

# Systemdesign

---

## Grafisches Eisenbahn Frontend (GEF)

<b>Author:</b>	<b>Nils Nesemann</b>
<b>Version:</b>	<b>0.6</b>
<b>Status:</b>	<b>freigegeben für Review</b>

## Inhalt

1	Einleitung.....	3
2	Architektur.....	4
2.1	Module .....	5
2.1.1	MainWindow .....	5
2.1.2	Interpreter .....	5
2.1.3	ConsoleWidget .....	5
2.1.4	ConfigWidget.....	5
2.1.5	GuiWidget.....	5
2.1.6	SerialInterface .....	5
2.1.7	FileLogging.....	5
2.2	Modulschnittstellen .....	5
3	System-Schnittstellen .....	6
3.1	Serielle Schnittstelle (RS232).....	6
3.2	Log Dateien.....	6
3.3	Config Datei .....	6
3.4	Benutzeroberfläche .....	6
4	Referenzdokumente.....	7

## Änderungen

Version	Datum	Author	Änderungen
0.0	12.12.2010	Nils Nesemann	Erstellung des Dokuments
0.1	15.12.2010	Nils Nesemann	Überarbeitung des Layouts
0.2	15.12.2010	Nils Nesemann	-Überarbeitung der Architekturbeschreibung -Überarbeitung der Systemschnittstellen
0.3	17.12.2010	Nils Nesemann	Einfügen einer Modulübersicht
0.4	17.12.2010	Robert Lucke	Korrektur gelesen
0.5	11.01.2011	Nils Nesemann	Layout überarbeitet
0.6	11.02.2011	Nils Nesemann	-Fehlerkorrektur -Reviewfreigabe

## 1 Einleitung

Das Systemdesign verschafft einen Überblick über das zu erstellende System und seine Architektur. Subsysteme bzw. Module und ihre Rolle werden in ihrer grundlegenden Funktion im Bezug zum Gesamtsystem erläutert.

## 2 Architektur

In Abbildung 1 wird die Architektur des „Grafischen Eisenbahn Frontends“ visualisiert.

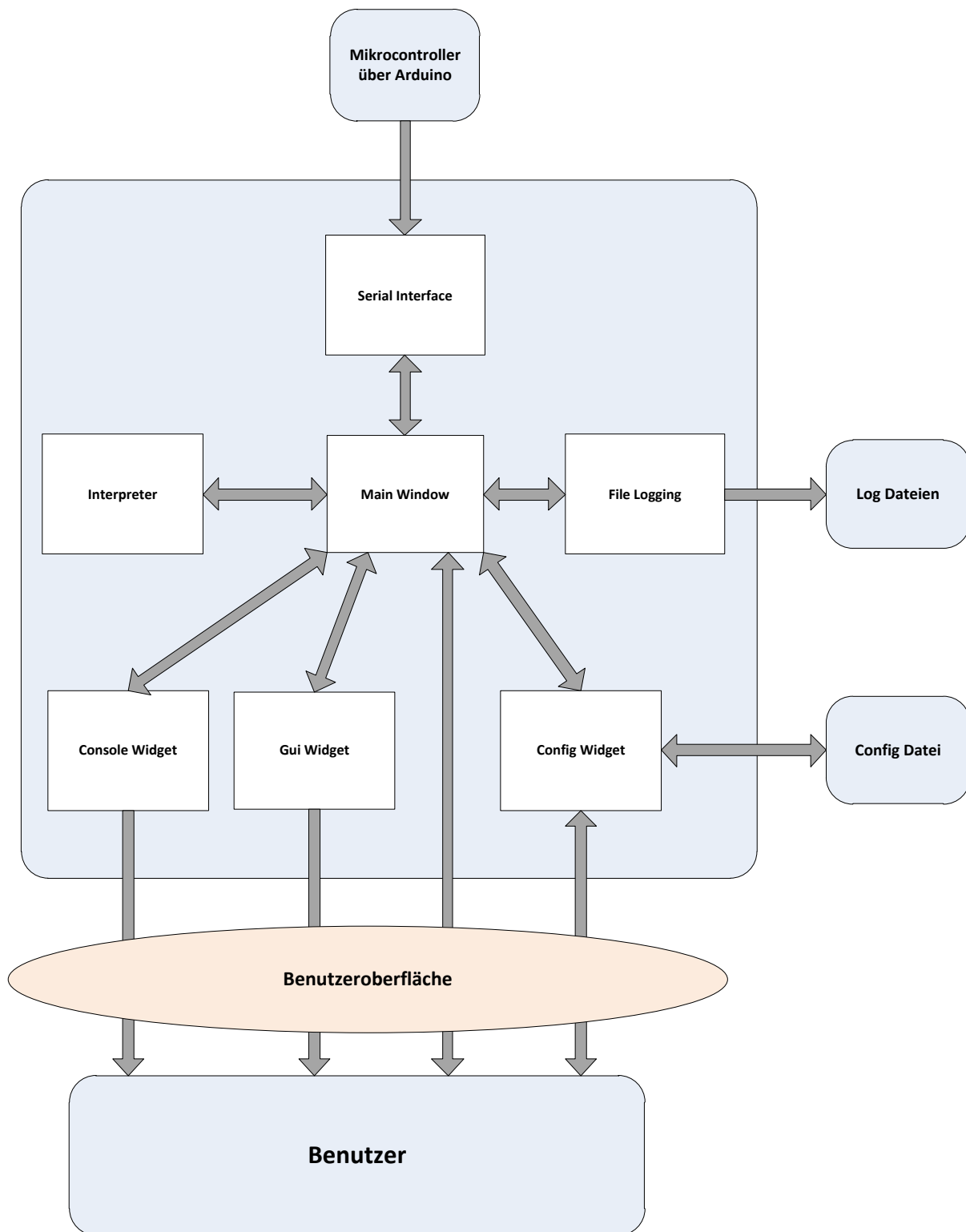


Abbildung 1: Systemarchitektur

Die Abbildung zeigt die einzelnen Module des Systems und einen Abstrakten Datenfluss zwischen diesen. Die zu erstellende Software ist im aktuellen Entwicklungsschritt lediglich ein darstellendes

Element und sieht keine Eingabe von Befehlen oder die Übermittlung von Befehlen an das Gleissystem vor.

