

# Konzeption Powerwall



Master Infrastructure Situation Display Observing Windows and Linux

Ein System zur Überwachung von vernetzten Rechnern

P. Brombosch, D. Krauss, F. Müller, Y. Noller, J. Scheurich  
H. Schäfer, S. Zillessen, A. Schneider, E. Doust

Universität Stuttgart  
Studenten der Fachrichtung Softwaretechnik

**Erstellt am:** 03. Februar 2013  
**Freigegeben am:** 10. Februar 2013  
**Version:** Version 0.1

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>4</b>
1.1	Überblick über das Dokument . . . . .	4
1.2	Leserkreis . . . . .	4
1.3	Namenskonventionen für dieses Dokument . . . . .	5
<b>2</b>	<b>Powerwall</b>	<b>6</b>
2.1	Aufbau der <i>Powerwall</i> . . . . .	6
2.2	<i>Operator-PC</i> . . . . .	7
2.3	Netzwerk . . . . .	7
2.4	Zugriffsmöglichkeiten auf das Grafik- <i>Cluster</i> . . . . .	7
<b>3</b>	<b>Layout</b>	<b>8</b>
3.1	Einschränkungen . . . . .	8
3.2	Layout-Idee . . . . .	8
<b>4</b>	<b>Konzeption</b>	<b>11</b>
4.1	Interaktion . . . . .	11
4.2	Einstellungen . . . . .	11
4.2.1	Einstellungen auf dem Operator-PC . . . . .	12
4.2.2	Einstellungen auf den Powerwall-Instanzen . . . . .	12
4.3	Status-Änderungen . . . . .	12

4.3.1	Aquise der Änderungen . . . . .	13
4.3.2	Verarbeitung von Änderungen . . . . .	13
4.3.3	Übertragung der Änderungen . . . . .	15
4.4	Operator-PC Anzeige . . . . .	15
<b>A</b>	<b>Anhang</b>	<b>17</b>
A.1	Begriffslexikon . . . . .	17
A.2	Versionshistorie . . . . .	23

# Abbildungsverzeichnis

2.1	Technischer Aufbau der Powerwall . . . . .	6
3.1	Layout-Konzept: Alle Elemente haben einen dezimalen Wert über den in ihrer übergeordneten Struktur eine eindeutige vertikale Anordnung bestimmt werden kann . . . . .	9
3.2	Drag & Drop: Die Kachel soll von rechts nach links, vor 8.0 verschoben werden .	10
3.3	Drag & Drop: Die verschobene Kachel hat den neuen Wert 7.0 erhalten . . . . .	10
4.1	Konzept: Die Struktur und der Datenfluss beim generieren des Layouts . . . . .	14
4.2	Darstellung: Skalierung und Anzeigen des Ausschnittes auf dem Operator-PC . .	16

# Kapitel 1

## Einleitung

### 1.1 Überblick über das Dokument

Einen wesentlichen Bestandteil des Projektes *MISD OWL* macht die *Powerwall* aus. Auf dieser soll die Anwendung *MISD OWL* funktionstüchtig sein und den aktuellen Stand von *zu überwachende Rechner* visualisieren.

Wichtig ist dabei, dass im Vergleich zur normalen *Desktop*-Anwendung der Platz auf der *Powerwall* sehr viel größer ist und man aus diesem Grund mehr Informationen gleichzeitig darstellen können soll. Einen weiteren großen Unterschied ergibt sich in der Steuerung der *Powerwall*: Diese erfolgt nur vom *Operator-PC*.

### 1.2 Leserkreis

Zum Leserkreis dieses Dokuments gehören:

- Die Entwickler des Systems
- Der Kunde
- Die Betreuer dieses Studienprojekts
- Die Gutachter des Reviews
- Personen, die dieses Projekt später weiterentwickeln, erweitern oder warten

## 1.3 Namenskonventionen für dieses Dokument

- Begriffe, die Referenzen auf das Begriffslexikon darstellen, werden *kursiv* geschrieben.
- Besonders wichtige Informationen oder hervorzuhebende Teile werden **fett** geschrieben.
- Verweise auf externe Informationen werden als Fußnoten dargestellt.

# Kapitel 2

# Powerwall

Bei der *Powerwall* um die es in erster Linie in diesem Projekt geht, handelt es sich um die *Powerwall* des *VISUS*. Diese befindet sich im Gebäude des *VISUS* im Keller.

## 2.1 Aufbau der *Powerwall*

"Die Projektion erfolgt durch zehn 4K-Projektoren, die in Kinos zum Einsatz kommen. Jeder dieser Projektoren hat eine Auflösung von 4.096 x 2.400 Pixeln.

