell wird keine bestimmte Middleware und keine Komponententechnologie (z. B. EJB) referenziert. Das Modell definiert nur die Serviceschnittstellen und legt fest, welche Untergruppe dieser Schnittstellen extern sichtbar sein soll. Die IBM Cúram Social Program Management-Umgebung sucht nach Middleware-Abhängigkeiten, indem notwendiger „Plumbing“-Code automatisch generiert wird. Die Entwickler müssen sich in der Regel nicht mit den Problemen der Middleware-Schnittstellen befassen.

## Implementierung

Nach der Modellierung der erforderlichen Geschäftsobjekte stellt der Entwickler eine Implementierung der modellierten Schnittstellen bereit. Alle Quellcode-Schnittstellendefinitionen und ein Teil der Implementierung werden von der IBM Cúram Social Program Management-Umgebung bereitgestellt.

Die Datenzugriffsoperationen auf Entitäten werden beispielsweise kostenlos generiert. Der Entwickler konzentriert sich auf die Codes und Geschäftslogik. Die Implementierungen von Geschäftsobjekten werden von den Entwicklern als normale Java-Objekte (POJOs) codiert. Die Verwendung von Code-generierten Geschäfts-schnittstellen und abstrakten Klassen stellt sicher, dass der Entwickler die notwendige Implementierung mit den korrekten, vom Modell angewendeten Schnittstellen bereitstellt.

## Erstellung

Die IBM Cúram Social Program Management-Umgebung bietet Funktionen zum Erstellen von entwickelten Anwendungen.

Mit bereitgestellten Scripts werden Codegeneratoren aufgerufen, die das Anwendungsmodell analysieren und den Entwurf der Zielanwendung generieren. Diese Code-generierte Ausgabe wird mit selbst erstellten Entwicklerimplementierungen von Geschäftsobjekten kompiliert. Alle notwendigen Middleware-Tools, die zum Erstellen der finalen Anwendung benötigt werden, werden ebenfalls automatisch aufgerufen.

Entwickler müssen Abhängigkeiten zwischen verschiedenen Quellcodes und kompilierten Anwendungsartefakten nicht berücksichtigen, da diese durch die bereitgestellten Scripts gehandhabt werden.

## Bereitstellung

Bei den IBM Cúram Social Program Management-Onlineanwendungen handelt es sich um auf Unternehmen abgestimmte Client/Server-Anwendungen. Die Implementierung solcher Anwendungen in der Produktionsumgebung kann eine komplexe Aufgabe sein, bei der viele Konfigurationsoptionen für die Implementierung über mehrere Netzwerknoden hinweg, Datenbankoptimierungen und Sicherheitseinrichtungen usw. notwendig sind. Entwickler benötigen in der Regel eine einfachere Implementierungskonfiguration, bei der Geschäftsobjekte als Bestandteil eines iterativen Implementierungszyklus getestet werden können.

Es werden mehrere Implementierungsalternativen für zu testende Geschäftsobjekte bereitgestellt. Die EJB-Implementierung mit einem einzigen Knoten lässt eine Implementierung der Anwendung in der ausgewählten Anwendungsserverumgebung der Organisation zu. Eine einfachere Onlineeinrichtung umfasst jedoch die Verwendung von RMI-IIOP-Middleware für die Client/Server-Kommunikation von IBM Cúram Social Program Management. Da RMI in allen Java Standard Edition-Implementierungen integriert ist, ist für diese Konfiguration keine Anwendungsserversoftware notwendig. Dies vereinfacht die Implementierung von Anwendungen aus Testzwecken. Außerdem werden weniger Anwendungsserverlizenzen benötigt.

IBM Cúram Social Program Management-Geschäftsobjekte können auch asynchron über eine Messaging-Middleware aufgerufen werden. Diese Art von Aufruf beruht auf der Workflow-Funktion von IBM Cúram Social Program Management, mit der „Prozessdefinitionen“ aus mehreren Schritten erstellt werden können, einschließlich Ausführung von Geschäftsobjekten und Interaktion mit Onlinebenutzern.

Darüber hinaus können Geschäftsobjekte „eigenständig“ implementiert werden, so dass Stapelmodusoperationen oder stapelorientierte Komponententests möglich sind. In dieser Konfiguration kann das Zielprogramm über die Befehlszeile oder über ein Script ausgeführt werden.

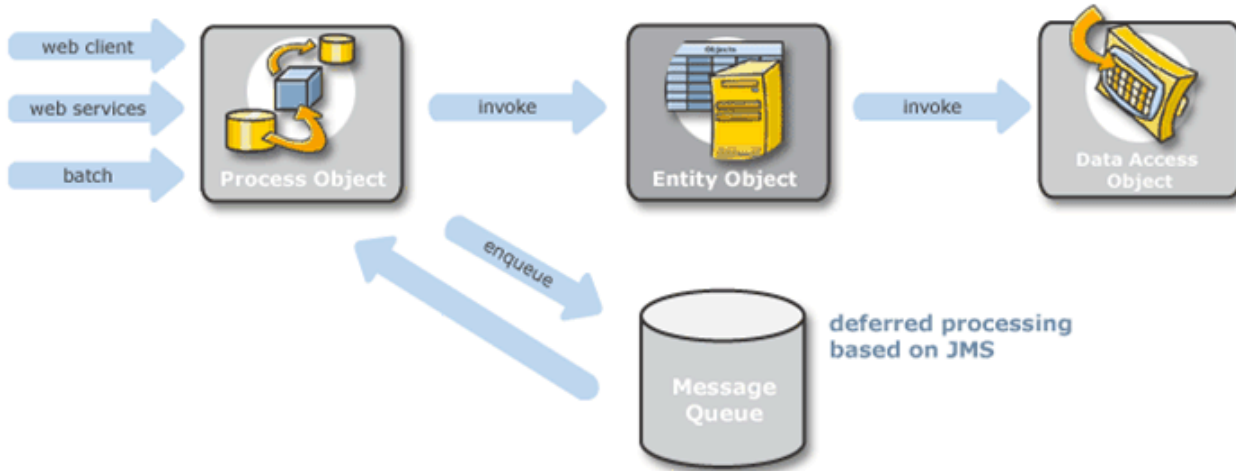


Abbildung 2. Aufrufoptionen

## Laufzeitarchitektur

Auf der logischen Entwurfsebene sind IBM Cúram-Anwendungen plattformunabhängig. Die tatsächliche Ausführung der Anwendung IBM Cúram muss in einer *echten* Umgebung implementiert werden.

Wie bereits erwähnt wurde, können Geschäftsobjekte auf unterschiedliche Weise implementiert werden. Die Client/Server-Umgebung ist eine der wichtigsten (und komplexeren) Möglichkeiten. Onlineanwendungen folgen der Java EE-Architektur. Dabei handelt es sich um eine moderne Architektur auf N-Ebene mit den separaten Ebenen "Darstellungsebene", "Anwendungsebene" und "Persistenzebene". Diese logische Architektur mit drei Ebenen ist die Standardarchitektur für die Entwicklung von Client/Server-Anwendungen. Mit der Trennung der Darstellung, Anwendungslogik und des persistenten Speichers können diese Ebenen relativ isoliert betrachtet werden und Entwürfe können einfacher gestaltet werden. IBM Cúram vereinfacht dieses Konzept sogar noch weiter, indem ein Großteil der Komplexität der Entwicklung einer N-Ebenen-Architektur ausgeblendet wird.

## Die Beziehung zur Java EE-Architektur

Berücksichtigen Sie die Schichtenarchitektur von Java Enterprise Edition (J2EE). Diese sind im Lernprogramm zu Java EE 5 im Thema 'Distributed Multitiered Applications' beschrieben.

