



## Anleitung: install\_ga-server-11.4 (32-Bit)

Stand 2021-03-22

Asus C:\\_ABLAGEN\AMEV-GA-Plattform\html\doc\GA-Server-11.4\_A12.docx

Überblick.....	3
Allgemeines.....	4
BIOS .....	4
<b>1 Suse 11.4 installieren.....</b>	<b>5</b>
1.1a Partition Standard .....	5
1.1b Partition für späteres Software-Raid1 (bei Bedarf).....	5
<b>1.2 Neuen Benutzer erstellen.....</b>	<b>6</b>
<b>2 Installation GA-Server.....</b>	<b>7</b>
2.1 Installation GA-Server via USB oder DVD.....	7
3.1 Desktop: Schaltflächen zeilenweise anordnen.....	8
3.2 Panel: Einige Nettigkeiten zu Prozessor, Netz, Festplatte, Klebezettel.....	8
3.3 pgAdmin III: Der PostgreSQL Administrator .....	9
3.4 Die Anwendung jLZHview: eine Desktop Visualisierung.....	10
3.5 Die Anwendung jLZHweb: eine Browser Visualisierung .....	10
3.6 Der Kommunikations-Prozess .....	10
<b>3.7 YaST: Software .....</b>	<b>11</b>
<b>3.8 YaST: Netzwerkgeräte.....</b>	<b>12</b>
3.8.0 Die Netzwerkadresse <i>localhost</i> .....	12
3.8.2 YaST: Netzwerk in einem 256-er Netz .....	12
3.8.3 YaST: Netzwerk in einem 8-er Netz .....	13
3.9 YaST: Netzwerkdienste.....	13
3.9.1 YaST: NTP-Einrichtung.....	13
3.10 <i>SSH, NX</i> : Fernzugriff mit einem Windows Rechner .....	14
3.11 Drucker installieren via YaST .....	14
3.12 Linux Passwort vergessen (c't 2015, Heft 5).....	14
4 Weitere Installationen von Suse .....	15
4.1 SLES 11.3 (SuseLinuxEnterpriseServer).....	15
4.1.1 SLES 11.3 Installation .....	15
4.1.2 SLES 11.3 GA-Server Installation .....	15
4.2 openSUSE 13.2 (Wisch- & Tupf-Technik) .....	16
4.2.1 openSUSE 13.2 Installation .....	16
4.2.2 openSUSE 13.2 GA-Server Installation.....	17
<b>4.3 System kopieren auf eine weitere Festplatte (via Life System).....</b>	<b>17</b>
<b>5 GA-Server parametrieren mit einem Objekt.....</b>	<b>18</b>
5.1.0 GA-Server installieren.....	18
5.1.1 Das Installationsverzeichnis <i>/home/lzh/gak_import/</i> .....	18
5.1.2 Die Parameter-Dateien .....	18
5.1.3 Die Basis-Datenbank-Tabellen .....	19
5.1.4 Der Datenbank-Import aus dem Installationsverzeichnis .....	19
<b>5.1.5 Erzeugen eines GA-Knoten aus dem Installationsverzeichnis.....</b>	<b>19</b>
<b>5.1.6 Wechseln zum GA-Knoten Benutzer.....</b>	<b>19</b>
5.1.7 Das ‚Jila Fenster‘ des GA-Knotens .....	19
5.1.8 Die Parameter-Dateien im GA-Knoten.....	20
5.1.9 Die serielle Schnittstelle <i>RS232</i> .....	20
A Anhang: <i>AMEV-GA-Plattform.de</i> .....	21
<b>B Anhang Linux Nomenklatur.....</b>	<b>22</b>
Allgemeines.....	22

Terminal öffnen.....	22
Die Bash: der Kommandozeileninterpreter.....	22
Pipes            von Befehlen.....	23
Netz.....	23
Windows.....	23
Zeiten.....	23
Sonstiges.....	23
Passwörter, Rechte, Benutzer.....	24
Routinen und Programme.....	24
Routinen und Programme und Skripte in Verzeichnissen.....	24
SecureShell.....	25
Dienste.....	25
Systemmeldungen Verzeichnisse.....	26
Systemroutinen.....	26
Prozesse beenden.....	26
PostgreSQL läuft nicht.....	26
PostgreSQL Datenbank Werkzeuge.....	26
PostgreSQL pg_hba.conf.....	28
Ports.....	28
Festplatten bearbeiten.....	28
Numerical Permissions.....	28
Festplattenkopie erstellen.....	29
Kopieren irgendwo hin, verpacken und wieder entpacken.....	29
Partition verpacken.....	29
<b>C Anhang: ITX Boards als GA-Server in Betrieb.....</b>	<b>30</b>
C.1.1 Mainboards für 32Bit mit 2/4 Kernen und maximal 4GB Memory.....	30
C.1.2 Gehäuse.....	30
C.1.3 Memory.....	30
C.1.4 Mini-Disk.....	30
C.1.5 Was kostet ein kompletter GA-Server?.....	30
D Anhang: NonRaid zu (software)Raid1 SuSE 10 1.....	31
E Anhang: SQL-Abfragen.....	32
E1.0 SQL-Import in die PostgreSQL-Datenbank.....	32
<b>F Anhang: Netzwerk und Zeitserver.....</b>	<b>34</b>
F1.1 Netzwerk Installation mit 1 Netzwerkkarte.....	34
F1.2 Netzwerk Installation mit 2 Netzwerkkarten.....	35
F1.3 Netzwerk Installation mit 2 Netzwerkkarten (Beispiel AMEV_BAcnet).....	36
F1.4 Zeitserver NTP.....	37
<b>G Anhang: Tomcat6 Installation.....</b>	<b>38</b>
<b>G1.0 tomcat6 Installation via Skript.....</b>	<b>38</b>
<b>G2.0 tomcat6 Installation händisch.....</b>	<b>39</b>
G2.1 tomcat6 Installation zu Fuß.....	39
G2.2 tomcat-users.xml: vorbereitete Datei kopieren.....	39
G2.3 tomcat-users.xml: Datei manuell bearbeiten.....	39
G2.4 tomcat6 starten.....	39
G2.5 Tomcat Manager via Firefox öffnen.....	39
G3.0 jLZHweb Installation der .war Datei.....	40
G3.1 jLZHweb Startseite einrichten.....	40
G3.2 jLZHweb Anwendung starten.....	41
G4.0 tomcat6 Bedienung und Einstellungen.....	41
G4.1 tomcat6 Einstellung via YaST.....	41
G5.0 Update jLZHweb.war.....	41
G6.0 Nachinstallation von Tomcat6.....	42
G6.1 Software Repository.....	42
G6.2 Tomcat6 installieren oder löschen.....	42
H Anhang: GA-Plattform-parametrieren.....	43
H7.0 GA-Knoten parametrieren.....	43
H7.1 Die Anwendung <i>GA-Plattform_nn.ods</i> .....	43
H7.2 Eine kurze Einführung als Überblick.....	44
I Anhang: Schemata.....	45
I8.0 Schemata erstellen.....	45
X Anhang: Melden Sie sich, falls.....	45

## Überblick

Im Wesentlichen gibt es drei Hürden zu absolvieren.

### Die Hürde mit der Hardware

Jeder handelsübliche PC (32Bit mit 2GB Memory, 200GB Disk etc.) kann verwendet werden. Das sind Hardware Boards mit RS232 oder LAN Schnittstelle als Verbindung zu den Schaltschränken diverser MSR-Hersteller und evtl. einer LAN Schnittstelle ins Internet bzw. Intranet als Verbindung zu einer Zentralen LeitTechnik.

### Die Hürde mit der EDV

Das ist der Download und die Installation der Software der *AMEV-GA-Plattform.de* auf einem 32Bit Rechner mit dem sehr stabilen Betriebssystem *openSUSE 11.4*.

Das lizenzkostenfreie Paket GA-Server Installation besteht aus der Leitzentrale (mit *PostgreSQL* Datenbank *LZH*, der Visualisierung *jLZHview*<sup>1</sup>, *jLZHweb* etc.) und/oder einem bzw. mehreren noch zu aktivierenden GA-Knoten, sowie die Tabellenkalkulation *LibreOffice*. Diese EDV-Hürde überspringt auch ein Nichtexperte in der EDV, wenn die in diesem Text vorgeschlagenen Komponenten und Anweisungen verwendet werden.

Geht aber doch etwas daneben, so wird die Installation einfach wiederholt.

### Die Hürde mit der MSR

Die Belegung der AMEV-GA-Plattform erfolgt via *Referenz-Datei*, das ist eine Zuordnungsliste (ähnlich der *BACnet* CSV-Dateien *EDE*, *StateText*, *Units*)<sup>2</sup> und dient für den Import der Datenpunkte des jeweiligen MSR-Fabrikates<sup>3</sup>.

### GA-Plattform parametrieren

Zur kompletten Parametrierung einer GA-Plattform können jetzt mit dem Werkzeug

*GA-Plattform\_nn.ods* (enthält Makros) alle erforderlichen Dateien erstellt werden.

