BTW 2003

26.-28. Februar 2003

The IOP Approach to Enterprise Frameworks

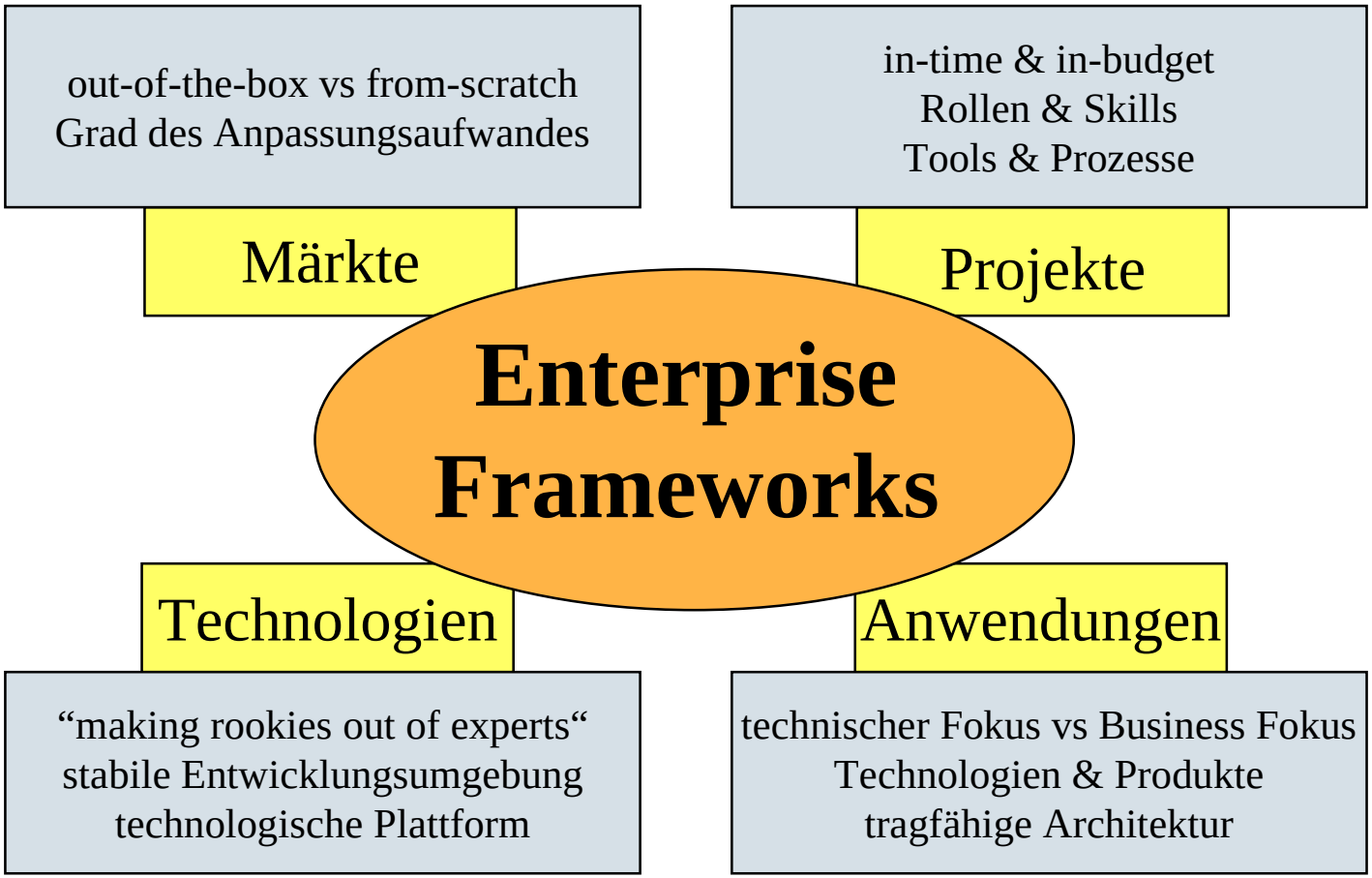
Stefan Schäfer, Dr. Udo Nink
(CronideSoft AG)
(udo.nink | stefan.schaefer)@cronidesoft.com

Cronide Soft AG

Inhalt

- Motivation
- Architektur
  - Entwurfsziele, Schichtenarchitektur, Systeme, Topologie, Architekturvergleich
- Bausteine
  - Objektmodell, Komponentenmodell, Design Repository, Resource Framework, Service Framework, Persistence Framework, Interaction Framework, Workflow Framework, Integration Framework, Code Generation Framework
- IOP-Components
- Anwendung - Entwicklungsprozess