Weitere Informationen zu verteilten mehrschichtigen Anwendungen finden Sie unter The Java EE 5 Tutorial .

Die clientseitige Darstellungsebene von IBM Cúram besteht aus HTML- und JavaScript-Benutzerschnittstellen, die von einem Standardbrowser auf dem Benutzerdesktop ausgegeben werden. Nur HTML- und JavaScript-Benutzerschnittstellen

werden mit den Clientgenerierungstools direkt unterstützt. Andere Clienttypen können auch mit generierten Server-Access-Beans entwickelt werden, um eine Verbindung zum Server herzustellen.

Die HTML-Benutzerschnittstellen werden während der Laufzeit von der serverseitigen Darstellungsebene generiert, die aus JavaServer Pages besteht. Browser-Clients kommunizieren mit dieser Ebene über HTTP, das aus Sicherheitsgründen im Allgemeinen mit SSL verschlüsselt ist.

Die Kommunikation der serverseitigen Darstellungsschicht mit der serverseitigen Geschäftslogik erfolgt über das RMI-IIOP-Protokoll. In der Regel werden Geschäftsobjekte in der Geschäftslogikebene als Session-EJBs dargestellt, obwohl diese einfache Java RMI-Objekte für die einfachere Implementierungsoption sein können, die während der Anwendungsimplementierung häufig verwendet wird. Auf jeden Fall sind Geschäftsobjekte letztendlich einfache Java-Objekte (POJOs) mit einer Verbindung über Middleware ("Plumbing"), die während der Anwendungserstellung transparent angegeben wird.

Das Back-End der IBM Cúram-Architektur besteht aus einer relationalen Datenbank und anderen Unternehmens- und traditionellen Anwendungen. Hier wird wieder die Middleware-Verbindung ("Plumbing") generiert, die für die Kommunikation mit dem EIS erforderlich ist.

## **Zusammenfassung der verwendeten Java-Technologien**

Eine Zusammenfassung der in IBM Cúram verwendeten Java-Technologien.

### **Enterprise Java Beans (EJB)**

In IBM Cúram Social Program Management wird Enterprise Java Beans für das Serverkomponentenmodell verwendet.

### **Java-Servlets**

Java-Servlets werden für die Darstellungsebene verwendet.

### **JavaServer Pages (JSP)**

JavaServer Pages werden zur Generierung der Benutzerschnittstelle verwendet.

### **Java Transaction API (JTA)**

Java Transaction API wird für Start- und Commitvorgänge verwendet.

### **Java Database Connectivity (JDBC)**

Java Database Connectivity wird für die Middleware verwendet, um mit der Anwendungsdatenbank zu kommunizieren.

### **Java Message Service (JMS)**

Java Message Service wird für die verzögerte Verarbeitung und für den Workflow in IBM Cúram Social Program Management verwendet.

### **Java Naming and Directory Interface (JNDI)**

Java Naming and Directory Interface wird in IBM Cúram Social Program Management sowohl für die Suche nach der Anwendungsinitialisierungszeit von Datenquellen und Warteschlangen als auch für die Suche nach Enterprise Java Beans in der Darstellungsebene verwendet.

### **Remote Method Invocation (RMI-IIOP)**

Remote Method Invocation over IIOP wird als Kommunikationsprotokoll zwischen der Darstellungs- und Anwendungsebene verwendet.



## Java EE-Entwurfsmuster

Viele der Oracle Java EE-Entwurfsmuster werden von IBM Cúram Social Program Management verwendet.

Das Struts-Framework in der Darstellungsebene von IBM Cúram Social Program Management verwendet das Front-Controller-Muster. Generierte Server-Access-Beans in der Darstellungsschicht implementieren das Service-Locator-Muster, um die Komplexität der Suche nach serverseitigen Objekten auszublenden.

Der IBM Cúram Social Program Management-Anwendungsserver verwendet das Session-Façade-Muster, um einen allgemein definierten Zugriff auf die Geschäftslogik bereitzustellen. Wertobjekte werden für die Kommunikation zwischen verschiedenen Geschäftsobjekten auf dem Server und zwischen Server und Webebene verwendet. Das Muster für das Datenzugriffsobjekt wird für jeden Datenbankzugriff im Anwendungsserver verwendet.

## Laufzeitarchitektur auf Darstellungsebene

Die Clientbenutzerschnittstelle wird während der Laufzeit mit JavaServer Pages (JSPs) als Servlets generiert. Diese Servlets werden in einem Web-Container ausgeführt, der sich zwischen dem Browser-Client und dem Back-End-Anwendungsserver von IBM Cúram Social Program Management befindet.

Im Browser-Client, einem Standard-Internet-Browser, werden dem Benutzer die HTML-Seiten angezeigt. Die durch die Benutzerinteraktion generierten Ereignisse führen zu einer HTTP-Anforderung, die an den Web-Server gesendet wird. Dies ist ein Standard-Web-Server wie IBM HTTP Server oder Apache Web Server. Eine von der Cúram-Umgebung generierte JSP wird in ein Java Servlet konvertiert, das die entsprechenden Parameter vom Web-Server akzeptiert und den Anwendungsserver aufruft.

Der Anwendungsserver von IBM Cúram Social Program Management verwendet generierte Server-Access-Beans, um die entsprechende Geschäftslogik des Servers aufzurufen und stellt außerdem die Logik für die Datenkonvertierung bereit. Das Ergebnis wird zurückgegeben und das Servlet, das dieses Ergebnis bearbeitet, generiert die neue HTML-Seite und gibt sie mithilfe des Web-Servers an den Web-Browser zurück.



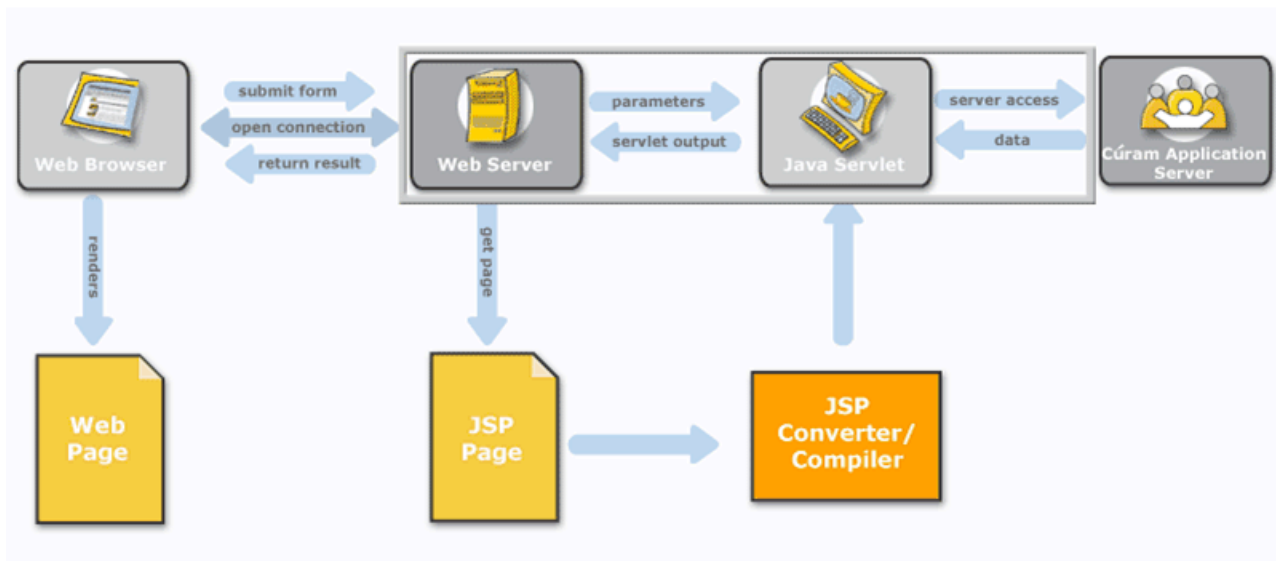


Abbildung 3. Client-Laufzeit-Architektur

In der folgenden detaillierten Ansicht der Webebene wird dargestellt, dass Clientanforderungen von einem Controller-Servlet abgefangen werden, das an das entsprechende JSP-Servlet gesendet wird. Server-Access-Beans konvertieren die Daten von HTML in eine geeignete Darstellungsform, die an den Back-End-Anwendungsserver gesendet werden kann. Während dieser Verarbeitung findet auch eine Datenvalidierung statt, die im Anwendungsmodell definiert wurde. Die Ergebnisse des Anwendungsservers werden für eine Anzeige formatiert und „Listendaten“ können nach einer vom Benutzer definierte Reihenfolge sortiert werden.

