



Johann Wolfgang Goethe Universität
Frankfurt am Main

Fachbereich Biologie und Informatik (15)
Lehrstuhl für Graphische Datenverarbeitung



Abschlussvortrag zur Diplomarbeit von Frank Bergmann

Konfiguration, Simulation und Visualisierung
von einfachen, dreidimensionalen
Reaktionsdiffusionssystemen



Überblick

- ◆ **Einleitung**
 - Motivation
 - Ziel der Arbeit
- ◆ Grundlagen
- ◆ Konzept
- ◆ Realisierung
- ◆ Zusammenfassung
- ◆ Ausblick
- ◆ Demonstration

Einleitung

- ◆ In biologischen Systemen diffundieren und interagieren Moleküle innerhalb eines abgeschlossenen Reaktionsvolumens
- ◆ Aus solchen Systemen sind im Laufe der Zeit so komplexe Strukturen wie lebende Zellen entstanden
- ◆ Motivation:
 - Simulation / Visualisierung von Reaktionsdiffusionssystemen verspricht Erkenntnisse bei der Untersuchung von räumlichen Effekten in metabolischen Prozessen
 - Damit kann ein besseres Verstehen und Verfolgen der Entstehung von dreidimensionalen Mustern erreicht werden

Ziel dieser Arbeit

- ◆ Ziel der Arbeit ist die Spezifikation und prototypische Implementierung:
 - Konfigurationstools:
 - ◆ zur Definition der Verteilung unterschiedlicher Molekülkonzentrationen, Membranen und Kanäle im Reaktionsvolumen
 - 3D – Simulationsalgorithmus (PDE Solver):
 - ◆ zur Simulation einfacher dreidimensionaler Reaktionsdiffusionssysteme
 - 3D Visualisierungskomponente:
 - ◆ Zur Darstellung der Simulationsergebnisse
 - ◆ Mit Untersuchung ob eine 3D-Visualisierung während der Laufzeit der Simulation durchgeführt werden kann.

Überblick

- ◆ Einleitung
- ◆ **Grundlagen**
 - Reaktionsdiffusionssysteme
 - Volumenvisualisierung
 - Systems Biology Workbench (SBW)
- ◆ Konzept
- ◆ Realisierung
- ◆ Zusammenfassung
- ◆ Ausblick
- ◆ Demonstration

Reaktion-Diffusion

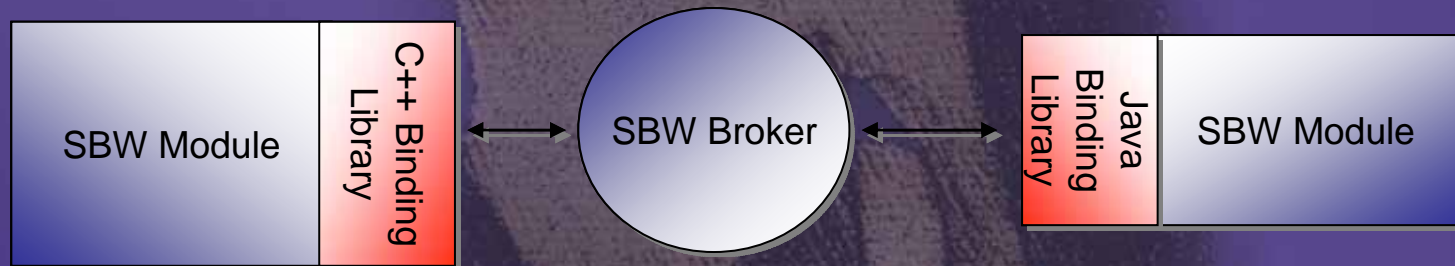
- ◆ In Ansammlungen von Elementen bewegen sich die einzelnen Elemente in zufälliger Weise (Brownsche Bewegung).
- ◆ Durch diese zufälligen Bewegungen breiten sich die Elemente aus.
- ◆ Resultiert diese Bewegung der einzelnen Elemente in einer gerichteten Bewegung der Gruppe spricht man von *Diffusion*.
- ◆ Falls diese einzelnen Elemente miteinander interagieren spricht man nicht mehr von *Diffusion*, sondern *Reaktion-Diffusion*.

Volumenvisualisierung

- ◆ Die Visualisierung von Daten, welche die dreidimensionale Struktur der Daten erhält, nennt man Volumenvisualisierung
- ◆ Üblicherweise werden Algorithmen zur Volumenvisualisierung in drei Kategorien unterteilt
 - Direct Volume Rendering (DVR)
 - Interactive Methods
 - Surface-Fitting Algorithms

Systems Biology Workbench (SBW)

- ◆ Software Framework das Plattform- und Sprachübergreifende Kommunikation zwischen Anwendungen ermöglicht