Die Geräte sind hochkant aufgestellt. Die paarweise Verwendung ermöglicht die 3D-Darstellung von Visualisierungen. Das Gesamtbild wird in 5 Streifen mit 4 Überblendbereichen erzeugt. Damit ergibt sich eine Gesamtauflösung von 10.800 x 4.096 Pixeln.<sup>1</sup>

Dies hat als Folge, dass eine Anwendung die auf der *Powerwall* visualisiert werden soll fünf

<sup>1</sup><http://www.visus.uni-stuttgart.de/institut/visualisierungslabor/technischer-aufbau.html>

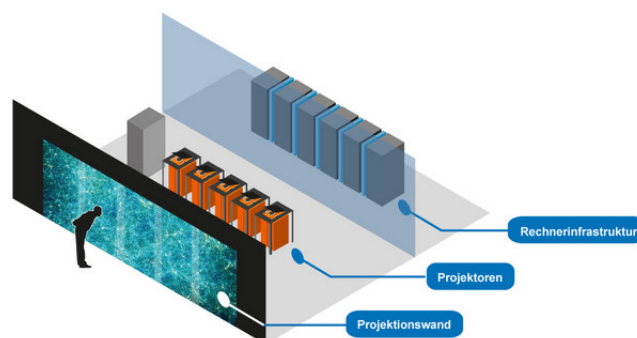


Abbildung 2.1: Technischer Aufbau der Powerwall

solche 4K-Projektoren ansteuern muss. Dies kann keine verfügbare Grafikkarte leisten. Aus diesem Grund steht im Hintergrund ein *Cluster*, das die Berechnung der Bilder übernimmt.

## 2.2 *Operator-PC*

Der *Operator-PC* steht im hinteren Bereich des Präsentations-Raumes. Er verfügt über zwei hochkant aufgestellte Monitore mit jeweils 1440 x 2560 Bildpunkte. Diese beiden Monitore sind als erweiterter Bildschirm konfiguriert. Damit hat der *Operator-PC* eine Auflösung von 2880 x 2560 Bildpunkte.

Der *Operator-PC* verfügt über normale Eingabemöglichkeiten (Tastatur, Maus).

## 2.3 Netzwerk

Das Grafik-*Cluster* und der *Operator-PC* sind über ein Gigabit-Netzwerk (1Gb/s) verbunden. Beide liegen im normalen Subnetz des *VISUS* und können daher auf unseren Server<sup>2</sup> zugreifen. Die Datentransferrate von 1Gb/s reicht theoretisch aus um ca. 125MB/s zu transportieren.

## 2.4 Zugriffsmöglichkeiten auf das Grafik-*Cluster*

Auf den Knoten des *Clusters* läuft Windows 7, 64Bit. Sie sind über die Power-Shell und über TeamViewer<sup>3</sup> erreichbar. Die Knoten heißen keshiki01-keshiki10. Wobei die ersten fünf Knoten für die ersten fünf Bild-Streifen zuständig sind. Die Knoten 06-10 sind für die 3D-Darstellung notwendig.

---

<sup>2</sup>acid.visus.uni-stuttgart.de

<sup>3</sup><http://www.teamviewer.com>



# Kapitel 3

## Layout

Das *Layout* beschreibt die Anordnung der *Kacheln* und *Organisationseinheiten* in der Visualisierung.

Hierbei haben wir uns auf ein einfaches TreeMap-Layout festgelegt, indem die hierarchische Struktur gut abgebildet werden kann. Innerhalb der einzelnen *Organisationseinheiten* können weiter *Organisationseinheiten* und *Kacheln* beliebig angeordnet werden. Diese Anordnung kann in sog. *Layouts* gespeichert werden.

### 3.1 Einschränkungen

Die vertikale Ausdehnung der TreeMap ist durch die Höhe der Anzeige beschränkt. Es wird kein vertikales Scrollen angeboten. Wenn zu wenig Platz auf der Darstellung vorhanden ist, so wird die horizontale Ausdehnung vergrößert.

### 3.2 Layout-Idee

*Organisationseinheiten* nehmen grundsätzlich immer die gesamte Höhe ein. Sie werden gemäß des TreeMap-Algorithmus ineinander geschachtelt.

Innerhalb einer jeden *Organisationseinheit* besteht ein spaltenorientiertes Layout.

Jede *Kachel* hat als Breite immer ein ganzzahliges Vielfaches der Kachelbreite in *Level S*. Diese Breite berechnet sich aus der Anzahl der angezeigten Buchstaben im Namen der *Kachel*, der

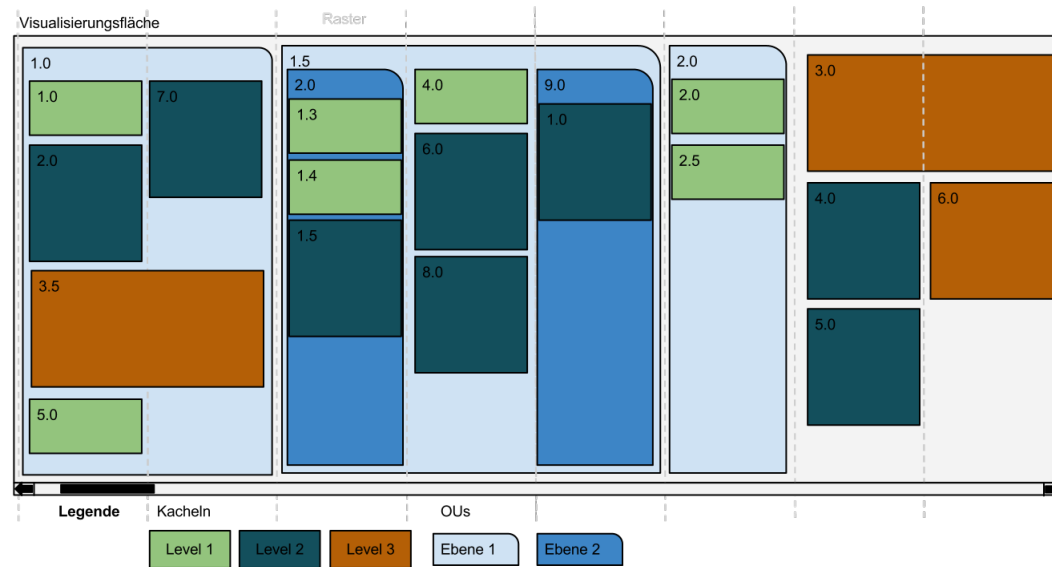


Abbildung 3.1: Layout-Konzept: Alle Elemente haben einen dezimalen Wert über den in ihrer übergeordneten Struktur eine eindeutige vertikale Anordnung bestimmt werden kann

Schriftart, und -größe sowie der Ränder um die *Kachel*.