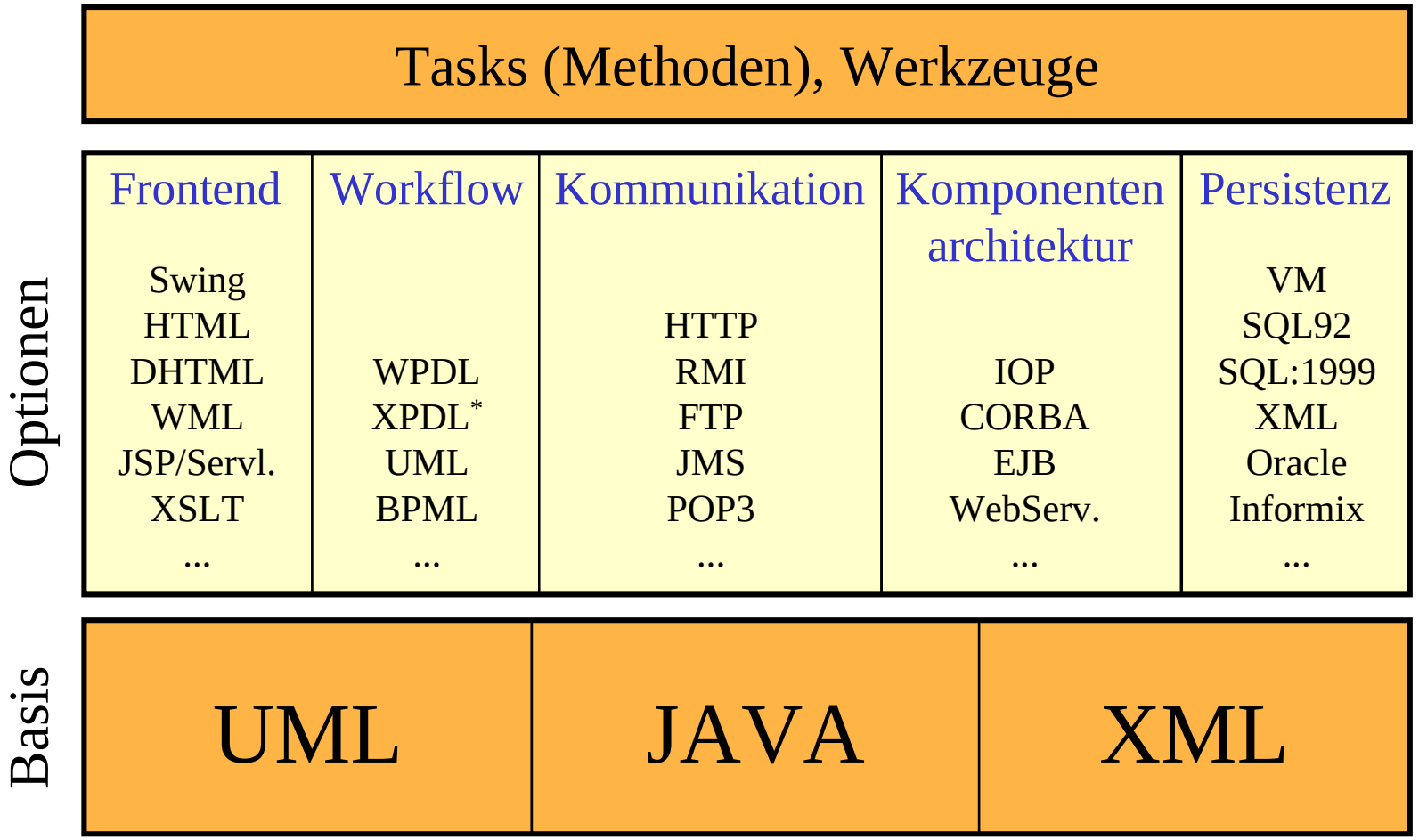
Motivation


out-of-the-box vs from-scratch
Grad des Anpassungsaufwandes
in-time & in-budget
Rollen & Skills
Tools & Prozesse
Märkte
Projekte
Enterprise Frameworks
Technologien
Anwendungen
“making rookies out of experts“
stabile Entwicklungsumgebung
technologische Plattform
technischer Fokus vs Business Fokus
Technologien & Produkte
tragfähige Architektur

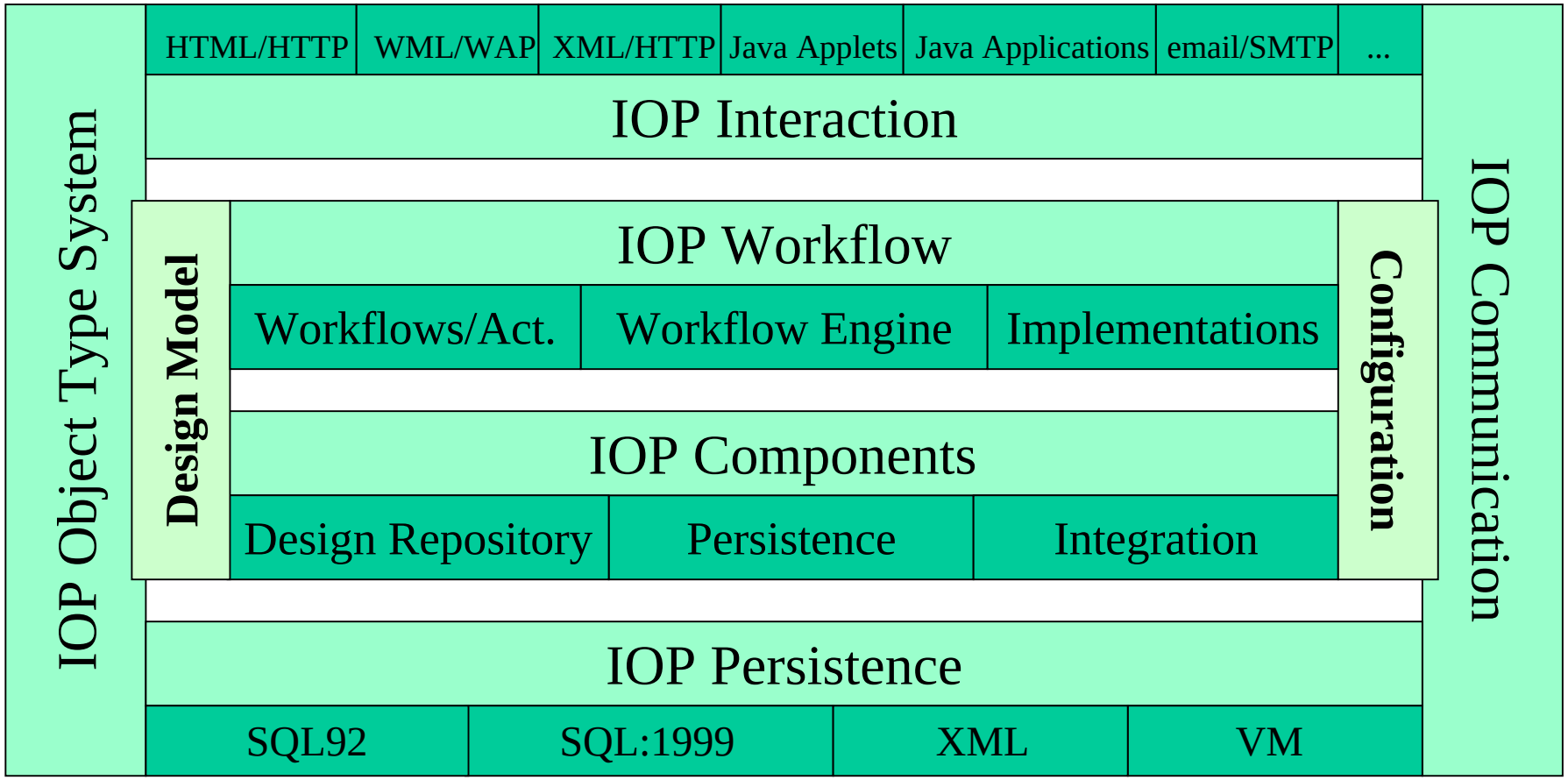
Motivation – Was ist IOP?

