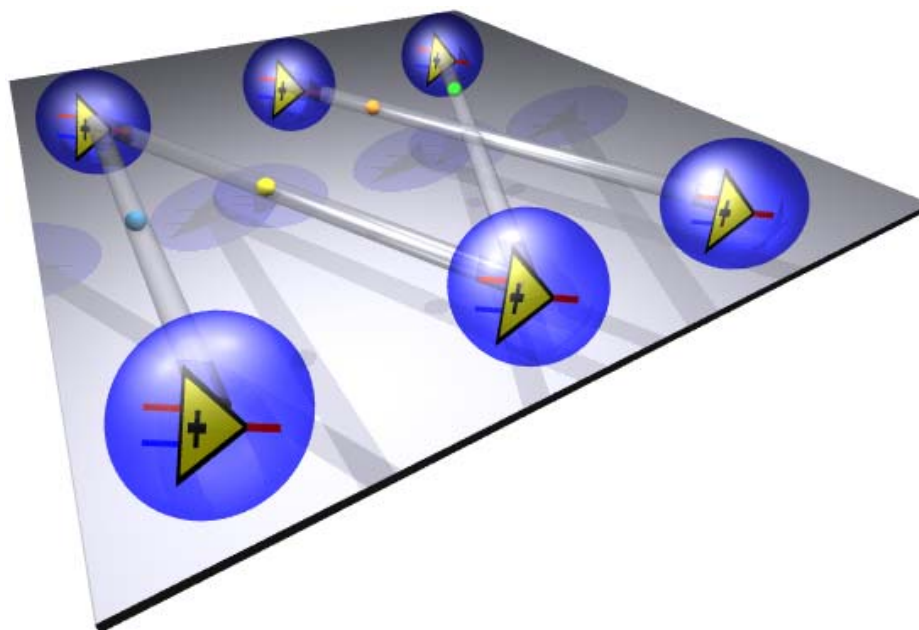


# Demo Sortieranlage

-

## Class Inheritance Toolset



# Sortieranlage

Das Demoprogramm Sortieranlage zeigt die Anwendung von Klassen und Vererbung. Die Kommunikation zwischen den Objekten erfolgt über Methodenaufrufe und Aggregation.