*Level S* - und *Level M* - *Kächeln* sind gleich breit. *Level M* ist höher als *Level S*. *Level L Kächeln* bleiben verändern ihre Höhe gegenüber *Level M* nicht mehr. Jedoch nehmen Sie in die Breite ein ganzzahliges Vielfaches der alten Breite ein um alle Visualisierungsplugins zu visualisieren.

In einer *Organisationseinheit* werden die *Kacheln* dann vertikal angeordnet. Das bedeutet, dass man von der obersten linken Ecke aus beginnend alle enthaltenden Elemente (*Kacheln* und *Organisationseinheiten*) durchnummerieren könnte. Um diese Anordnung kümmert sich unser Layout-Algorithmus.

Für die Reihenfolge der Anordnung erhält jedes Element (*Kachel* oder *Organisationseinheit*) einen dezimalen Wert. Dieser bestimmt in aufsteigender Sortierung innerhalb der übergeordneten *Organisationseinheit* die Anordnung auf der Oberfläche.

Beim Verschieben eines Elementes auf die Position  $i$  wird dieser dezimale Wert auf

$$\frac{v_{i-1} + v_i}{2}$$

gesetzt ( $v_k$  bestimmt den Wert der  $k$ -ten *Kachel* auf der aktuellen Ebene).

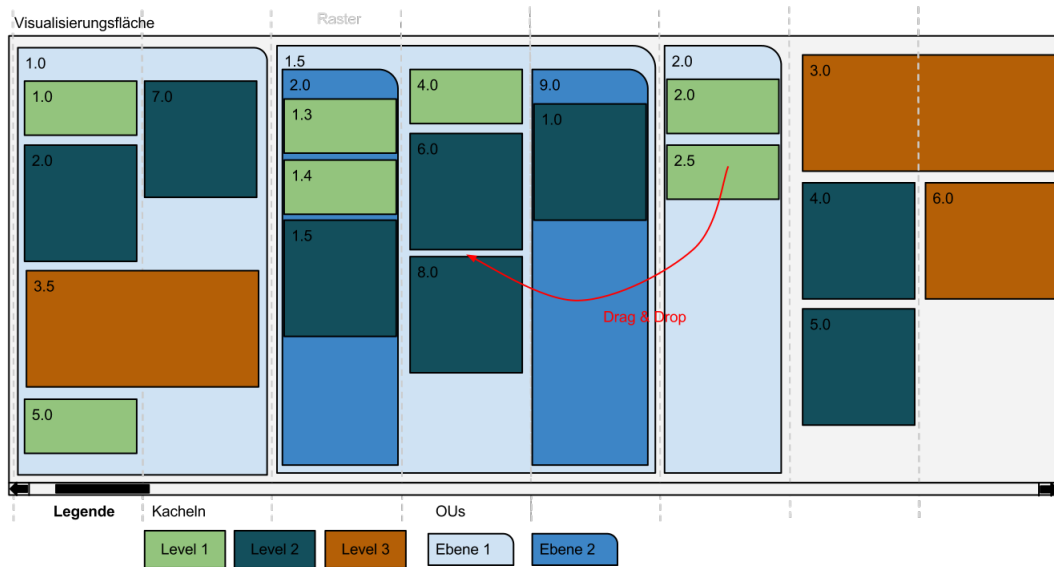


Abbildung 3.2: Drag & Drop: Die Kachel soll von rechts nach links, vor 8.0 verschoben werden