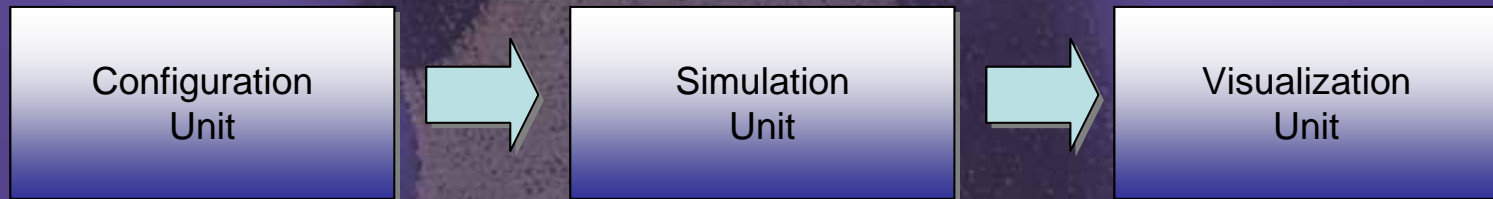
- ◆ Durch Binding-Libraries ermöglicht SBW den einfachen Zugriff auf Anwendungen unterschiedlichster Art. Im Moment stehen Simulations-, Modellierungs- und Optimierungsmodule zur Verfügung.

Überblick

- ◆ Einleitung
- ◆ Grundlagen
- ◆ **Konzept**
 - Konfigurierung
 - Simulation
 - Visualisierung
- ◆ Realisierung
- ◆ Zusammenfassung
- ◆ Ausblick
- ◆ Demonstration

Konzept

- ◆ Dreiteilung des Projekts in Konfiguration, Simulation und Visualisierung



- ◆ Aufgaben der Konfiguration
 - Festlegen des Reaktionsvolumens
 - Auswählen des Simulationsmodus
 - Einfügen der Elemente
 - Abspeichern der Konfiguration in einem Standardformat

Konzept – Konfiguration

- ◆ Elemente
 - Reaktionsvolumen
 - Molekülkonzentrationen der Species X und Species Y
 - Membranen
 - Kanäle
- ◆ Dateiformate
 - cellML
 - ◆ Standardisiertes Dateiformat zum Austausch von zellulären und subzellulären Prozessen
 - ◆ Hauptaugenmerk liegt auf der mathematischen Beschreibung der Modelle
 - ◆ Modell definiert als Netzwerk von wieder verwendbaren Komponenten (bestehend aus Variablen und Gleichungen)
 - SBML
 - ◆ Standardisiertes Dateiformat zum Austausch von biologischen Modellen
 - ◆ Natives Dateiformat für SBW Module
 - ◆ Fokus auf Pragmatismus



Konzept – Simulation

- ◆ Aufgaben der Simulation
 - Einlesen der Konfiguration
 - Initialisieren des Reaktionsvolumens
 - Initialisieren des ReactionPlugins
 - Solange nicht abgebrochen wird
 - ◆ Einen Schritt Simulieren
 - ◆ Abspeichern / Weitergeben der Simulationsdaten
- ◆ Plugin System zur Bereitstellung von unterschiedlichen Simulationsmodi
 - Reine Diffusion
 - Brusselator Reaktionsdiffusionssystem
 - SBW/Jarnac

Konzept – Simulation – Plugins

1 / 3

◆ Reine Diffusion

- Nur der Diffusionsteil der Reaktions-Diffusions-Gleichung wird ausgewertet
- Dies geschieht durch „Nullsetzen“ der Reaktionsgleichungen in der allgemeinen Gleichung:

$$\frac{\partial x}{\partial t} = F_x(x, y) + D_x \nabla^2 x$$

$$\frac{\partial y}{\partial t} = F_y(x, y) + D_y \nabla^2 y$$



$$\frac{\partial x}{\partial t} = rD_x \nabla^2 x$$

$$\frac{\partial y}{\partial t} = rD_y \nabla^2 y$$

Konzept – Simulation – Plugins

2 / 3

◆ Brusselator Reaktionsdiffusionssystem

- Reaktionsdiffusionssystem (nach Nicolis et. al 1977) bestehend aus zwei Chemikalien welche in vier Reaktionen miteinander interagieren:



- Daraus ergeben sich die folgenden Gleichungen für die Konzentrationsveränderungen:

$$\text{rate of change } X = A - (B+1)X + X^2Y \text{ and}$$

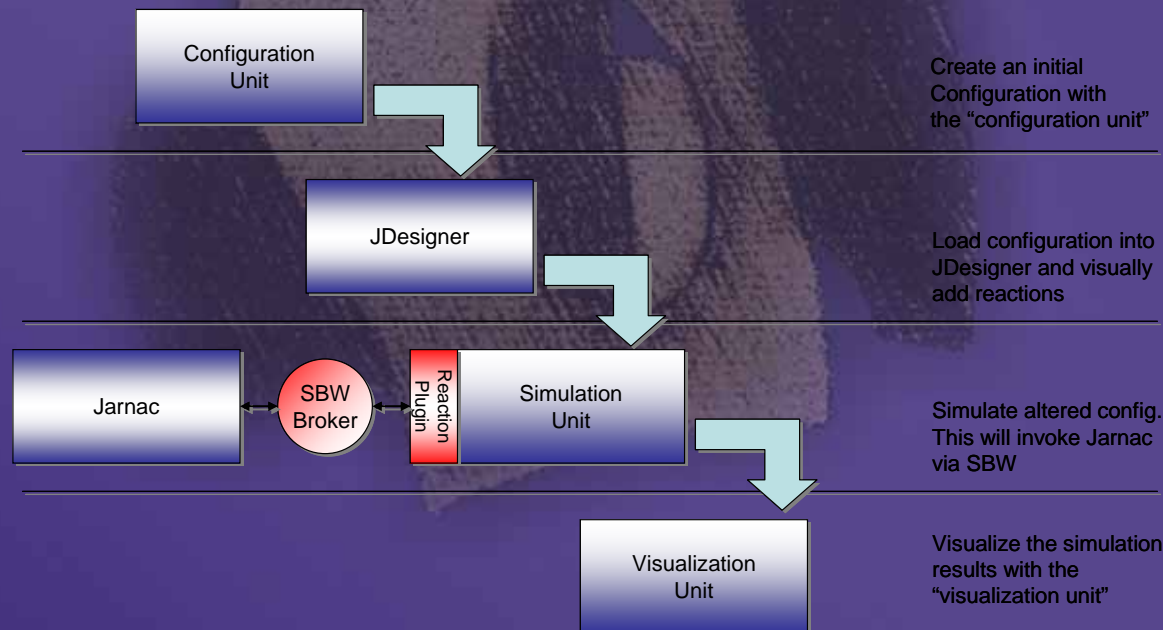
$$\text{rate of change } Y = BX - X^2Y$$

Konzept – Simulation – Plugins

3 / 3

◆ SBW/Jarnac

- Die Idee hierbei ist eine mittels des Konfigurationstools generierte Konfiguration in JDesigner nachzubearbeiten und später durch Jarnac berechnen zu lassen



Konzept – Visualisierung 1 / 2

- ◆ Zum Testen, ob eine 3D Visualisierung während der Durchführung der Simulation möglich/sinnvoll ist, sind zwei Ausführungsmodi vorgesehen:
 - Online: Visualisierung während der Simulation
 - Offline: Visualisierung von abgespeicherten Simulationsdaten
- ◆ Aufgaben der Visualisierung
 - Laden der Konfiguration / Laden der Simulationsdaten
 - Berechnen und Anzeigen einer ersten Iteration / Anzeigen der ersten abgespeicherten Iteration
 - Abgeben der Kontrolle an den Benutzer:
 - ◆ Navigation durch die generierte Szene
 - ◆ Automatisches Abspielen weiterer Iterationen
 - ◆ Verändern des Schwellenwertes für den die Szene erstellt wurde
 - ◆ Auswählen einer anderen Visualisierung
 - ◆ „Clippen“ des Reaktionsvolumen entlang der drei Hauptachsen

Konzept – Visualisierung 2 / 2

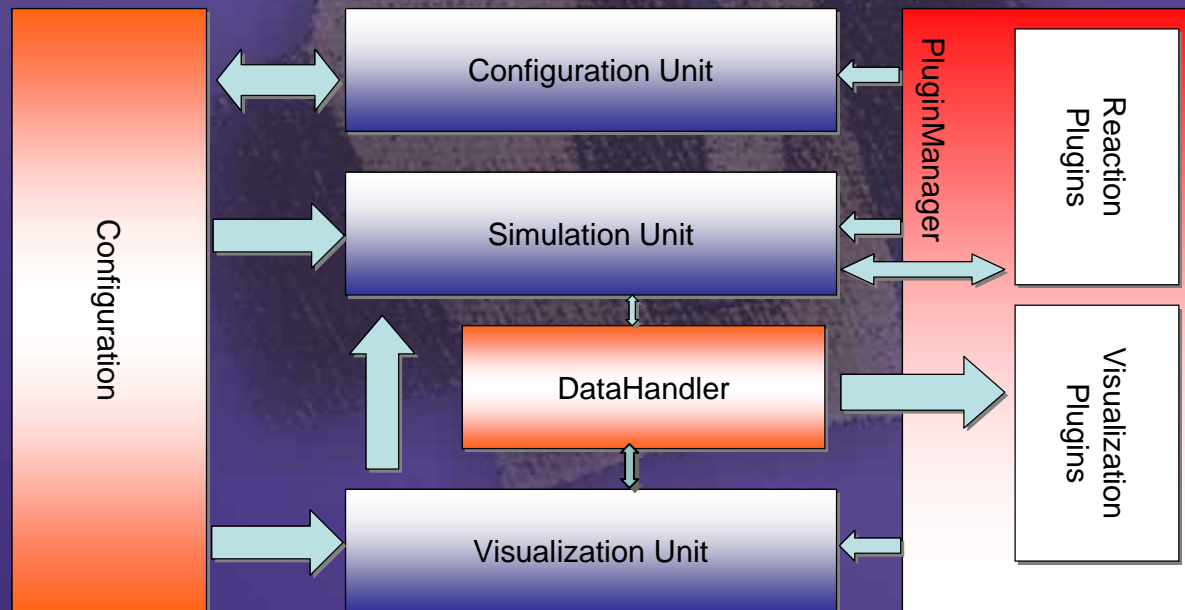
- ◆ Plugin System für verschiedene Visualisierung der Simulationsdaten:
 - Umsetzung von Standardalgorithmen der Volumenvisualisierung
 - ◆ Marching Cube
 - ◆ Marching Tetrahedra
 - ◆ TexturePlugin
 - ◆ Opaque Cubes