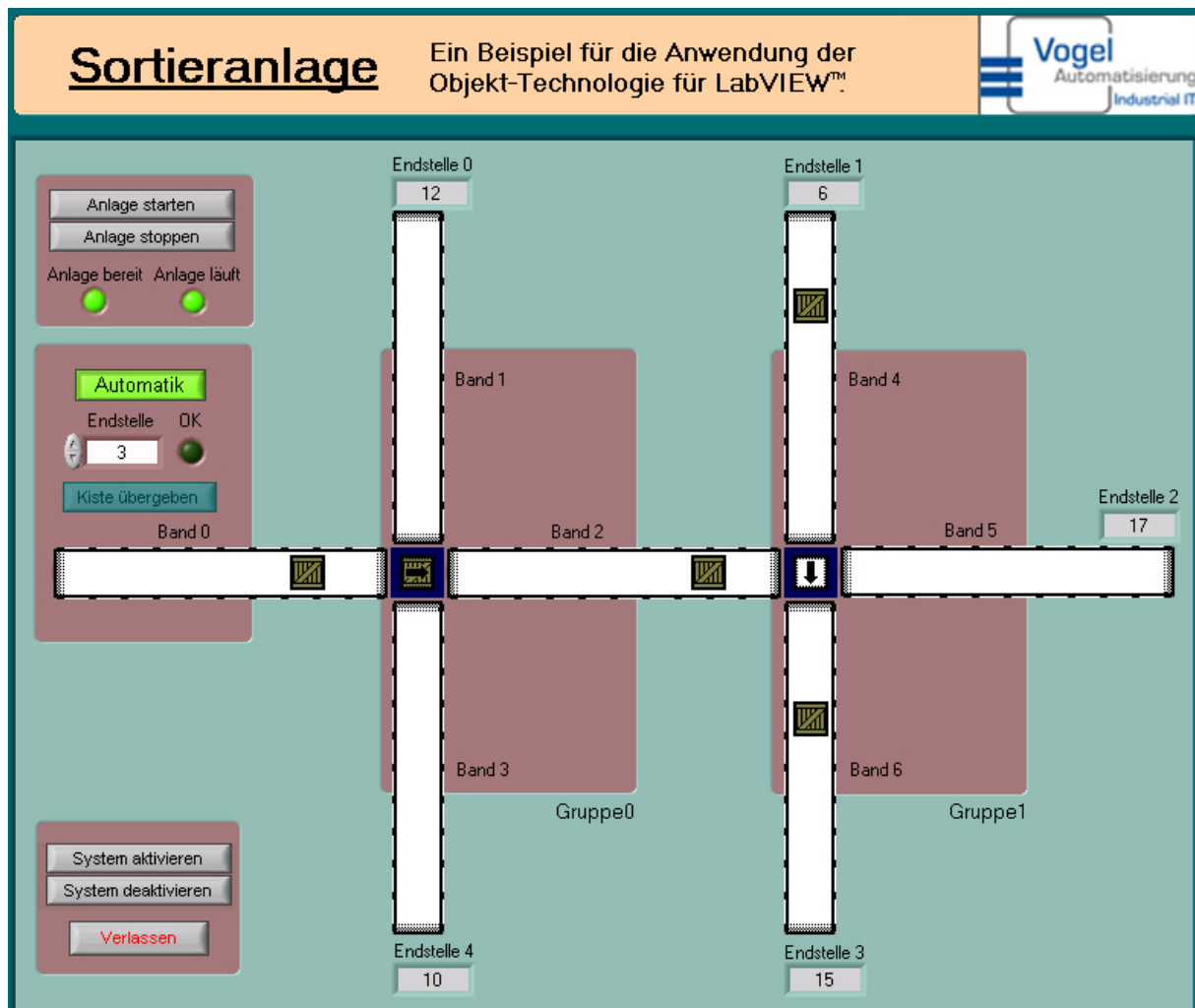
An einer zentralen Stelle werden Kisten aufgegeben, gemäß ihrer Zieladresse weitergeleitet und an Endstellen gesammelt.

Die Sortieranlage ist aus unabhängigen Objekten aufgebaut. Diese repräsentieren Transportbänder, Verteiler, Endstellen und Gruppen. Sie werden allgemein als Transportelemente bezeichnet. Die Objekte transportieren, verteilen, oder zählen die Kisten oder fassen Transportelemente zu einem komplexen Transportelement (Gruppe) zusammen.

Die Objektorientierung erlaubt eine beliebige Erweiterung der Sortieranlage. Es ist möglich zusätzliche Klassen von Transportelementen zu kreieren.

Die Objekte besitzen Indikatoren auf dem Frontpanel des Hauptprogramms (Hauptbildschirm). Damit kann die Anlage als übersichtliches Bild dargestellt werden.


Die Bedienung erfolgt vom Hauptprogramm aus.



## Bedienung

| <i>Bedien-<br/>Anzeigeelement</i> | <i>bzw.</i> | <i>Funktion</i>   |
|-----------------------------------|-------------|---|
| System aktivieren                 |             | Einrichten der Objektstruktur der Sortieranlage                 |
| System deaktivieren               |             | Objektstruktur auflösen, Objekte löschen                        |
| Anlage bereit                     |             | Objektstruktur der Sortieranlage ist eingerichtet               |
| Anlage starten                    |             | alle Transportbänder anfahren                                   |
| Anlage stoppen                    |             | alle Transportbänder anhalten                                   |
| Anlage läuft                      |             | die Transportbänder sind angefahren                             |
| Kiste übergeben                   |             | eine Kiste an BandO übergeben                                   |
| Endstelle                         |             | Nummer der Endstelle zu der die Kiste transportiert werden soll |
| Automatik                         |             | ständig Kisten mit zufälligem Ziel an BandO übergeben           |
| OK                                |             | Kiste konnte übergeben werden                                   |
| Verlassen                         |             | Beenden des Programms   |

Zum Start der Anlage sind folgende Schritte erforderlich (siehe Tabelle):

1. Starten des Hauptprogramms (  )
2. System aktivieren
3. Anlage starten

Danach können Kisten von Hand oder per Automatik übergeben werden.