## 2.1 Module

### 2.1.1 MainWindow

Das Modul MainWindow ist das zentrale Steuermodul der Anwendung. Der komplette Datenfluss wird über das Modul geleitet und kann so flexibel an die anderen Module verteilt werden. Dies ist nötig um zum Beispiel Rohdaten von der Seriellen Schnittstelle in einer Datei loggen zu können. Zudem werden sämtliche andere Module von dem MainWindow initialisiert und gesteuert.

### 2.1.2 Interpreter

Der Interpreter dient zur Aufbereitung der Daten welche über die serielle Schnittstelle empfangen werden. Sämtliche Rohdaten werden an den Interpreter übergeben. Sobald sich aus den Rohdaten ein gültiges Telegramm ergibt, gibt der Interpreter diese zurück.

### 2.1.3 ConsoleWidget

Das ConsoleWidget stellt die Benutzerkonsole(vorerst nur Ausgabe) dar. Ereignisse des Schienennetzes und des grafischen Frontends sowie Fehlermeldungen werden hier übersichtlich ausgegeben.

### 2.1.4 ConfigWidget

Die gesamte Eingabe und Verwaltung von Konfigurationsdaten übernimmt das ConfigWidget. Eine grafische Oberfläche ermöglicht dem Benutzer die Konfiguration. Die Konfiguration wird über das Laufzeitende hinaus in einer Datei abgelegt (siehe 3.3 Config Datei).

### 2.1.5 GuiWidget

Das GuiWidget zeichnet als Teil der Benutzeroberfläche ein rudimentäres Abbild des Schienennetzes und der enthaltenen Elemente (Sensoren, etc.). Ereignisse auf dem Schienennetz (z.B. Sensor aktiviert) werden für den Benutzer anschaulich dargestellt. Ebenso werden aktuelle Zugdaten in Textform ausgegeben.

### 2.1.6 SerialInterface

Die serielle Schnittstelle ist die Schnittstelle zum Gleissystem (über den Arduino). Sie dient als Low-Level Treiber zum empfangen von Daten über den COM-Port.

### 2.1.7 FileLogging

Das Logging-Modul stellt Funktionen bereit um Events und Daten in Dateiform mitzuschneiden.

## 2.2 Modulschnittstellen

Alle Module (das Hauptfenster ausgenommen) sollen einheitlich über die gleichen Schnittstellen zum Konfigurieren, Initialisieren, Starten und Stoppen verfügen.

Darüber hinaus sind die Schnittstellen im jeweiligen Moduldesign definiert.

## 3 System-Schnittstellen

### 3.1 Serielle Schnittstelle (RS232)

Siehe Schnittstellendokument

### 3.2 Log Dateien

Die zu erstellenden Logdateien sollen folgende Daten (je nach Konfiguration durch den Nutzer) enthalten können:

- Datum in der Form: JJJJ-MM-TT
- Uhrzeit in der Form: HH-mm-SS-ssss (Stunde-Minute-Sekunde-Millisekunde)
- Empfangene Rohdaten in der Form: [01 02 05 FF E1]
- Interpretierte Telegramme in der beispielhaften Form: „Sensor 1: aktiv“
- Statusmeldungen der Anwendung selbst. Zum Beispiel: „Interpreter: Status: Gestartet“

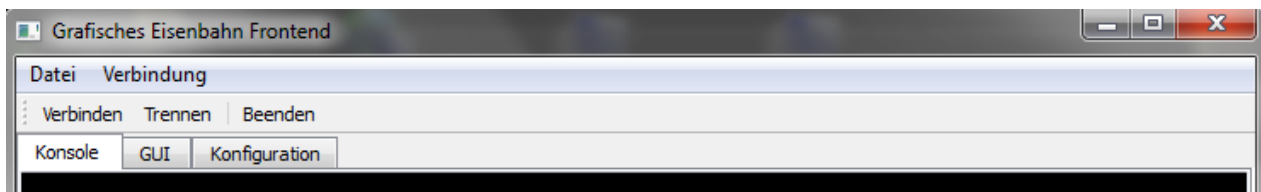
Die Dateien selbst sind als Textdateien mit der Endung „.log“ abzuspeichern.

### 3.3 Config Datei

Die Systemkonfiguration ist als \*.ini Datei abzuspeichern. Die Jeweils benötigten Parameter sind in im jeweiligen Moduldesign zu definieren.

### 3.4 Benutzeroberfläche

Die Benutzeroberfläche setzt sich aus den Modulschnittstellen zusammen.



**Abbildung 2: Schema der Benutzeroberfläche**

Abbildung 2 zeigt wie die einzelnen Modulschnittstellen anzuordnen sind. Die Funktionen des Hauptfensters (hier schematisch „Verbinden, Trennen, Beenden“) sollen aus einer Symbolleiste aufrufbar sein. Modulspezifische Funktionen sind in den zugehörigen Tabs abrufbar.

Die Modulspezifischen Benutzerschnittstellen sind im jeweiligen Moduldesign definiert.

## **4 Referenzdokumente**

Pflichtenheft

Interface Dokumente

Moduldesign