Überblick

- ◆ Einleitung
- ◆ Grundlagen
- ◆ Konzept
- ◆ **Realisierung**
 - Design des Systems
 - Konfiguration
 - Simulation
 - Visualisierung
- ◆ Zusammenfassung
- ◆ Ausblick
- ◆ Demonstration

Realisierung

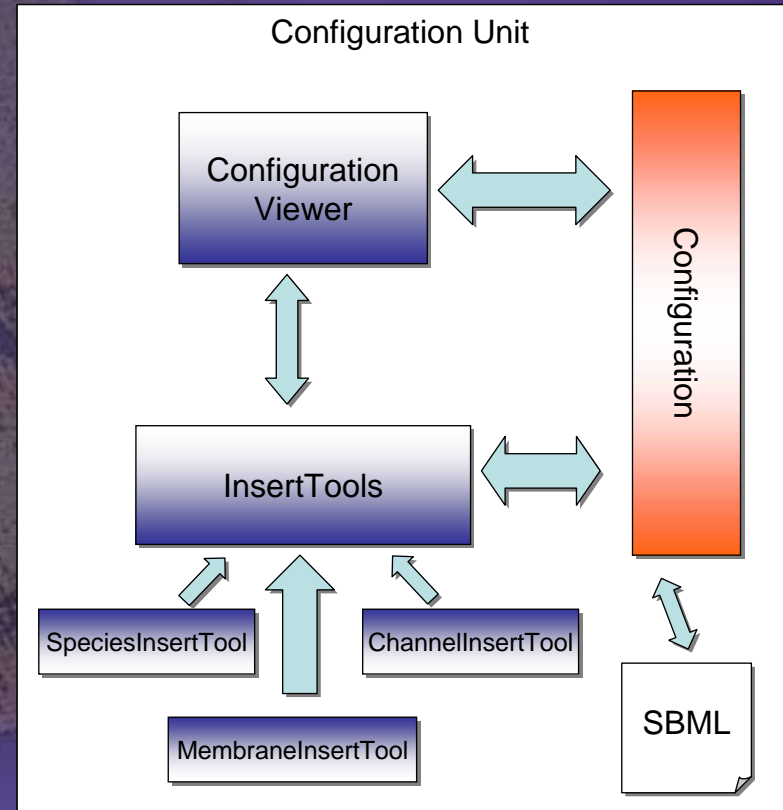
- ◆ Umsetzung auf der Basis von C++ erweitert um Qt und OpenSceneGraph
- ◆ Design des Systems



Realisierung – Konfiguration 1 / 2

◆ Überblick

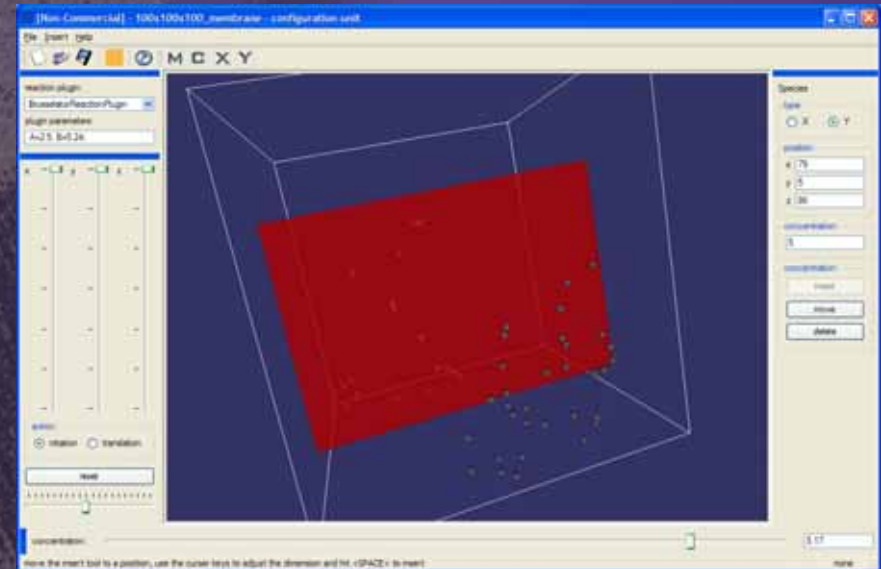
- Spezialisierte Werkzeuge zum Einfügen der verschiedenen Elemente
- Konfigurations-Datenstruktur basierend auf einer Hashmap von Elementen
- Serialisation der Konfiguration als SBML level 2, erweitert um Annotations welche Positionsinformationen enthalten, die noch nicht in der SBML Spezifikation enthalten sind