- IOP ist ein Enterprise Java Framework
- IOP kapselt und komplettiert Application Server
- IOP selbst ist kein Application Server
  - läuft standalone oder in Application Servern
- IOP ist eine Plattform für
  - Consulting + Entwicklung
  - Schulung + Coaching
- IOP abstrahiert grundsätzlich von Technologien + Produkten + Standards
- Einheitliches Objektmodell (von GUI bis Persistence)
- Unterteilung in Domänen- und Anwendungscode

Architektur – Entwurfsziele



Architektur – Schichtenarchitektur

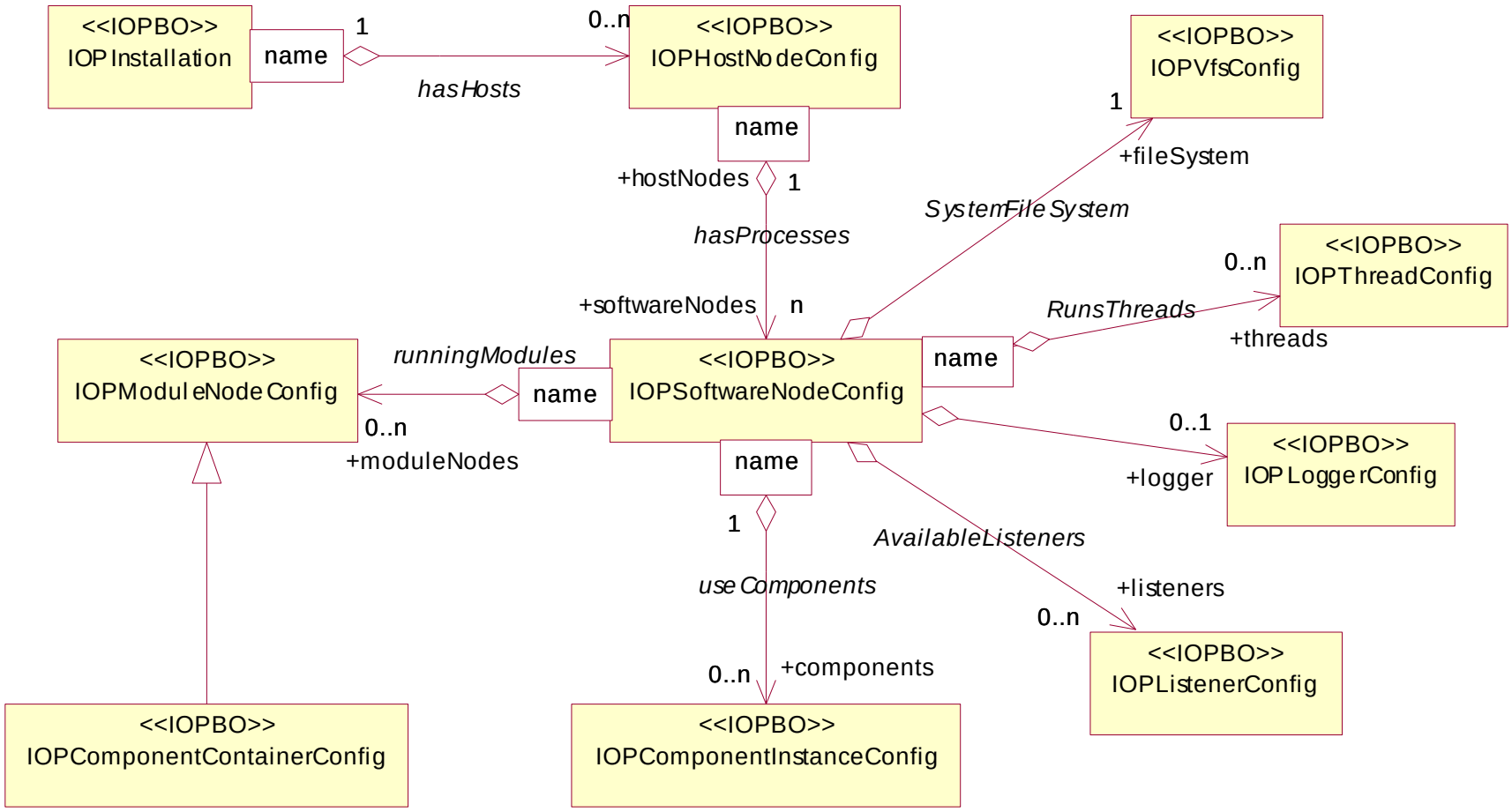


Cronide Soft AG

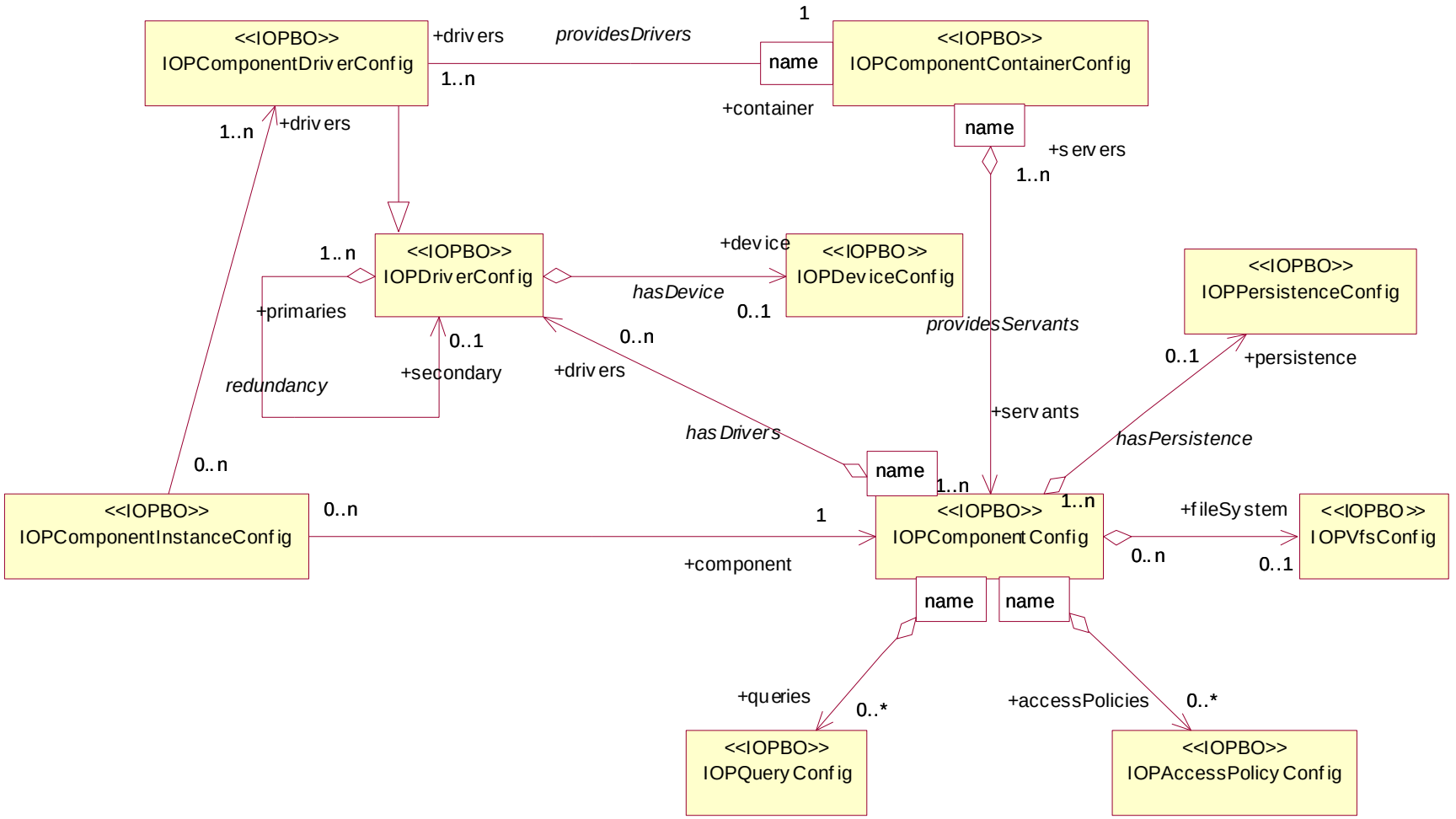
Architektur – Systeme

IOP Boot System	IOP Configuration System	IOP Session System	IOP Logger System	IOP File System	IOP Driver System	IOP Component System
Start	Installation Topologie Optionsauswahl Anfragen	Sessions User-Info Laufzeitstatus	Protokolle Nachrichten Stufen Kategorien	VFS Dateisysteme	APIs Protokolle Geräte Mapping	Verteilung Container Instanzen
IOP Kernel System	Verwaltung der Systeme, run Levels					
IOP Workflow System	**IOP Resource System**	**IOP Object Type System**	**IOP Transaction System**	**IOP Statistics System**	**IOP Dispatcher System**	**IOP Thread System**
Ablaufmaschine technischer Wf Interaction Wf Business Wf	dynamische IOP Ressourcen	Typsystem Objekttypen Metaobjekte Mappings	Tx-Klammern JTS JTA	statistische Laufzeitdaten Auswertung Analyse	Delegation von Requests an Systeme, Wf, Komponenten	konfigurierbare System- + Application Threads