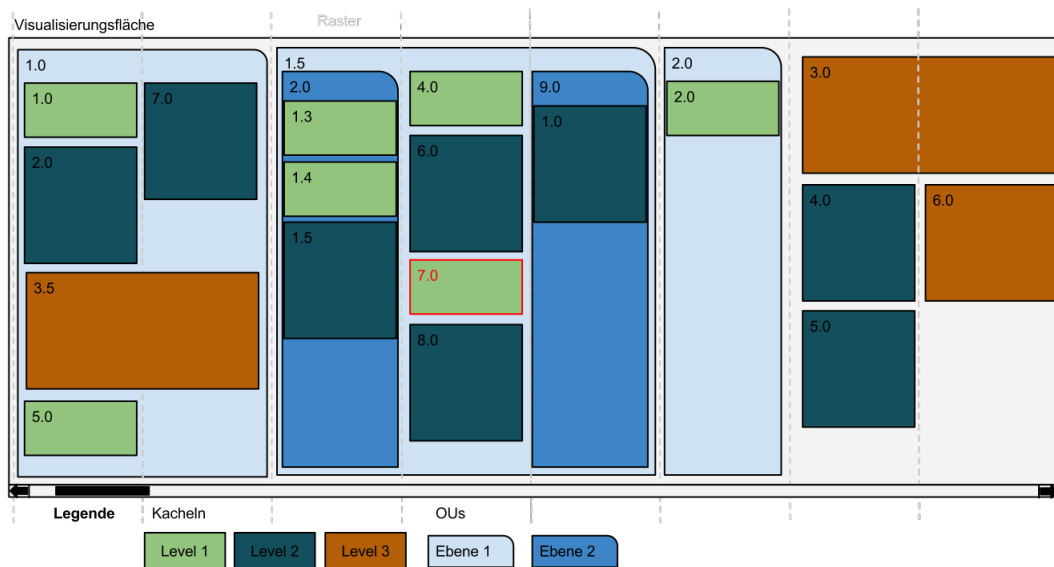


Abbildung 3.3: Drag & Drop: Die verschobene Kachel hat den neuen Wert 7.0 erhalten

# Kapitel 4

## Konzeption

### 4.1 Interaktion

Die Interaktion mit der *Powerwall* ist nur über den *Operator-PC* möglich. Veränderungen am Status der Visualisierung (z.B. Kacheln vergrößern oder verschieben, Modifizieren von Einstellungen, etc.) findet nur dort über die gegebene Tastatur und die Maus statt.

Daraus lässt sich unser grundlegendes Konzept ableiten: Es werden nur auf dem *Operator-PC* durchgeführte Status-Änderungen an die *Powerwalls* weitergeleitet.

Auf den *Powerwall*-Knoten läuft unser normaler *Desktop-Client*, wobei über eine Konfigurations-Datei festgelegt wird, dass es sich hierbei um eine *Powerwall*-Visualisierung handelt.

### 4.2 Einstellungen

Um einen reibungslosen Ablauf zu gewährleisten müssen einige besonderen Einstellungen vorgenommen werden.

Um die Einstellungen global (auf dem *Operator-PC* und auf den *Powerwalls*) gleich zu behalten können alle folgenden Einstellungen auch für alle Systeme gleichermaßen vorgenommen werden. Die Anwendung muss dann jedoch mit einem Kommandozeilenparameter gestartet werden. Dazu wird die Anwendung mit `MISD.Client.exe operator` (um sie als Operator auszuführen) oder `MISD.Client.exe powerwall <IP-Adresse> <x-offset> <y-offset>` (um sie als *Powerwall* des *Operator-PCs* mit der IP-Adresse <IP-Adresse> und dem angegebenen Offset

auszuführen) ausgeführt.

### 4.2.1 Einstellungen auf dem Operator-PC

Auf dem Operator-PC wird in der Konfigurations-Datei festgelegt, dass es sich bei diesem Gerät um den Operator-PC (`IsOperator=true`) handelt. Zusätzlich muss dann noch die Gesamtauflösung der zu steuernden *Powerwall* angegeben werden (`PowerwallResX=10800` und `PowerwallResY=4096`). Diese Größe wird für die Skalierung der Ansicht verwendet.

Der *Operator-PC* muss über eine festgelegte IP-Adresse verfügen. Diese darf sich nicht verändern, da diese auf den *Powerwall*-Instanzen konfiguriert wird.

### 4.2.2 Einstellungen auf den Powerwall-Instanzen

Die Konfigurations-Datei der *Powerwall*-Instanzen muss das Attribut `IsPowerwall=true` setzen. Damit wird der *Powerwall*-Modus aktiviert.

Zusätzlich muss die IP-Adresse des *Operator-PCs* bei den *Powerwalls* angegeben werden (`OperatorIP="192.168.178.12"`). Diese wird für die Übertragung der Status-Änderungen benötigt.

Darüber hinaus muss der horizontale Offset der Anzeige definiert werden (`PowerwallOffsetX=0`). Dieser beschreibt den Offset zwischen dem lokalen Bild (dem Streifen, den dieser Beamer produziert) zum gesamten Bild.

## 4.3 Status-Änderungen

Die relevanten Status der Anwendung ergeben sich aus folgenden Bereichen:

- User-Spezifischen Visualisierungs-Einstellungen (Schriftgröße, Schriftart, *Plugin*-Prioritäten-Liste, ...)
- Status-Änderungen an *Organisationseinheiten* (auf- und zuklappen)
- Status-Änderungen an *Kacheln*

Da es sich im Gesamten lediglich um Visualisierungs-Eigenschaften handelt müssen keine Daten vom *Operator-PC* an die *Powerwalls* übertragen werden. Die eigentliche Daten-Akquise der *Clients* läuft selbstständig von den *Cluster-Knoten* zum *Server* ab.

### 4.3.1 Akquise der Änderungen

Die relevanten Status-Änderungen (siehe oben) werden aktiv von den *Kacheln* oder *Organisationseinheiten* gehandhabt. Sobald eine *Kachel* oder eine *Organisationseinheit* ihren Status ändert informiert sie den *LayoutManager* über diese Statusänderung. Dazu stellt dieser die Methode *SetMSState* und *SetOState* zur Verfügung.

Der *LayoutManager* übergibt die Änderungen an das aktuelle *Layout* und informiert alle registrierten *EventHandler*, dass Änderungen am *Layout* stattgefunden haben.