Die Software wurde entwickelt und getestet unter

- Linux 32 Bit:       OpenSuse 11.4       LibreOffice 3.3.2
- Linux 64 Bit:       Ubuntu 18.4       LibreOffice 6.0.7.3
- Windows 32 Bit:   Windows 7        OpenOffice 4.1.6

Die Umsetzung der Datenpunkte einer MSR-Liste in die Form einer Referenz-Datei, erfolgt via Tabellenkalkulation, die Konfiguration erfolgt via Makro und Ausführung der Programme *ref2cfg\_lin32.exe*, *ref2cfg\_lin64.exe*, *ref2cfg\_win32.exe* usw.

Zu konfigurieren sind diverse Einstellungen wie das Netzwerk, sowie die Parameter-Dateien

- Datenbank: *gak.cfg*
- GA-Knoten: *xxxx.ref*, *gak.cfg*, *anrufer.chk*, *vbp\_XXXX.001*, *iz04ddev.bn*

Mit wenigen Grundkenntnissen in der MSR sowie Unterstützung via Software und mit vielen Beispielen diverser MSR-Fabrikate gelingt das auch (es ist *XXXX:=Objekt* z.B. *AMEV*).

Die Belegung eines GA-Knoten erfolgt via Konfigurationsdatei *gak.cfg*. Diese Datei wird in das GA-Knoten Verzeichnis */gak/cfg/gak.cfg* kopiert und der GA-Knoten wird gestartet.

Die Belegung einer Leitzentrale erfolgt via Import-Verzeichnis */gak\_import/XXXX/gak.cfg*.

Der GA-Knoten ist das Bindeglied zwischen der MSR und dem Datenbank-Server und ist meist im Schaltschrank vor Ort untergebracht.

### Schemata

Das sind via Grafikprogramm oder Tabellenkalkulation beliebige bzw. nach Vorgaben z.B. der VDI gezeichnete Hintergrundbilder der Anlage. Diese werden in das Verzeichnis

*/srv/ftp/schemata/* kopiert. Via *jLZHview* oder *jLZHweb* werden die Schemata in der

Datenbank-Tabelle *SCHEMATA* dem betreffenden Objekt und Benutzer zugeordnet werden.

Der Schema-Editor der *jLZHview* oder *jLZHweb* unterstützt den Import beliebiger Schemata in vielen Formaten z.B. *gif*, *jpg*, *png*, (kein *wmf*) und die Positionierung der Datenpunktfenster z.B. mit Copy/Paste oder SQL-Import.

---

<sup>1</sup> Download *AMEV-GA-Plattform.de* unter (4) *Zugangsoftware und Wartung: jLZH\_Handbuch.pdf*

<sup>2</sup> Für MSR-Fabrikate mit *BACnet-Protokoll* kann via Tabellen-Kalkulation sehr einfach aus den *BACnet* CSV-Dateien (*EDE*, *StateText*, *Units*) eine eindeutige Referenz-Datei erzeugt werden. Fragen Sie uns...

<sup>3</sup> Andere MSR-Protokolle als das einheitliche *BACnet-Protokoll* werden als Dateien in den Formaten (*csv*, *pdf*, *xlsx*) ausgeliefert.

## Allgemeines

Die Installationsversion-11.4 enthält alle Pakete für openSUSE 11.4 und GA-Server (32Bit). Alle erforderlichen Komponenten sind auf USB bzw. DVD enthalten.<sup>4</sup>

Auch weitere Komponenten (*Openoffice, PDF* u.a.) werden mitinstalliert.

Als Hardware sind ITX-Boards mit Atom Prozessoren mit 2/4 Kernen getestet und in Betrieb. Ebenfalls getestet wurden andere Boards (auch HP Proliant SLES11.3) oder als virtuelle Maschine mit diversen Prozessoren. Auch alte oder gebrauchte Hardware kann verwendet werden... das ist kein Mangel, denn die DDC kann ja auch bereits 5, 10 oder 20 Jahre alt sein. Keine Angst bei der Installation, im schlechtesten Fall wird einfach neu installiert!

Bitte immer OHNE NETZWERK installieren, es ist ja alles Erforderliche vorhanden, eine Nachinstallation oder ein Update z.B. von Firefox ist abschließend immer via YaST möglich.

- Im 0.Schritt: BIOS des Rechners prüfen (z.B. Datum) (Taste F2 oder DEL je nach BIOS)
- Im 1.Schritt: Betriebssystem installieren (ca. 15-30 Minuten)
- Im 2.Schritt: GA-Server installieren (ca. 10 Minuten)
- Im 3.Schritt: Einstellungen und Nachbearbeitung (ca. 10-20 Minuten)
- Im 4.Schritt: Objekt installieren (Testbeispiel amev)
- Im 5.Schritt: GA-Server anschließen an USV, Router und DDC (Schaltschrank)

## BIOS

Das BIOS wird beim Start via <F2-Taste> geöffnet und sieht bei verschiedenen Mainboards immer etwas anders aus! Aber jetzt NUR die nachfolgend gezeigten Einstellungen nachprüfen und einstellen. Sonst... das BIOS ist sehr nachtragend und mag dann nur noch Experten...

**Beispiel:** ITX-Boards mit Atom Prozessor

```
Datum / Uhr          einstellen
ist beim Neustart die Uhr wieder falsch: neue Knopfbatterie einbauen.
Bootreihenfolge:    Hard Drive          <zeigt Festplatte>
Boot to Network     <Disable>
```

Auch die Reiter können anders bezeichnet sein:

### Reiter Main

```
System Date         prüfen
System Time         prüfen
```

### Reiter Configuration

```
Onboard Devices    Serial Ports          <enabled>
                   On board lan          <enabled>
                   USB ports          <Enable All>
                   Numlock           <On>

SATA Drives
Event logging       <enabled>
Video               <Auto>
Fan Control & Real-Time Monitoring
PCI/PCIe Add-In Slots Not Populated
```

### Reiter Security

```
Passwords nur bei Bedarf setzen
```

### Reiter Power

```
After power failure <Power off>
```

### Reiter Boot

```
Boot device Order für USB bzw. DVD:
Boot USB Devices first <Disable> → <enabled>
```

### Reiter Save and Exit

<sup>4</sup> DVD/USB *openSUSE 11.4* und *install-GA-Server-11.4*.

Download via *AMEV-GA-Plattform.de* und als Abbild auf DVD/USB brennen.

Keine Angst wegen dem Supportende, es sind für diese Anwendung keine Updates erforderlich! SLES 11.3 kann auch verwendet werden (Lizenz beachten), die Installation ist ähnlich, jedoch mit zusätzlicher Nachbearbeitung (für *Experten*).

## 1 Suse 11.4 installieren

DVD/USB *openSUSE-11.4-DVD-i586.iso* einlegen (getestet mit externem DVD Laufwerk).<sup>5</sup>  
Die Installationszeit dauert ca. 30 Minuten, davon die ersten 10 Minuten am Bildschirm.

### Start der Installation<sup>6</sup>

Sprache German/Deutsch (mit F2 Taste zu Beginn der Installation auswählen)  
Systemanalyse ...  
Neuinstallation  mit Häkchen [] Automatische Konfiguration verwenden  
Zeitzone Europa  
 **Gnome-Desktop**<sup>7</sup> → weiter  
 *Partitionsbasierend* → *Partitionsaufbau erstellen...* → weiter  
 *Benutzerdefinierte Partitionierung (für Experten)* → weiter  
Im Baum *Festplatten* → *sda* anklicken (falls alte Installation, jetzt löschen)

### 1.1a Partition Standard

Hinzufügen →  *Primäre Partition* (in jedem Schritt)

Lfd	Größe (MIN)	Einhängepunkt	FS-Typ	Fstab-Optionen <sup>8</sup>	Bestätigen mit
(a)	4 GB ( 2 GB)	swap	Swap	Gerätename=DISK00	Beenden
(b)	20 GB (10 GB)	/	ext4	Gerätename=DISK01	Beenden
(c)	10 GB ( 4 GB)	/home	ext4	Gerätename=DISK02	Beenden
(d)	Rest <sup>9</sup>	/lzh-daten	ext4	Gerätename=DISK03	Beenden

Lfd: (a) Name **swap** in Schaltfläche Dateisystem auswählen  
(b) Name / wird automatisch angezeigt  
(c) Name /home auswählen  
(d) Name /lzh-daten eintippen  
(d) Kann auch auf eine 2.Festplatte, falls vorhanden. Damit gibt es eine Trennung der Plattenzugriffe in Programme und Datenbank.

HINWEIS: Die Größe der Partition (a) bis (c) ist nur ein Vorschlag!

**Übernehmen** → weiter

Installieren, Reboot mit Dialog *Neuer Benutzer*

**Weiter geht es im Abschnitt 1.2 Neuen Benutzer erstellen.**

### 1.1b Partition für späteres Software-Raid1 (bei Bedarf)

Mit diesem Partitionstyp kann nachträglich bei Bedarf auch ein *Software-RAID1* erstellt werden. Das System wird erst mal auf einer oder mehreren Festplatten erstellt und getestet bzw. in Betrieb genommen. Dann kann daraus jederzeit ein Software-RAID1 erstellt werden d.h. auf ein bereits laufendes System kann nachträglich ein RAID1 installiert werden.