Architektur – Topologie

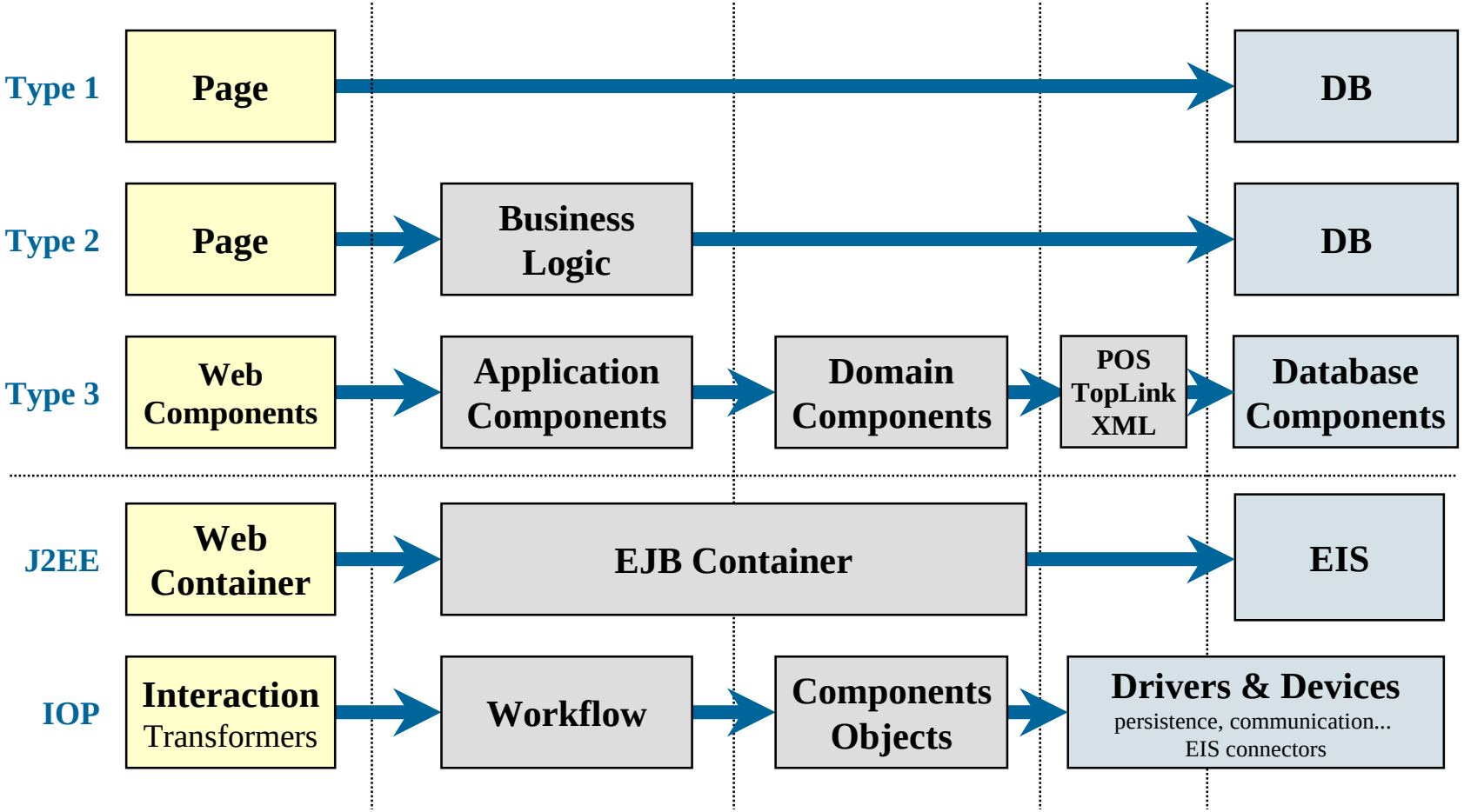


Architektur – Topologie (fortgesetzt)

