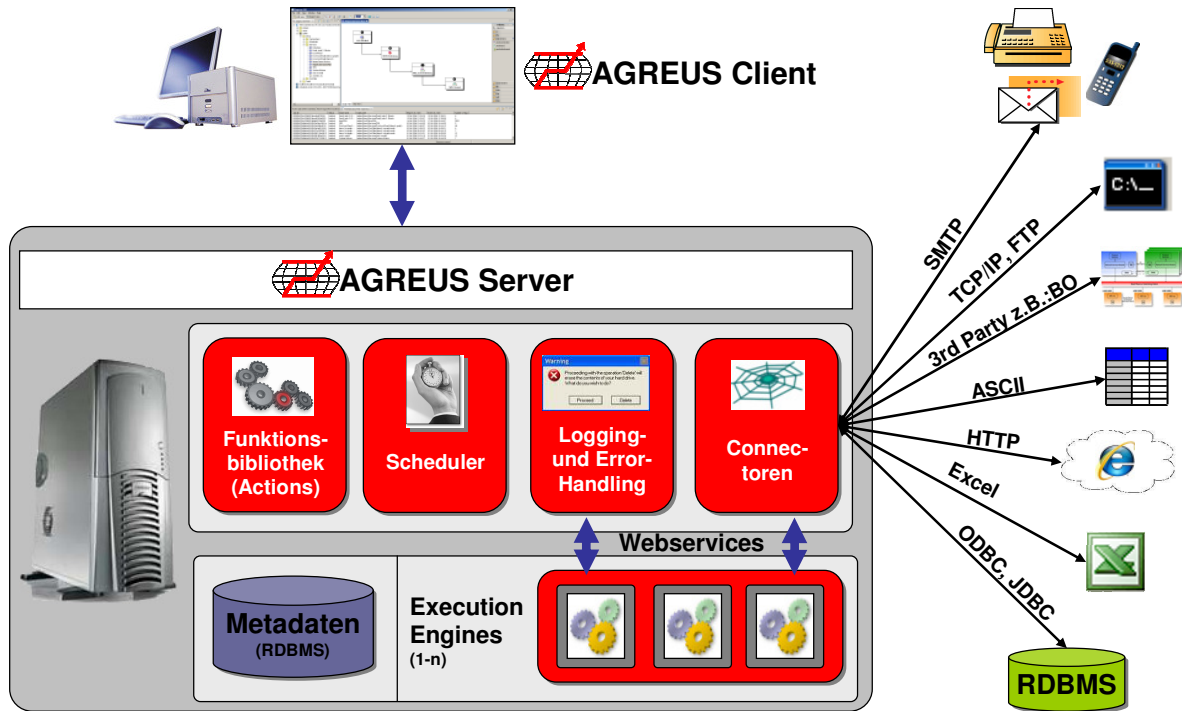


Technische Übersicht

ZAGREUS ist eine zentrale Plattform zur Automatisierung von Unternehmensabläufen. Mit Hilfe einer grafischen Oberfläche werden komplexe Abläufe ohne Programmierung entwickelt und in Produktion genommen. Der Ablaufprozess wird mit einem Ablaufplan (Schedule) versehen und wird ab dieser Zeit periodisch von ZAGREUS als Job durchgeführt.



Technische Voraussetzungen für eine ZAGREUS Installation

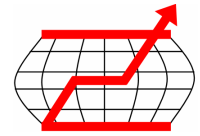
ZAGREUS ist eine reine **Client-Server Lösung**, die aufgrund der geringen Systemanforderungen grundsätzlich in jedem Unternehmen installiert werden kann.

Systemvoraussetzungen ZAGREUS-Server:

Betriebssysteme:	Windows:	Win2000, Win2003, WinXP (Prof.)
	LINUX:	keine Einschränkungen bekannt
	UNIX:	keine Einschränkungen bekannt
Speicherplatz Platte:		mindestens 200 MB
RAM:		mindestens 32 MB pro Execution Engine
Prozessoren:		Pentium II (oder höher) oder vergleichbare Prozessoren
Prozessoren:		Pentium II (oder höher) oder vergleichbare Prozessoren

Systemvoraussetzungen ZAGREUS-Client:

Betriebssysteme:	Windows:	Alle Versionen der letzten 10 Jahre
	LINUX:	keine Einschränkungen bekannt
	UNIX:	keine Einschränkungen bekannt
Speicherplatz Platte:		mindestens 30 MB
RAM:		mindestens 512 MB
Prozessoren:		Pentium II (oder höher) oder vergleichbare Prozessoren



