

# **Evaluation von Open-Source-Software für System Dynamics hinsichtlich deren Integrierbarkeit**

**Axel Hummel<sup>1</sup>, Heiko Kern<sup>1</sup>, Christian Böhme<sup>2</sup>,  
René Keßler<sup>2</sup> und Arndt Döhler<sup>2</sup>**

**<sup>1</sup> Betriebliche Informationssysteme, Universität Leipzig**

**<sup>2</sup> Intershop Communications AG**

**9. Workshop Informationssysteme mit Open Source (ISOS 2012)**

**19. September 2012**

# Agenda

---

## 1) Einleitung und Motivation

## 2) Anforderungen an eine Simulationsumgebung

## 3) Evaluation von Simulationsumgebungen für System Dynamics

- Evaluationsparameter
- Evaluationsdurchführung
- Evaluationsergebnisse

## 4) Fazit

# Einleitung – Problemstellung

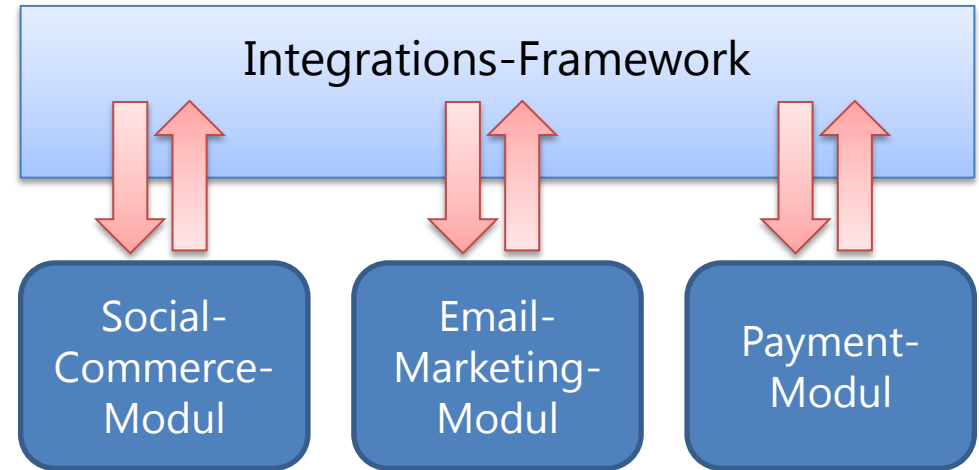
- **Die optimale Konfiguration eines Online-Shops ist eine schwierige Aufgabe**
  - Hohe Anzahl von Konfigurationsparametern
  - Abhängigkeiten zwischen den Konfigurationsparametern
- **Shop-Manager entscheiden auf Basis ihrer bisherigen Erfahrungen**
  - Subjektive Entscheidungen
  - Effekte der Entscheidungen sind kaum vorhersehbar



# Einleitung – Zielsetzung und Lösungsansatz

## – Zielsetzung

- Entwicklung eines Simulations-Frameworks zur Entscheidungsunterstützung von Shop-Managern



## – Lösungsansatz

- Entwicklung von atomaren Simulationsmodulen
- Kombination von verschiedenen Simulationsmodulen zur Definition komplexer E-Commerce-Szenarios

# Einleitung – Simulations-Framework

---

- **Simulations-Framework verfügt über**
  - Zentrale Benutzeroberfläche
    - **Konfiguration der Gesamtsimulation**
    - **Darstellung der Simulationsergebnisse**
  - Verschiedene Laufzeitumgebungen zur Ausführung der Simulationsmodule
    - **System Dynamics**
    - **Agentenbasierte Simulation**
  - Implementierung erfolgt in Java
- **Offene Frage**
  - Welche Laufzeitumgebung zur Ausführung von System Dynamics-Simulationen ist für das Simulations-Framework besonders geeignet?

## – **Simulationssoftware für System Dynamics**

1. Klassische Programmiersprachen (C, C++, Java, ...)
2. Spezialisierte Programmiersprachen
3. Integrierte Simulationsumgebungen

## – **Für System Dynamics existieren heutzutage leistungsfähige Simulationsumgebungen**

- Modelleditor
- Laufzeitumgebung
- Auswertung der Simulationsergebnisse

# Anforderungen an Simulationsumgebung

---

- **Primäre Anforderungen**

- Plattformunabhängige Laufzeitumgebung (Windows, Linux, Mac OS)
- Integrierbare Laufzeitumgebung
  - **Modelle laden und ausführen**
  - **Ergebnisse auslesen**
  - **Idealerweise mittels Java-API**
- Akzeptabler Preis

- **Sekundäre Anforderungen**

- Komfortabler Modelleditor

# Evaluationsparameter

---

- **Modelleditor**
  - Graphisch oder textuell
- **Funktionalitäten der integrierten Laufzeitumgebung**
  - Modelle laden
  - Modelle ausführen
  - Ergebnisse zurückgeben
- **Arten der Integration**
  - Batch-Aufruf, Programmierschnittstelle (API)
- **Technologie / Programmiersprache für Integration**
- **Betriebssystem**



# Evaluationsdurchführung

---

- **Auswahl von 10 Simulationsumgebungen für System Dynamics**
  - 6 weit verbreitete kommerzielle Simulationsumgebungen
    - **Vensim, Powersim, iThink, STELLA, AnyLogic, Consideo Modeler**
  - 4 verfügbare Open-Source-Simulationsumgebungen
    - **Sphinx SD Tools, MapSim, System Dynamics, Simantics System Dynamics**
  
- **Evaluationsablauf**
  - Umsetzung eines konkreten Szenarios
  - Auswertung der Dokumentation

