

Vorlesung Wintersemester 2011/12

Workflow-Management-Systeme

Kapitel 2:
Aspekte und Komponenten von Workflow-
Management-Systemen

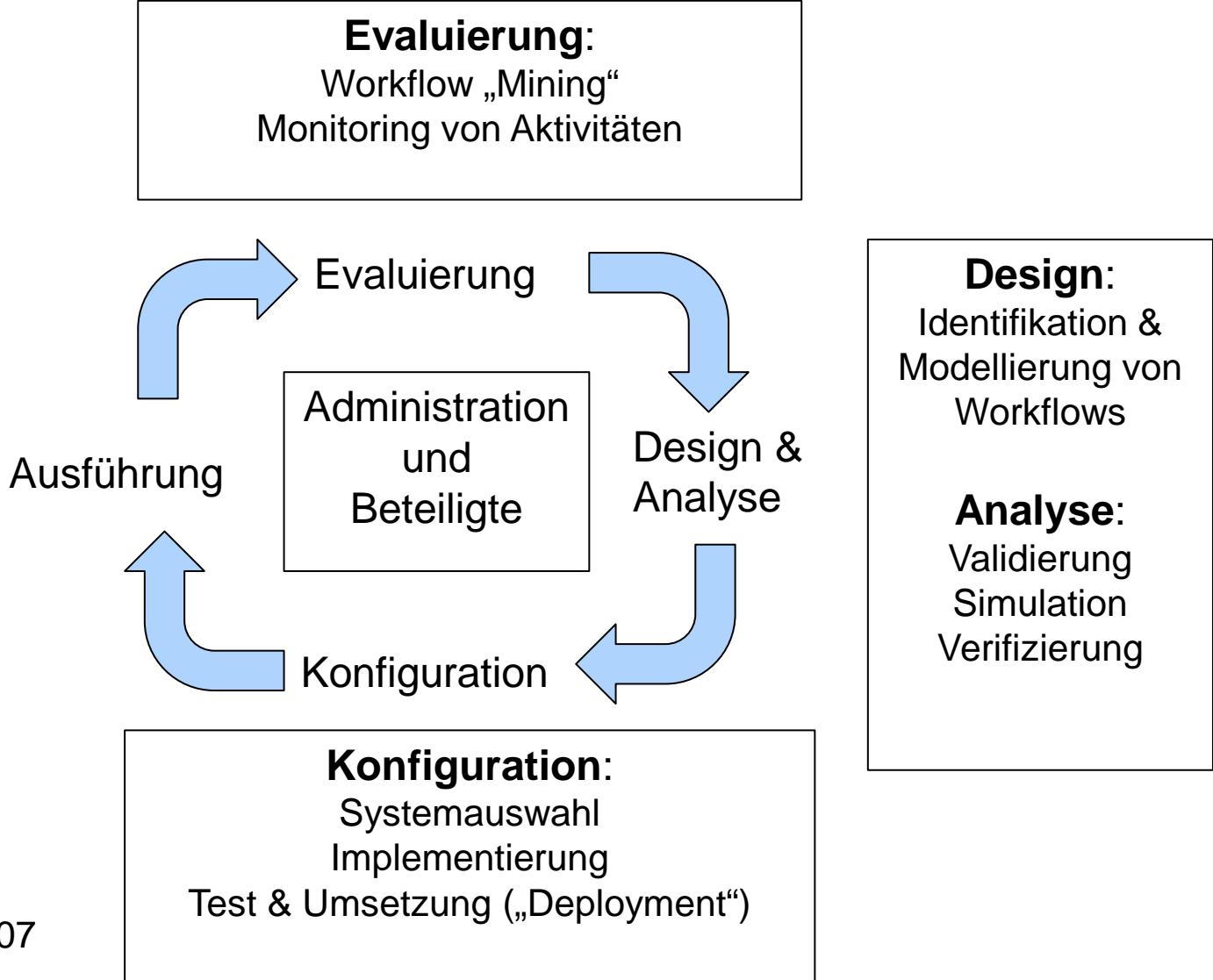
Überblick Kapitel 2

- ◆ Lebenszyklus von Workflows
- ◆ Überblick Modellierung und Ausführung
- ◆ Workflow-Aspekte
 - Begriffe
 - Architekturansätze
- ◆ Integration von WfMS in Anwendungssysteme
- ◆ Referenzmodell der WfMC
 - Überblick
 - Komponenten und Schnittstellen
 - Anmerkungen

Lebenszyklus von Workflows (1)

Lebenszyklus
Überblick
Aspekte
Integration
Referenzmodell

Ausführung:
Durchführung
(„Enactment“)
Monitoring
Wartung



Quelle: M. Weske, 2007

Lebenszyklus von Workflows (2)

◆ Design und Analyse

- Design:
 - Identifikation, Abgrenzung, Spezifikation von Zielen
 - Modellierung der Workflow Schemata typischerweise in graph. Notation (Kommunikationsbasis mit Anwendern)
- Analyse:
 - Validierung (Feedback der Anwender)
 - Simulation (Prozessverhalten, z.B. Dauer, Lastverteilung)
 - Verifizierung (z.B. Erkennung von Deadlocks)

◆ Konfiguration

- Systemauswahl (z.B. WfMS), Spezifikation von technischen Informationen, Integration existierender Systeme, Spezifikation von Benutzerinteraktionen
- Implementierung
- Test und Umsetzung („Deployment“)

Lebenszyklus von Workflows (3)

◆ Ausführung

- Durchführung („Enactment“): korrekte Ausführung („Orchestration“) von Workflowinstanzen gemäß der spezifizierten Bedingungen (zeitl. Ablauf, Rechte für Endbenutzer, etc.)
- Monitoring (z.B. Statusinformationen von aktiven Instanzen, Events in Log-Files)
- Betrieb und Wartung

◆ Evaluierung

- Workflow Mining (Auswertung von Log-Files zum Erkennen von Verhaltensmustern)
- Monitoring von Aktivitäten (z.B. Identifikation von Engpässen)

Zyklische Struktur mit logischen Abhängigkeiten

Kein striktes Vorgehensmodell

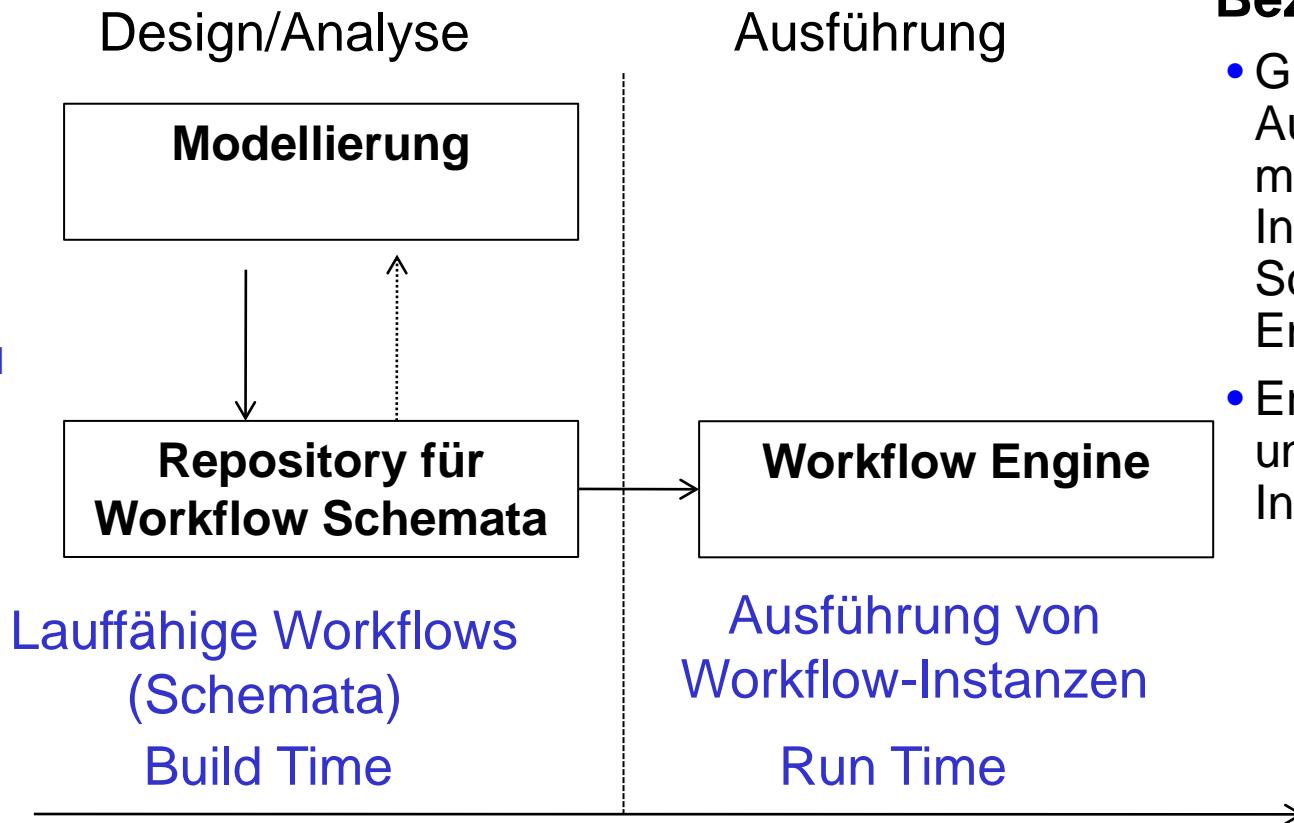
Vergleich Luftfahrt und Workflow-Management

Luftfahrt-Management	Workflow-Management
Gesamtes Flugnetz einer Luftfahrtgesellschaft (alle Flugverbindungen)	Gesamtheit aller Workflow-Modelle (Schemata) eines Unternehmens
Einzelstrecke (z.B. Karlsruhe-Mailand)	Workflow-Modell (Schema) eines Prozesses (z.B. Reisekostenabrechnung)
Gesamtheit aller Ressourcen (Flugzeuge, Piloten, Flugbegleiter, Bodenpersonal)	Ressourcen-Modell (Personen, externe Applikationen)
Zugeordnetes Flugzeug (z.B. HH1007 Karlsruhe-Mailand, 27.10.2010)	Workflow-Instanz (z.B. Reisekostenabrechnung Nr. 08/007 vom 18.10.2010)
Flugüberwachung der fliegenden Jets in der Flugleitzentrale	Monitoring der aktiven Workflow-Instanzen in Echtzeit
Analyse von Kennzahlen über Flugverspätungen, Störungen, Passagierzahlen u.a.	Analyse von ausgeführten Workflow-Instanzen hinsichtlich Durchlaufzeit, Liegezeit, Kosten u.a.