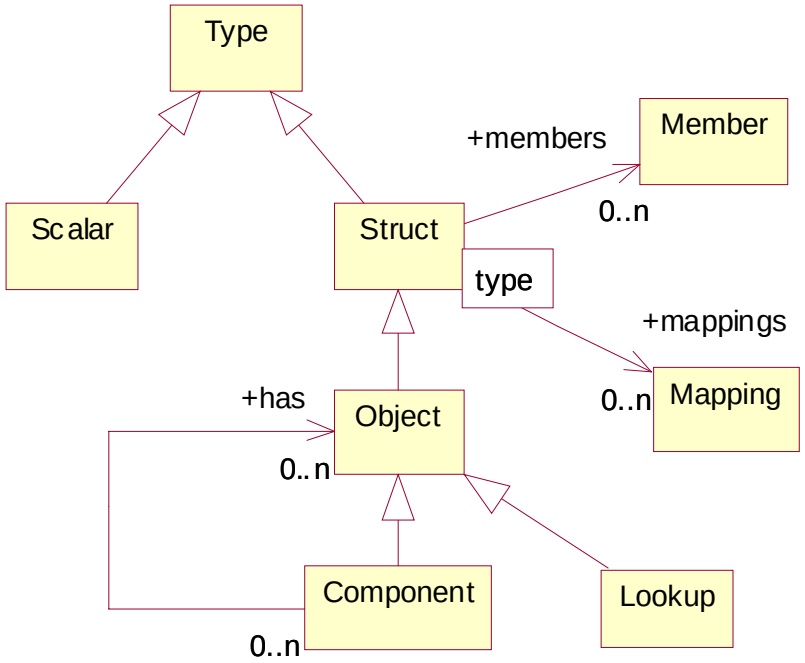


Cronide Soft AG

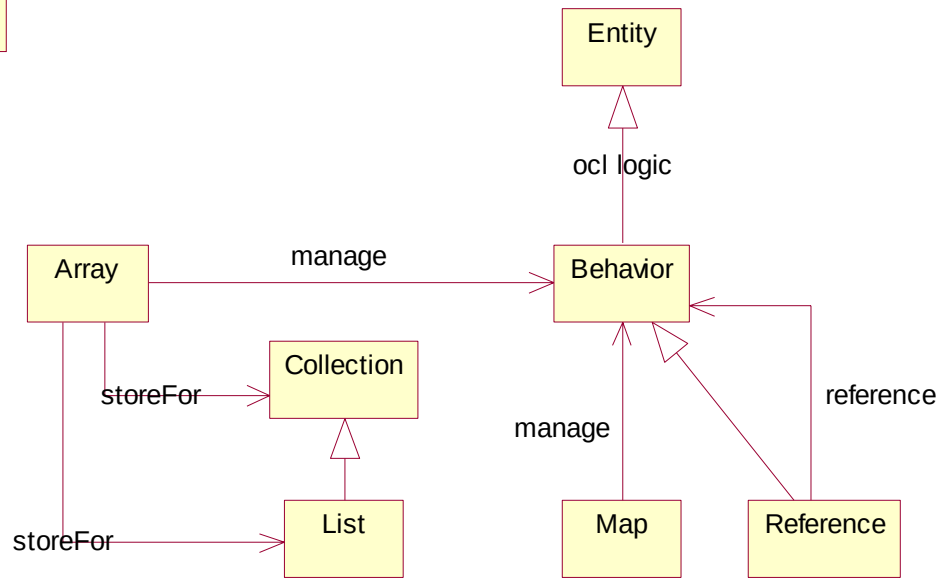
Architektur – Vergleich



Bausteine – Objektmodell



- Erweitertes Java-Typsystem
- höhere Ausdrucksstärke
- höhere Effizienz (Member)
- typsicher



- Häufige Objektmuster
- Graphschnittstelle
- Anfragen + Navigation

Cronide Soft AG

Bausteine – Komponentenmodell

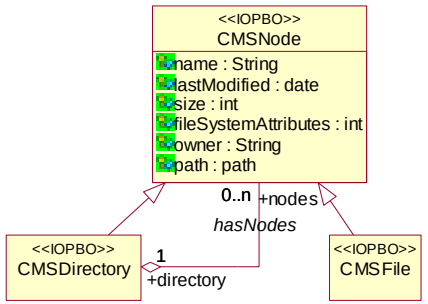
Komponenten modell	J2EE/EJB	IOP Components
Komponenten architekturen	EJB	IOP, CORBA, EJB
Resource Management	Threads, Connections, Komponenten	IOP Resourcen (Components, File Systems, Workflows, Threads, Treiber + Listener, ...)
Workflow	n.a.	IOP Workflow
Kommuni kation	RMI, RMI-IIOP, JMS	beliebige Treiber (HTTP, JMS, RMI, ...)
Programmier modell	beliebige Java-Klassen, kaum Vorstrukturierung	model-driven, globales Objektmodell + -Zugriff

Bausteine – Web Services vs IOP Components

- Abgrenzung Web Services
  + inter-Anwendungskommunikation
  + Publikation + Aufruf von Diensten
  + verteilte Kommunikation mittels spezifizierter Komponentenschnittstellen + XML-Nachrichten
  - Eignung für intra-Anwendungskommunikation?
  - weder Unterstützung für Implementierung von Diensten noch Assemblierung komplexer Dienste
- geplanter Einsatz von Web Services in IOP
  – WSDL für Workflows, Activities, Komponenten generieren
  – Treiber für die Einbindung von Web Services