Realisierung – Konfiguration 2 / 2

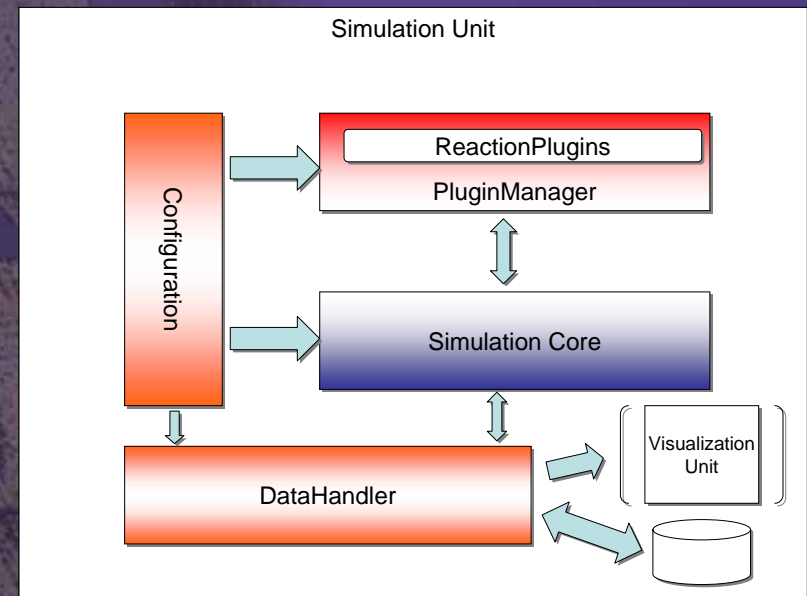
◆ Programm

- Beachtung von gängigen Standards
 - ◆ Dockwindows
 - ◆ Tool & Menu Bars
 - ◆ Statusleiste
 - ◆ Drag & Drop Unterstützung zum Öffnen neuer Dokumente
 - ◆ Recent-Files
 - ◆ Apspeichern von Position & Größe des Programmfensters



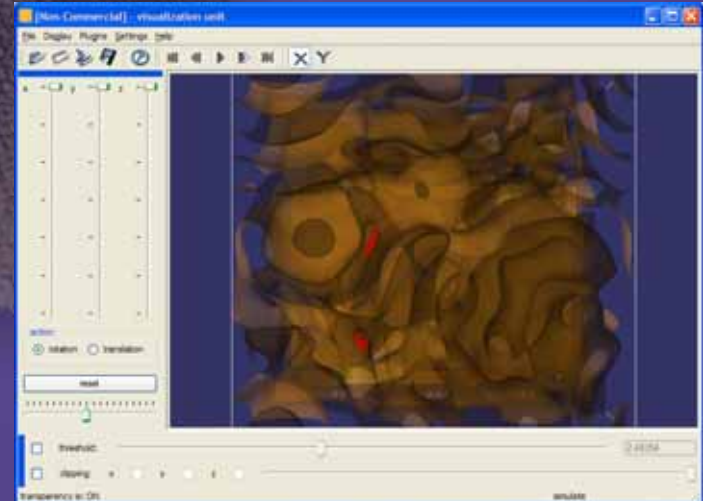
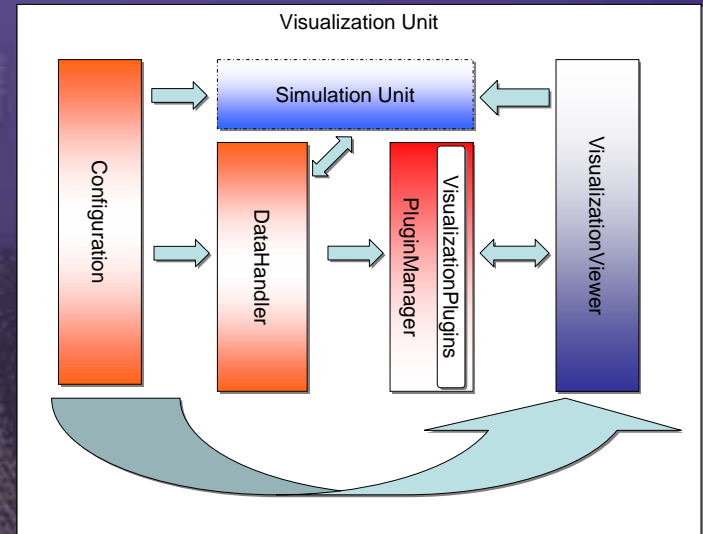
Realisierung – Simulation