Quelle: nach Gadatsch 2007

Überblick Modellierung – Ausführung (1)

Lebenszyklus
Überblick
Aspekte
Integration
Referenzmodell



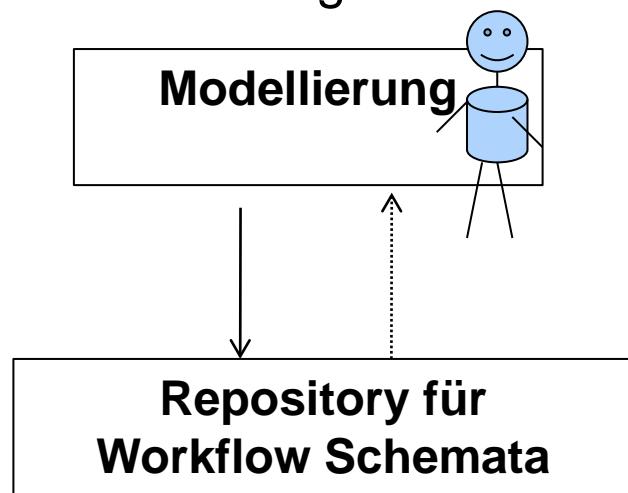
Beziehungen:

- Gleichzeitige Ausführung mehrerer Instanzen eines Schemas in Engine
- Engine führt unterschiedliche Instanzen aus

Quelle: M. Weske, 2007

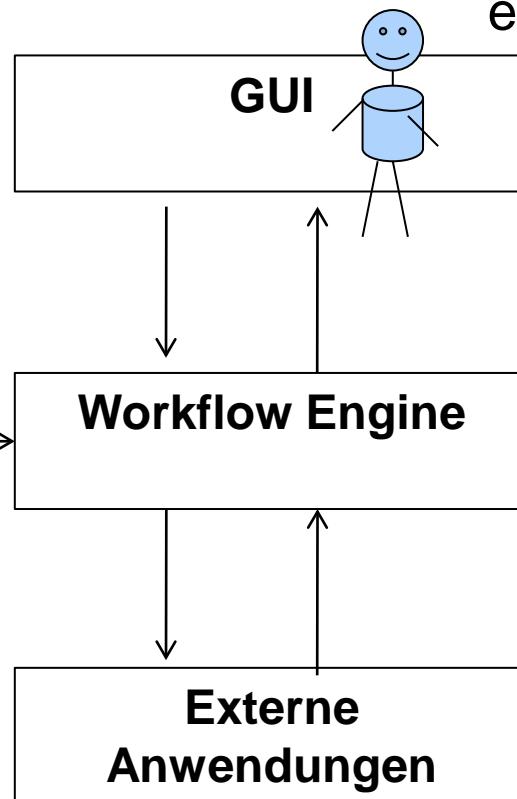
Überblick Modellierung – Ausführung (2)

Workflow-Designer



Lauffähige Workflows
(Schemata)

Endbenutzer (Akteur)
eines Workflows



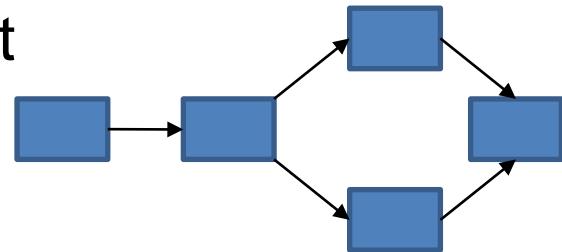
Ausführung von
Workflow-Instanzen

Quelle: M. Weske

Erstellung eines lauffähigen Workflows

Grobes Vorgehen:

1. Definition des Workflowmodells
(Schema) und anderer Daten mit
einem Modellierungstool
2. Spezifikation von Ressourcen
(z.B. Personen)
3. Implementierung oder Integration
der von den einzelnen
Arbeitsschritten ausgeführten
Programme



Ausführung einer Workflow-Instanz

1. Initialisieren und Starten einer neuen Workflow-Instanz
(gesteuert durch ein Ereignis, z.B. Eintreffen einer Dienstreisegenehmigung oder eines Auftrages)
2. Solange Arbeitsschritte auszuführen sind und kein „Terminate“-Arbeitsschritt erreicht wird:
 - Ausführung eines Arbeitsschrittes (automatisiert, semi-automatisiert)
 - Das weitere Vorgehen nach Beendigung des Arbeitsschrittes erfolgt nach den Vorgaben des Workflow-Schemas (Bedingungen, Verzweigungen, ...)
3. Terminierung der Workflow-Instanz

Workflow-Aspekte (1)

- ◆ **Funktionaler Aspekt:** beschreibt die funktionalen Einheiten, d.h. die Struktur des Workflows mit elementaren und zusammengesetzten (d.h. Sub-Workflows) Arbeitsschritten
- ◆ **Verhaltensbezogener Aspekt:** konstituiert den Kontrollfluss zwischen den elementaren und zusammengesetzten Arbeitsschritten eines Workflows unter Berücksichtigung von kausalen und temporalen Zusammenhängen
- ◆ **Datenbezogener Aspekt:** betrifft den Datenfluss eines Workflows

Workflow-Aspekte (2)

- ◆ **Organisations-Aspekt:** beschreibt die organisationsbezogenen Inhalte mit der Definition von Organisationsstrukturen und deren Population, sowie die Festlegung, wer die verschiedenen Operationen eines Workflows ausführen kann/darf
- ◆ **Operationaler Aspekt:** regelt die Einbindung von Applikationen in einen elementaren Arbeitsschritt. Applikationen sind Programme oder Programmsysteme, die für die Ausführung eines Workflows benötigt werden

Die genannten Aspekte bilden das Fundament eines WfMSs.

Erweiterungen sind möglich und in vielen Anwendungsfällen notwendig, z.B. ist in vielen Umgebungen ein Sicherheitsaspekt erforderlich, um Zugriffskontrolle zu regeln.

Implementierungsmodell eines WfMS (1)

- ◆ Ziel: Beschreibung des implementierungs-unabhängigen Teils der Architektur eines WfMSs
- ◆ Spezifikation der wesentlichen Module oder Komponenten eines WfMSs
- ◆ Die wesentlichen funktionalen Komponenten der Architektur eines WfMSs umfassen Module, welche die Aspekte eines WfMSs realisieren
- ◆ Ergänzend ist ein Steuermodul (nicht unbedingt zentral!) erforderlich sowie
- ◆ Hilfsmodule, welche allgemeine Dienste zur Verfügung stellen

Implementierungsmodell eines WfMS (2)

Steuermodul

funktionaler Aspekt

verhaltensbezogener
Aspekt

datenbezogener
Aspekt

organisatorischer
Aspekt

operationaler
Aspekt

weitere Aspekte

Datenbankzugriff

weitere Hilfsmodule

Wie kommt man zur Architektur eines Workflow Management Systems?

- ◆ Beim Übergang vom Implementierungsmodell zur Implementierungsarchitektur sind folgende Entscheidungen zu treffen:
 - Werden Module durch Basissysteme (z.B. Betriebssystem, Middleware) unterstützt?
 - Wie werden die (persistennten) Daten einer Komponente verwaltet?
 - Wie wird die Kommunikation zwischen den Modulen realisiert?
- ◆ Architekturansätze unterscheiden sich stark hinsichtlich dieser drei Fragen, daher keine allgemeine Implementierungsarchitektur
- ◆ Die Konkretisierung der Implementierungskonzepte erfolgt in der eigentlichen Implementierungsphase

Anforderungen an WfMS

◆ **Funktionale** Anforderungen

- Werden vor allem durch die Workflow-Sprache und ihre Ausdrucksfähigkeit und durch Vorgaben bezüglich der Benutzerschnittstellen bestimmt

◆ **Nicht-Funktionale** Anforderungen

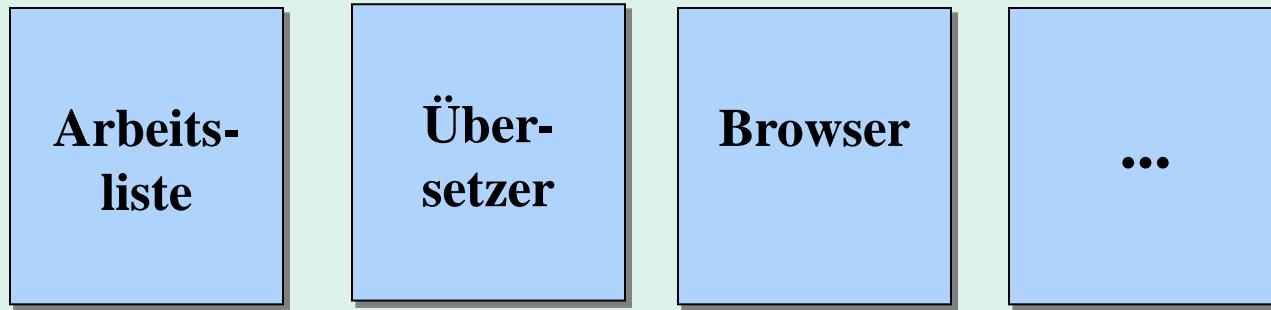
- Werden vor allem durch das beabsichtigte Einsatzgebiet eines Workflow-Management-Systems beeinflusst

Funktionale Anforderungen

- ◆ Implementierung des Workflow-Meta-Schemas
 - Funktionsaspekt
 - Verhaltensaspekt
 - Informationsaspekt
 - Organisationsaspekt
- ◆ Applikationsintegration (Operationsaspekt): Nutzung der Eigenschaften externer Programme, z.B. hinsichtlich transaktionaler Ausführung
- ◆ Bereitstellung von Benutzerschnittstellen
 - Endbenutzer (Anwender), Entwickler, Administrator, etc.