Es ist möglich, die Anlage im Betrieb zu stoppen und wieder zu starten.

Das Programm ist mit folgenden Schritten zu beenden:

1. Anlage stoppen
2. System deaktivieren
3. Verlassen

## Struktur und Funktionsweise der Sortieranlage

### Hierarchie der Objekte

Zur Darstellung im Bereich der Objektorientierung hat sich die grafische Notation UML (*Unified Modelling Language*) etabliert. UML erlaubt eine kompakte Darstellung der verschiedenen Aspekte wie Klassen und Ableitung, Abhängigkeiten zwischen den einzelnen Objekten, deren Kommunikation usw. Dafür stehen verschiedene Diagrammtypen mit genormten Diagrammelementen zur Verfügung.

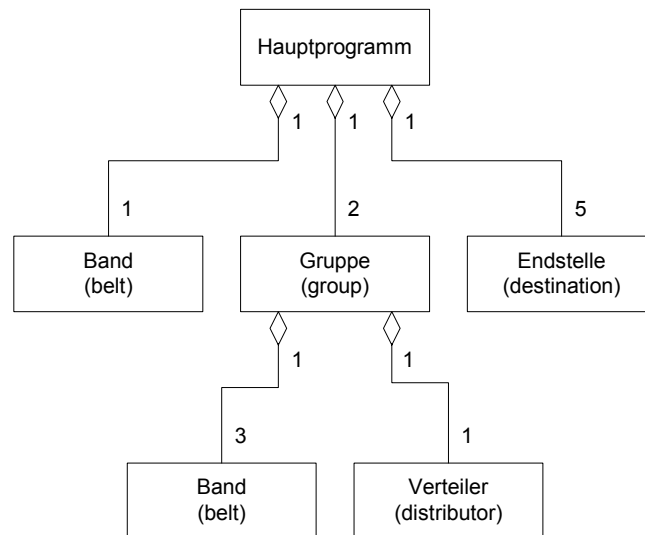
Assoziationsdiagramme verdeutlichen allgemein Beziehungen zwischen Objekten. Sie können genutzt werden, um spezielle Zusammenhänge wie die Kommunikation in bestimmten Situationen darzustellen. Die beteiligten Objekte sind als Rechtecke dargestellt, die Beziehungen durch Verbindungslinien. Daran ist die Art der Beziehung (z.B. Methodenaufruf) und durch einen Pfeil die Wirkungsrichtung angetragen.

Ein Aggregations- oder Kompositionsdiagramm zeigt die Zugehörigkeiten der einzelnen Objekte zur Laufzeit. Die Raute zeigt auf das übergeordnete Element.

Die Zahlen an den Verbindungslinien geben die Anzahl der zugeordneten Objekte des Typs an ( \* → beliebig). Durch diese Zusammenfassung vereinfacht sich die Darstellung erheblich.

Die Hierarchie der Objekte der Sortieranlage ist im Kompositionsdiagramm der Gesamtanlage dargestellt. Dem Hauptprogramm direkt untergeordnet sind: ein Transportband, zwei Gruppen und fünf Endstellen. Eine Gruppe wiederum besteht aus drei Transportbändern und einem Verteiler. Jedes Unterobjekt ist seinem Aggregatobjekt fest zugeordnet.

Dem Hauptprogramm sind nur die direkt untergeordneten Objekte bekannt. Die Funktionalität zur Verwaltung der Unterobjekte der Gruppen (kreieren, vernichten, etc.) ist in den Gruppen gekapselt.

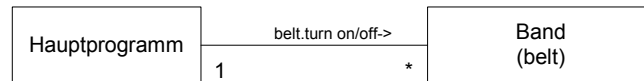


Kompositionsdiagramm Gesamtanlage

## Kommunikation

Das Assoziationsdiagramm für das Starten und Stoppen von Transportbändern zeigt die dafür erforderliche Kommunikation. Das Hauptprogramm führt einen Methodenaufruf (*belt.turn on* bzw. *belt.turn off*) für jedes zu startende bzw. stoppende Band durch.