Die Anleitung befindet sich als Download unter

[http://wiki.linux-club.de/opensuse/NonRaid zu %28software%29Raid1 SuSE 10 1](http://wiki.linux-club.de/opensuse/NonRaid%20zu%20software%20Raid1%20SuSE%2010%201)

Grösse	Einhängepunkt	FS-Typ	Fstab-Optionen	Bestätigen mit
512 MB	/boot	ext4	Gerätename=DISK00	Beenden
16 GB	/	ext4	Gerätename=DISK01	Beenden
10 GB	/home	ext4	Gerätename=DISK02	Beenden
Rest	Extended			Beenden
2-8 GB	Swap	Swap	Gerätename=DISK03	Beenden
Rest	/lzh-daten	ext4	Gerätename=DISK04	Beenden

Bei SLES 11.3 ist der FS-Typ ext3.

Schön und manchmal ist das sehr praktisch. Mit diesem Partitionstyp ist das 1:1 umsetzbar. Die Anleitung muss nur geringfügig angepasst werden und wenn die Partitionen bereits wie hier gezeigt aussehen, ist alles weniger verwirrend (für Nichtexperten). OpenSUSE 11.4 und SLES11.3 wurde getestet!

<sup>5</sup> Mit dieser Installation getestete Boards siehe *Anhang C ITX Boards als GA-Server in Betrieb*.

<sup>6</sup> Installation kann auch mit gestecktem Netzkabel ausgeführt werden.

<sup>7</sup> KDE wird bei dieser Installation NICHT unterstützt.

<sup>8</sup> Derart parametriert kann die Festplatte dupliziert werden (siehe *Festplattenkopie erstellen*).

<sup>9</sup> z.B. Festplatte mit 250 GB. Via *Erweiterte Partition* können weitere Partitionen erstellt werden

## 1.2 Neuen Benutzer erstellen

Der Benutzer (BN) und das Passwort (PW) der Installation sind erstmal vordefiniert.

- BN: *dummy* PW: *dummy* (Hilfsbenutzer nur für die Installation, später entfernbar)  
[✓] Alle Häkchen entfernen → weiter
- BN: *root* PW: *@@@<sup>10</sup>* (*root* ist immer Chefin/ Chef)  
Es kann hier ein eigenes PW verwendet werden. Aber NICHT vergessen!

Die Meldungen „*Das Passwort ist zu einfach*“ → ignorieren!

Die Installationspasswörter können später sehr einfach via *YaST<sup>11</sup>* geändert werden.

Im Dialog die Schaltfläche [*Installieren*] betätigen (kommt zweimal vor).

### Die Installation ist ab jetzt automatisch (ca.10 bis 15 Minuten)<sup>12</sup>

- (1) *Installation durchführen (mit Diashow, Details,...)*
- (2) *Das System wird neu gestartet*
- (3) *Automatische Konfiguration*
- (4) **Der Dialog zeigt den Benutzer: *dummy***

## Weiter geht es im Abschnitt 2 Installation GA-Server.

### Nach dem Neustart noch einige Hinweise:

- (1) Ein Klick mit der rechten Maustaste auf den Desktop Hintergrund öffnet ein Auswahlmeneü, nachfolgend im Text als Kontextmeneü bezeichnet.
- (2) Es kann z.B. jetzt (oder später) die gesamte Festplatte getestet werden:  
Kontextmeneü → *Im Terminal öffnen*  
Folgende Befehle eintippen  
> `su root` (switch user *root* mit PW: *@@@* der Installation)  
> `badblocks -sv /dev/sda`
- (3) Falls ein anderer Hintergrund gewünscht ist:  
Kontextmeneü → *Hintergrund der Arbeitsfläche* ändern (Reiter Farbe, Schrift)
- (4) Falls im Dateimanager die Symbolansicht dauerhaft lästig ist:  
Meneü Reiter *Bearbeiten* → *Einstellungen*  
Reiter *Ansichten* → *Neue Ordner anzeigen mit:* → *Listenansicht* auswählen  
Reiter *Listenspalten* [✓] Häkchen setzen bei *Besitzer, Ort, Zugriffsrechte*  
→ *Aktualisieren*
- (5) Die automatisch für jeden Benutzer von *openSUSE* erzeugten Unterverzeichnisse können gelöscht werden (*Bilder, Musik, Öffentlich, Videos* und *Vorlagen*)

Weitere Details hierzu siehe nachfolgende Abschnitte.

<sup>10</sup> Tastenkombination @:=Alt Gr + q

<sup>11</sup> *YaST*: erreichbar via Schaltfläche *Rechner*

<sup>12</sup> Falls die Installation von *openSUSE* Fehler registriert, ist das Installationsmedium bzw. die Festplatte zu prüfen.

## 2 Installation GA-Server

Der erste Anmelde-Dialog zeigt den Benutzer: *dummy*

Anmelden als *Andere...*

BN: *root* PW: *@@@* (oder eigenes PW)

*Persönlicher Ordner* öffnen (Doppelklick auf die Schaltfläche)

### 2.1 Installation GA-Server via USB oder DVD

Den Datenträger mit dem Installationsskript<sup>13</sup> einlegen.

Es öffnet sich der Geräte-Ordner im Dateimanager und zeigt die gepackte Datei für die Installation mit dem Schriftsatz LATIN9<sup>14</sup> für die Datenbank.

`install_ga-server-11.4_LAT9_20yy-mm-tt.amev.tar.gz` (yy:Jahr, mm:Monat, tt:Tag)

- Diese Datei markieren und via Kontextmenü *Kopieren nach* → *Persönlicher Ordner*  
Danach den Datenträger aushängen (entfernen).

- Schaltfläche *Persönlicher Ordner* öffnen via Doppelklick<sup>15</sup>

- Die Installationsdatei markieren und via Kontextmenü → *Hier entpacken*

- Im Verzeichnis eine der beiden Dateien öffnen via Doppelklick

`install_ga-server-11.4.sh` `.sh`

(ohne logging)

`install_ga-server-11.4-mit-logging.sh`

Es öffnet sich ein Dialog → *im Terminal ausführen*

Es öffnet sich das Skript → alle Fragen mit *j* (*ja*) bestätigen

Das Skript läuft sichtbar und automatisch durch in ca. 5 bis 8 Minuten.

**Die Passwörter der Installation werden automatisch gesetzt** (also KEINE Eingabe).

**Nach dem Neustart** öffnet sich automatisch der Desktop des Benutzers *lzh*.

Der Benutzer *lzh* hat fast alle Rechte, aber keine *root* Rechte.

Bei Bedarf können via *YaST* weitere Benutzer mit eingeschränkten Rechten erstellt werden.

**Weiter geht es mit 3 System Einstellungen und Nachbearbeitung**

HINWEIS: Diese Installation kann auch via *Terminal* oder via *SSH* aus der Ferne ausgeführt werden. Das Netzwerk ist vorab zu Parametrieren (*YaST*).

- Via *Terminal* entpacken

`tar -tvf <datei>`

`list all files`

`tar -xf <datei>`

`extract all files`

Wenn das Entpacken misslingt, so ist die Datei beschädigt und die Installation ist abzubrechen!  
Ein neuer Download der Datei ist erforderlich.

- Via *Terminal* installieren

Kontextmenü → *Im Terminal öffnen*

Verzeichnis auflisten mit dem Befehl

`> ls -la`

(`la`: list all)

In das Verzeichnis der ausgepackten Datei wechseln mit

`> cd /root/install_ga-server-114_20yy-mm-tt.amev`

und den Befehl für eine der beiden Dateien eintippen

`> sh install-ga-server-114.sh`

(ohne logging)

`> sh install-ga-server-114-mit-logging.sh`

...

<sup>13</sup> Download via Internet *AMEV-GA-Plattform.de* → (2.2) Skript *LAT9...*

<sup>14</sup> Eine Installation in *UTF8* kann durch Export und anschließendem Import der Datenbank ausgeführt werden.  
*UTF8* ist ohne deutsche Sonderzeichen, analog dem GA-Knoten.

<sup>15</sup> Der Doppelklick ist wichtig um tatsächlich ins Verzeichnis zu wechseln.

### 3 System Einstellungen und Nachbearbeitung

Alles ist an erforderlicher Software installiert und kann getestet werden.

Die folgenden beiden Schritte beleben den GA-Server für den realen Einsatz in einem Schaltschrank mit einer DDC:

YaST Netzwerkgeräte (siehe Abschnitt 3.8)

GA-Server parametrieren (siehe Abschnitt 5)

Allerdings sind je nach installiertem System und Hardware (Mainboard) abschließend noch einige Einstellungen und Nacharbeiten für den Benutzer *lzh* auszuführen. Der „Nichtexperte“ sollte sich jetzt etwas Zeit nehmen um sich einen Überblick zu verschaffen...

#### 3.1 Desktop: Schaltflächen zeilenweise anordnen

Für den Benutzer *lzh* liegen alle wichtigen Programme als Schaltfläche auf dem Desktop und werden mit Doppelklick geöffnet.<sup>16</sup> Siehe auch Kontrollzentrum, Arbeitsflächen-Effekte. Die Schriftgröße des Desktop via Kontextmenü „Hintergrund der Arbeitsfläche ändern“.

- Reiter Schriftarten: die Größen 10 auf Größe 8 setzen falls zu groß.