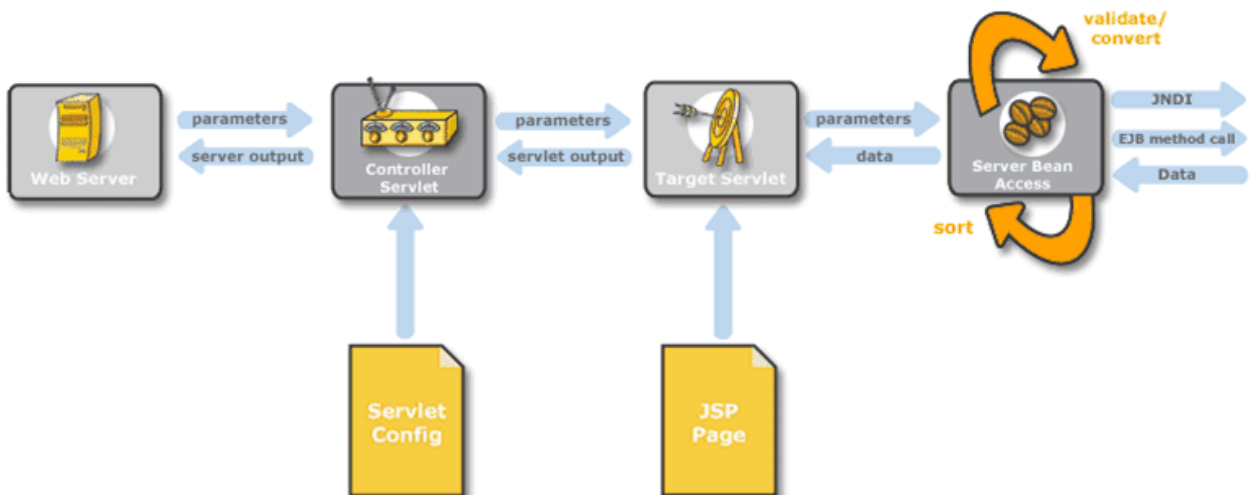


Abbildung 4. Client-Laufzeit-Architektur (Ebene 2)

## Das Struts-Framework

Das Apache Struts Framework ist ein Open-Source-Framework zum Erstellen von Webanwendungen auf der Grundlage von Standardtechnologien wie Java-Servlets, Java-Beans, ResourceBundle und XML.

Das Struts-Framework unterstützt Anwendungsarchitekturen auf der Basis von Modell-2-Ansätzen, eine Variante des Model View Controller-Entwurfsmusters (MVC). Struts bietet eine eigene Controller-Komponente und kann in andere Technologien integriert werden, um das Modell (Model) und die Anzeige (View) bereitzustellen. Struts kann für das Modell mit allen Datenzugriffstechnologien interagieren, einschließlich Enterprise Java-Beans, JDBC und Object Relational Bridge. Struts ist für die Ansicht mit JavaServer Pages, Velocity Templates, XSLT und anderen Darstellungssystemen kompatibel.

IBM Cúram Social Program Management verwendet Struts, generiert aber viele der erforderlichen Komponenten. Einige angepasste JSP-Tags sind im Lieferumfang von IBM Cúram Social Program Management enthalten. Es werden statt Struts-Form-Beans die eigenen Server-Access-Beans verwendet.

Struts stellt die Front-Controller-Komponente bereit und mandatiert das Dateiformat `struts-config`.

## Laufzeitarchitektur auf Geschäftslogikebene

Der Anwendungsserver von IBM Cúram Social Program Management kann logisch in drei Ebenen aufgeteilt werden. Eine Remote-Schnittstellenebene bearbeitet die Interaktion mit der Middleware. Die Geschäftsobjektebene enthält die Geschäftslogik der Anwendung als einfache Java-Klassen. Die Datenzugriffsebene führt die Interaktion mit der Anwendungsdatenbank oder mit anderen unternehmensweiten Informationssystemen aus.

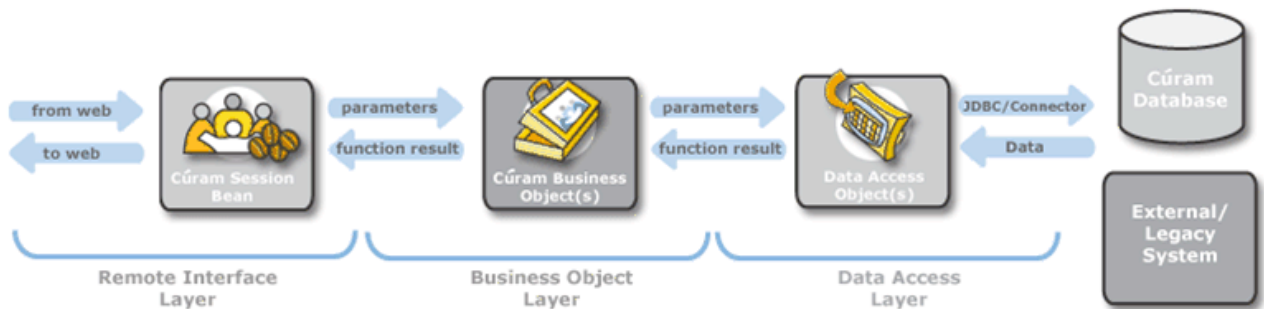


Abbildung 5. Server-Laufzeit-Architektur

## Remote-Schnittstellenebene

Mit der Remote-Schnittstellenebene werden verschiedene Aspekte der Client/Server-Interaktion gesteuert. Es werden Middleware-Probleme außerhalb der Geschäftslogik der Anwendung behandelt und der Code der Geschäftslogik wird so abgeschirmt, als ob er kein Teil einer Client/-Server-Umgebung wäre.

Die Funktionen der Remote-Schnittstellenebene lauten wie folgt:

### Lebenszyklus-Management des Geschäftsobjekts

Erstellen/Suchen geeigneter Geschäftsobjekte zum Bedienen von Clientanforderungen

### Berechtigung

Prüfen der Berechtigungen eines Clients zum Ausführen einer bestimmten Serverfunktion

### Transaktionssteuerung

Starten, Durchführen von Commits und Rollbacks von atomaren Transaktionen

### Marshalling von Argumenten

Übersetzen verschiedener Datenformate bei der Client/Server-Übertragung

### Letzte Ausnahmebehandlung

Behandlung von Ausnahmebedingungen, die vom Anwendungscode nicht richtig behandelt wurden, und Protokollierung dieser Ausnahmebedingungen an einem zentralen Standort sowie Durchführung einer einheitlichen Berichterstellung für Clientanwendungen.