Funktionale Komponenten eines WfMSs

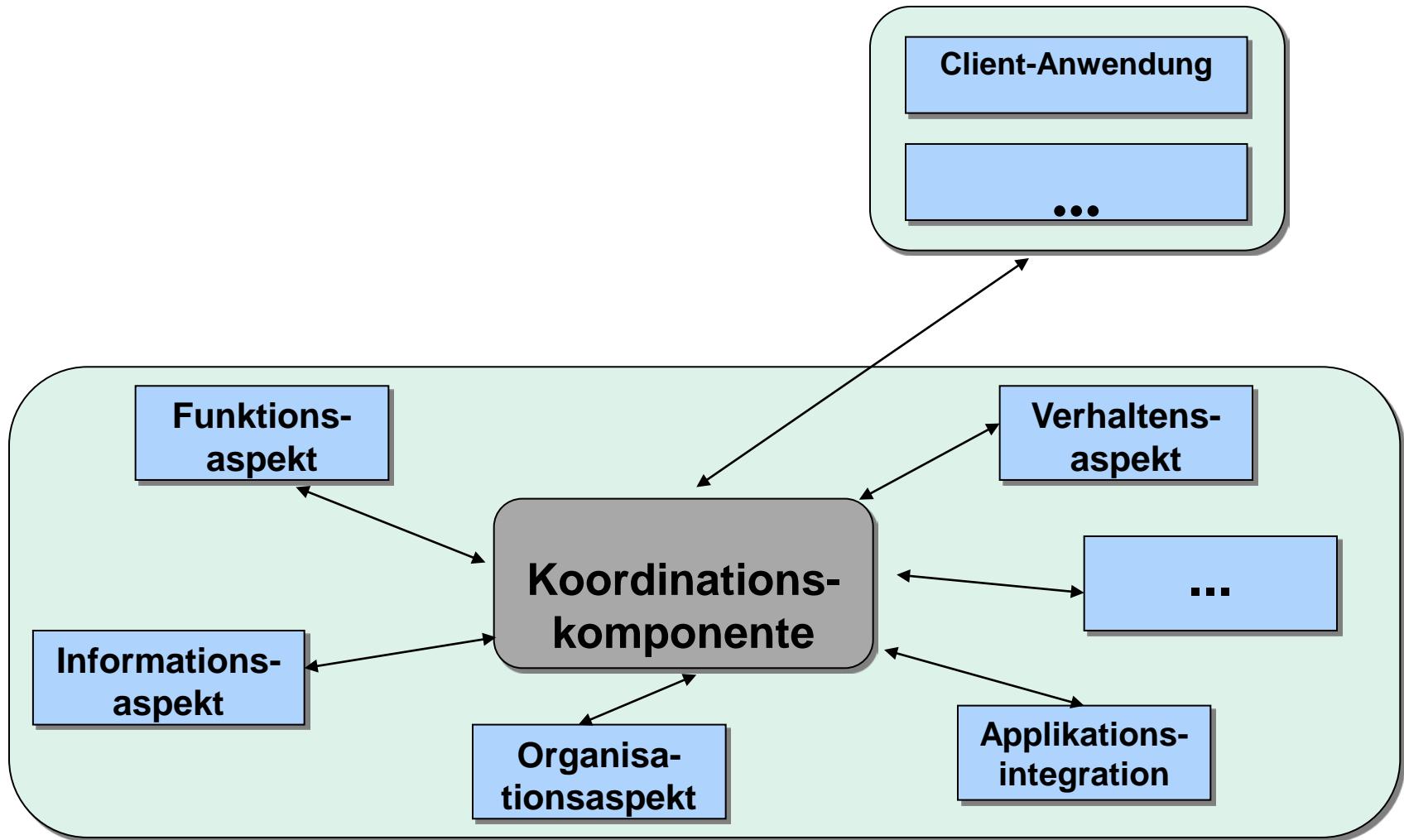
Werkzeug-System



Workflow-Management-Kernsystem



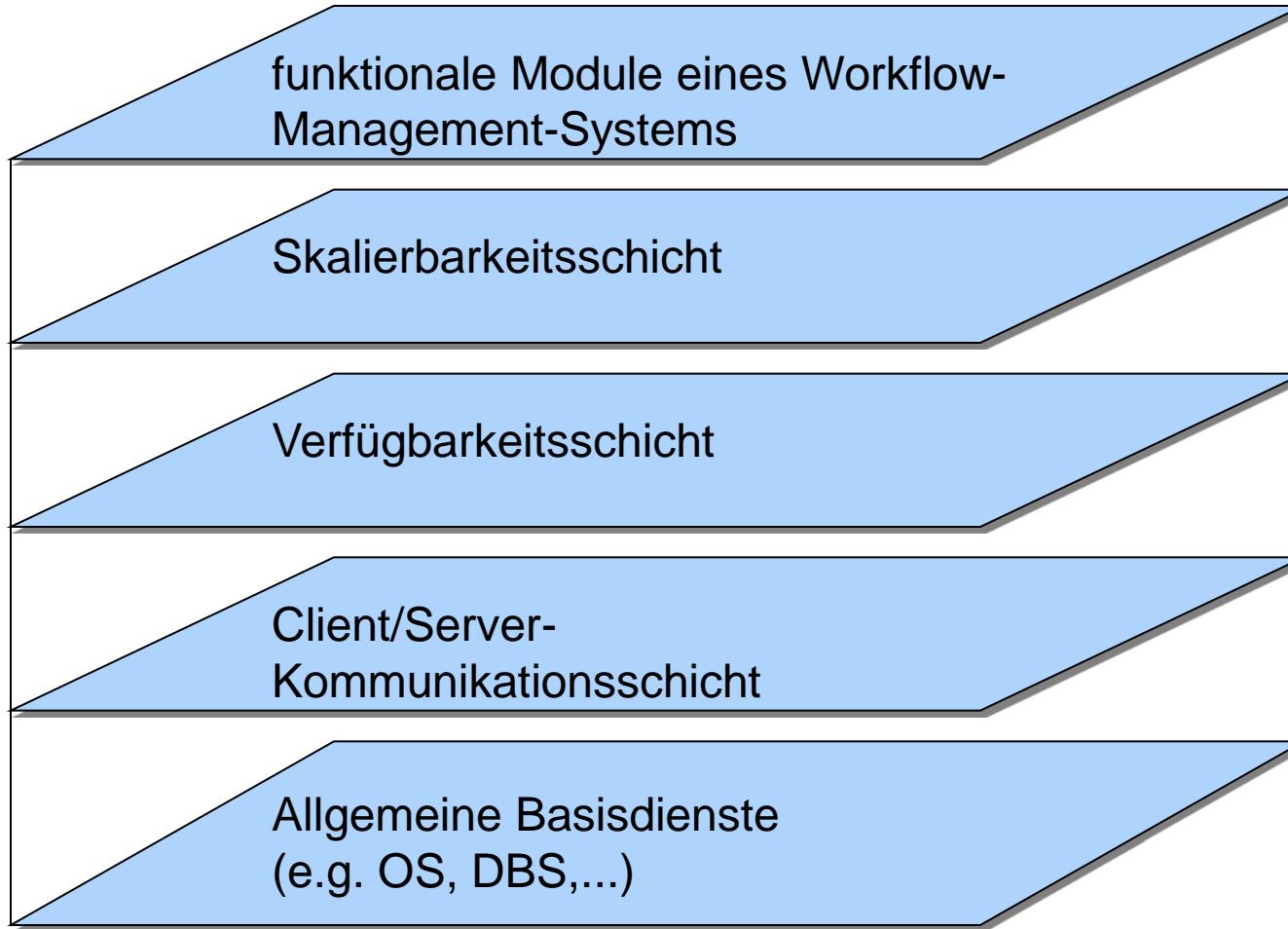
Komponentenarchitektur eines WfMSs



Nicht-funktionale Anforderungen

- ◆ Offenheit der Systemimplementierung auf verschiedenen Ebenen
 - Funktionale Erweiterbarkeit und Konfigurierbarkeit
 - Unterstützung von verteilten heterogenen Einsatzumgebungen
- ◆ Zuverlässigkeit des Systems
 - Verfügbarkeitsgarantien für Workflow-Instanzen
 - Transparenz von Systemfehlern
- ◆ Analysierbarkeit von Workflow-Management-Systemen
- ◆ Skalierbarkeit
- ◆ Berücksichtigung organisatorischer Gegebenheiten