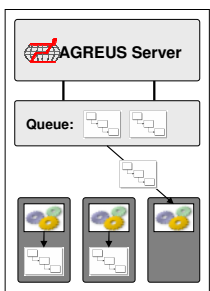
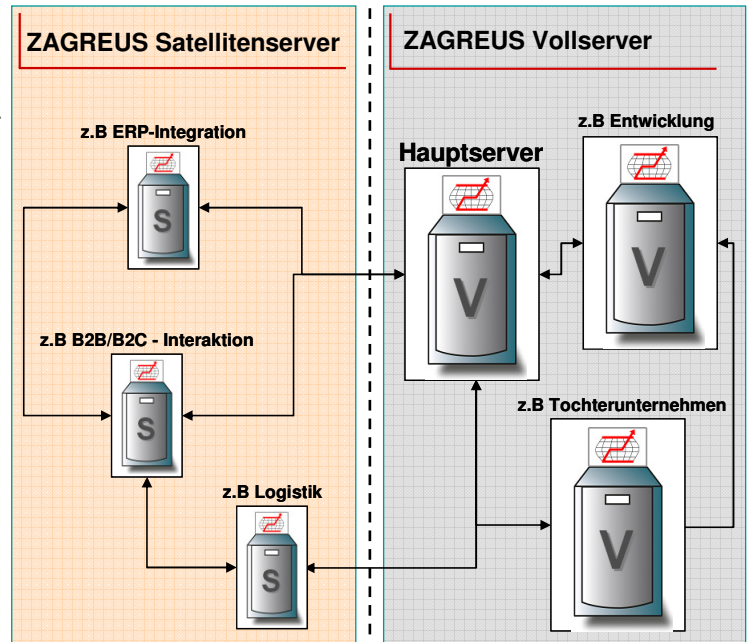
Die ZAGREUS Systemkomponenten

ZAGREUS besteht aus **vier Hauptkomponenten**. Neben ZAGREUS Server und Client sind noch **Execution Engines**, eine **zentrale Metadatenbank** und der **ZAGREUS Client** wichtige Faktoren für die größtmögliche Flexibilität, Performance, Sicherheit und einfache Bedienung.

ZAGREUS Server

Der ZAGREUS Server wird als Service (Windows) oder Prozess (LINUX, UNIX) installiert. Der Installationsprozess benötigt nur ein paar Minuten. Die Aufgaben des ZAGREUS Server sind:

- **Ausführen** von Jobs (zeit- und eventgesteuert)
- **Queueing**: Verwaltung der Jobs
- **Logging** der Serveraktivitäten
- **Error-Handling**: Reaktion im Fehlerfall
- Verteilung von **Aufgaben** an und **Kommunikation** mit ZAGREUS Komponenten im Netzwerk
- **SOA**: Webservice-Provider und Interaktion mit Webservices auf Basis einer verteilten ZAGREUS Systemarchitektur
- **Monitoring** aller vergangenen, aktiven und zukünftigen Jobs



ZAGREUS Execution Engines

Der ZAGREUS Server übergibt einen neuen Job zur Abarbeitung an eine freie Execution Engine. Es können beliebig viele Execution Engines pro Server im Einsatz sein, über die die **Parallelisierung** von Prozessen erfolgt.

ZAGREUS Metadaten

Alle für die Arbeit mit ZAGREUS notwendigen Daten, werden in einer **zentralen Metadatenbank** gespeichert. Das ermöglicht die einfache Entwicklung von Prozessen (z.B. Wiederverwendbarkeit von Objekten), das reibungslose Zusammenspiel aller Komponenten und eine hohe Flexibilität der gesamten Lösung.

ZAGREUS Client

Für die einfache Entwicklung komplexer Abläufe steht der ZAGREUS Client zur Verfügung. Seine Aufgaben sind:

- **Intuitives grafisches Interface** für die Erstellung von automatischen Services oder Unternehmensabläufe
- **Verwaltung der ZAGREUS-Objekte** über einen einfach zu bedienenden Browser in der gesamten ZAGREUS Systemarchitektur
- „**Schedulen**“ der Abläufe

The screenshot shows the ZAGREUS Client interface. The main window displays a graphical workflow for processing vehicles. The workflow includes steps like **5.1 z:in** (Hole Fahrzeugklassen), **5.2 z:log** (Fahrzeug: Itzag PROD), **5.3 z:in** (Einzelfahrzeug), **5.3.1 z:in** (z:parse), **5.3.2** (Schreibe ASCII), and **5.3.3** (Ergebnissätze). Below the workflow, there are buttons for **db:commit** and **db:close**. The interface also features a **Palette** on the right with various actions like **Auswahl**, **common attributes**, **z**, **file**, **mail**, **mail:attachment**, **mail:close**, **mail:connection**, **mail:getinfo**, **mail:getmsg**, and **mail:send**. At the bottom, there is a table for **Aktive Jobs [localhost]** and **Execution Engines [localhost]**.

Id	Port	Status	verfügbar	Job ID	Gestartet	TimeOut	Params
1	5001	Frei	Ja				->ms32M ->ms512M, (derby, mstr8.1.0,db, fop, bo)
2	5002	Frei	Ja				->ms32M ->ms512M, (derby, mstr8.1.0,db, fop, bo)