Änderungen in der Anordnung (z.B. Drag & Drop von *Kacheln*) werden auf gleichem Wege an das *Layout* übergeben. Der einzige Unterschied ist, dass hierbei das Drag & Drop Event diese Änderung auslöst. Für das Ändern der Reihenfolge stellt die Klasse *LayoutManager* die Methode *MoveBefore*, *MoveBetween* und *MoveAfter* zur Verfügung. Über diese werden die Sortier-Werte der entsprechenden Elemente dann gesetzt.

### 4.3.2 Verarbeitung von Änderungen

Die *TreeView*, welches sich um das Layouten der *Kacheln* und *Organisationseinheiten* kümmert stellt diese durch ein Binding auf das *DataModel* dar. Dabei handelt es sich immer um die Darstellung von Elementen des Types *TilableElement*. Diese Klasse besitzt zwei Spezialisierungen *MonitoredSystem* und *OrganisationalUnit*. Für das Layout und die Powerwall ist das Attribut *SortingProperty* von Bedeutung. In ihm wird die Reihenfolge der *Kacheln* gespeichert.

Bei der Darstellung der *ObservableCollection* wird eine sog. *SortDescriptions*<sup>1</sup> verwendet. Dazu wird ein eigener *Converter* geschrieben, der eine Sortierung von Elementen aufgrund der *SortingProperty* unterstützt<sup>2</sup>.

---

<sup>1</sup><http://msdn.microsoft.com/en-us/library/system.componentmodel.sortdescription.aspx>

<sup>2</sup><http://stackoverflow.com/questions/5722835/how-to-sort-treeview-items-using-sortdescriptions-in-xam>

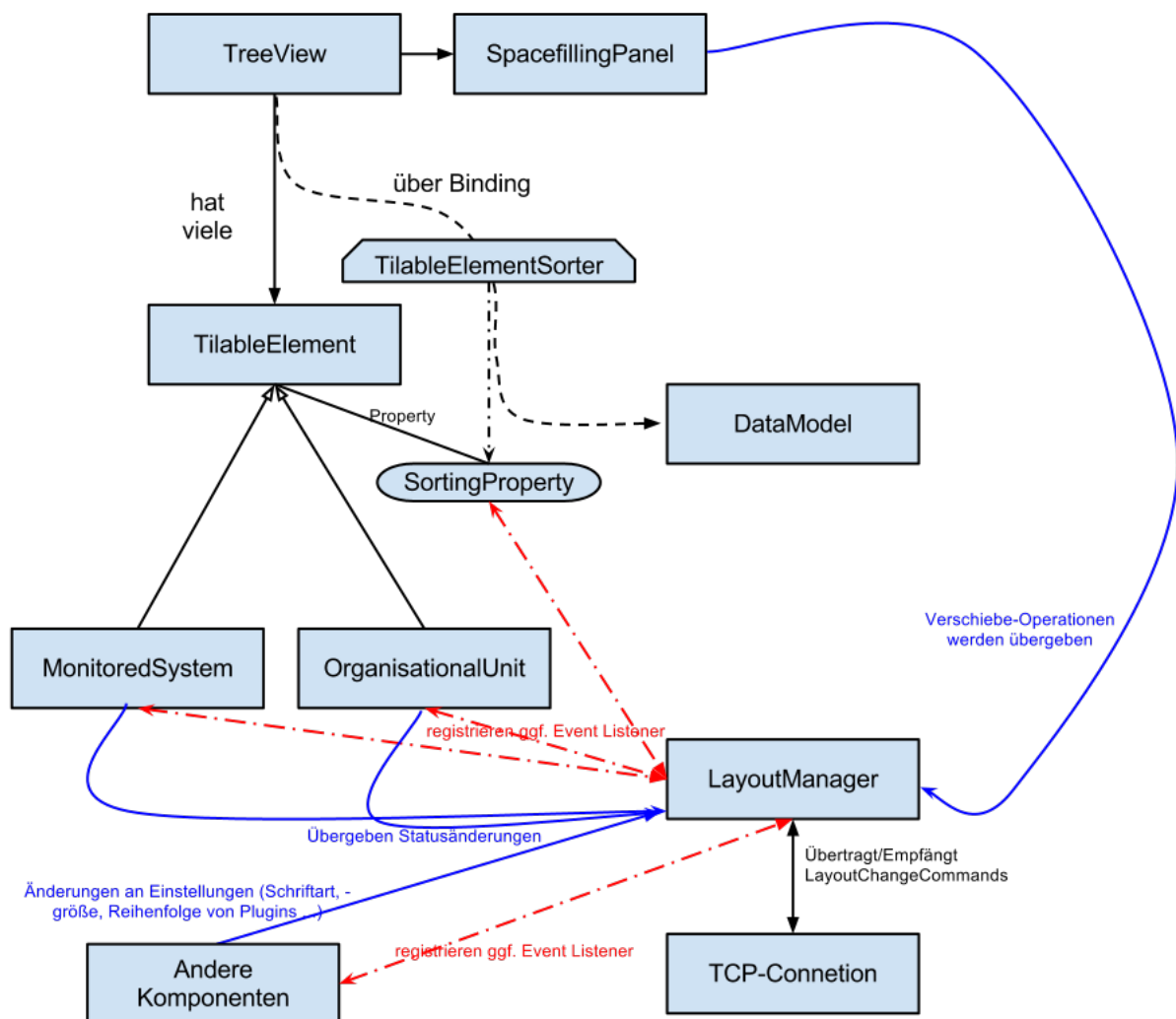


Abbildung 4.1: Konzept: Die Struktur und der Datenfluss beim generieren des Layouts

### 4.3.3 Übertragung der Änderungen

Der `LayoutManager` nimmt auch bei der Übertragung der Änderungen zu der *Powerwall* eine zentrale Rolle ein.