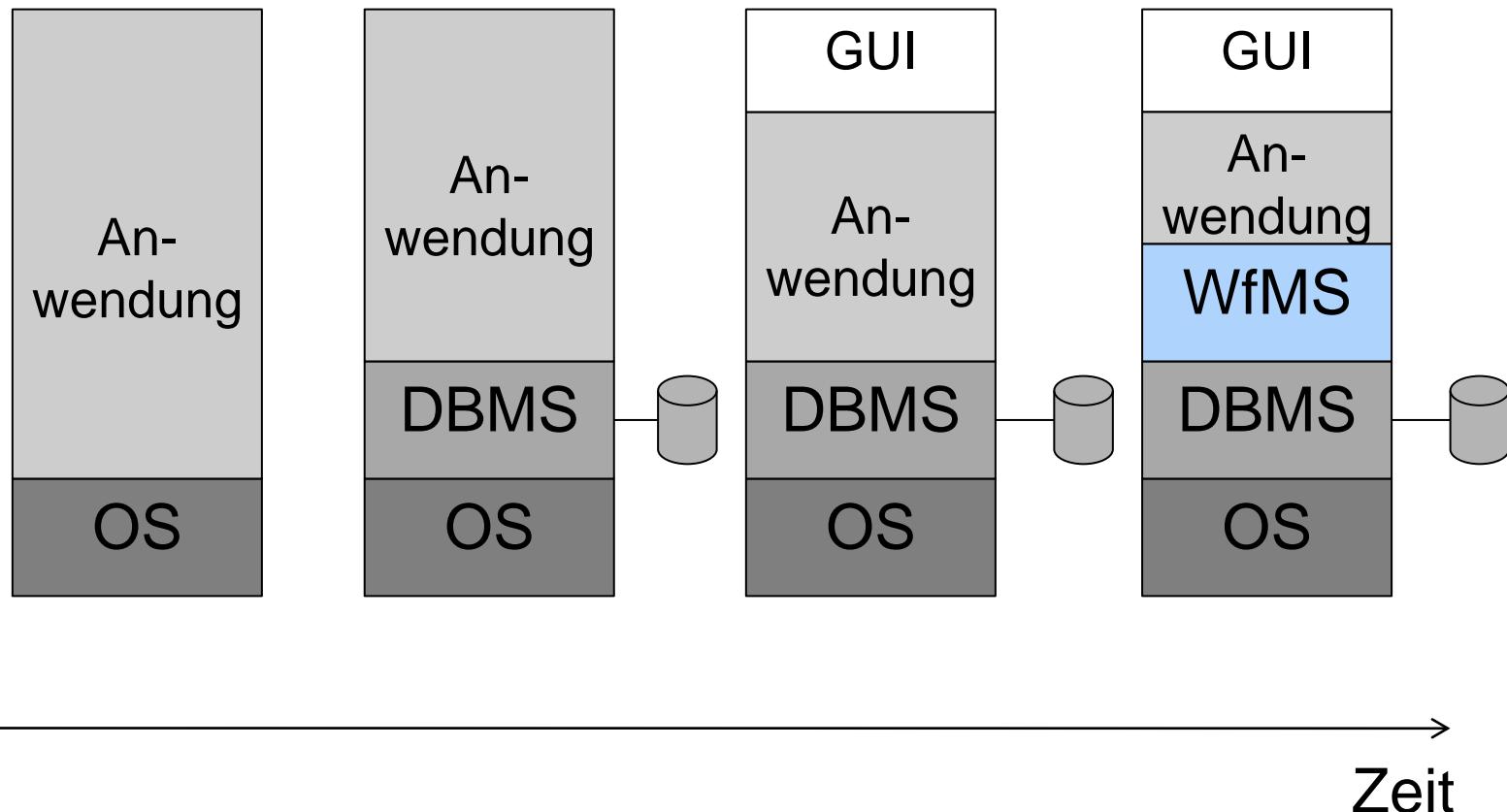
Schichtenarchitektur eines WfMSs



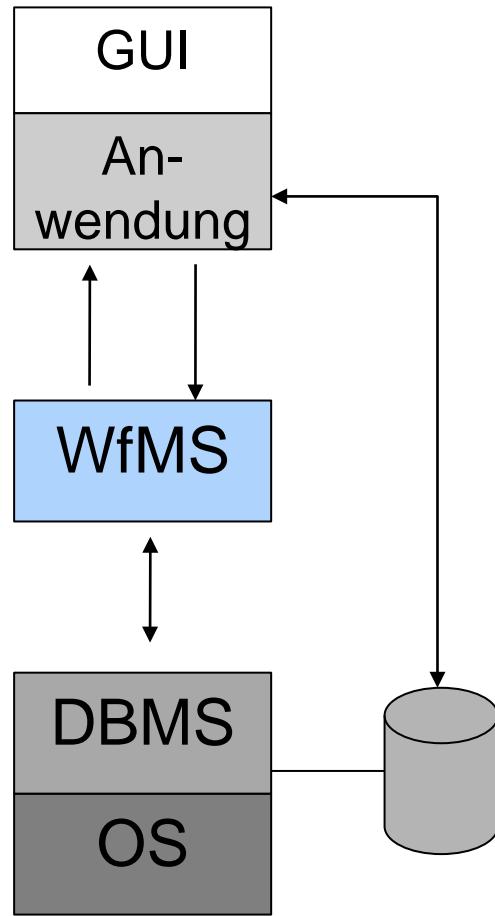
Einbindung von WfMS in Anwendungssysteme (1)

Evolution von Anwendungssystemen

Lebenszyklus
Überblick
Aspekte
Integration
Referenzmodell

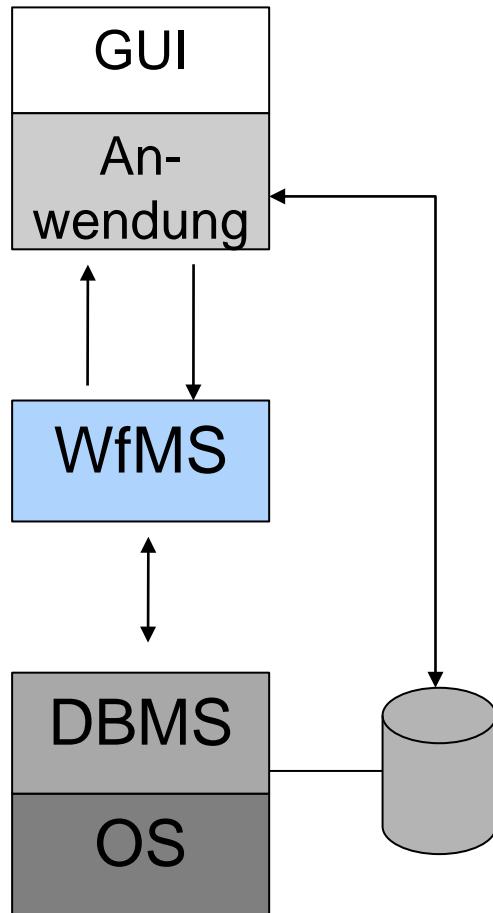


Einbindung von WfMS in Anwendungssysteme (2)



- ◆ Explizite Trennung von
 - Workflow-Management
 - Anwendung
- ◆ Workflow Management
 - Zuständig für „Logistik“ des Workflows
 - Zeitliche Steuerung
 - Aufruf von Anwendungen
 - Berücksichtigung von „Constraints“
 - Zuordnung von Endbenutzern (Anwender, Akteure)
- ◆ Anwendung
 - Ausführen von Arbeitsvorgängen
 - automatisiert – semi-automatisiert

Einbindung von WfMS in Anwendungssysteme (3)



Vorteile Trennung Workflow-Management und Anwendungslogik

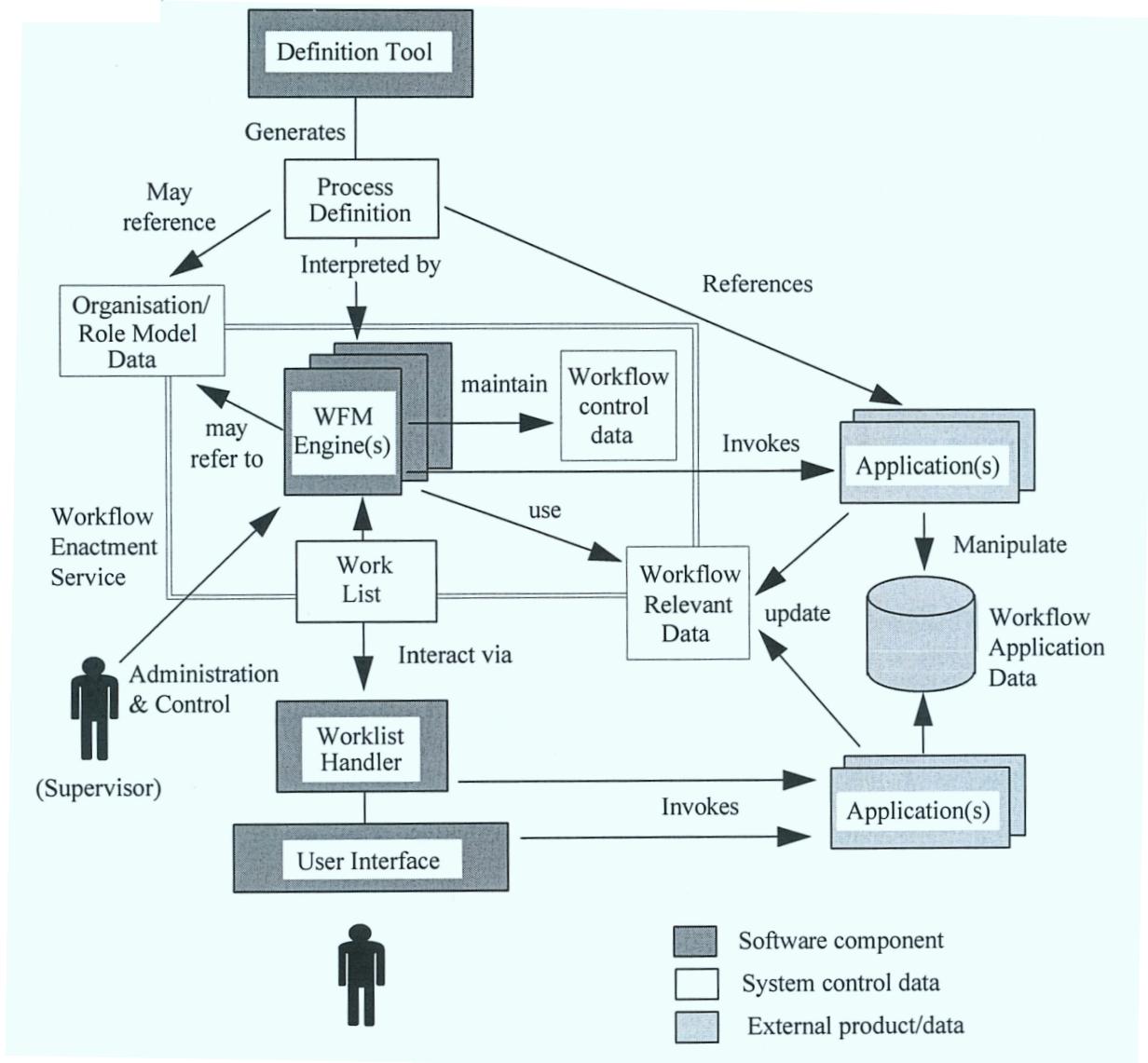
- ◆ Komplexitätsreduktion
- ◆ Wiederverwendung generischer Prozessfunktionalitäten
- ◆ Einfache Prozessanalyse
- ◆ Einfache, modulare Anwendungsentwicklung

Referenzmodell der WfMC - Überblick (1)