## Geschäftsobjektebene

Die Geschäftsobjektebene enthält die Anwendungsgeschäftslogik, einschließlich der „Prozessobjekte“, die die geschäftsspezifische Funktionalität implementieren, sowie der „Entitätsobjekte“, die indirekt die Datenzugriffe ausführen.

Diese Ebene ist die einzige Ebene in der IBM Cúram Social Program Management-Architektur, die eine Logik enthält, die direkt von den Anwendungsentwicklern codiert wird. Bestimmte Muster werden nach der Konvention in „sofort einsatzfähigen“ Anwendungen verwendet. Die Geschäftsobjektebene unterteilt sich in die Fassaden- und Serviceebene. Die verschiedenen Objekttypen im folgenden Diagramm werden zusammen als „Geschäftsobjekte“ bezeichnet:

Die Interaktion zwischen der Fassaden- und Serviceebene wird im folgenden Diagramm dargestellt.

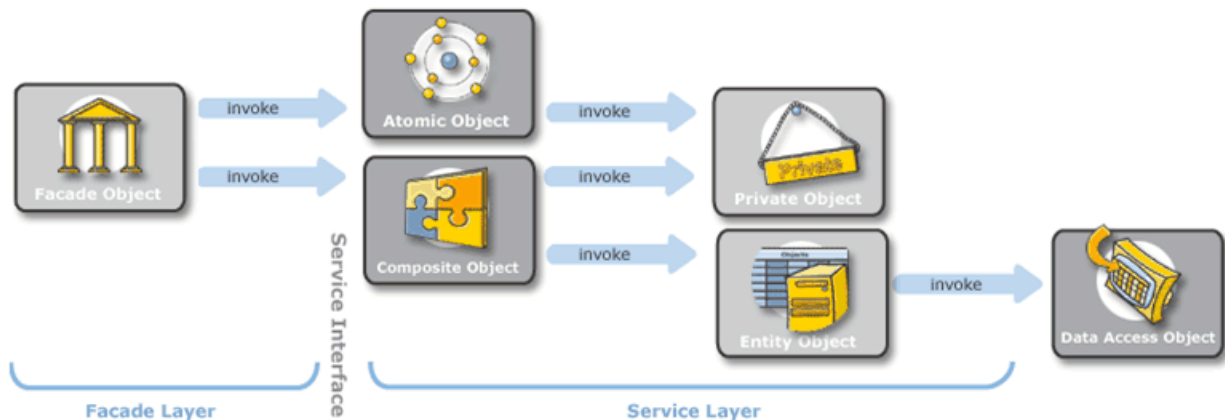


Abbildung 6. Geschäftsobjektebene

## Fassadenebene

Eine Fassadenebene ist eine Geschäftskomponente auf höherer Ebene, die komplexe Interaktionen zwischen Geschäftskomponenten niedrigerer Ebenen enthält und zentralisiert.

- Die Fassadenebene stellt die öffentliche (ferne) Schnittstelle zum Server dar.
- Fassadenobjekte können eine schnittstellenspezifische Verarbeitung durchführen.
- Fassadenobjekte sequenzieren Aufrufe zur Serviceebene.
- Fassadenobjekte sind „Prozessobjekte“.

## Serviceebene

Die Serviceebene enthält die Kerngeschäftsprozesse von IBM Cúram Social Program Management. Folgende Prozesstypen werden in der Serviceebene verwendet:

- „Atomare“ Prozesse sind Einheiten der niedrigsten Ebene, die den Server in einem konsistenten Zustand verlassen.
- „Zusammengesetzte“ Prozesse sind Fertigprozesse, die mehrere atomare (unteilbare) Operationen sequenzieren.

## Factory-Methoden

IBM Cúram Social Program Management verwendet Factory-Methoden, um eine Schnittstelle für die Erstellung von Geschäftsobjekten bereitzustellen, ohne dabei ihre konkrete Klasse angeben zu müssen. Alle Geschäftsobjekte (Prozesse und Entitäten) werden erstellt, indem die Factory-Methoden aufgerufen werden.

- Factorys unterstützen optional eine sehr differenzierte Tracefunktion mithilfe von dynamischen Java-Proxys.
- Factorys sind für die Fehlerbehebung und das Verständnis des Steuerungsablaufs nützlich.

## Connectors

Connectors von IBM Cúram Social Program Management sind eine Gruppe von Tools, die die Anwendungsintegration vereinfachen. Anwendungsentwickler können mit Connectors Objekte entwickeln, die Daten darstellen, die von traditionellen oder anderen Anwendungsplattformen auf dieselbe Art und Weise wie bei der Verwendung von Geschäfts- oder Entitätsobjekten von IBM Cúram Social Program Management abgerufen werden oder persistent gespeichert sind.

Connectors machen außerdem das Schreiben von Code auf „niedriger Ebene“, um auf traditionelle Systeme zuzugreifen, überflüssig. Der Mechanismus der Kommunikation mit der traditionellen Plattform wird dem Entwickler nicht angezeigt.

Connectors werden für Folgendes bereitgestellt:

- IBM MQSeries
- Web-Services
- SQL

---

## Entwicklungsarchitektur

Die Entwicklungsumgebung von IBM Cúram Development Environment setzt sich aus einer Cúram-Serverentwicklungsumgebung (SDEJ) und einer Cúram-Cliententwicklungsumgebung (CDEJ) zusammen.

Der Entwicklungsansatz von IBM Cúram hat folgende wichtige Funktionen:

- Einen auf einem Metamodell basierenden Entwicklungsansatz
- Anwendungsmodell auf der Basis von UML
- Codegeneratoren zur Erstellung von wichtigen Teilen der Client/Server-Anwendungen
- Vereinfachter Code der selbstgeschriebenen Geschäftslogik
- Vereinfachte Entwicklung von Benutzerschnittstellen
- Übergeordnete Geschäftsinfrastruktur

## Serverentwicklungsumgebung

In der Serverentwicklungsumgebung wird ein modellbasierter Ansatz für die Entwicklung verwendet. Ein Anwendungsmodell wird mit einem UML-Modellierungstool geschäftszentriert und plattformunabhängig definiert.

Das Modell ist der wichtigste Baustein für die Codegeneratoren, da alle notwendigen Entitäts- und Prozessobjekte definiert werden.

Die Generatoren erstellen die erforderlichen Klassen und Dateien für die Anwendungsstruktur. Diese Struktur enthält außerdem den Code der Remote-Schnittstellenebene, der Datenzugriffsebene und der Geschäftsobjektebene sowie selbst geschriebenen Code.

### Anwendungsmodell

Das Anwendungsmodell ist die Grundlage für die Serverentwicklung und besteht aus mehreren Elementen.

#### Domänen:

Anwendungsspezifische Datentypen

#### Entitäten:

Die modellierten und von der Anwendung persistent gespeicherten Objekte

#### Prozesse:

Zusammengehöriger Satz von Aktivitäten zur Erfüllung eines Geschäftsziels

#### Wertobjekte:

Werden als Nachrichten durch die Anwendung übergeben

### Anwendungsmodelldomänen

Anwendungsmodelldomänen werden in Bezug auf „grundlegende“ Datentypen (Zeichenfolge, ganze Zahlen usw.) oder andere Domänen definiert. Diese Domänen haben anwendungsspezifische Typnamen wie „SOCIAL\_SECURITY\_NUMBER“ oder „PAYMENT\_AMOUNT“ und bilden zusammen eine Baumstruktur, die „Domänenhierarchie“ genannt wird.

Domänen können zugehörige Validierungen wie Großschreibung, Bereichsprüfungen, Codetabellen, Musterübereinstimmungen oder angepasste Validierungen haben. Attribute von Entitäten und Wertobjekte werden als Domänen angegeben.