Die Transportbänder der Gruppen können ebenfalls gestartet und gestoppt werden. Dafür stellen die Gruppen entsprechende Methoden zur Verfügung. Ein direkter Zugriff vom Hauptprogramm aus ist nicht möglich, da diese Transportbänder dort nicht bekannt sind.

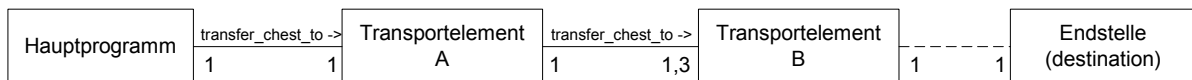


Assoziationsdiagramm  
Anlage starten/stoppen

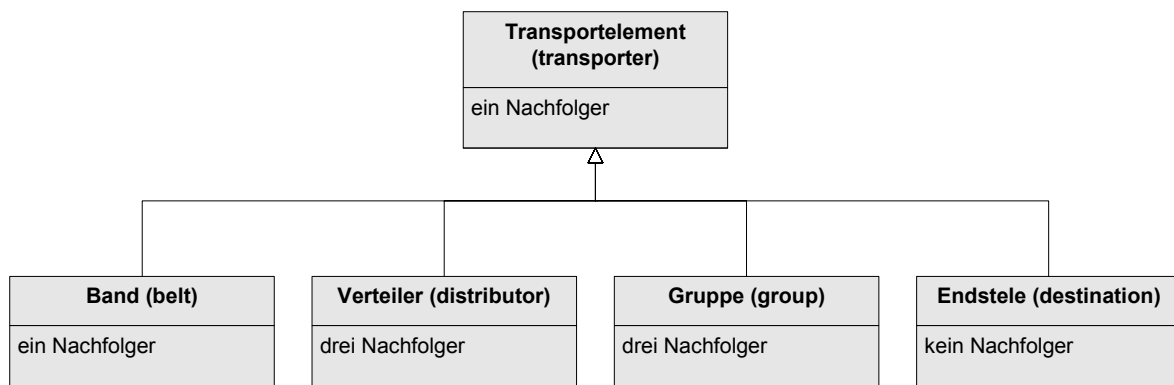
Der Kistentransport ist ebenfalls in einem Assoziationsdiagramm dargestellt. Allgemein betrachtet ergibt sich folgender Zusammenhang: Das Hauptprogramm übergibt die Kiste das Transportelement A. Dies geschieht durch den Methodenaufwurf *transporter.transfer\_chest\_to*. Dieses Transportelement kann einen oder drei Nachfolger haben. Im ersten Fall handelt es sich um ein Transportband im letzteren um einen Verteiler oder eine Gruppe (siehe vereinfachtes Vererbungsdiagramm).

Transportelement A gibt die Kiste, wiederum mittels Methodenaufwurf, an Transportelement B weiter. Das Hauptprogramm ist der Vorgänger, B der Nachfolger von A.

Jeder Zweig der Sortieranlage wird durch eine Endstelle abgeschlossen. Endstellen haben keinen Nachfolger.