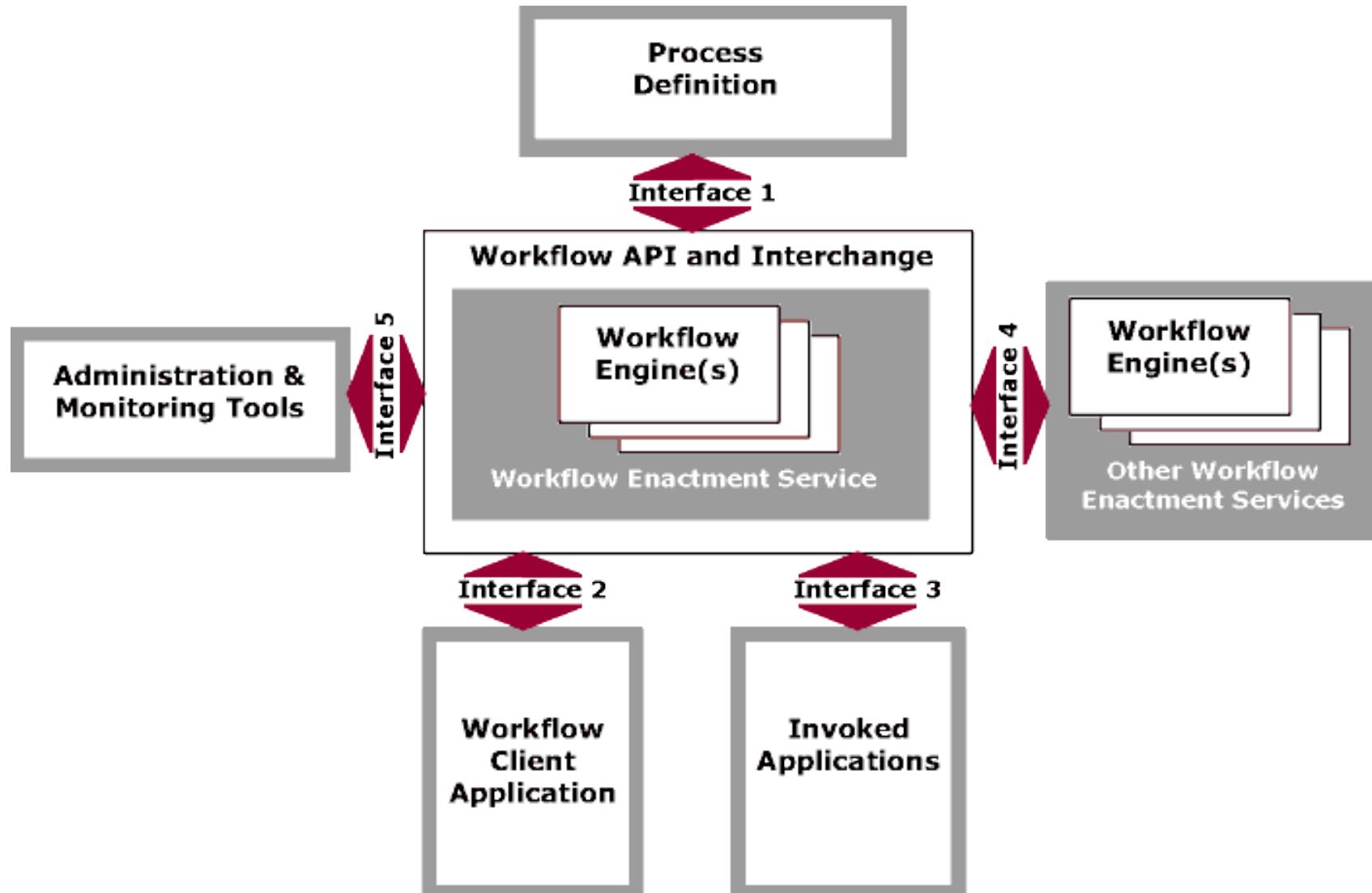
- ◆ WfMC (Workflow Management Coalition)
 - Non-Profit-Organisation zur Standardisierung
 - davon ca. 50 Prozent WfMS-Anbieter (Forschung ca. 25 Prozent)
- ◆ Referenzmodell von 1995
- ◆ Ziel:
 - Interoperabilität
 - Festlegung von standardisierten Komponenten und Schnittstellen zur Abdeckung der Basisfunktionalität eines WfMS

Generisches WF-Management System (WfMC)

Komponenten
und deren
Zusammenspiel
für
Basisfunktionalität



Referenzmodell der WfMC - Überblick (2)



Referenzmodell der WfMC – Komponenten (1)

- ◆ „Workflow Enactment Service“ (Ausführung)
 - Workflow Engine(s) als „Herz des Systems“
 - ggf. mehrere Engines (z.B. aus Gründen der Performanz), nicht relevant für Endbenutzer (Anwender)
 - Ziel der Steuerung des Prozessablaufes: „*right activities at right order to right people*“
 - Aufgaben:
 - Initialisieren (→ „Process Definition Tools“) und Beenden von Workflowinstanzen
 - Planung des Ablaufs („Routing“) von Instanzen
 - Zuordnung von Tasks zu Ressourcen (gemäß Ressourcenklassifikation → „Process Definition Tools“)
 - Starten von Anwendungen zur Ausführung von Aktivitäten (→ „Invoked Applications“)
 - Verwaltung von Prozess-Variablen und Triggern

Referenzmodell der WfMC – Komponenten (2)

◆ „Process Definition Tools“

- Prozessdefinitionen i.E.S.
 - Erstellung und Spezifikation von Prozessen: zeitliche (z.B. Sequenz, Parallelität) und konditionale (z.B. Verzweigungen) Abfolge von Komponenten (z.B. Aktivitäten), Constraints, etc.
 - Sprachen: z.B. Petri-Netze, BPMN (Business Process Model and Notation)
 - Spezifikation von Aktivitäten (z.B. Rollen zur Ausführung, aufzurufende Anwendungen sowie Übergabeparameter)
 - Überprüfung auf syntaktische Korrektheit und Konsistenz
- Ressourcenklassifikation
 - z.B. Rollen, organisatorische Einheiten
- Workflow-Analyse
 - Semantische Korrektheit, Simulation zur Analyse des Prozessverhaltens

Referenzmodell der WfMC – Komponenten (3)

◆ „Workflow Client Application“

- Präsentation der Aufgaben („*work items*“) an Endbenutzer (Anwender) in Form von sog. Arbeitsliste („*worklists*“)
- Informationen zu „*work items*“
- Initiieren und Beenden von Aufgaben
- Übergabe von Zustandsparametern (→ Workflow Engine)

◆ „Invoked Applications“

- Anwendungen i.E.S. (Start durch → Workflow Engine) sowie Anwendungs-Parameter (Austausch mit → Workflow Engine);
- Konfigurationsdateien
- Datenzugriff
- Automatisierte sowie interaktive Anwendungen

Referenzmodell der WfMC – Komponenten (4)

◆ „Other Workflow Enactment Service(s)“

- Workflow Interoperabilität: Verknüpfung von autonomen Workflow Systemen
- Sequentielle und hierarchische Ausführung von Teilprozessen (Sub-Workflows)
- Beispiele: bereichs- bzw. unternehmensübergreifende Kopplung

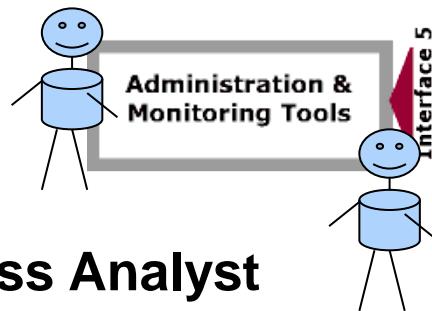
◆ „Administration & Monitoring Tools“

- Operationales Management
 - Verwaltung von Endbenutzern (Stammdaten, Verfügbarkeiten)
 - Zustandsinformationen einer Prozessinstanz
- Beobachtungs- und Report-Tools
 - Verarbeitung der vergangenheitsbezogenen „Rohdaten“ (→ Workflow Engine) zur Berechnung von Performanz-Indikatoren; Reports

Einbindung von Rollen am Beispiel des Referenzmodells der WfMC

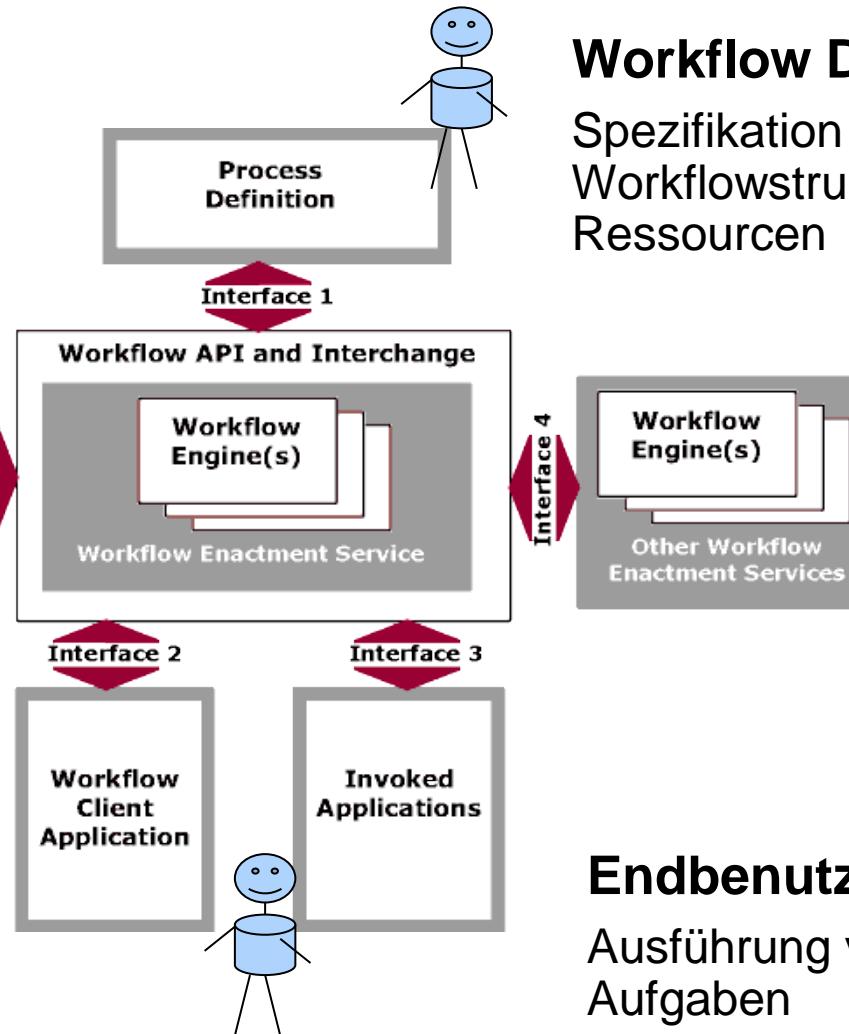
Administrator

Verwaltung von Ressourcen und Authorisierungen
Beobachtung (Monitoring)
Instandhaltung lauffähiges System