Wenn es über die Einstellungen gewünscht ist, dann erstellt dieser einen TCP-Server, der es ermöglicht, dass sich die *Powerwall-Clients* mit diesem verbinden. Sobald eine Änderung stattgefunden hat, wird diese Änderung in eine `LayoutChangeCommand`-Instanz verpackt und über den TCP-Server an alle verbundenen *Clients* gesendet.

In regelmäßigen Abschnitten (alle 50 übertragenen `LayoutChangeCommands`) wird außerdem ein komplettes Abbild des aktuellen Layouts auf dem *Operator-PC* an die *Powerwall-Clients* gesendet um zu verhindern, dass der Status der Anzeigen divergiert.

Der `LayoutManager` auf den *Powerwall-Clients* erhält über die TCP-Schnittstelle die `LayoutChangeCommands` und führt diese auf sein aktuelles geladenes `Layout` aus.

Dadurch können alle Änderungen übertragen werden, ohne viel Datentransfer zu erzeugen.

Die betroffenen Oberflächen-Elemente (insb. `TilableElement` und die Eigenschaft `SortingProperty` sowie die `MonitoredSystems` und die `OrganisationalUnits`) registrieren sich selbst `Event-Handler` auf die verschiedenen Veränderungen die im `LayoutManager` auftreten können, wodurch sie auf der *Powerwall* dann alle Änderungen am Layout erhalten, die auch auf dem *Operator-PC* durchgeführt wurden.

## 4.4 Operator-PC Anzeige

Auf dem *Operator-PC* steht uns eine Auflösung von 2.880 x 2.560 Bildpunkten zur Verfügung. Auf der *Powerwall* hingegen eine Auflösung von 10.800 x 4.096 Bildpunkten.

Auf dem *Operator-PC* wird nach unserer Konzept-Idee nahezu der gleiche Bildschirm Inhalt zu sehen sein, wie auf der *Powerwall*.

Dazu wird die Anzeige auf dem *Operator-PC* um den Faktor 0.5 skaliert, sodass der gesamte Bildausschnitt der *Powerwall* auf dem *Operator-PC* noch 5.400 x 2.048 Bildpunkte groß ist. Da auf der *Powerwall* keine Menüleisten und keine Fensterdekorationen angezeigt werden, können wir also auf den beiden Monitoren des *Operator-PCs* mehr als die Hälfte des Inhaltes auf der *Powerwall* darstellen. Wie geplant, wird es hier nur ein scrollen in horizontale Richtung geben.



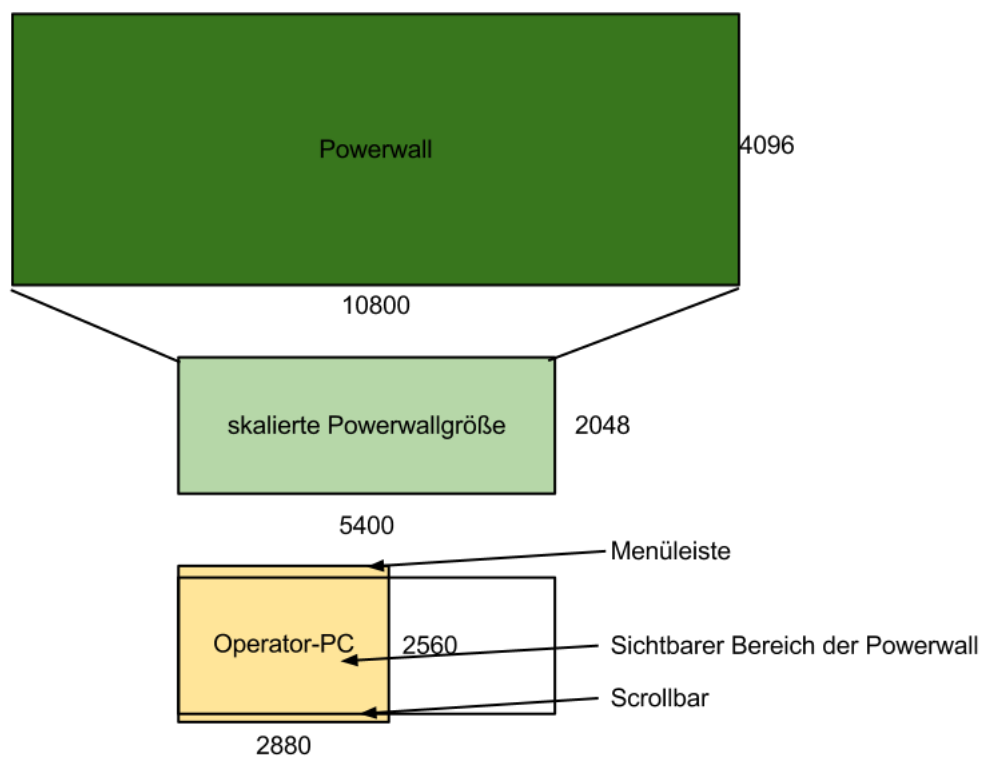


Abbildung 4.2: Darstellung: Skalierung und Anzeigen des Ausschnittes auf dem Operator-PC