- ◆ Realisierung als plattformunabhängiges Kommandozeilenprogramm für größtmögliche Einsetzbarkeit auf verschiedenen Systemen
- ◆ Verschiedene Operationsmodi:
 - Initialisierung von Konfigurationsdatei
 - Initialisierung von vorherigem Simulationslauf
 - Abänderung einer laufenden Simulation durch Angabe einer neuen Konfigurationsdatei

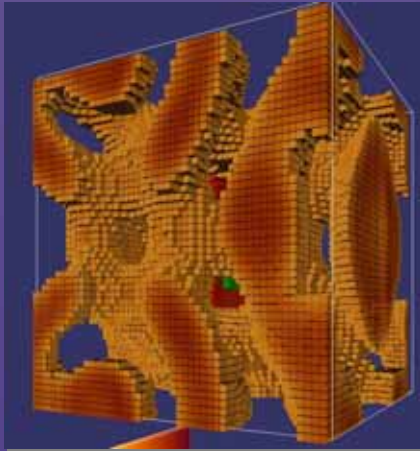


Realisation - Visualisierung 1 / 3

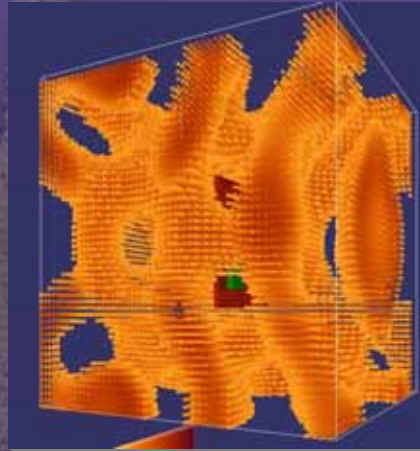
- ◆ Online- und Offline-Modus unterscheiden sich hauptsächlich in der Ansteuerung des DataHandlers. Entweder wird dieser durch den Simulator gefüllt, oder es werden Simulationsdaten früherer Simulationen in ihn geladen.
- ◆ Die Programmoberfläche wurde möglichst einheitlich zur „configuration unit“ gehalten. Mit der gleichen Unterstützung der gängigen Windows Features.