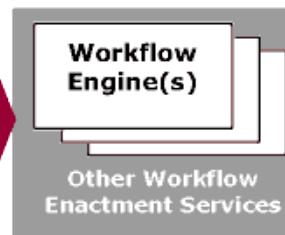
Prozess Analyst

Performanzanalysen



Workflow Designer

Spezifikation der Workflowstruktur & Ressourcen



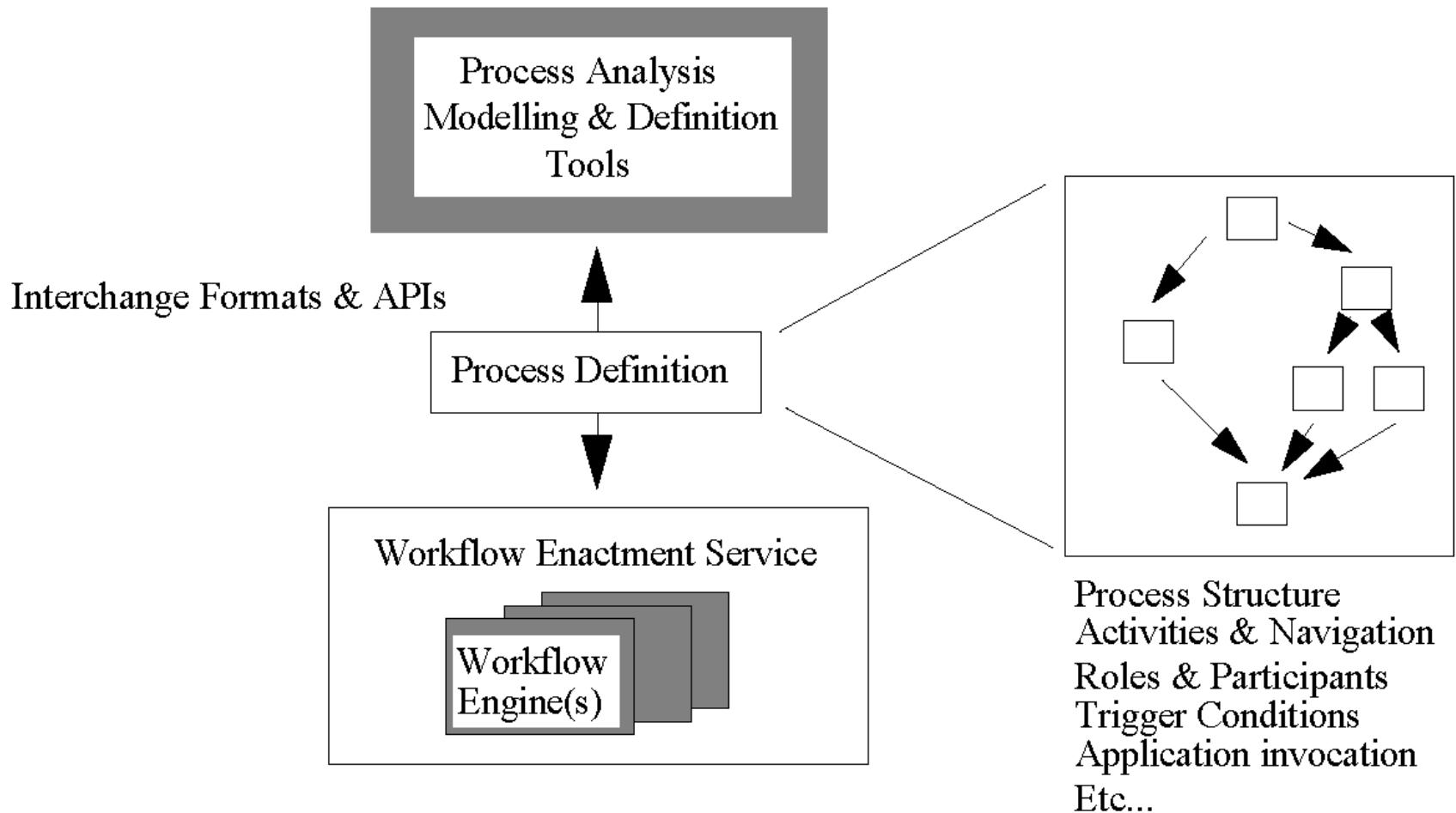
Endbenutzer

Ausführung von Aufgaben

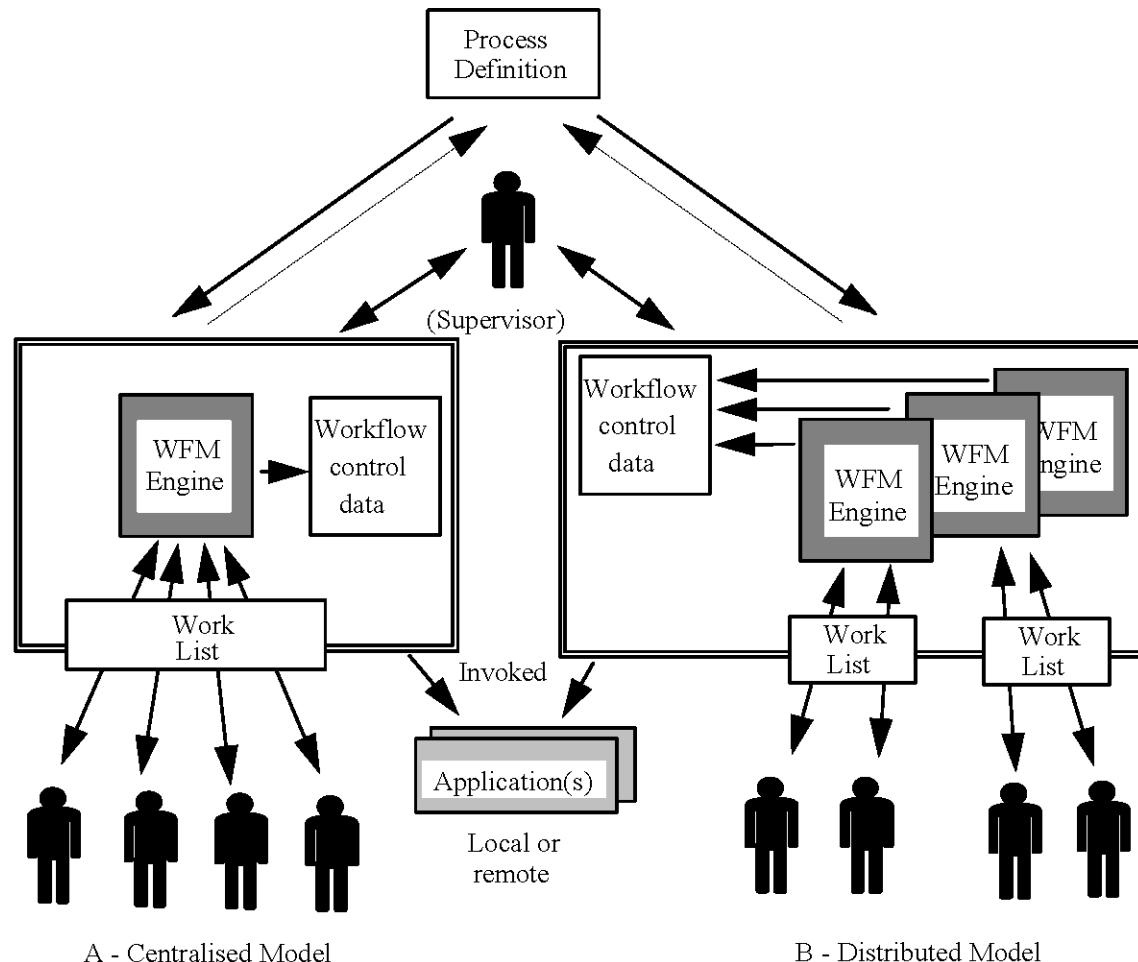
Referenzmodell der WfMC – Schnittstellen (1)

- ◆ Generisches Ziel Interoperabilität
 - Standards für API und Datenaustausch
 - Verknüpfung von Ausführungsmaschinen unterschiedlicher Hersteller
 - Unabhängigkeit der Anwendungen von konkreten WfMS
- ◆ Schnittstelle 1: Workflow Process Definition Read/Write-Interface
 - Verbindung von Workflow-Modellierungswerkzeugen mit der zentralen Komponente - Austausch von Workflow-Schemata
- ◆ Schnittstelle 2: Notifikationsdienst auf Clientseite (Workflow Client Application Programming Interface)
– Problem: ein Eingangskorb für mehrere Systeme