# Anhang A

## Anhang

### A.1 Begriffslexikon

<b>Begriff</b>	<b>Client</b>
<b>Bedeutung</b>	Der Client ist ein Desktop oder eine Powerwall, welcher auf den Webservice der MISD-Software zugreift und die graphische Aufbereitung der Informationen anzeigt.
<b>Abgrenzung</b>	Ein Client kann gleichzeitig eine Workstation sein, muss das aber nicht.
<b>Gültigkeit</b>	Bei der Entwicklung, Nutzung und Erweiterung der MISD-Software.
<b>Bezeichnung</b>	Ein Client ist durch den Benutzer und den Rechner, von dem aus er benutzt wird, eindeutig gekennzeichnet.
<b>Querverweise</b>	Desktop, Powerwall, Server, Workstation
<b>Seiten</b>	11, 13, 15

<b>Begriff</b>	<b>Cluster</b>
<b>Bedeutung</b>	Als Cluster wird eine Gruppe von zu überwachenden Rechnern bezeichnet, die entweder mit Hilfe eines Cluster-Managers wie dem HPC Cluster Manager oder dem Bright Cluster Manager betrieben wird.
<b>Abgrenzung</b>	Eine Workstation, die nicht über eine Cluster-Management-Lösung überwacht wird, gehört nicht zu einem Cluster.
<b>Gültigkeit</b>	Bei der Entwicklung, Nutzung und Erweiterung der MISC-Software.
<b>Bezeichnung</b>	Jedes Cluster ist mit seinen Elementen in der Datenbank eindeutig benannt und gespeichert.
<b>Querverweise</b>	Workstation, Client, zu überwachender Rechner
<b>Seiten</b>	1, 7, 13

<b>Begriff</b>	<b>Desktop</b>
<b>Bedeutung</b>	Der Desktop ist ein Desktop-PC, welcher auf den Webservice der MISC-Software zugreift und die graphische Aufbereitung der Informationen anzeigt. Der Überbegriff ist Client.
<b>Abgrenzung</b>	Ein Desktop ist keine Powerwall. Ein Desktop kann gleichzeitig eine Workstation sein, muss das aber nicht.
<b>Gültigkeit</b>	Bei der Entwicklung, Nutzung und Erweiterung der MISC-Software.
<b>Bezeichnung</b>	Ein Desktop ist durch den Benutzer und den Rechner, von dem aus er benutzt wird, eindeutig gekennzeichnet.
<b>Querverweise</b>	Client, Powerwall, Server, Workstation
<b>Seiten</b>	4, 11

<b>Begriff</b>	<b>Kachel</b>
<b>Bedeutung</b>	Eine Kachel bezeichnet die grafische Repräsentation eines einzelnen zu überwachenden Rechners. Diese stellt das System in einem rechteckigen Rahmen in verschiedenen Detailstufen dar.
<b>Abgrenzung</b>	Eine Kachel repräsentiert nur einen Rechner, entspricht diesem jedoch nicht. Interaktion mit einer Kachel hat keine Einfluss auf den repräsentierten Rechner.
<b>Gültigkeit</b>	Bei der Entwicklung, Nutzung und Erweiterung der MISD Software.
<b>Querverweise</b>	Cluster, Workstation, Visualisierungsplugin
<b>Seiten</b>	8, 9, 12, 13

<b>Begriff</b>	<b>Layout</b>
<b>Bedeutung</b>	Ein Layout ist die Anordnung der Kacheln und Organisationseinheiten auf der Oberfläche sowie der Level der einzelnen Kacheln. Derartige Layouts können gespeichert und wieder geladen werden. Dadurch können Nutzungsmuster wiederholt angewendet werden, ohne erneut konfigurieren zu müssen.
<b>Abgrenzung</b>	Zum Layout gehören nicht die abgebildeten Farben der Kacheln und auch nicht die dargestellten Werte.
<b>Gültigkeit</b>	Bei der Entwicklung, Nutzung und Erweiterung der MISD Software.
<b>Querverweise</b>	Kachel, Organisationseinheiten
<b>Seiten</b>	8

<b>Begriff</b>	<b>Level</b>
<b>Bedeutung</b>	Ein Level ist eine Detailtiefe der Repräsentation eines zu überwachenden Rechners in der Benutzeransicht der MISD-Software. Es werden 3 Level zur Verfügung gestellt. Je nach Höhe des Level, werden mehr oder weniger Informationen auf einer Kachel angezeigt.
<b>Abgrenzung</b>	Die Level beeinflussen nicht die Daten der Software. Sie sind rein graphische Elemente.
<b>Gültigkeit</b>	Bei der Entwicklung, Nutzung und Erweiterung der MISD Software.
<b>Bezeichnung</b>	Die Level werden so entworfen, das ein späteres Einfügen von weiteren Leveln möglich ist.
<b>Seiten</b>	8, 9

<b>Begriff</b>	<b>MISD OWL</b>
<b>Bedeutung</b>	Master Infrastructure Situation Display-Observing Windows and Linux
<b>Abgrenzung</b>	MISD-OWL bezeichnet in diesem Dokument ausschließlich die im Studienprojekt 2012 der Universität Stuttgart entstandene Software.
<b>Gültigkeit</b>	Bei der Entwicklung, Nutzung und Erweiterung der MISD Software.
<b>Bezeichnung</b>	Der Name MISD-OWL ist im Studienprojekt festgelegt.
<b>Querverweise</b>	System
<b>Seiten</b>	4, 21