Assoziationsdiagramm  
Kistentransport



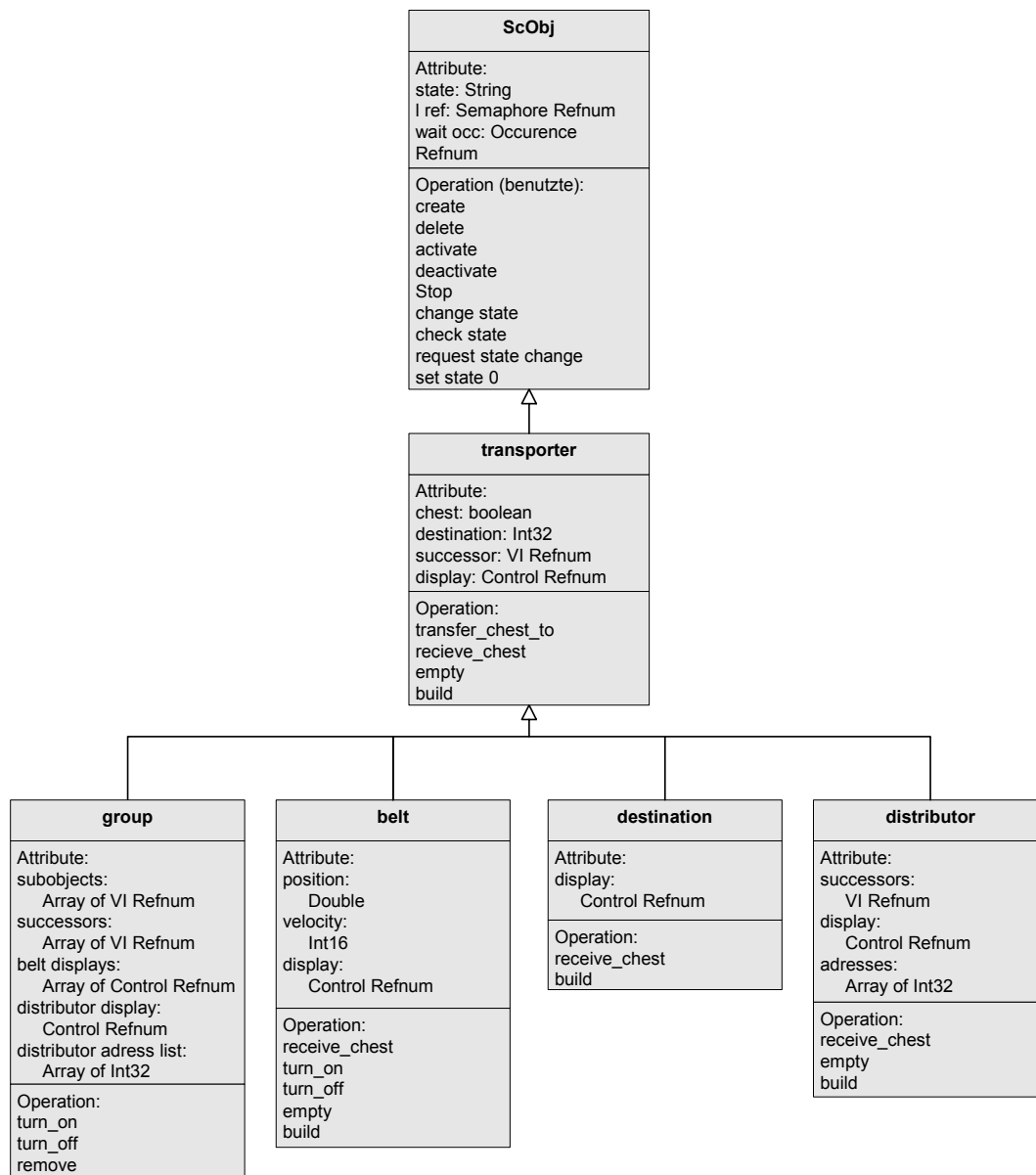
Vereinfachtes Vererbungsdiagramm

## Ableitung der Klassen

Dem Vererbungsdiagramm kann man entnehmen, wie die Klassen eines Projektes erzeugt wurden und welche Attribute und Methoden sie besitzen. Eine Klasse wird durch einen Kasten mit bis zu drei Feldern dargestellt. Das obere Feld enthält den Klassennamen, das mittlere die Attribute und das untere die Methoden.

Die Ableitungsbeziehung zwischen Oberklasse und abgeleiteter Klassen ist durch einen Pfeil dargestellt, dessen Spitze zur Oberklasse zeigt.

Bei der Ableitung werden alle Methoden und Attribute vererbt. Eine Klasse besitzt also zusätzlich zu den eigenen Methoden und Attributen die der Oberklasse.



Vollständiges Vererbungsdiagramm

## Funktion der Transportelemente

Eine Analyse der Funktionsweise der verschiedenen Transportelemente zeigt, daß alle über einen bestimmten Grundaufbau und gemeinsame Eigenschaften verfügen und daß sich diese gut mit einem Zustandsautomaten realisieren lassen. Deshalb wurde die gemeinsame Oberklasse *transporter* von der Klasse *ScObj* abgeleitet. Von dieser werden wiederum alle spezialisierten Transportelement-Klassen abgeleitet. Sie erben damit die einheitliche Grundfunktionalität.