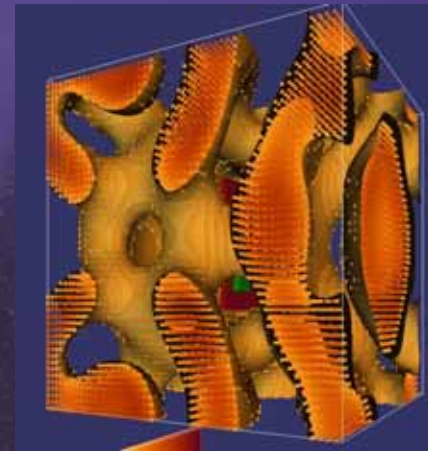
Realisation - Visualisierung 2 / 3



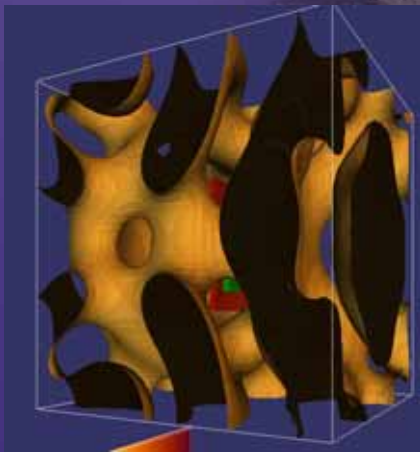
SimpleCubePlugin



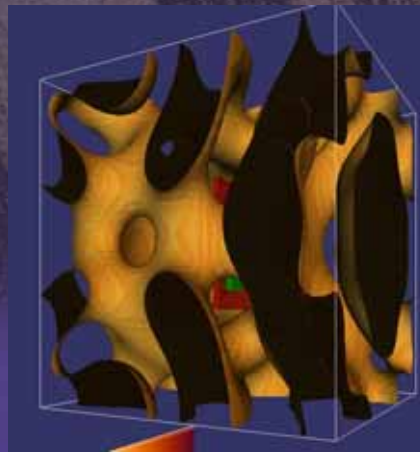
PointCloudPlugin



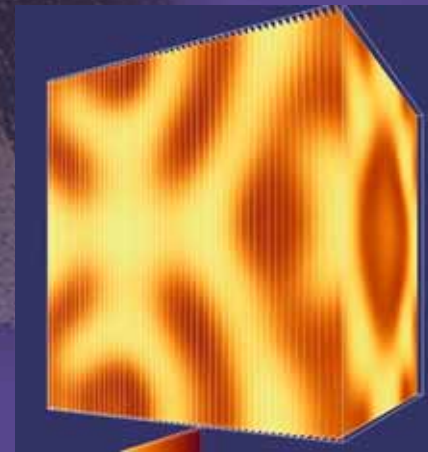
ComboPlugin



MarchingCubePlugin

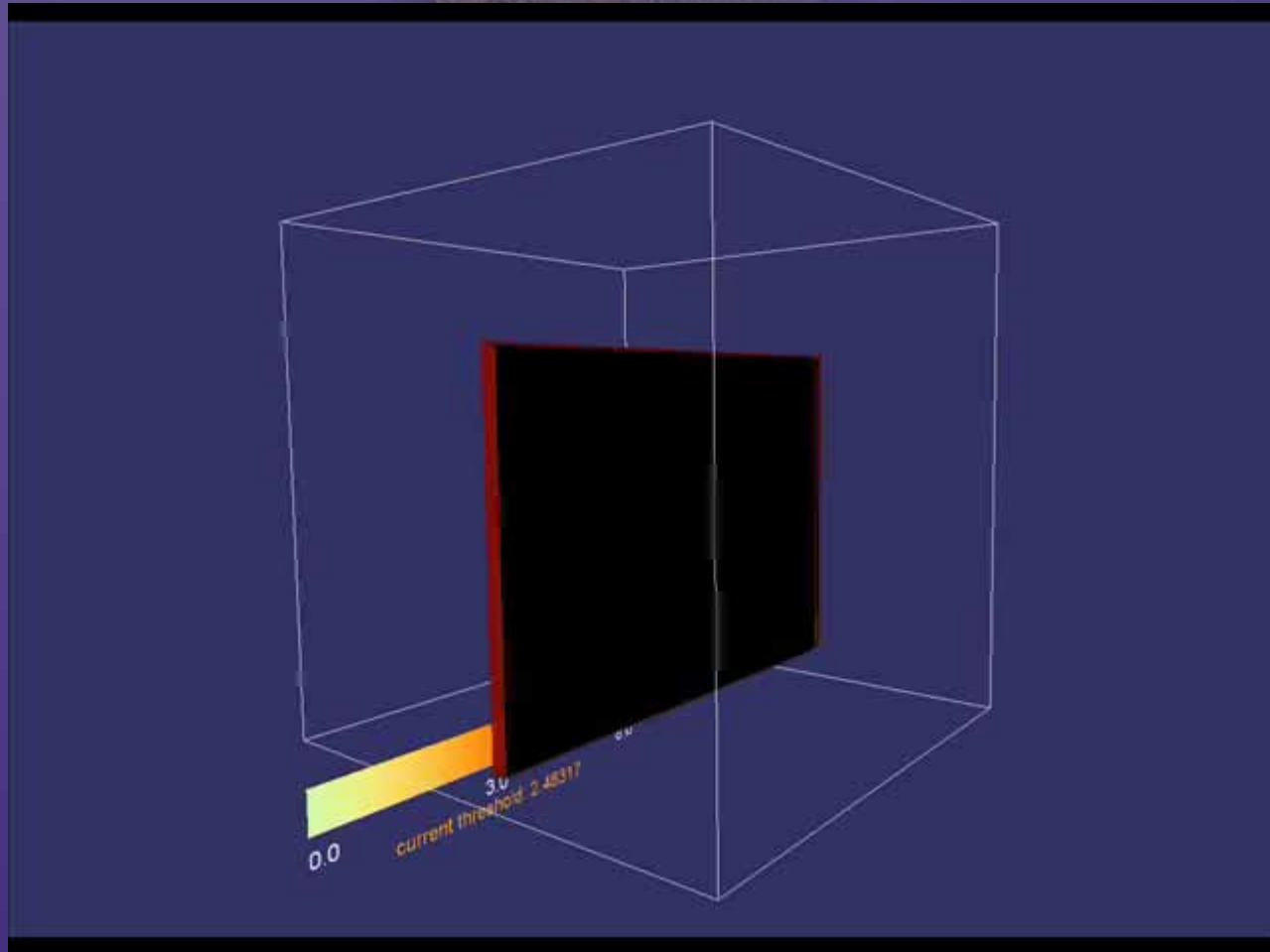


MarchingTetrahedronPlugin



TexturePlugin

Realisation - Visualisierung 3 / 3



Probleme & Limitationen

- ◆ 3D Simulation & Visualisierung
 - Große Anzahl von Elementen
 - viele Berechnungen
 - viel Speicherplatz benötigt
 - Somit ist selbst auf aktuellen Computern höchstens die Simulation und Visualisierung von Reaktionsvolumen bis zu 200x200x200 sinnvoll möglich.
- ◆ Qt-Multithreading
 - Die Idee war die Generierung der Szene durch die einzelnen Plugins in einen separaten Thread auszulagern.
 - Durch die Verwendung des QProgressDialog hätte dann die Generierung der Szene einfach abgebrochen werden können, falls sie zu viel Zeit und Ressourcen verbraucht hätte. Dieser Dialog hätte nur erscheinen sollen, nach dem eine einstellbare Zeitspanne überschritten wurde.
- ◆ SBW-Performance
 - Durch die interne Message Struktur von SBW musste für jede Berechnung eine Nachricht vom Simulator über den Broker an Jarnac gesendet werden.
 - Das führte im Endeffekt zu immens hohen Laufzeiten, selbst für kleine Reaktionsvolumen.