Anordnung der Schaltflächen auf dem Desktop (Beispiel zeilenweise):

kp.exe	GNOME Terminal	Xterm	GNOME System Monitor	PuTTY
Firefox	Persönlicher Ordner	var_log	LibreOffice Calc	
Dokumentation	pgAdmin	jLZHview	workspace	
Take Screenshot				
Müll	Session Logout			

Hinweise:

- *Firefox* dient zur händischen Installation des Web-Servers *tomcat6* (Anhang G) Bei Bedarf auf den Desktop ziehen aus der Schaltfläche *Rechner* und dann öffnen (ohne Netzwerk).  
Im Menü *Bearbeiten* *Einstellungen*  
Reiter *Allgemein* *Leere Seite anzeigen*  
Reiter *Erweitert* *Update* Häkchen entfernen (nichts aktualisieren)  
via Menü | Ansicht | Symbolleiste | Häkchen setzen/entfernen
- Systemmeldungen öffnen via Doppelklick auf den Desktop-Ordner */var\_log* oder via Terminal als user root  
`su`  
> `gedit /var/log/messages` Editor *gedit* wird geöffnet  
> `tail /var/log/messages` Datei wird fortlaufend aktualisiert

#### 3.2 Panel: Einige Nettigkeiten zu Prozessor, Netz, Festplatte, Klebezettel

In einem Panel können zusätzliche Funktionen sehr übersichtlich dargestellt werden. Abhängig von der verwendeten Hardware sind die voreingestellten Funktionen noch zu bearbeiten, zu löschen oder zu ergänzen mit einem *rechte Maustaste Klick* im Panelbereich.

- *Zum Panel hinzufügen...* Rand oben anklicken  
Es öffnet sich ein Auswahl-Dialog für diverse Anzeigen
  - *Augen* falls die Maus verschwunden ist
  - *Farbenblind Applet* falls die Augen schwächer wurden
  - *Klebezettel* für fortlaufende Notizen...
  - *Hardware Sensor Monitor* Temperatur von CPU, Festplatte...  
Reiter *Allgemein* → *Aktualisierung* - Intervall (sek) 10  
Reiter *Sensoren* → Häkchen setzen bei gewünschter Funktion
  - *Systemmonitor* (Reiter *Prozessor, Speicher, Netzwerk,...Festplatte*)  
Farbenreihenfolge systematisieren grün, rot, blau, gelb  
Hintergrund immer schwarz oder grau.  
*Aktualisierungsintervall des Systemmonitors:* 5000 Millisekunden oder mehr  
Bei Bedarf ändern auf 500 Millisekunden.
  - ...weitere bei Bedarf
- *Aus Panel entfernen* (im Panel betreffendes anklicken)

Nicht jede Hardware wird durch das Installationsskript unterstützt.

Für Experten: Eine Paketsuche im Internet z.B. unter *Linux-monitoring sensors*

<sup>16</sup> Analog kann der Desktop für den Benutzer GA-Knoten eingerichtet werden.

### 3.3 pgAdmin III: Der PostgreSQL Administrator

Der Abschnitt 3.3 bis 3.6 ist Lesestoff.

**Weiter geht es im Abschnitt 3.7 YaST.**

Die Verwaltung der Datenbank ist für den Administrator reserviert und wird mit Doppelklick auf die Schaltfläche *pgAdmin* geöffnet.

Es zeigt sich erst mal ein ‚*Tip of the Day...*‘ (informativ für Neulinge von PostgreSQL)

Im Baumfenster erscheint der *Server als ‚LZH: Leitzentrale Haustechnik‘.*

Mit Doppelklick öffnet sich der Anmeldedialog *Zum Server verbinden*

Die Passwordeingabe ist bereits getätigt, da es Sicherheitsbedenken beim Benutzer *lzh* nicht geben sollte, er ist ja der Hauptbenutzer, also auch ein Datenbank Benutzer. Andernfalls ist diese Schaltfläche auf dem Desktop zu sperren bzw. die Benutzerrechte zu entfernen.

Andere Benutzer können via *YaST Control Center* → *Auswahl Sicherheit und Benutzer* mit weniger Rechten hinzugefügt werden.

HINWEIS:

Die Datenbank wird täglich um 2:05 durch einen *cron* auf *vacuumdb* gewartet (entlüftet).

Siehe Details als Benutzer *root* im Verzeichnis */etc/crontab*

Eine Sicherung erfolgt via *pgAdmin* oder via Kommando *pg\_dumpall*

Das Passwort des Datenbank Benutzers *lzh* darf nicht einfach verändert werden, da es Abhängigkeiten gibt zu den Prozessen *kp.exe*, *pg\_vacuum*, *Transfer\_MW*, ...

Die Datenbank sollte möglichst nur von Experten administriert werden, denn man kann ja alles (kaputt) machen. Als weitere Übung kann aber immer noch das gesamte System in ca. 30 Minuten neu aufgesetzt werden. Fortgeschrittene mit Datenbank Kenntnissen wissen sowieso schon mehr.

Falls die Datenbank bearbeitet werden muss, zum Beispiel für ein *REINDEX* der Tabellen *datenpunkte* oder *messwerte*, so ist der Prozess *kp.exe* (im lila Fenster) vorher anzuhalten mit *stopj* und nach der Indizierung wieder mit *exit* zu starten. Ein *REINDEX* wird meist mit dem *SQL-Editor* in *pgAdmin* ausgeführt und kann bei vielen Daten sehr lange dauern!

#### REINDEX

Ein *REINDEX* wirkt von Zeit zu Zeit beschleunigend bei SQL-Abfragen!

Beispiel für ein *REINDEX* der Tabelle *messwerte* mit etwa 60.000.000 Datensätzen

```
REINDEX INDEX mw_erfassungszeit          etwa 20 Minuten
REINDEX INDEX mw_obj_adr_eref            etwa 160 Minuten
```

Folgender Prozess ist vor dem *REINDEX* zu deaktivieren:

Falls der *kp.exe* aktiv ist, so wird dieser mit *STOPJ* beendet.

Nach dem *REINDEX* ist der zuvor gestoppte Prozess wieder zu aktivieren (*exit* eintippen).

#### pgAdmin III bzw. kp.exe arbeitet nicht

Manchmal kann der *pgAdmin* bzw. der *kp.exe* nicht starten.

```
/etc/init.d/postgresql restart          Neustart
```

Falls das nicht hilft

```
su postgres                             BN: postgres
/usr/local/pgsql/bin/postmaster -D /lzh-daten/pgdata0117
```

Falls das auch nicht hilft, so sollte die gemeldete Antwort genau gelesen werden:

#### Beispiel kp.exe startet nicht

(kein lila Bildschirm)

- (1) *su postgres* (als Benutzer *postgres* anmelden)
- (2) */etc/init.d/postgresql status* (Zustand anzeigen)
- (3) */usr/local/pgsql/bin/postmaster -D /lzh-daten/pgdata01/*
- (4) *rm /lzh-daten/pgdata01/postmaster.pid*
- (5) *rm /tmp/.s.PGSQL.5432\** auch *.s.PGSQL.5432.lock*
- (6) --->(3) wiederholen und dann beenden
- (7) --->(2) wiederholen mit *start*
- (8) *kp* Fenster verlassen mit *exit*

Weitere Details siehe Anhang bzw. im Internet unter [www.postgresql.org](http://www.postgresql.org)

<sup>17</sup> Es ist *-D* : spezifizy Data Directory

### 3.4 Die Anwendung jLZHview: eine Desktop Visualisierung

Die Bediensoftware der LeitTechnik ist die Anwendung *jLZHview* unter Eclipse Java /Linux:

- *jLZHview* eine Desktop Visualisierung der AMEV-GA-PLATTFORM

Mit einem Doppelklick auf die Schaltfläche *jLZHview* wird die Anwendung gestartet.

Ein Doppelklick im Baumfenster (links) auf

*GA-Server\_AMEV: PostgreSQL via Localhost*

öffnet die Anwendung für den Benutzer `postgres` mit Passwort `lzh.wks.`<sup>18</sup>

- Weitere voreingestellte *Leitzentralen* können entfernt bzw. erweitert werden.
  - Menü *Verwaltung* → *Leitzentralen* öffnet den Dialog *Leitzentralen*: diesen Eintrag markieren und via Schaltfläche *Bearbeiten* kann der Eintrag geändert werden.
  - Menü *Benutzer-Privilegien* öffnet die Benutzerverwaltung
- Weitere Details sind in der Hilfe der *jLZHview* zu finden.

HINWEIS: das Testbeispiel Objekt *amev* ist teilweise vorinstalliert

→ also weiter geht's mit dem üblichen *Versuch & Irrtum...*

### 3.5 Die Anwendung jLZHweb: eine Browser Visualisierung

Eine weitere Bediensoftware der LeitTechnik ist die Anwendung *jLZHweb* unter Eclipse Java

- *jLZHweb* eine Browser Visualisierung der AMEV-GA-PLATTFORM

Die Anwendung startet via URL im Browser (*Chrome, Firefox* u.a.)

Der Webserver *tomcat6* ist bereits vorinstalliert und aktiv.

Die *jLZHweb* wird bei Bedarf wie im *Anhang F* beschrieben installiert, vereinfacht mit einem Skript als Benutzer *root* (siehe G1.0 Tomcat6 Installation via Skript)

Die Anwendung kann lokal oder aus der Ferne via Browser gestartet werden.