# Evaluationsergebnisse – kommerzielle Simulationsumgebungen

	Vensim 5.11a	Powersim Studio 9	iThink / STELLA 9.1.4 / 9.1.4	AnyLogic 6.7.1	Consideo Modeler 7.5.1
Modelleditor	Graphisch	Graphisch	Graphisch	Graphisch	Graphisch
Modelle laden	Ja	Ja	Ja	Ja	Nein
Modelle ausführen	Ja	Ja	Ja	Ja	Nein
Ergebnisse zurückgeben	Ja	Ja	Ja	Ja	Nein
Arten der Integration	Batch-Aufruf, API	Batch-Aufruf, API	Batch-Aufruf, API	Batch-Aufruf, API	Keine
Technologie	C++, C#	.NET	.NET	Java	(Java)
Betriebssystem	Windows, Mac OS	Windows	Windows, Mac OS	Windows, Linux, Mac OS	Windows, Linux, Mac OS
Ausschlusskriterium	C++, C#	.NET	.NET	Hoher Preis	Fehlende Integration

# Evaluationsergebnisse – kommerzielle Simulationsumgebungen

	Vensim 5.11a	Powersim Studio 9	iThink / STELLA 9.1.4 / 9.1.4	AnyLogic 6.7.1	Consid Modeler 7.5.1
Modelleditor	Graphisch	Graphisch	Graphisch	Graphisch	Graphisch
Modelle laden	Ja	Ja	Ja	Ja	Nein
Modelle ausführen	Ja	Ja	Ja	Ja	Nein
Ergebnisse zurückgeben	Ja	Ja	Ja	Ja	Nein
Arten der Integration	Batch-Aufruf, API	Batch-Aufruf, API	Batch-Aufruf, API	Batch-Aufruf, API	Keine
Technologie	C++, C#	.NET	.NET	Java	(Java)
Betriebssystem	Windows, Mac OS	Windows	Windows, Mac OS	Windows, Linux, Mac OS	Windows, Linux, Mac OS
Autoren- kriterium	C++, C#	.NET	.NET	Hoher Preis	Fehler- Integration

# Evaluationsergebnisse – Open-Source-Simulationsumgebungen

	<b>Sphinx SD Tools</b> <b>0.7b</b>	<b>MapSim</b> <b>4.1</b>	<b>System Dynamics</b> <b>1.3</b>	<b>Simantics System Dynamics</b> <b>1.4</b>
Modelleditor	Graphisch	Textuell	Graphisch	Graphisch
Modelle laden	Ja	Ja	Ja	Ja
Modelle ausführen	Ja	Ja	Ja	Ja
Ergebnisse zurückgeben	Ja	Ja	Ja	Ja
Arten der Integration	API	Batch-Aufruf, API	Batch-Aufruf	Batch-Aufruf, API
Technologie	Java	.NET	Java	C
Betriebssystem	Windows, Linux, Mac OS	Windows	Windows, Linux, Mac OS	Windows, Linux, Mac OS
Ausschlusskriterium	—	.NET	Reifegrad, API	C

# Evaluationsergebnisse – Open-Source-Simulationsumgebungen

	<b>Sphinx SD Tools</b> <b>0.7b</b>	<b>MapSim</b> <b>4.1</b>	<b>System Dynamics</b> <b>1.3</b>	<b>Simantics System Dynamics</b> <b>1.4</b>
Modelleditor	Graphisch	Textuell	Graphisch	Graphisch
Modelle laden	Ja	Ja	Ja	Ja
Modelle ausführen	Ja	Ja	Ja	Ja
Ergebnisse zurückgeben	Ja	Ja	Ja	Ja
Arten der Integration	API	Batch-Aufruf, API	Batch-Aufruf	Batch-Aufruf, API
Technologie	Java	.NET	Java	C
Betriebssystem	Windows, Linux, Mac OS	Windows	Windows, Linux, Mac OS	Windows, Linux, Mac OS
Ausschlusskriterium	—	.NET	Reifegrad, API	C

# Fazit – Sphinx SD Tools

- **Sphinx SD Tools wurden als favorisierte Lösung ausgewählt**
- **Sphinx SD Tools haben sich bewährt**
  - Mehrere Simulationen erfolgreich entwickelt (Payment-Modul, E-Mail-Marketing-Modul, Recommendation-Modul, ...)
  - Laufzeitumgebung per Java-API in Simulations-Framework integriert
  - API wurde erweitert



# Fazit – Open-Source vs. kommerzielle Simulationsumgebungen

- **Für unsere Zwecke sind Open-Source-Simulationsumgebungen die bessere Alternative**
  - Plattformunabhängigkeit
  - Gute Integrationsfähigkeit
- **Modelleditor**
  - Kaum Defizite (graphischer Editor, Syntaxprüfung, ...)
  - Probleme bei sehr umfangreichen System Dynamics-Modellen
- **Laufzeitverhalten der Simulationen**
  - Beide Lager bieten gute Performance
- **Auswertungskomponente**
  - Kommerzielle Simulationsumgebungen bieten mehr Möglichkeiten

# Vielen Dank für Ihre Aufmerksamkeit!

## Kontaktinformationen

Axel Hummel

Betriebliche Informationssysteme

Universität Leipzig

Augustusplatz 10-11, Raum P-906

04109 Leipzig, Germany

Tel.: +49 341 9732360

[hummel@informatik.uni-leipzig.de](mailto:hummel@informatik.uni-leipzig.de)

<http://bis.informatik.uni-leipzig.de/AxelHummel>



Forschungsprojekt SimProgno

<http://www.simprogno.de>

Gefördert vom Bundesministerium  
für Bildung und Forschung

GEFÖRDERT VOM



Bundesministerium  
für Bildung  
und Forschung