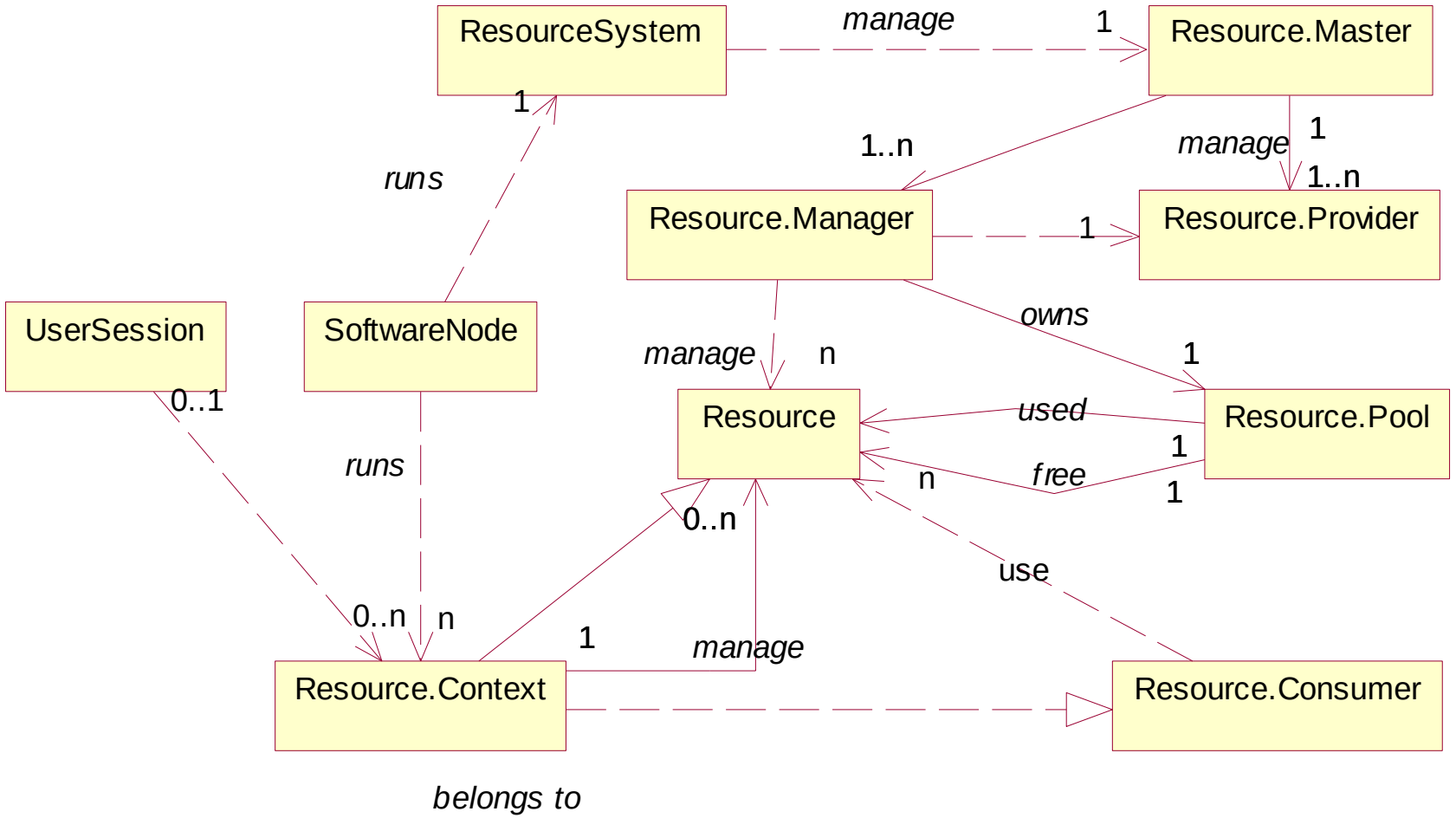
Bausteine – Design Repository

- hält alle statischen Design-Informationen
- reichert diese an um Mapping-Informationen für die Zielsprachen Java + SQL
- Derzeit werden das UML-Klassenmodell und das UML-Komponentenmodell unterstützt
- Dynamische Design-Informationen sind mittels WPDL unterstützt, für XPDL in Arbeit und für UML Aktivitätsmodelle geplant
- verwendet IOP Design Component



Cronide Soft AG

Bausteine – Resource Framework



Cronide Soft AG

Bausteine – Service Framework

- Sammlung von Mikro-Frameworks für partielle, meist technische Lösungen
  - filesystem, localization, logging, persistence, session, workflow, ...
- Device/Driver-Konzept als wichtigstes Entwurfsmuster
- Devices kapseln low-level APIs (Protokolle)
  - ejb, ftp, http, pop3, rmi, smtp, corba, ...
- Drivers nutzen Devices und kapseln high-level APIs
  - component, persistence, virtual file system, workflow, ...
- Mögliche Kombinationen können pro instanziiertem Dienst konfiguriert werden
  - z.B. VFS mit gekapseltem ftp- und http-Server + lokalem Dateisys.

Bausteine – Persistence Framework

- Persistente Speicherung von Java-Objekten
- Angelehnt an SUN JDO + objektrelationale Mappings
- Verfügbare Mappings
  - IOP Virtual Memory Persistence, IOP XML Persistence, IOP SQL92 Persistence, IOP SQL:1999 Persistence
- Mapping ist pro Komponente konfigurierbar
- Mappings werden aus UML + Konfiguration generiert
  - Datenbankskripte, Objektzugriff, Anfragetransformation
- unterstützte Konzepte
  - Objektversionierung, multi-Objekt Aktionen, Objektgraphen, automatische Erkennung von Änderungen, automatisches Delete bei Kompositionen, XML-Import/Export, relationale + komplexe Ergebnismengen, optimierter Speicherungsalgorithmus

Bausteine – Interaction Framework

- Definiert Zugriffspunkt auf IOP-Systeme von außen
- Basiert auf einer Service-Handler-Architektur
- Transformiert Anfragen an + Antworten von IOP-Dispatcher
- Erlaubt Kombination von
  - Kommunikationsprotokollen: HTTP, JMS, RMI, POP3, ...
  - und Formaten: HTML, WML, XML, Java Objects, Messages, ...
  - durch Konfiguration für verschiedene Server
    - Tomcat, JBoss, Oracle iAS, ...
- Erweiterbar durch anpassbare technische Workflows

Bausteine – Workflow Framework

- Basiert auf der Referenzarchitektur der WfMC + OMG
- Workflow-Spezifikation via WPDL, XPDL in Arbeit
- Konfigurierbar: verteilte Ausführung, Auditing, Statistiken
- Participants werden auf Organizational Model abgebildet
- Robustheit + Skalierbarkeit durch transaktionale Workflows und disconnected Session Management
- Unterstützung verschiedener Workflow-Ebenen
  - technischer Workflow, Interaction Workflow, Business Workflow
- Unterteilung in einfache, effiziente Core Engine und spezialisierter Workflow Engine