<b>Begriff</b>	<b>Operator-PC</b>
<b>Bedeutung</b>	Der Operator-PC bedient die Powerwall. Es handelt sich dabei um einen speziellen Computer, der im Raum der Powerwall steht und über den die Interaktion mit der Powerwall stattfindet. Auf ihm werden Menüleisten angezeigt. Auf der Powerwall nicht.
<b>Querverweise</b>	Powerwall, Client
<b>Seiten</b>	1, 4, 7, 11–13, 15

<b>Begriff</b>	<b>Organisationseinheit</b>
<b>Bedeutung</b>	Eine Organisationseinheit ist eine Menge von Workstations, die einen technischen, lokalen, oder ähnlichen Zusammenhang haben. Jede Workstation kann Mitglied von genau einer Organisationseinheit sein. Die Organisationseinheiten sind entlang einer Hierarchie ineinander geschachtelt.
<b>Abgrenzung</b>	Eine Organisationseinheit muss nicht der Hierarchie des Active Directory entsprechen. Das Active Directory wird nur zur Initialisierung der zu überwachenden Rechner verwendet.
<b>Gültigkeit</b>	Bei der Entwicklung, Nutzung und Erweiterung der MISD Software.
<b>Bezeichnung</b>	Jede Organisationseinheit muss über einen Namen oder eine ID eindeutig identifizierbar sein.
<b>Querverweise</b>	Workstation, Active Directory
<b>Seiten</b>	8, 9, 12, 13

<b>Begriff</b>	<b>Plugin</b>
<b>Bedeutung</b>	Ein Plugin realisiert die Überwachungsfunktionalität für einen speziellen Bereich. Es kann aus den verschiedenen Datenerfassungsmodulen und einem Visualisierungsmodul bestehen. Zusätzlich wird dem Plugin ein Datensatz zugeordnet. Für das CPU-Plugin wären das alle Kenngrößen, Metriken und Standardwerte.
<b>Abgrenzung</b>	Ein Plugin besteht aus Kenngrößen, kann in Einzelfällen aber auch nur eine Kenngröße enthalten.
<b>Gültigkeit</b>	Bei der Entwicklung, Nutzung und Erweiterung der MISD Software.
<b>Bezeichnung</b>	Jedes Plugin ist in der Datenbank eindeutig benannt.
<b>Seiten</b>	12

<b>Begriff</b>	<b>Powerwall</b>
<b>Bedeutung</b>	Eine Powerwall bezeichnet ein besonders hochauflösendes Display, auf welchem aufgrund der sehr hohen Auflösung sehr viele Informationen gleichzeitig angezeigt werden können. Der Überbegriff ist Client.
<b>Abgrenzung</b>	Im Falle von MISD wird die Anwendung auf die gegebenen Powerwalls angepasst. Eine Powerwall ist kein Desktop.
<b>Gültigkeit</b>	Bei der Entwicklung, Nutzung und Erweiterung der MISD Software.
<b>Querverweise</b>	Client
<b>Seiten</b>	1, 4, 6, 11–13, 15

<b>Begriff</b>	<b>Server</b>
<b>Bedeutung</b>	Der Server ist das zentrale Element der MISD Software. Auf ihm laufen die Webservices, werden die Daten der Workstations und Cluster gesammelt und in einer Datenbank gespeichert. Der Server stellt außerdem Daten für die Benutzerschnittstelle zur Verfügung.
<b>Abgrenzung</b>	Der Server bezieht sich immer auf den Server der MISD Software.
<b>Gültigkeit</b>	Bei der Entwicklung, Nutzung und Erweiterung der MISD Software.
<b>Bezeichnung</b>	Der Server wird bei der Kommunikation mit den Workstations eindeutig erkennbar und zertifiziert sein.
<b>Querverweise</b>	<i>MISD OWL</i>
<b>Seiten</b>	13

<b>Begriff</b>	<b>VISUS</b>
<b>Bedeutung</b>	Visualisierungs Institut der Universität Stuttgart
<b>Querverweise</b>	MISD
<b>Seiten</b>	6, 7

<b>Begriff</b>	<b>zu überwachender Rechner</b>
<b>Bedeutung</b>	Die zu überwachenden Rechner sind die Gesamtmenge der über das System erfassten Geräte. Dazu zählen Workstations und Cluster-Nodes.
<b>Abgrenzung</b>	Ein zu überwachender PC muss kein Client sein, kann dies jedoch. Ein PC, der nicht durch einen Dienst vom System überwacht wird, fällt nicht in diese Kategorie.
<b>Gültigkeit</b>	Bei der Entwicklung, Nutzung und Erweiterung der MISD Software.
<b>Bezeichnung</b>	Alle zu überwachenden PCs sind in der Datenbank eindeutig benannt und gespeichert.
<b>Querverweise</b>	Cluster, Workstation
<b>Seiten</b>	4

## A.2 Versionshistorie

### Version 0.1

<b>Datum</b>	02.02.2013
<b>Änderungen</b>	Initiale Version
<b>Bearbeiter</b>	Sebastian Zillesen