Im Folgenden wird die Vererbungshierarchie von Domänenklassen, die bei "root" beginnen, mit jeder niedrigeren Ebene und eingerückten Aufzählungszeichen, die von der vorhergehenden höheren Ebene vererbt wird, dargestellt:

- root
  - STRING
    - FIRSTNAME
    - SURNAME
    - ADDRESSLINE
  - INT\_64
    - UNIQUE\_ID
      - PERSON\_ID
  - DATE
    - DATE\_OF\_BIRTH

## Anwendungsmodellentitäten

Entitäten haben Attribute, die als Domänentypen definiert sind, sowie primäre und alternative Schlüssel und Fremdschlüsselbeziehungen.

Für die Stiloperationen „Create, Read, Update, Delete“ (CRUD; Erstellen, Lesen, Aktualisieren, Löschen), die als stereotypische Methoden in UML angegeben werden, sind keine Implementierungen erforderlich. Andere Geschäftsmethoden können auch verwendet werden, wenn Signaturen in UML verwendet werden. Komplexe Datenbankoperationen können in SQL angegeben werden.

Der persistente Speicher besitzt Pre- und Post-Zugriffs-Exitpunkte.

## Anwendungsmodellprozesse

Beliebige Geschäftsfunktionen werden im IBM Cúram Social Program Management-Modell als Methoden von Prozessklassen dargestellt. Methodenargumente werden als Wertobjekte modelliert.

Das Modell definiert die Schnittstelle, aber nicht die Implementierung von Prozessobjekten.

## Server-Code-Generierung

Ein plattformspezifisches Modell wird automatisch anhand des plattformunabhängigen Metamodells generiert. Der Code wird für eine EJB-Serveranwendung automatisch generiert.

Auf der Abbildung werden die wichtigsten generierten Artefakte des Prozesses zur Server-Code-Generierung dargestellt.

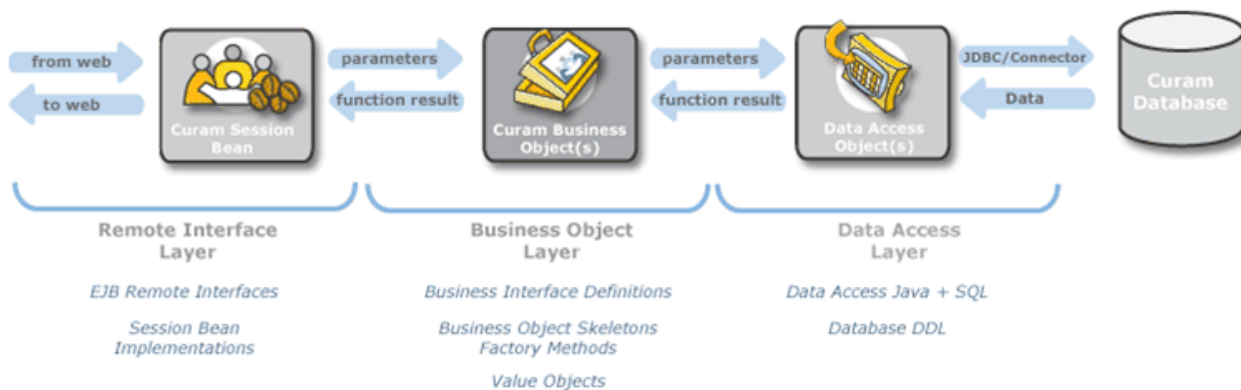


Abbildung 7. Server-Code-Generierung

## Cliententwicklungsumgebung

Der IBM Cúram-Client enthält von JSPs generierte HTML-Seiten. Die JSPs werden anhand von XML-Anzeigedefinitionen generiert und die Formatierung der Anzeigeseiten werden von Formatvorlagen bestimmt. Die XML-Anzeigedefinitionen sind von der Darstellungsebene unabhängig und das für IBM Cúram spezifische Format wird UIM-Format (UIM: User Interface Metadata, Benutzerschnittstellenmetadaten) genannt.

Die automatische Datenvalidierung/-konvertierung basiert auf Anwendungsmodelldefinitionen, die angepasste Widgets und JavaScript-Exitpunkte unterstützen.

## Entwicklung der Benutzerschnittstelle

Mithilfe der Entwicklungsumgebung von IBM Cúram Social Program Management soll die Erstellung einer Benutzerschnittstelle vereinfacht werden. Sie können Kundenseiten bestimmten Back-End-Serverschnittstellen zuordnen. Da die Metadaten zu diesen Schnittstellen bereits im Anwendungsmodell erfasst wurden, können diese Informationen für die Erstellung der Benutzerschnittstelle verwendet werden.

Der Großteil der übrigen Arbeit für Kundenentwickler besteht darin, die Felder für die einzelnen Kundenseiten aufzulisten. Standardwidgettypen werden für Felder und Steuerelemente in der Anzeige je nach den Informationen der den Feldern zugeordneten Datentypen bereitgestellt.

Die Felder haben ein Rasterlayout in den Clustern und Listen, die zusammen mit der allgemeinen Seitenhierarchie in XML angegeben sind. Die Widgettypen werden von den Verbindungen automatisch bestimmt.

Beispiel des UIM-Codes (User Interface Metadata) für das Feld `firstname`:

```
<FIELD LABEL="Field.Label.FirstName">
<CONNECT>
<SOURCE NAME="Interface1" PROPERTY="firstForename"/>
</CONNECT>
a
</FIELD>
```

Das Attribut LABEL des Elements FIELD beschreibt den Beschriftungstext, der diesem Feld zugeordnet wird, wenn eine Kundenseite angezeigt wird. Der Wert "Field.Label.FirstName" ist eine Referenz auf den tatsächlichen Beschriftungswert in einer anderen Eigenschaftendatei. Die Verwendung von Zeichenfolgen, die in Eigenschaftendateien externalisiert werden, ermöglicht eine einfache Suche nach den Kundenanwendungen. Das Element SOURCE beschreibt, woher die Feldinhalte stammen, wenn die Seite angezeigt wird. Das Attribut NAME des Elements SOURCE gibt eine bestimmte Back-End-Schnittstelle an, die im Anwendungsmodell definiert ist. Das Attribut PROPERTY gibt ein bestimmtes Datum an, das von der Back-End-Schnittstelle zurückgegeben wird.

Die genaue Position des Feldes auf der Anzeige wird nicht angegeben. Es gibt keine Informationen zum Datentyp des Feldes oder zu den zugeordneten HTML-Steuerelementen. Diese Informationen werden während der Erstellungszeit der Anwendung automatisch angegeben. Durch die Vereinfachung der UIM-Seiten können Sie einfach aus Vorlagen kopiert bzw. eingefügt werden, sodass der Entwickler produktiver arbeiten kann.