Überblick

- ◆ Einleitung
- ◆ Grundlagen
- ◆ Konzept
- ◆ Realisierung
- ◆ **Zusammenfassung**
- ◆ Ausblick
- ◆ Demonstration

Zusammenfassung

- ◆ Das Ziel der Konfiguration, Simulation und Visualisierung einfacher, dreidimensionaler Reaktionsdiffusionssysteme konnte erreicht werden.
- ◆ Zwar mussten etliche vereinfachende Annahmen getroffen werden, diese erlaubten aber eine hilfreiche Visualisierung von Reaktionsdiffusionsprozessen.
- ◆ Durch die Dreiteilung des Problems sind drei Programme entstanden, die auch durch Programme dritter erweitert oder ersetzt werden können. Dies ist unter anderem auch durch das Verwenden des standardisierten Dateiformates SBML erreicht worden.
- ◆ Schließlich ist das entwickelte Programmpaket leicht erweiterbar, durch das implementierte PluginSystem für Reaktionsgleichungen und Visualisierungen.

Überblick

- ◆ Einleitung
- ◆ Grundlagen
- ◆ Konzept
- ◆ Realisierung
- ◆ Zusammenfassung
- ◆ **Ausblick**
- ◆ Demonstration

Ausblick

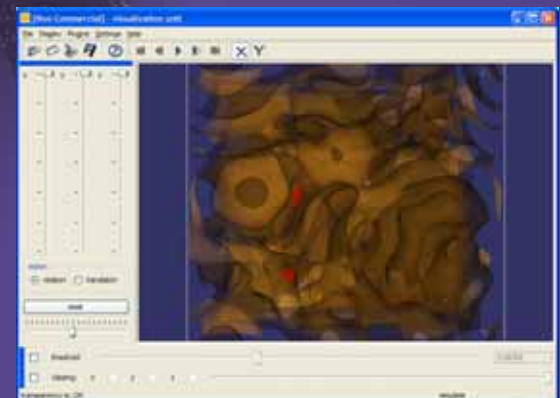
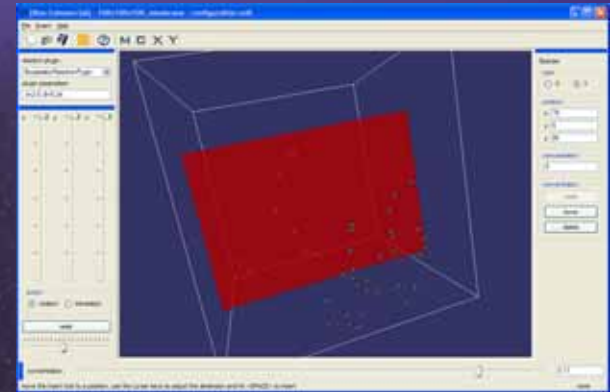
- ◆ Optimierung des Simulationskerns
- ◆ Ändern von Simulationsannahmen
 - Einheitliche Diffusionsrate im ganzen Reaktionsvolumen
 - Unbegrenztes Vorkommen von Substraten
 - Membranen
 - ◆ im Moment nicht durchlässig
 - ◆ Ändern von Reaktionen an Membranen
 - ◆ Bewegen der Membranen
- ◆ Integration bestehender Volumenvisualisierungsbibliotheken: vtk / OpenDX

Überblick

- ◆ Einleitung
- ◆ Grundlagen
- ◆ Konzept
- ◆ Realisierung
- ◆ Zusammenfassung
- ◆ Ausblick
- ◆ **Demonstration**
 - Konfiguration und Visualisierung von Diffusion
 - Visualisierung einer Brusselator-Simulation

Demonstration

- ◆ Diffusion (live)
 - Erstellung eines Reaktionsvolumens
 - Einfügen einiger Moleküle
 - Simulation von Diffusion
- ◆ Brusselator (von Datei)
 - Demonstrieren der 3D VisualizationPlugins



Danksagung

◆ KGI:

- Prof. Herbert M. Sauro
- Vijay Chikarmane, Alpan Raval, Cameron Wellock, Anastasia Deckard, Sri Rama Krishna Paladugu, Abhishek Agrawal

◆ AGC:

- Prof. Dr.-Ing. Detlef Krömker
- Dipl.-Wirtsch.-Inform. Daniel F. Abawi
- Dipl.-Biol. Jens Barthelmes

◆ Freunde:

- Matthias Pfeiffer, Martin Klossek, Christoph Karwoth

Vielen Dank für Ihre Aufmerksamkeit

- ◆ Mehr Informationen zu dieser Arbeit, sowie die PDF Version der Diplomarbeit und eine Programmversion finden Sie unter:

<http://public.kgi.edu/~fbergman/thesis/results.html>

- ◆ Für Fragen und Anmerken bin ich zu Erreichen unter:

frank_bergmann@kgi.edu