Schnittstelle 1 – Workflow Process Definition



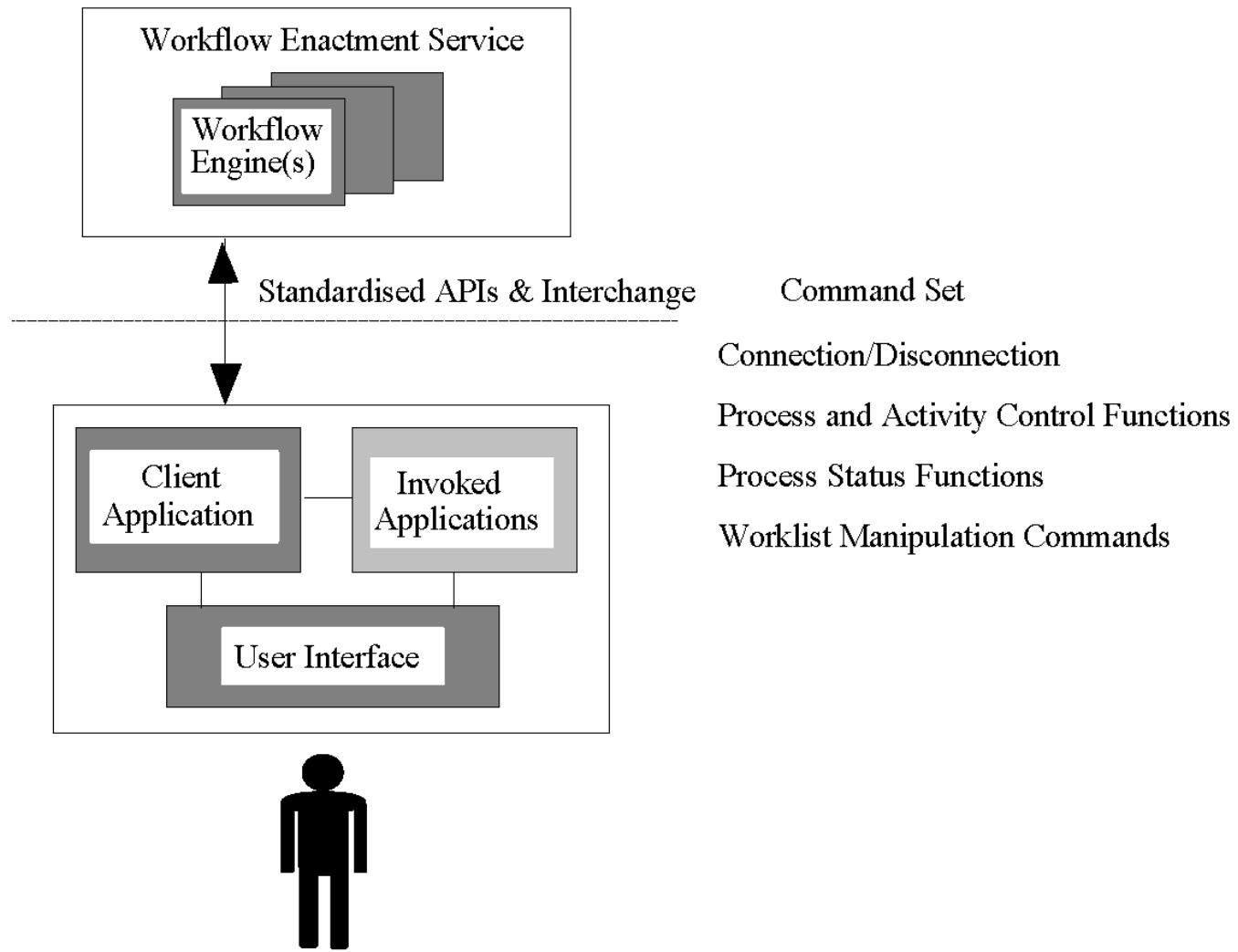
Worklists in WfMS



A - Centralised Model

B - Distributed Model

Schnittstelle 2 – Client API



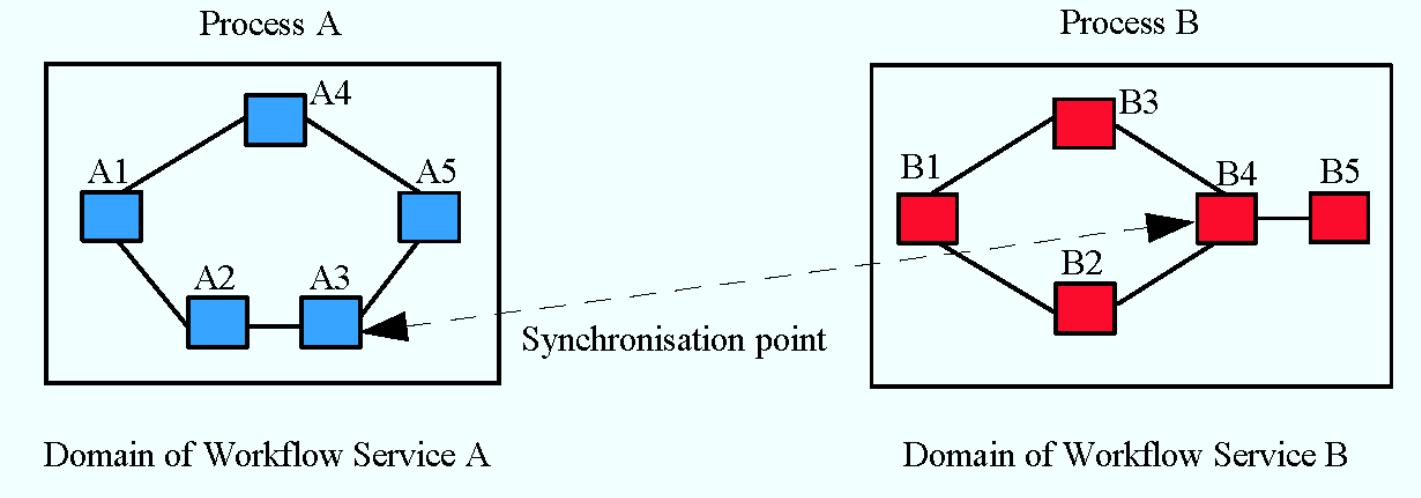
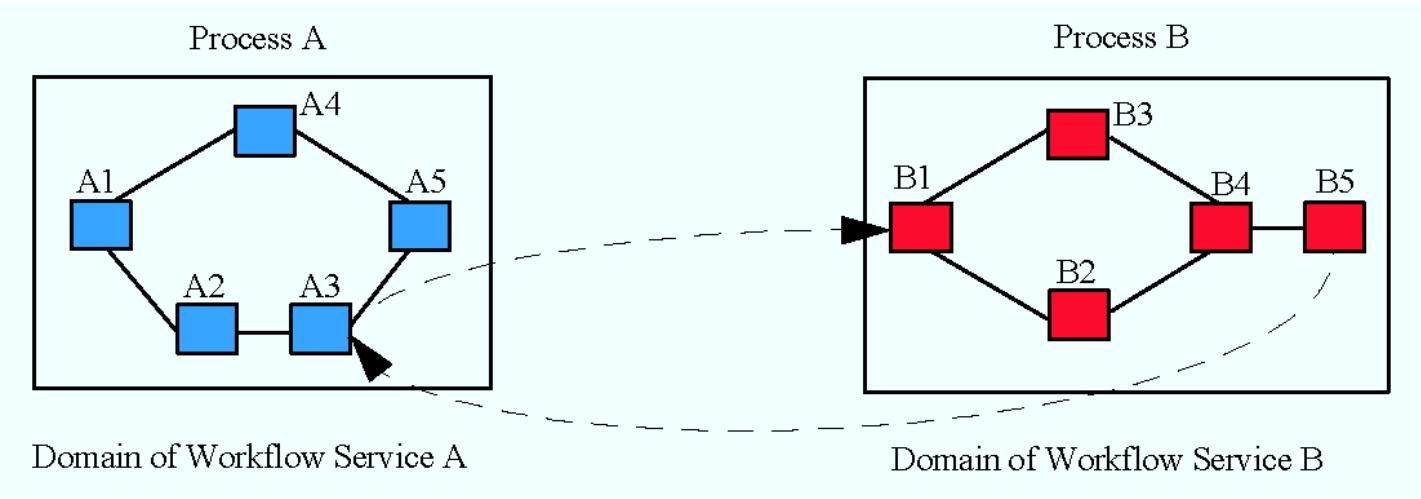
Referenzmodell der WfMC – Schnittstellen (2)

- ◆ Schnittstelle 3: Programmierschnittstelle für den Applikationsaufruf
- ◆ Schnittstelle 4: Austausch von workflow-relevanten Daten zwischen verschiedenen, autonomen Workflow-Engines. Ziel ist die Übergabe eines Subworkflows zur Ausführung an ein anderes WfMS
- ◆ Schnittstelle 5: Administrations- und Monitoring-Werkzeuge anderer Hersteller zur Information über den aktuellen Ausführungszustand von Workflows und zur Administration laufender Workflows

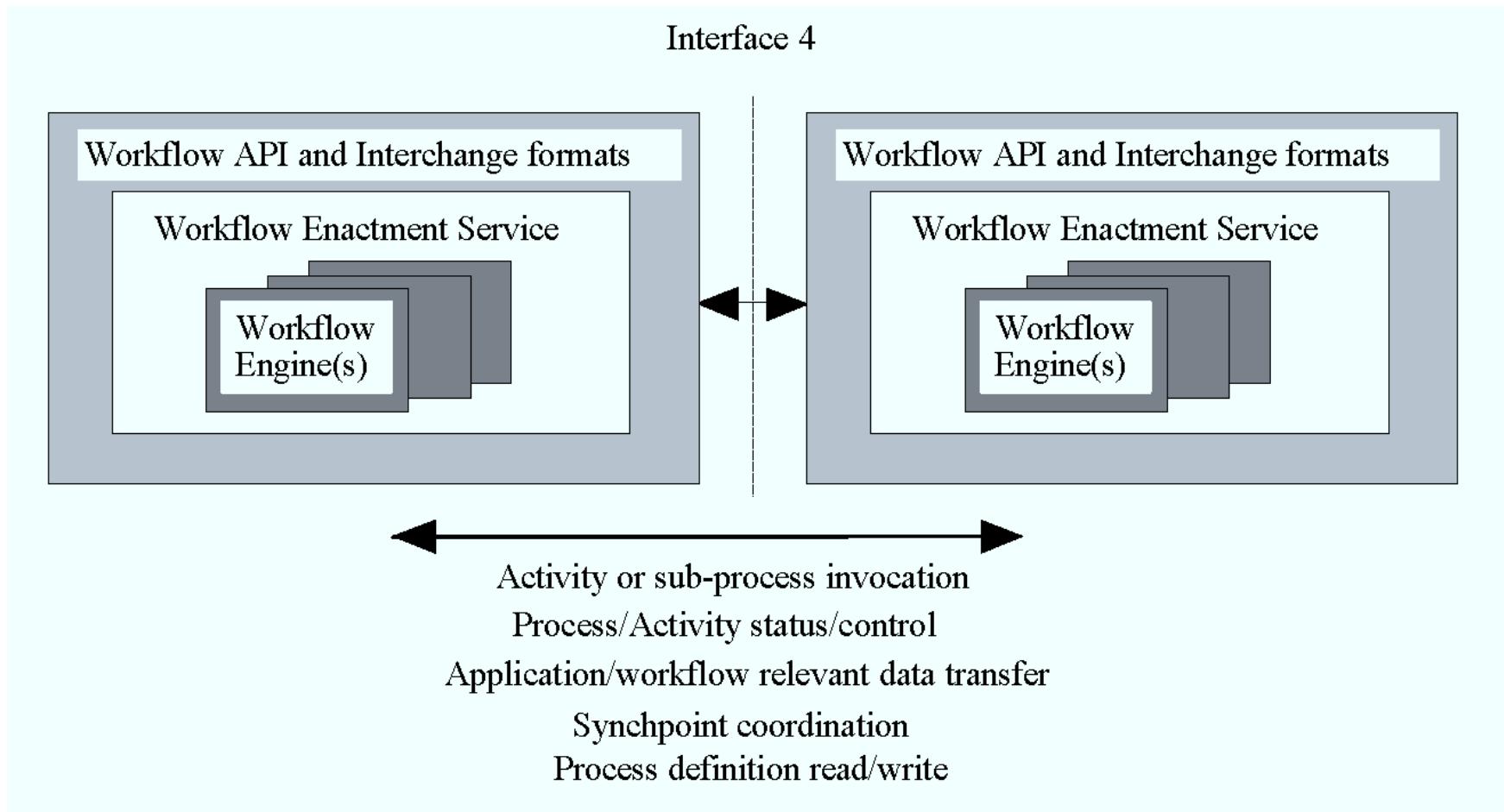
Schnittstellen sind nicht völlig unabhängig voneinander spezifizierbar