HINWEIS:

1. Angehalten wird der Webserver *tomcat6* mit `rctomcat6 stop` (BN: *root*)  
oder via *YaST* Gruppe *System* → *Systemdienste* → *Anhalten*  
In beiden Fällen ist *tomcat6* nach einem *reboot* wieder aktiv.
2. Das voreingestellte interne Netzwerk ist `127.0.0.0 mask 255.255.0.0`  
zu finden im Verzeichnis  
`/lzh-daten/pgdata01/pg_hba.conf` als user `postgres` anmelden
3. Falls der Webserver NICHT gewünscht wird, so wird er via *YaST* Gruppe *Software* deinstalliert.

### 3.6 Der Kommunikations-Prozess

Der *Kommunikations-Prozess* (*kp.exe*) kommuniziert mit *GA-Knoten* und mit der Datenbank *PostgreSQL*. Es können auf einem *GA-Server* mehrere *GA-Knoten* installiert werden.

Ein Doppelklick auf die Desktop Schaltfläche öffnet das *KP-Terminal* (lila Fenster)

und kann jederzeit wieder geschlossen werden z.B. via [x] in der oberen Ecke rechts.

Diese Schaltfläche sollte nicht gelöscht werden (*KP-Terminal* Befehl: `xterm -e screen -x`)

Falls das lila Fenster nicht sichtbar ist, bzw. der *kp*-Prozess nicht startet,

- so ist die Datenbank nicht aktiv (siehe 3.3 pgAdmin III)
- so fehlt i.a. die Bibliothek `libcapi20.so.3`.  
(bei *OpenSUSE 11.4* immer installiert, nicht bei *SLES 11.3*)  
Diese Bibliothek ist bei der Installation erforderlich, auch wenn kein ISDN verwendet wird und wird nachinstalliert via *YaST*:

*Software installieren* (DVD *openSUSE-DVD-i586-Build0024* einlegen)

Im Suchfenster des Software-Verwalters

`capi4linux` eintippen und dort das *Häkchen* setzen → *Anwenden*, fertig.

Danach das *KP-Terminal* verlassen mit dem Befehl `exit` (eintippen).

Das lila Fenster erscheint.

HINWEIS:

Manchmal kommt es vor, dass der *KP-Prozess* sich aufhängt. (lila Fenster NICHT sichtbar)

---

<sup>18</sup> Es kann gespielt werden... und wenn doch etwas zerstört wurde, so kann im schlechtesten Fall die *jLZHview* neu installiert werden (nur auspacken!) oder die gesamte AMEV-GA-Plattform neu installiert werden.

## Mit der Tastatur Eingabe

STOPJ wird der *kp* gestoppt

exit wird der *kp* wieder gestartet (läuft in einer Schleife)

## Falls dies nicht klappt (eher selten)

YaST runlevel *kp* anhalten

ps -x alle *kp* Prozesse beenden mit kill *prozess ID*

YaST runlevel *kp* starten (lila Fenster sichtbar)

## Oder via Terminal

ps -x alle *kp* Prozesse beenden mit kill *prozess ID*

screen -l öffnet screen (lock)

kpstart startet die *kp.exe*

## 3.7 YaST: Software

Ein Blick in das System schadet nie, es muss ja nicht gleich alles verändert werden.

Via Schaltfläche *Rechner* wird YaST als Benutzer *root* geöffnet.

a) Einen Benutzer mit weniger Rechten erstellen als der Benutzer *lzh*

b) *Software-Repositories*

[✓] alle Häkchen entfernen. Das Erforderliche ist vorhanden und im laufenden Betrieb ist ein automatisches Update meist störend, besonders wenn kein Internet vorhanden ist.

Wird jedoch via Internet aktualisiert, so wird die gewünschte Software anschließend via Schaltfläche *Software installieren* im System aktiviert.

Für Nichtexperten: Updates downloaden und anschließend mit dem *Software-Verwalter* das Gewünschte aktualisieren.

c) *Software installieren oder löschen (Software-Verwalter)*

Mal reinschauen was installiert ist: z.B. via Suchfenster folgendes eintippen:

*java-1\_6\_0-sun* ist vorhanden → Häkchen gesetzt

*java-1\_6\_0-openjdk* falls vorhanden → Häkchen entfernen und deinstallieren

*vsftp*

*tomcat6*

*capi4linux* (muss derzeit noch installiert sein, obwohl die ISDN-Zeit vorbei ist)

*nxserver*

System → *Systemdienste (Runlevel)*

Mal reinschauen was aktiviert ist: z.B.

*tomcat6* (anhalten, wenn *jLZHweb* nicht benutzt wird)

Keine Angst, im schlechtesten Fall muss neu installiert werden!

## 3.8 YaST: Netzwerkgeräte

### 3.8.0 Die Netzwerkadresse *localhost*

Der Name *localhost* bezeichnet die virtuelle IP-Adresse 127.0.0.1 für den Zugriff auf Server, die lokal auf einem Rechner laufen.

Wie GA-Server in ein Netzwerk eingebunden werden wird im nächsten Abschnitt gezeigt.

#### GA-Server

Der GA-Server wird als lokale Leitzentrale (LZH) verwendet.

Hier kommunizieren ein oder mehrere GA-Knoten via *localhost* mit der lokalen PostgreSQL-Datenbank.

In die Datenbank-Tabelle GAKNOTEN sind alle IP-Adressen der GA-Knoten einzutragen.

- AMEV\_1 via 127.0.0.1#4404 (unser Beispiel)
- XY3A\_1 via 127.0.0.1#4405 (weiteres Beispiel, es wird der Port erhöht)
- etc.

Im GA-Knoten zu parametrieren sind jeweils die Dateien

- /cfg/anrufer.chk mit Zeile: 001:: L 3 040 002 10000 127.0.0.1#4403
- /lz/vbp/vbp\_AMEV.001 mit Zeile: LZ1:: 040 002 127.0.0.1#4403

#### ZLT-Server

Der GA-Server kann auch als übergeordnete Zentrale-LeitTechnik (ZLT) verwendet werden.

Hier kommunizieren die lokalen GA-Server mit der übergeordneten PostgreSQL- bzw. Oracle-Datenbank via Netzwerk.

In die Datenbank-Tabelle GAKNOTEN sind **alle** IP-Adressen der lokalen GA-Server einzutragen.

- GA-Server AMEV mit 172.31.10.50#4404 (unser Beispiel)
- GA-Server XY3A mit 172.31.10.88#4404 (weiteres Beispiel)

### 3.8.1 YaST Netzwerkeinstellungen

#### Reiter **Globale Optionen**

Traditionelle Methode mit ifup

Ipv6 aktivieren → Häkchen weg

Zu sender Hostname: AUTO

Standard-Route über DHCP ändern → Häkchen setzen

#### Reiter **Übersicht → Bearbeiten**

Es können eine, oder falls vorhanden, mehrere Netzwerkkarten parametrieren werden. Meist wird eine Netzwerkkarte für die Verbindung zur ZLT (Zentrale LeitTechnik) verwendet. Eine zweite Netzwerkkarte dient der Verbindung zur DDC (Schaltschrank), z.B. für BACnet, MODBUS/IP<sup>19</sup>

### 3.8.2 YaST: Netzwerk in einem 256-er Netz

Dieses Beispiel zeigt die Einträge in einer Netzwerkkarte in einem 256-er Netz.

Statisch zugewiesene IP-Adresse

IP-Adresse: 172.31.10.50

Subnetzmaske: 255.255.255.0 oder kurz /24 (256-er Netz)

Host strassenname.12345.geb.eub (siehe Hostname)

→ weiter

#### Reiter **Hostname/DNS**

Hostname: strassenname (vorinstalliert)

Domänenname: 12345.geb.eub (vorinstalliert)

Nameserver 1: 172.31.10.1 (IP vom Router)

Domänensuche: 12345.geb.eub (vorinstalliert)

<sup>19</sup> Als Netzwerk verwenden viele DDC'en die Schnittstelle RS232 bzw. via Adapter RS485 (große Reichweite ~ 300-800 m). Es gibt Mainboards mit 2 bzw. 4 x RS232 on Board.

Notiz: Der Host- und Domänenname wird automatisch geändert, wenn ein GA-Knoten erzeugt wird (siehe GA-Knoten).

### Reiter **Routing**

*Standard-IPv4-Gateway:* 172.31.10.1 (IP vom Router)  
*Gerät:* - oder eth0 (siehe Übersicht)

Mit **OK** werden die Netzwerkeinstellungen gespeichert.

### 3.8.3 YaST: Netzwerk in einem 8-er Netz

Dieses Beispiel zeigt die Einträge in einer Netzwerkkarte in einem 8-er Netz.

#### ⊙ *Statisch zugewiesene IP-Adresse*

*IP-Adresse:* 172.31.10.50  
*Subnetzmaske:* 255.255.255.248 oder kurz /29 (8-er Netz)  
*Host* strassenname.12345.geb.eub (siehe *Hostname*)  
→ weiter

### Reiter **Hostname/DNS**

*Hostname:* strassenname (vorinstalliert)  
*Domänenname:* 12345.geb.eub (vorinstalliert)  
*Nameserver 1:* 172.31.10.49 (IP vom Router)  
*Domänensuche:* 12345.geb.eub (vorinstalliert)

### Reiter **Routing**

*Standard-IPv4-Gateway:* 172.31.10.49 (IP vom Router)  
*Gerät:* eth0 (siehe Übersicht)

Mit **OK** werden die Netzwerkeinstellungen gespeichert.