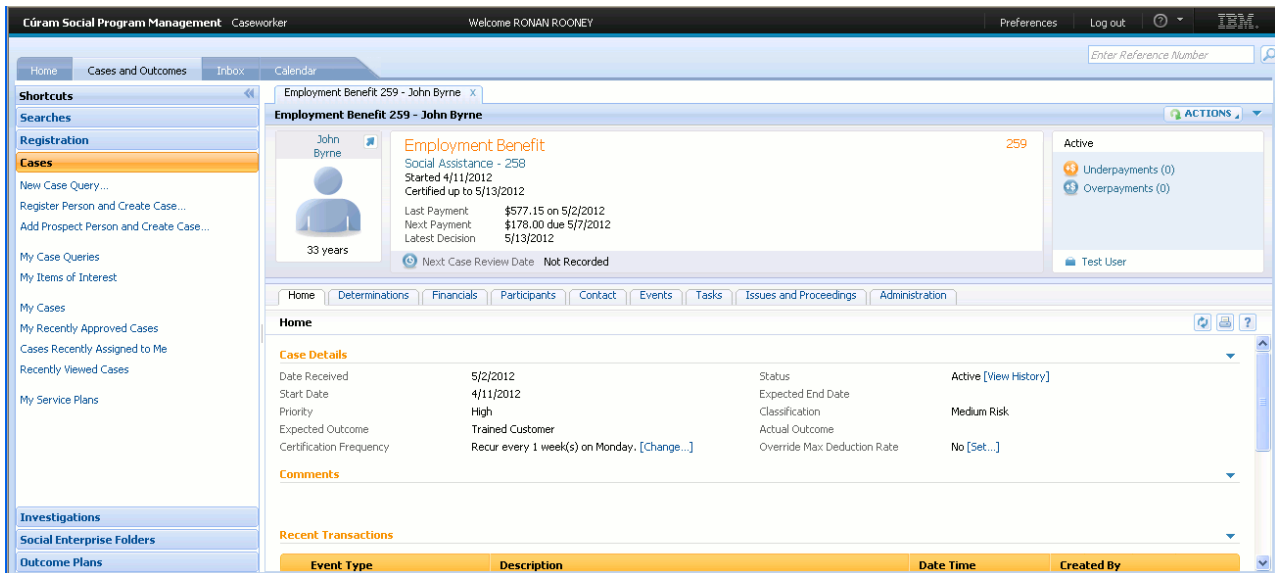


Abbildung 8. Beispielbenutzerschnittstelle in IBM Cúram

## Client-Code-Generierung

Im folgenden Diagramm werden die wichtigsten generierten Artefakte des Client-Code-Generierungsprozesses dargestellt.

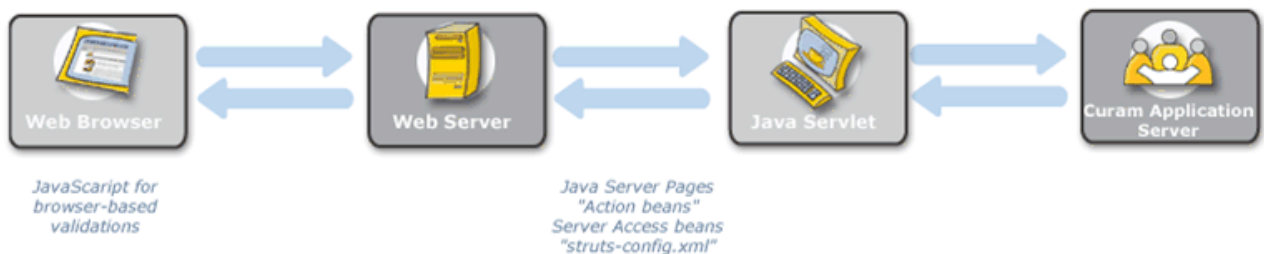


Abbildung 9. Client-Code-Generierung

## Geschäftsinfrastruktur

IBM Cúram stellt eine umfassende Geschäftsinfrastruktur bereit.

### Entwicklungsumgebung für Regeln

Cúram Express Rules (kurz CER) ist eine Sprache für die Definition von Fragen, die gestellt werden können, und die Bestimmung der Antworten auf diese Fragen. Die Umgebung für die Entwicklung von CER-Regeln ist der CER Rules Editor.

In CER wird für jede Frage Folgendes angegeben:

- Den Namen der Frage
- Den Datentyp, der die Antwort auf die Frage bereitstellt
- Die Regeln zum Bereitstellen der Antwort (wenn die Frage gestellt wurde)



Die Antwort auf eine Frage kann eine einfache Ja/Nein-Antwort sein, zum Beispiel auf die Frage "Ist diese Person berechtigt, Leistungen zu erhalten?". Mit CER können Sie jedoch auch komplexe Antworttypen definieren. Zum Beispiel würde die Antwort auf die Frage "Welche Gruppen von Personen im Haushalt haben einen dringenden Bedarf?" mit einer Liste von Haushaltsgruppen beantwortet, bei der jede Haushaltsgruppe aus einer Liste von Personen besteht.

Die Umgebung für die Entwicklung von CER-Regeln ist der CER Rules Editor. Dieser Editor bietet sowohl technischen als auch geschäftlichen Benutzern eine benutzerfreundliche Umgebung und Schnittstelle, um einen Regelsatz und zugehörige Regelklassen zu erstellen, zu bearbeiten und zu validieren. Weitere Informationen zu CER Rules Editor finden Sie in den Handbüchern *IBM Cúram Express Rules Reference Manual* und *Working With Cúram Express Rules*.

## Verarbeitung der Anspruchs- und Leistungsberechtigung

Die Eligibility and Entitlement Engine stellt einen Mechanismus zur Bestimmung der Anspruchs- und Leistungsberechtigung in einem Fall über beliebige Zeitspannen bereit. Durch diese Bestimmung wird Cúram das automatische Beurteilen und erneute Beurteilen von Fällen ermöglicht, wobei Gesetzesänderungen, Änderungen der Sätze, Änderungen der Bedingungen bei den Bürgern und andere Komplexitäten berücksichtigt werden.

In Eligibility and Entitlement wird für diese Funktionalität Cúram Express Rules verwendet. Die CER-Engine unterstützt diesen Prozess durch Befolgen der im CER Rules Editor definierten Regeln und der für einen Fall erfassten Angaben. Da die CER-Regeln flexibel sind, können sie zum Definieren von Anzeigeregeln verwendet werden. So können einem Geschäftsbenutzer die Informationen zu einem Fall übermittelt werden. Diese Regeln sind knapp oder ausführlich, je nachdem wie sie benötigt werden.

Weitere Informationen zur Bestimmung von Anspruchs- und Leistungsberechtigungen finden Sie im *Cúram-Handbuch zu den Grundlagen der Anspruchsberechtigung und Leistungshöhe mit Cúram Express Rules (CER)*.

## XML und Drucken

Die Druckverarbeitung von IBM Cúram Social Program Management basiert auf der Zusammenführung von Daten mit Dokumentvorlagen, um eine finale Dokumentausgabe zu produzieren.

Dokumentvorlagen werden in XSL (Extensible Stylesheet Language) erstellt und in der Anwendungsdatenbank gespeichert. Geschäftsprozesse, mit denen Druckausgaben erstellt werden, produzieren Daten in Form eines XML-Dokuments und senden diese Daten zusammen mit der ID des entsprechenden Stylesheets, um diese Daten zu formatieren, an einen Druckserver.

Das Erzeugen von XML-Daten in IBM Cúram Social Program Management-Prozessen ist einfach, da jedes „Wertobjekt“ in XML-Daten konvertiert werden kann. Da die Kommunikation zwischen Geschäftsprozessen und Datenzugriffsobjekten in Form von Wertobjekten vorliegt, können alle Daten in XML konvertiert werden. Dies ist so einfach wie der Zugriff auf die Anwendungsdatenbank.

Der XML-Server von IBM Cúram Social Program Management kann anhand der XML-/XSL-Definitionen PDF-, RTF-, HTML- und Textdokumente erstellen.