Bausteine – Integration Framework

- Konzepte, Implementierungen + Workflows für die Integration von Fremdsystemen
- Zugang über Treiber oder Integrationskomponenten
- gemeinsame Code-Basis mit IOP Interaction
  - Formate, Kommunikationsprotokolle, Services, Transformationen
- Wiederverwendung von IOP Interaction
  - Registrierung eines Listeners an einem Integration Hub
- Integrationskomponenten können als Persistence Manager fungieren
  - Integrationsdaten als Geschäftsobjekte

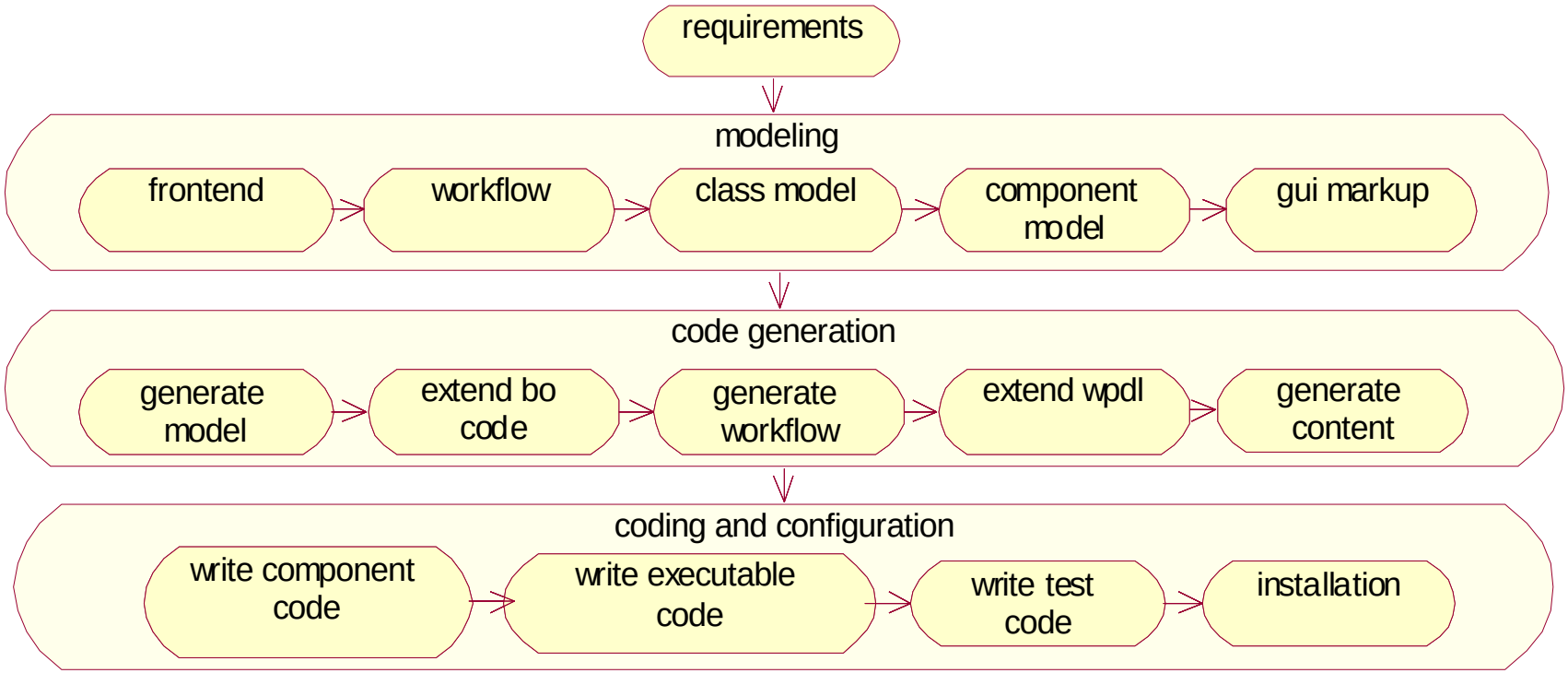
Bausteine – Code Generation Framework

- Satz von Compilern ausgehend von Design-Modellen
- Sowohl für Anwendungsentwicklung als auch zur Framework-Entwicklung eingesetzt
  - dadurch kontinuierliches Testen der Konzepte an IOP
- Alle Design-Informationen werden im IOP Object Type System und IOP Design Repository abgelegt
- Design bleibt auf Geschäftsobjektebene, generiert werden:
  - Java-Klassen mit inneren Schnittstellen zu Entity, Behavior, Collections, Maps und Referenzierung
  - SQL-Skripte für Typen, Tabellen, Indizes, Table Spaces
  - DTD/XMLSchema zur XML-Darstellung von Objektgraphen
  - IOP Workflow Format für Ablaufstrukturen von Workflows

IOP-Components

IOP Design Component Modellelemente UML 1.3 + Mapping	**IOP ID Component** OID-Vergabe	**IOP Workflow Definition Component** Prozessdefinitionen WPDL workflows, act. ...	**IOP Workflow Instance Component** Instanziierung Konfigurierung laufzeitoptimiert	**IOP Localization Component**
IOP Component System			Verwaltung der Komponenten	
IOP Content Management Component multi-provider Sites bulk Loading auch für GUI	**IOP Content Run-Time Component** laufzeitoptimiert logisch/physisch multi Channel	**IOP Actor Component** Akteure Participants Accounts	**IOP Access Component** Access Control Lists Accessor/Accessible Access Levels	**IOP Organization Component** Organizational Units Rollen Gültigkeiten

Anwendung – Entwicklungsprozess



Cronide Soft AG