Die gemeinsamen Eigenschaften sind:

1. ein Transportelement kann Kisten empfangen (vom Vorgänger)
2. ein Transportelement kann Kisten weitergeben (an den Nachfolger)
3. ein Transportelement kennt seinen Nachfolger
4. ein Transportelement kennt das Ziel der aktuellen Kiste
5. ein Transportelement kann zu einem Zeitpunkt höchstens eine Kiste haben

Die von *transporter* abgeleiteten Klassen erweitern diese Grundfunktionalität um spezifische Eigenschaften oder schränken sie ein.

Die zur Realisierung der Grundfunktionalität erforderlichen Attribute werden ebenfalls an die Unterklassen vererbt. Sie sind in *transporter.i* attribute zusammengefasst. Das Boolean *chest* zeigt an, ob das Objekt eine Kiste hat, in *destination* wird die Zieladresse dieser Kiste abgelegt und *successor* ist eine VI-Referenz auf das Nachfolger-Objekt.

Die Methoden der Klasse *transporter* dienen der Einrichtung eines Objektes der Klasse (*build*), dem Empfang einer Kiste (*receive\_chest*) der Weitergabe an den Nachfolger (*transfer\_chest\_to*) und der Freigabe des Transportelementes für den Empfang weiterer Kisten (*empty*).

Ein Transportband (*belt*) transportiert eine am Bandanfang übergebene Kiste zum Bandende und übergibt sie an das nachfolgende Transportelement. Die Bewegung wird auf dem Hauptbildschirm durch einen modifizierten Slide-Indikator angezeigt.

Die Objekte der Klasse *belt* besitzen zusätzliche Attribute wie aktuelle Position der Kiste (*position*), Geschwindigkeit des Bandes (*velocity*) und eine Referenz auf das Anzeigeelement des Bandes (*display*).

Die geerbte Funktionalität wurde um Methoden erweitert, die die Bewegung des Bandes (*move*), das Anfahren (*turn on*) und das Anhalten (*turn off*) realisieren.

Einige der von der Klasse *transporter* geerbten Methoden wurden erweitert und sind deshalb sowohl in der Oberklasse *transporter* als auch in der abgeleiteten Klasse *belt* aufgeführt.

Ein Verteiler (*distributor*) besitzt drei Anschlüsse für nachfolgende Transportelemente. Aus dem Kistenziel (*destination*) wird die Richtung der Weitergabe ermittelt. Die Darstellung auf dem Hauptbildschirm erfolgt durch einen Picture-Ring-Indikator.

Die Attribute der Klasse *distributor* enthalten die VI-Referenzen auf die Nachfolgerobjekte (*successors*), eine Adressenliste (*adress list*) und eine Referenz auf das Anzeigeelement des Verteilers (*display*).

Eine Endstelle (*destination*) schließt einen Zweig der Sortieranlage ab. Sie sammelt und zählt eintreffende Kisten. Die Kisten werden nicht weitergegeben. Ein Numeric-Indikator auf dem Hauptbildschirm zeigt die Anzahl der empfangenen Kisten an.

Die Klasse *destination* hat als Attribut eine Referenz auf das Anzeigeelement der Endstelle (*display*).

Die Sortieranlage beinhaltet eine sich wiederholende Struktur aus einem Verteiler und drei Transportbändern. Diese werden als Gruppe (*group*) zusammengefasst. So wird der Aufwand bei der Erzeugung, Verwaltung und beim Löschen der Objekte entsteht gekapselt und die Übersichtlichkeit erhöht.

Die Attribute der Klasse *group* enthalten die VI-Referenzen auf die Nachfolger und Unterobjekte der Gruppe (*successors, subobjects*) Referenzen auf die Anzeigeelemente (*distributor display, belt displays*) und eine Kopie der Adressenliste des Verteilers (*distributor adress list*).

Die geerbte Funktionalität wurde erweitert um Methoden zum Starten (*turn on*) und Stoppen (*turn off*) der Bänder und zum Entfernen der Gruppe aus dem Speicher (*remove*).