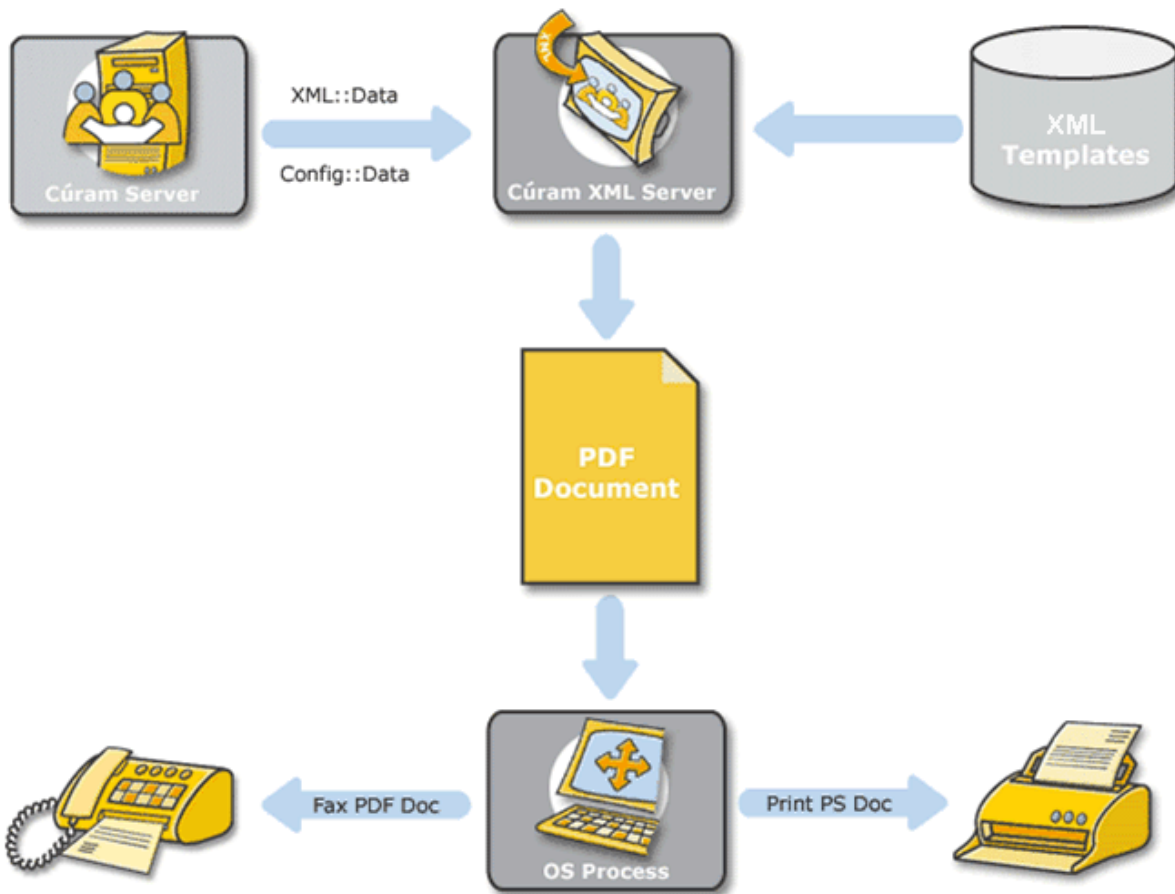


Abbildung 10. XML und PDF-Druck

## Workflow

Das Workflow-Management-System von IBM Cúram basiert auf den Standards von Workflow Management Coalition und ermöglicht Organisationen, Geschäftsprozesse in einzelne Aktivitäten aufzugliedern und dann zwischen ihnen flexible Beziehungen zu erstellen.

Die prozedurale Automatisierung eines Geschäftsprozesses erfolgt durch die Verwaltung der Abfolge von Arbeitsaktivitäten und die Zuordnung der entsprechenden Benutzer- bzw. Systemressourcen, die mit den verschiedenen Aktivitätsschritten verknüpft sind.

Das Workflow-Management-System von IBM Cúram enthält ein interaktives Prozessdefinitionstool, mit dem die Workflowaktivitäten und -übergänge definiert werden, sowie eine Workflow-Engine, mit der der Workflow in der Produktionsumgebung verwaltet wird. Außerdem ist eine Verwaltungskomponente zum Pflegen von Instanzen der Workflowprozessdefinitionen enthalten.

So wird sichergestellt, dass Organisationen ihre Prozesse flexibel bereitstellen und verwalten sowie schneller auf Änderungen reagieren können.

In den Workflowaktivitäten ist ein automatisierter Schritt (Aufruf eines Cúram-Geschäftsprozesses) oder ein manueller Schritt (Erstellung einer manuell auszuführenden Aufgabe für einen Benutzer) enthalten.



---

## Bemerkungen

Die vorliegenden Informationen wurden für Produkte und Services entwickelt, die auf dem deutschen Markt angeboten werden. Möglicherweise bietet IBM die in dieser Dokumentation beschriebenen Produkte, Services oder Funktionen in anderen Ländern nicht an. Informationen über die gegenwärtig im jeweiligen Land verfügbaren Produkte und Services sind beim zuständigen IBM Ansprechpartner erhältlich. Hinweise auf IBM-Lizenzprogramme oder andere IBM Produkte bedeuten nicht, dass nur Programme, Produkte oder Services von IBM verwendet werden können. Anstelle der IBM Produkte, Programme oder Services können auch andere, ihnen äquivalente Produkte, Programme oder Services verwendet werden, solange diese keine gewerblichen oder anderen Schutzrechte von IBM verletzen. Die Verantwortung für den Betrieb von Produkten, Programmen und Services anderer Anbieter liegt beim Kunden. Für die in diesem Handbuch beschriebenen Erzeugnisse und Verfahren kann es IBM Patente oder Patentanmeldungen geben. Mit der Auslieferung dieses Handbuchs ist keine Lizenzierung dieser Patente verbunden. Lizenzanforderungen sind schriftlich an folgende Adresse zu richten (Anfragen an diese Adresse müssen auf Englisch formuliert werden):

IBM Director of Licensing

IBM Europe, Middle East & Africa

Tour Descartes

2, avenue Gambetta

92066 Paris La Defense

France

Trotz sorgfältiger Bearbeitung können technische Ungenauigkeiten oder Druckfehler in dieser Veröffentlichung nicht ausgeschlossen werden.

Die hier enthaltenen Informationen werden in regelmäßigen Zeitabständen aktualisiert und als Neuausgabe veröffentlicht. IBM kann ohne weitere Mitteilung jederzeit Verbesserungen und/oder Änderungen an den in dieser Veröffentlichung beschriebenen Produkten und/oder Programmen vornehmen. Verweise in diesen Informationen auf Websites anderer Anbieter werden lediglich als Service für den Kunden bereitgestellt und stellen keinerlei Billigung des Inhalts dieser Websites dar.

Das über diese Websites verfügbare Material ist nicht Bestandteil des Materials für dieses IBM Produkt. Die Verwendung dieser Websites geschieht auf eigene Verantwortung.

Werden an IBM Informationen eingesandt, können diese beliebig verwendet werden, ohne dass eine Verpflichtung gegenüber dem Einsender entsteht. Lizenznehmer des Programms, die Informationen zu diesem Produkt wünschen mit der Zielsetzung: (i) den Austausch von Informationen zwischen unabhängig voneinander erstellten Programmen und anderen Programmen (einschließlich des vorliegenden Programms) sowie (ii) die gemeinsame Nutzung der ausgetauschten Informationen zu ermöglichen, wenden sich an folgende Adresse:

IBM Corporation  
Dept F6, Bldg 1  
294 Route 100  
Somers NY 10589-3216  
U.S.A.

Die Bereitstellung dieser Informationen kann unter Umständen von bestimmten Bedingungen - in einigen Fällen auch von der Zahlung einer Gebühr - abhängig sein.