Ein einfacher Test der Netzwerkeinstellung kann über *Firefox* oder *Netzwerkdienste* erfolgen.

**WICHTIG:** wird ein anderes Netzwerk als 172.31.0.0 verwendet, so ist dieses auch in der PostgreSQL-Datei /lzh-daten/pgdata01/pg\_hba.conf einzutragen!

### 3.9 YaST: Netzwerkdienste

Es können mehrere Netzwerkdienste parametrisiert werden (z.B. ptbtime1.ptb.de).

Das Beispiel zeigt den Netzwerkdienst *NTP*.

Die Meldung *Datei /etc/ntp.conf wurde manuell geändert...* kann ignoriert werden:

die Netzwerkdienste wurden bereits während der Installation eingerichtet.

#### 3.9.1 YaST: NTP-Einrichtung

### Reiter **Allgemeine Einstellungen**

NTP-Daemon starten

#### ⊙ *jetzt und beim Systemstart*

*Richtlinie zur Laufzeitkonfiguration:* Automatisch

*Synchronisationsintervall in Minuten:* 5

*Synchronisierungstyp:* Server

*Adresse:* ptbtime1.ptb.de

oder die *Zentrale LeitTechnik* z.B. 172.31.10.10

→ markieren → *Bearbeiten* → *Test*

Wenn der Test meldet: *Der Server ist erreichbar und antwortet korrekt*, so kommt der Rechner ins Internet. Andernfalls Netzwerkgeräte prüfen (vergesslich gewesen?).

Der Router selbst sollte vorher schon richtig konfiguriert sein!

### 3.10 SSH, NX: Fernzugriff mit einem Windows Rechner

Fernzugriffe auf den GA-Server sind bereits installiert. Es ist jetzt noch der Fernzugriff auf dem eigenen Windows Rechner zu installieren. Unter Linux geht ja schon alles, allerdings sollte noch der Benutzer `root` gesperrt werden! (siehe B Anhang, Dienste `/etc/ssh/sshd_config`) Einige der genannten Anwendungen sind kommerziell, können aber privat genutzt werden.

- a) PuTTY  
download <http://www.putty.org>
- b) Secure Shell  
download <http://www.openssh.com>  
*B Anhang, Dienste*
- c) NoMachine: NX Client for Windows  
download <http://www.nomachine.com>  
Nichtkommerziell frei und auf 2 User beschränkt
- d) Oder andere...

Die folgenden Abschnitte sind Lesestoff.

**Weiter geht es im Abschnitt 5 GA-Server parametrieren mit einem Objekt**

GA-Server sind sowohl via SSH oder NX-Client direkt erreichbar.

1. Teste aus einem Windows 7 Netzwerk einen GA-Server mit der IP 172.31.10.50  
Systemsteuerung | Netzwerkstatus | LAN-Verbindung | Eigenschaften  
| Auswahl Internetprotokoll Version 4 | Eigenschaften |
  - IP-Adresse automatisch beziehen
  - Folgende IP-Adresse verbinden  
IP-Adresse: 172.31.10.51 (erste freie Adresse im GA-Server)  
Subnetzmaske: 255.255.255.0  
Nach dem Test wieder zurücksetzen.
2. SSH Secure Shell Client: Host name 172.31.10.50, User name lzh, Port number 22

### 3.11 Drucker installieren via YaST

Das Beispiel zeigt die Installation eines Netzwerkdruckers.

**YaST → Hardware**

Gruppe Drucker: Druckerkonfiguration → Hinzufügen

Es sind einige selbsterklärende Einträge auszufüllen, bzw. kann danach gesucht werden.

Beispiel:      Netzwerkdrucker HP ColorLaserJet cp1515n  
                 IP-Adresse:            172.31.10.115:9100  
                 Suche nach dem Drucker hp cp1515n

### 3.12 Linux Passwort vergessen (c't 2015, Heft 5)

Das sollte NICHT vorkommen, aber...

- (1) beim Start von Linux schnell die `<Esc>` Taste drücken
- (2) Boot Menü erscheint: `Linux...` auswählen und Buchstabe `<e>` eintippen
- (3) An den String `linux /boot/vmlinuz....` den String `init=/bin/sh` anhängen
- (4) Starten mit Tastaturkombination `Strg+x`
- (5) die Boot Shell öffnet sich
- (6) `passwd username` eintippen (z.B. `lzh`)
- (7) neues password eintippen
- (8) falls die Meldung `Authentication token manipulation error` erscheint,  
vorher `mount -o remount,rw /` eintippen

Die Tastatur beim Bootvorgang stellt sich oft auf eine US Tastatur.  
(siehe B Anhang Linux Nomenklatur)

## 4 Weitere Installationen von Suse

Die Installation von Suse Distributionen ist derzeit zwingend, da die Anwendungen auch das ISDN Protokoll benutzen. Inzwischen gibt es nur noch wenige ISDN Anschlüsse.

### 4.1 SLES 11.3 (SuseLinuxEnterpriseServer)

Die Installation des SLES 11.3 (Lizenz beachten) verläuft ähnlich openSUSE 11.4.

Die Installation erfolgt wieder ohne Netzwerk von DVD.

Wir geben eine (sehr) kurze Anleitung.

#### 4.1.1 SLES 11.3 Installation

Medienprüfung → *weiter*

⊙ *Physischer Computer Installationseinstellungen* (hier aufpassen)

Reiter *Experten* → Schaltfläche *Partitionierung* (analog *openSUSE11.4*)

(Die Passworteingabe für den Benutzer `root` erfolgt erst gegen Ende der Installation)

Hostname über DHCP

Firewall deaktivieren

Test der Internetverbindung (muss nicht sein)

Konfiguration von Netzwerk Services (kostet irgendwann)

⊙ *Konfiguration überspringen*

*Neuer lokaler Benutzer: dummy* (analog *openSUSE11.4*)

Klon kann jetzt erstellt werden: `/root/autoinst.xml`

Hostname: `strasse.eub` (oder anders)

Die Installation dauert etwas länger als bei *openSUSE11.4*.

#### 4.1.2 SLES 11.3 GA-Server Installation

Der *GA-Server* wird analog *openSUSE11.4* installiert mit demselben Skript

```
install_ga-server-114_2015-mm-tt_eub.tar.gz (mm:Monat, tt:Tag)
```

Das Skript wurde derzeit nicht auf SLES angepasst, da es selten angewendet wird. Daher sind einige wichtige Nacharbeiten erforderlich, da das System noch „nicht sauber“ ist. Sogar Nichtexperten können als „Saubermänner“ mitmachen.

Nach dem Neustart sind wichtige Nacharbeiten via *YaST Software* erforderlich

(Die YaST Oberflächen sehen bei SLES etwas anderes aus).

Via *YaST Software* ist zu installieren (Häkchen setzen)

[✓] capi4linux ISDN

[✓] java-1\_6.0-sun

[✓] vsftpd

[✓] tomcat6 bei Bedarf

[ ] nxserver von openSUSE 11.4 ist zu deinstallieren (läuft nicht auf SLES)

Die NoMachine Version 5 kann danach dann installiert werden mit der Internetsuche

```
Download NoMachine for Linux-i686 Version 5.0.53_1
```

Das Paket `nomachine_5.0.53_1_i685.rpm` herunterladen. Installationsanleitung ebenda.

Dieses Paket ist frei für 3 users und wurde von uns getestet.

Und nicht vergessen: den NoMachine Client auf dem Windows Rechner.

## 4.2 openSUSE 13.2 (Wisch- & Tupf-Technik)

Die Installation des openSUSE 13.2 verläuft ähnlich openSUSE 11.4. Allerdings ist diese Version stark gewöhnungsbedürftig. Die allgemeine Entwicklung von Betriebssystemen setzt derzeit eben auf die „Wisch- & Tupf-Technik“, obwohl für unsere Art von Anwendungen ein klassischer Desktop geeigneter ist.

HINWEIS: Die Test-Hardware hatte eine CPU 4xD525MW mit 1,8GHz. Der Bildschirm Acer P223w (kein Touch) taugte nicht zur Installation (blieb dunkel). Ein älterer Samsung mit Standardauflösung 1024x768 ging dann...

Die grafische Oberfläche (Gnome) ist deutlich langsamer als bei 11.4 oder SLES 11.3

### 4.2.1 openSUSE 13.2 Installation

Booten mit `openSUSE-13.2-DVD-i586.iso`

...warten (Der Bildschirm bleibt einige Minuten dunkel)

Netzwerk kann jetzt schon installiert werden (muss aber nicht)

Initializing, Language, Systemüberprüfung (keine Häkchen setzen)

Online-Repositories vor der Installation...

Zusatzprodukte

Weiter

Expertenmodus, Festplatte sda (Reihenfolge nach Gerätenamen DISK00, DISK01,...)

<input checked="" type="radio"/> Betriebssystem	/	Ext4	Gerätenamen DISK01
<input checked="" type="radio"/> Daten und Programme	/home	Ext4	Gerätenamen DISK02
<input checked="" type="radio"/> Daten	/lzh-daten	Ext4	Gerätenamen DISK03
<input checked="" type="radio"/> Swap	swap		Gerätenamen DISK00
<input type="radio"/> Raw Volume	nur bei Bedarf		

Zeitzone

Gnome Desktop wählen

Neuer Benutzer

Installation bestätigen

Neustart (Geduld...)