Vorgehensweise: minimale Menge an Funktionen bzw. Funktionalität standardisieren

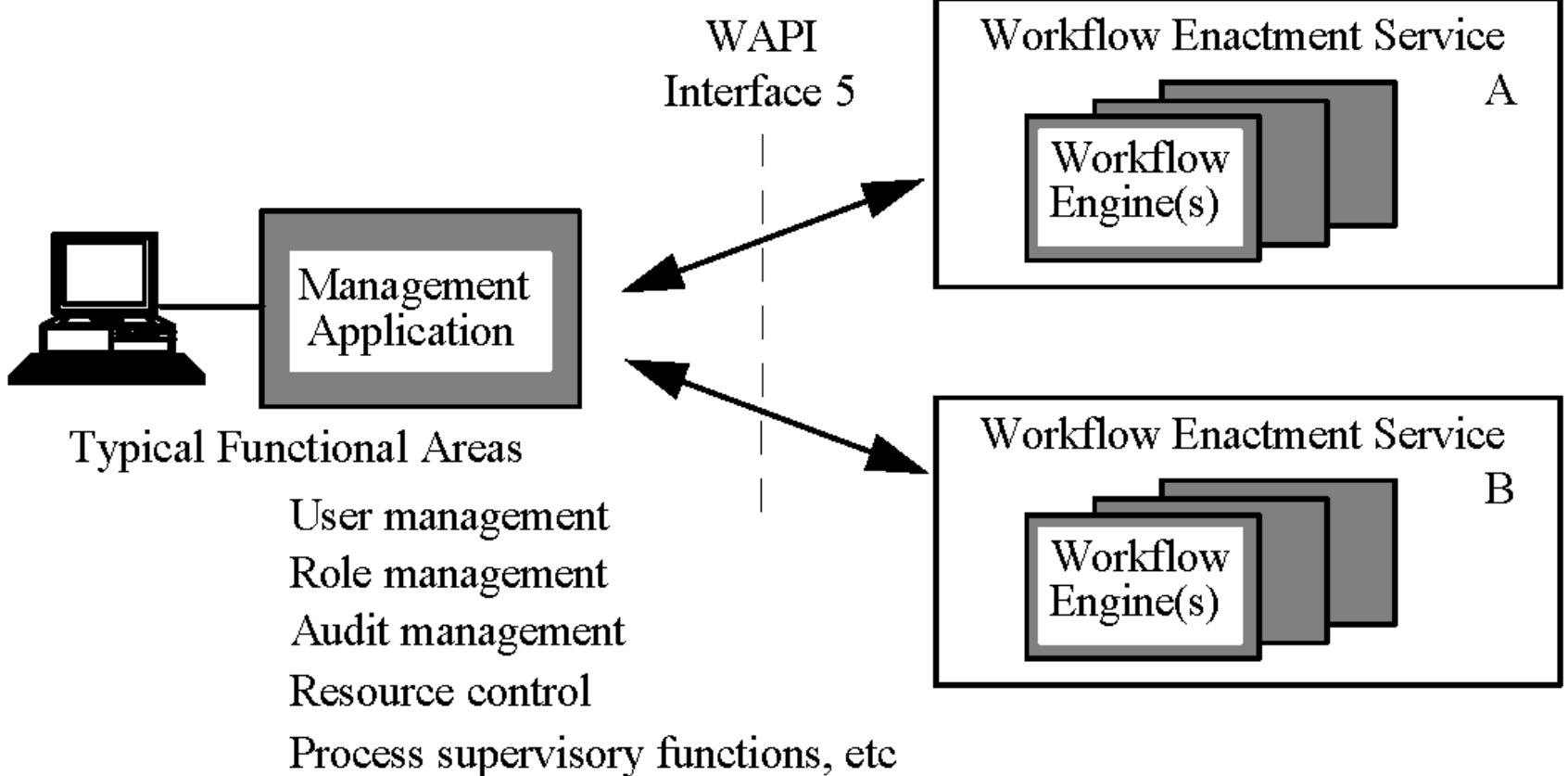
Koordination zwischen Workflow Engines (Bsp.)



Schnittstelle 4 – Workflow Engines



Schnittstelle 5 – Monitoring



Design-Primitive des Referenzmodells

- ◆ Abstraktion
 - Beschreibung der wesentlichen Komponenten und Schnittstellen, unabhängig von technischer Umsetzung
 - Spezifikation der Interfaces (abstrakte Terme und konkrete „Bindings“)
 - Konkrete Bindings konnten im Lauf der Zeit den Entwicklungen angepasst werden (API in C → Middleware → Web Services)
- ◆ Lebenszyklus eines Prozesses
 - Phasen des Lebenszykluses tragen zum Gesamtbild bei
- ◆ Integrierte Sicht von Daten, Prozess und Organisation

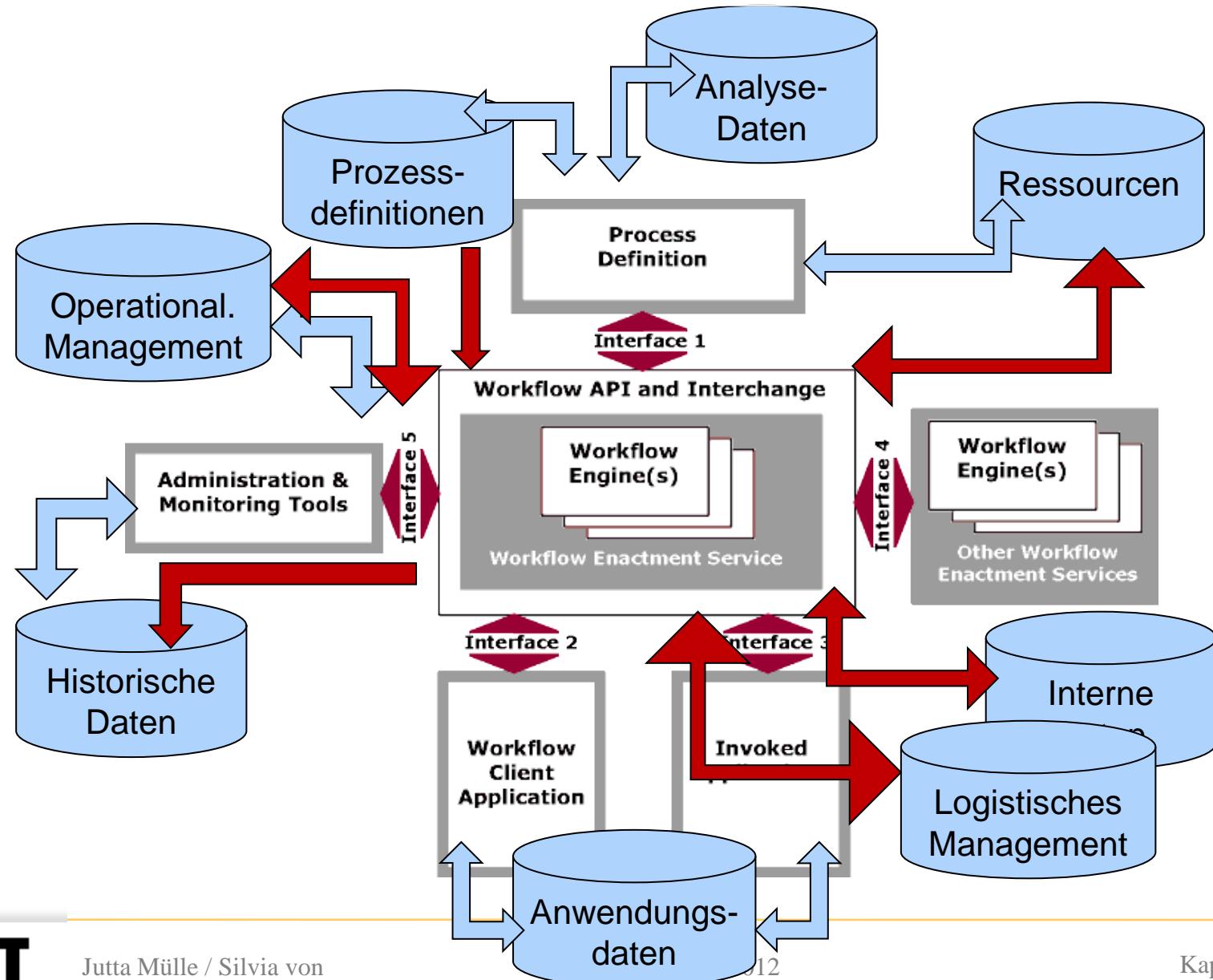
Einige Schwachstellen des Referenzmodells

- ◆ Datensicht etwas vernachlässigt
- ◆ Mangelnde Spezifikation der Schnittstellen „nach außen“
 - Schnittstelle 3 : Synchronisation von Anwendungen mit Arbeitslisten und Engine (keine ACID-Eigenschaften)
 - Schnittstelle 4: lose Kopplung (z.B. unternehmens-übergreifend) nicht adäquat unterstützt, keine Datenstruktur-Spezifikation
- ◆ Zu geringe Flexibilität
 - Adaptive Workflows → Anpassungen zur Laufzeit
 - Ad-hoc-Entscheidungen (Problem lokale Autonomie)
- ◆ Fokusierung auf ablauforientierte WF-Paradigmen
 - Eingeschränkter Nutzen für neue Paradigmen z.B. Daten-zentrierte Workflows

Klassifikation von Daten in einem WfMS (v.d. Aalst)

- ◆ Prozess-Definitionen (Schemata), inkl. Definitionen der Aktivitäten
- ◆ Ressourcen (Rollen, Organisationen)
- ◆ Analysedaten
- ◆ Operationales Management (technische Parameter)
- ◆ Historische Daten: Nachverfolgung einer Prozessinstanz
- ◆ Anwendungsdaten
 - Zugriff nur über Applikation, nicht über Engine
 - Masterdaten (generisch, z.B. Stammdaten) und Prozess-Instanzdaten (spezifisch)
- ◆ Interne Daten: technische Daten (z.B. Netzwerkadressen)
- ◆ Logistisches Management (z.B. Prozesszustände inkl. Attribute, Ressourcenzustände)

Daten am Beispiel des WfMC Referenzmodells



Exemplarische Fragen Kapitel 2

- ◆ Was ist der Unterschied zwischen einem Workflow-Schema und einer Workflow-Instanz?
- ◆ Welche Vorteile ergeben sich durch die Trennung Anwendung und WfMS in Anwendungssystemen?
- ◆ Beschreiben Sie die Komponenten des Referenzmodells der WfMC
- ◆ Welche Aspekte unterscheidet man bei WfMS?
- ◆ Welche nicht-funktionalen Anforderungen an die Architektur von WfMS sind zu beachten?