Die Bereitstellung des in diesem Dokument beschriebenen Lizenzprogramms sowie des zugehörigen Lizenzmaterials erfolgt auf der Basis der IBM Rahmenvereinbarung bzw. der Allgemeinen Geschäftsbedingungen von IBM, der IBM Internationalen Nutzungsbedingungen für Programmpakete oder einer äquivalenten Vereinbarung.

Alle in diesem Dokument enthaltenen Leistungsdaten stammen aus einer kontrollierten Umgebung. Die Ergebnisse, die in anderen Betriebsumgebungen erzielt werden, können daher erheblich von den hier erzielten Ergebnissen abweichen. Einige Daten stammen möglicherweise von Systemen, deren Entwicklung noch nicht abgeschlossen ist. Eine Gewährleistung, dass diese Daten auch in allgemein verfügbaren Systemen erzielt werden, kann nicht gegeben werden. Darüber hinaus wurden einige Daten unter Umständen durch Extrapolation berechnet. Die tatsächlichen Ergebnisse können davon abweichen. Benutzer dieses Dokuments sollten die entsprechenden Daten in ihrer spezifischen Umgebung prüfen.

Alle Informationen zu Produkten anderer Anbieter stammen von den Anbietern der aufgeführten Produkte, deren veröffentlichten Ankündigungen oder anderen allgemein verfügbaren Quellen.

IBM hat diese Produkte nicht getestet und kann daher keine Aussagen zu Leistung, Kompatibilität oder anderen Merkmalen machen. Fragen zu den Leistungsmerkmalen von Produkten anderer Anbieter sind an den jeweiligen Anbieter zu richten.

Aussagen über Pläne und Absichten von IBM unterliegen Änderungen oder können zurückgenommen werden und repräsentieren nur die Ziele von IBM.

Alle von IBM angegebenen Preise sind empfohlene Richtpreise und können jederzeit ohne weitere Mitteilung geändert werden. Händlerpreise können u. U. von den hier genannten Preisen abweichen.

Diese Veröffentlichung dient nur zu Planungszwecken. Die in dieser Veröffentlichung enthaltenen Informationen können geändert werden, bevor die beschriebenen Produkte verfügbar sind.

Diese Veröffentlichung enthält Beispiele für Daten und Berichte des alltäglichen Geschäftsablaufs. Sie sollen nur die Funktionen des Lizenzprogramms illustrieren und können Namen von Personen, Firmen, Marken oder Produkten enthalten. Alle diese Namen sind frei erfunden; Ähnlichkeiten mit tatsächlichen Namen und Adressen sind rein zufällig.

## COPYRIGHTLIZENZ:

Diese Veröffentlichung enthält Musteranwendungsprogramme, die in Quellsprache geschrieben sind und Programmier Techniken in verschiedenen Betriebsumgebungen veranschaulichen. Sie dürfen diese Musterprogramme kostenlos kopieren, ändern und verteilen, wenn dies zu dem Zweck geschieht, Anwendungsprogramme zu entwickeln, zu verwenden, zu vermarkten oder zu verteilen, die mit der Anwendungsprogrammierschnittstelle für die Betriebsumgebung konform sind, für die diese Musterprogramme geschrieben werden. Diese Beispiele wurden nicht unter allen denkbaren Bedingungen getestet. IBM kann daher die Zuverlässigkeit, Wartungsfreundlichkeit oder Funktion dieser Programme nicht garantieren oder implizieren. Die Beispielprogramme werden ohne Wartung (auf "as-is"-Basis) und ohne jegliche Gewährleistung zur Verfügung gestellt. IBM übernimmt keine Haftung für Schäden, die durch Ihre Verwendung der Musterprogramme entstehen.

Kopien oder Teile der Musterprogramme bzw. daraus abgeleiteter Code müssen folgenden Copyrightvermerk beinhalten:

© (Name Ihres Unternehmens) (Jahr). Teile des vorliegenden Codes wurden aus Musterprogrammen der IBM Corp. abgeleitet.

© Copyright IBM Corp. \_Jahreszahl oder Jahreszahlen eingeben\_. Alle Rechte vorbehalten.

Wird dieses Buch als Softcopy (Book) angezeigt, erscheinen keine Fotografien oder Farbabbildungen.

---

## Hinweise zur Datenschutzrichtlinie

IBM Softwareprodukte, einschließlich Software as a Service-Lösungen ("Softwareangebote"), können Cookies oder andere Technologien verwenden, um Informationen zur Produktnutzung zu erfassen, die Endbenutzererfahrung zu verbessern und Interaktionen mit dem Endbenutzer anzupassen oder zu anderen Zwecken. In vielen Fällen werden von den Softwareangeboten keine personenbezogenen Daten erfasst. Einige der IBM Softwareangebote können Sie jedoch bei der Erfassung personenbezogener Daten unterstützen. Wenn dieses Softwareangebot Cookies zur Erfassung personenbezogener Daten verwendet, sind nachfolgend nähere Informationen über die Verwendung von Cookies durch dieses Angebot zu finden.

Je nachdem, welche Konfigurationen implementiert wurden, ist es möglich, dass dieses Softwareangebot Sitzungscookies und persistente Cookies zum Erfassen der Namen, Benutzernamen, Kennwörter, Profilnamen oder anderer personenbezogener Daten einzelner Benutzer für die Sitzungsverwaltung, Authentifizierung, Single-Sign-on-Konfiguration oder für einen besseren Bedienungskomfort und/oder andere Zwecke der Nutzungsverfolgung bzw. funktionale Einsatzmöglichkeiten. Diese Cookies oder ähnliche Technologien können nicht inaktiviert werden.

Wenn die für dieses Softwareangebot genutzten Konfigurationen Sie als Kunde in die Lage versetzen, personenbezogene Daten von Endbenutzern über Cookies und andere Technologien zu erfassen, müssen Sie sich zu allen gesetzlichen Bestimmungen in Bezug auf eine solche Datenerfassung, einschließlich aller Mitteilungspflichten und Zustimmungsanforderungen, rechtlich beraten lassen.

Weitere Informationen zur Nutzung verschiedener Technologien, einschließlich Cookies, für diese Zwecke finden Sie in der "IBM Online-Datenschutzerklärung, Schwerpunkte" unter <http://www.ibm.com/privacy> und in der "IBM Online-Da-

tenschutzklärung" unter <http://www.ibm.com/privacy/details> im Abschnitt "Cookies, Web-Beacons und sonstige Technologien" und unter "IBM Software Products and Software-as-a-Service Privacy Privacy Statement" unter <http://www.ibm.com/software/info/product-privacy>.

---

## Informationen zu Programmierschnittstellen

In der vorliegenden Veröffentlichung werden vorgesehene Programmierschnittstellen dokumentiert, mit deren Hilfe Kunden Programme für den Zugriff auf IBM Cúram Social Program Management-Services schreiben können.

---

## Marken

IBM, das IBM Logo und [ibm.com](http://www.ibm.com) sind eingetragene Marken der International Business Machines Corporation in den USA und/oder anderen Ländern. Weitere Produkt- und Servicennamen können Marken von IBM oder anderen Unternehmen sein. Eine aktuelle Liste der IBM Marken finden Sie auf der Website "Copyright and trademark information" unter <http://www.ibm.com/legal/us/en/copytrade.shtml>.

Apache ist eine Marke der Apache Software Foundation.

Microsoft und Internet Explorer sind Marken der Microsoft Corporation in den USA und/oder anderen Ländern.

Oracle, Java und alle Java-basierten Marken sind eingetragene Marken der Oracle Corporation und/oder ihrer verbundenen Unternehmen.

Sonstige Namen können Marken der jeweiligen Rechtsinhaber sein. Weitere Firmen-, Produkt- und Servicennamen können Marken oder Servicemarken anderer Unternehmen sein.







Gedruckt in Deutschland