#### 4.2.2 openSUSE 13.2 GA-Server Installation

Der *GA-Server* wird analog *openSUSE11.4* installiert mit demselben Skript

```
install_ga-server-114_2015-mm-tt_eub.tar.gz (mm:Monat, tt:Tag)
```

Das Skript wurde derzeit nicht auf *openSUSE 13.2* angepasst, da es aus unserer Sicht nicht geeignet ist. Daher sind eine Reihe wichtiger Nacharbeiten erforderlich, da das System noch „nicht sauber“ ist.

Hier sind Nichtexperten keine geeigneten „Saubermänner“.

Nach dem Neustart sind wichtige Nacharbeiten via *YaST Software* erforderlich (Die YaST Oberflächen sehen bei *openSUSE13.2* echt anderes aus).

Das Skript kann nur via *Terminal* (Suchbegriff) gestartet werden

```
cd /root/install_ga-server-114
sh install_ga-server-114-mit-logging.sh
```

Alles mit *j* (*ja*) beantworten etc.

Neustart

Wenn alles geklappt hat, dann auf *Aktivitäten* klicken Suchbegriff <Desktop> etc.

Suchbegriff <YaST> Software installieren (wie bei SLES 11.3)

```
libcapi20-3
vsftp
tomcat6
nxserver 3.5
```

usw.

Leider gibt es das Panel für die *cpu*, *Temperatur* etc. nicht (wir fanden es noch nicht). Wir beobachten die weitere Entwicklung. Vielleicht gibt es eine Version mit wahlweiser Umschaltung. Unser Favorit derzeit bleibt *openSUSE11.4*.

#### 4.3 System kopieren auf eine weitere Festplatte (via Life System)

Ist eine Festplatte einmal hergestellt, so kann diese sehr locker mit einem *Linux Life System* dupliziert werden. Das reduziert erheblich den Arbeitsaufwand, da hier nur wenige händische Arbeitsschritte erforderlich sind (Serienproduktion).

Eine solche binäre Festplattenkopie dauert je nach Größe der Festplatte schon mal 90 Minuten (bei 500 GB). Dabei werden ALLE *Bytes* der Quellfestplatte übertragen, d.h. die Zielfestplatte muss mindestens gleich groß sein. Gleichzeitig hat man damit einen Schreibtest und kann sich den *badblocks* schon mal schenken.

Die Quellfestplatte sollte Gerätenamen z.B.: *DISK00*,... haben, sonst gibt es Arbeit beim Booten, was nicht jeder sofort kann.

Die Zielfestplatte muss mindestens gleich groß sein, kann aber ein anderes Fabrikat sein.

Beide Festplatten in einen Testrechner einbauen und mit einem *Linux Life System* von CD oder USB starten z.B. mit *Ubuntu 12.0ff* das für Ungeübte so ziemlich verständlich daherkommt.

Ist das Life System hochgefahren, so sind folgende Programme erforderlich, welche je nach Life System woanders zu suchen sind (sic!).

- *gparted* (zeigt die Details der beiden Festplatten: *sda* und *sdb*)
- *Terminal* (zum Ausführen von Kommandos)

Folgender Befehl startet den Kopiervorgang im *Terminal*:

```
sudo dd if=/dev/sda of=/dev/sdb bs=4096 (das Terminal bleibt stumm)
HINWEIS: if:inputfile, of:outputfile, bs:blocksize
```

Folgender Befehl prüft den Kopierfortschritt:

```
sudo kill -SIGUSR1 $(pidof dd) (ein 2.Terminal öffnen!)
(mehrfach aufrufbar)
```

Also echt aufpassen, von wo (*if*: Quelle) nach wo (*of*: Ziel) kopiert wird.

Hier hilft der Partitionierer (*gparted*). Die Zielfestplatte ist ja meist leer!

Vorab kann noch getestet werden mit *badblocks -sv /dev/sda* bzw. *sdb*

Hinweis: Falls auf der Quell-Festplatte bereits ein Netzwerk konfiguriert wurde, so ist das Netzwerk auf der Zielfestplatte neu zu konfigurieren (andere Netzwerkkarte!).

## 5 GA-Server parametrieren mit einem Objekt

Ein GA-Server enthält kein, ein oder mehrere Objekte, die *PostgreSQL* Datenbank, die *jLZHview* und die *jLZHweb*, sowie weitere Anwendungen (siehe auch *jLZH\_Handbuch.pdf*). Dieser Abschnitt der händischen Belegung der Datenbank und des GA-Knoten ist aufwändiger! Wird jedoch die Anwendung **GA-Plattform\_nn.ods** auf den Desktop kopiert, ausgepackt und bearbeitet, so ist die Parametrierung via Menü Steuerung nahezu automatisch, sowohl der Datenbank, als auch eines GA-Knotens (siehe H7.1 Die Anwendung *GA-Plattform\_ nn.ods*).

### 5.1.0 GA-Server installieren

Mit dem vorinstallierten *BACnet*-Beispiel *AMEV*<sup>20</sup> kann der GA-Server ausprobiert werden. Installiert werden muss jedoch noch der GA-Knoten.

Weiter geht es zu **(5.1.5) Erzeugen eines GA-Knoten**<sup>21</sup>.

#### 5.1.1 Das Installationsverzeichnis /home/lzh/gak\_import/

Aus dem Installationsverzeichnis wird die Datenbank und ein GA-Knoten parametriert. Das Installationsverzeichnis beginnt immer mit dem Objektname

/home/lzh/gak_import/AMEV/	AMEV:=Objektname	zwingend (4 Zeichen)
cfg/	Konfiguration	zwingend
iz/	InselZentrale	falls BACnet
_logfiles	Logging Dateien	zwingend
lz/vbp/	VerbindungsProtokoll	zwingend
schemata/	Hintergrundbild	falls vorhanden
sql/	SQL-Dateien	falls vorhanden

#### 5.1.2 Die Parameter-Dateien

Die Parameter-Dateien für das *BACnet*-Beispiel *AMEV* sind im Verzeichnis

/home/lzh/gak_import/amev/	beginnt immer mit dem Objektname
/home/lzh/gak_import/amev_BACnet/	oder mit einer beliebigen Erweiterung
<b>mit den Parameter-Dateien</b>	
o cfg/anrufer.chk	Anruferliste (bei Bedarf anpassen)
db_atb.txt	Attribute <sup>22</sup> (Meldetexte der Datenbank)
db_dim.txt	Dimensionen (Einheiten der Datenbank)
gak.cfg	Ablaufdatei (Konfiguration der Datenpunkte)
amev_1.ref	Referenzdatei (nicht erforderlich, nur Doku)
o iz/iz04ddev.bn	BACnet Netzwerk (falls vorhanden)
o linux/hostname	Hostname (nur wenn nicht via YaST)
o lz/vbp/vbp_AMEV.001	Verbindung zur Datenbank
o schemata/ Hintergrundbild	beginnt immer mit dem Objektname
AMEV_HK1_image001.gif	Schema- oder Hintergrundbild
o sql/	<b>Kann auch via <i>jLZHview Schemata-Editor</i> erstellt werden</b>
INSERT_2-FENSTER.SQL	Koordinaten der Datenpunkte
INSERT_3-FENSTERSYMBOLS.SQL	Symbole (gibt es automatisch)
<b>Kann auch via <i>jLZHview</i> geändert werden</b>	
UPDATE_4-DATENPUNKTE_BESCHREIBUNG.SQL	Erweiterung falls >24 Zeichen

**HINWEIS:** Die Verzeichnisstruktur sollte bei vielen Objekten strukturiert werden  
Verschiedene Bezeichnungen sind ausführbar (Beispiele)

```
/home/lzh/gak_import/amev.VORSCHAU_0  
/amev.ZEITPLAN_0
```

Es kann auch ein übergeordnetes Verzeichnis erstellt werden

```
/home/lzh/gak_import/AMEV_Ablage/amev.VORSCHAU_0  
/AMEV_Ablage/amev.ZEITPLAN_0  
/AMEV_Ablage/...etc.
```

<sup>20</sup> Für GA-Server im realen Betrieb sollte jedoch der Benutzer *amev* gelöscht werden.

userdel -rf amev löscht den Benutzer *amev* mit allen Verzeichnissen (ausführen als BN:root)

<sup>21</sup> Allerdings gibt es hierfür keine echten Daten, da die Anbindung an eine MSR (DDC oder Schaltschrank) fehlt.

<sup>22</sup> Hier nicht erforderlich, da die Attribute und Dimensionen mit den Einträgen in der Datenbank übereinstimmen.

### 5.1.3 Die Basis-Datenbank-Tabellen

Im ersten Schritt sind in dieser Reihenfolge folgende Tabellen in der Datenbank einzutragen

(1)	strassen	AMEV Str.	Name der Straße
(2)	objekte	AMEV	Objektname 4 Zeichen maximal (0-9, A-Z)
(3)	gaknoten	AMEV_1	Objektname lt. Konvention ergänzt mit _1
(4)	schemata	AMEV_...	Hintergrundbild beginnt mit dem Objektnamen

Diese Eintragungen können mit der Anwendung *jLZHview* via Dialog erzeugt werden.

Alternativ auch aus der vorbereiteten SQL-Datei via SQL Import.<sup>23</sup>

```
/home/lzh/gak_import/AMEV_STRASSE+OBJEKT+GAKNOTEN+SCHEMA_INSERT.SQL
```

### 5.1.4 Der Datenbank-Import aus dem Installationsverzeichnis

Sind die *Basis-Datenbank-Tabellen* (5.1.3) eingetragen, so werden jetzt die Datenpunkte, die Schemata, Fenster und Fenstersymbole in die Datenbank als Benutzer *lzh* importiert.

In einem *Terminal* werden der Reihe nach die Befehle (1, 2, 3 bzw. 5) ausgeführt

Aufruf mit Parameter	erforderlicher Verzeichnisinhalt
1_import-datenpunkte.sh amev	gak.cfg

und falls Hintergrundbilder und Datenpunkt-Fenster vorhanden sind

2_import-schemata.sh amev	AMEV_HK1_image001.gif
3_import-fenster+symbole.sh amev	INSERT_2-FENSTER.SQL
	INSERT_2-FENSTERSYMBOLS.SQL
5_import-schemata+fenster.sh amev	wie (2) + (3)

Danach ist die Datenbank bestückt.

### 5.1.5 Erzeugen eines GA-Knoten aus dem Installationsverzeichnis

Der GA-Knoten ist für das Beispiel *AMEV* noch NICHT installiert, also selber machen!

In einem *Terminal* werden folgende Anweisungen eingetippt

```
cd /home/lzh/gak_import/amev    wechselt ins Installationsverzeichnis
su root                        dann das Passwort für root eintippen
erzeuge_gaknoten             eintippen
```

**erzeugt den GA-Knoten als Benutzer amev**

Das Skript erzeugt den GA-Knoten als Benutzer *amev*.

Es stoppt für die 2-malige Eingabe des Passwortes für den GA-Knoten Benutzer.

```
@amev                z.B. als Passwort (merken!)
```

Es wurden alle erforderlichen Dateien aus dem Installationsverzeichnis übernommen und der GA-Knoten Benutzer *amev* installiert.

Nach dem erforderlichen Neustart öffnet sich immer der Desktop des Benutzers *lzh*.

### 5.1.6 Wechseln zum GA-Knoten Benutzer

Via Desktop des Benutzers *lzh* wird via Schaltfläche

*Session Logout Dialog* → *Benutzer wechseln*

der Auswahldialog für alle installierten Benutzer sichtbar.

```
GA-Knoten amev                auswählen
```

Der Desktop des GA-Knoten Benutzers *amev* wird analog dem Benutzer *lzh* eingerichtet.

Via *Terminal* *htop* oder *ps -x* sind alle Prozesse sichtbar, so auch die *gak.exe*.

### 5.1.7 Das ‚lila Fenster‘ des GA-Knotens

Das ‚lila Fenster‘ des GA-Knotens wird sichtbar via

Desktop: Schaltfläche *GA-Knoten* öffnen mit Doppelklick

Terminal: *screen -x* eintippen

Geschlossen wird das ‚lila Fenster‘ mit dem (x) in der oberen Ecke rechts.

Beendet und gestartet wird der Prozess *gak.exe* mit

```
stopj                beendet gak.exe (stop ja)
exit                 startet gak.exe (Schleifenausgang im Skript gakstart)
```

Falls der GA-Knoten abgestürzt ist z.B. wegen fehlerhafter *gak.cfg*, so hilft

```
ps -x                listet alle Prozesse mit PID, TTY, STAT, TIME, COMMAND
kill [pid]           löschen via pid die Prozesse gak.exe, gakstart, screen...
```

<sup>23</sup> Via Desktop *pgAdmin III* öffnen, anmelden, Menü *SQL* öffnen und Inhalt der SQL-Datei via *Copy/Paste* ausführen.

```
screen -l          startet einen screen
gakstart          startet den GA-Knoten (das 'lila Fenster' erscheint)
```

### 5.1.8 Die Parameter-Dateien im GA-Knoten

Die Verzeichnisstruktur im GA-Knoten beginnt immer mit dem Objektnamen

- /home/**amev**/ (beginnt mit dem Objektnamen)
- /home/**amev**/gak/cfg/ (GA-Knoten Unterverzeichnis)
  - anrufer.chk Anruferliste (bei Bedarf anpassen)
  - db\_atb.txt Attribute (Meldetexte der Datenbank)
  - db\_dim.txt Dimensionen (Einheiten der Datenbank)
  - gak.cfg Ablaufdatei (Konfiguration der Datenpunkte)
  - amev\_1.ref Referenzdatei (nicht erforderlich)
- gak/iz/iz04ddev.bn BACnet Netzwerk (falls vorhanden)
- gak/lz/vbp/vbp\_AMEV.001 Verbindung zur Datenbank *LZH*

Diese Parameter-Dateien

- sind Text-Dateien oder sogenannte Laufzeitdateien und genau auf Spaltenpositionen fixiert!
- sollten mit dem Texteditor *gedit* überprüft werden, falls der GA-Knoten nicht läuft.

Das Werkzeug *GA-Plattform\_nn.ods* ist als download auf *AMEV-GA-Plattform.de* verfügbar. Damit wird vieles nahezu automatisiert erzeugt (mit Beispielen von vielen MSR-Fabrikaten). Für *MSR*-Fabrikate mit *BACnet-Protokoll* kann via Tabellen-Kalkulation sehr einfach aus den *BACnet* CSV-Dateien *EDE*, *StateText*, *Units* eine eindeutige Referenzdatei erzeugt werden. Fragen Sie uns...

### 5.1.9 Die serielle Schnittstelle RS232

Die serielle Schnittstelle ist *RS232*<sup>24</sup> bzw. via Adapter *RS232/RS485* (siehe auch Anhang B). Da oft mehrere Schnittstellen physikalisch auf einem Mainboard vorhanden sind, kann die „Richtige“ mit einem „*RS232 Check Tester*“ vorab ermittelt werden.

Internet: "*RS232 DB25 Male/Female 18 LED Multi-Line Status Tester Adapter, TS-R18*"

Die Schnittstelle *RS232* wird in der Datei *gak.cfg* in der IZ-Zeile definiert als

```
/dev/ttySi      für i=0,1,... (abhängig von der Anzahl)
```

Also einfach ausprobieren, indem der Tester auf die jeweilige Schnittstelle gesteckt wird.

Zeigen die Dioden 2:TD und 4:RTS Dauerlicht, so ist die Schnittstelle verfügbar.

Zeigen die gegenüber liegenden Dioden 2:TD und 4:RTS auch kurzes Blinklicht, so ist der GA-Knoten *gak.exe* aktiv und die Einstellung in der *gak.cfg* ist richtig.

In der *gak.cfg* ist die IZ-Zeile für *i=0* /dev/ttyS0 (Standard)

In der *gak.cfg* ist die IZ-Zeile für *i=1* /dev/ttyS1

Werden mehrere Schnittstellen *RS232* verwendet, so sollten für den Test die jeweils anderen GA-Knoten gestoppt werden.<sup>25</sup>

<sup>24</sup> BACnet benötigt eine separate Netzwerkkarte.

<sup>25</sup> GA-Knoten öffnen z.B. via *ssh* oder *PuTTY* und den GA-Knoten mit *STOPJ* beenden.

## **A Anhang: AMEV-GA-Plattform.de**

Ab 2020 ist die Webseite *AMEV-GA-Plattform.de* im Internet verfügbar. Alle Updates werden immer dort abgelegt (jeweilige Version beachten).

Ein Gebäude-Automations-Server (*GA-Server*) kommuniziert mit Gebäude-Automationsstationen (*AS*, auch Schaltschrank genannt oder auch *DDC*) in den Liegenschaften.

Das besondere an einem *GA-Server* ist, dass er gleichzeitig sowohl *GA-Knoten* als auch *Zentrale LeitTechnik (ZLT)* ist. Es können mehrere *GA-Knoten* gleichzeitig installiert werden mit unterschiedlichen *MSR*-Fabrikaten.

Ebenso können Zählvorrichtungen unabhängig vom installierten *MSR*-Fabrikat via Schnittstelle aufgeschaltet werden.

Wird z.B. vor Ort ein Monitor installiert (Touchscreen) oder ein Laptop angeschlossen, so kann der Benutzer dieses Vorort-System genauso bedienen wie die übergeordnete Zentrale LeitTechnik.

Die Bediensoftware ist die *jLZHview* unter *Eclipse Java*: eine detailreiche Visualisierung von Schemata, Tabellen und Zeitreihen mit Statistik sowie Zählerverbrauchsanalysen zum

- überwachen von Betriebsmeldungen, Grenzwerten, Datenübertragung
- stellen und schalten von Betriebszuständen
- analysieren des Anlagenbetriebs mit einer Vielzahl numerischer Methoden,
- wie Ableitungen, gleitende Mittelwerte, Interpolationen, Integrationen, Korrelationsmatrizen, Eigenpaare etc.

Das gesamte System der Hard- und Software ist fernwartbar.

Ein Zugriff auf die *GA-Server* als auch auf die *Zentrale LeitTechnik* kann

- via *jLZHview* (Desktop Server-Client)
- via *jLZHweb* (Web Browser)
- via *SSH-Client (Terminal)*

installiert werden.

Die implementierte Benutzerverwaltung ist datenbankgestützt und kann sehr kleinteilig parametrisiert werden: so hat beispielsweise ein Benutzer XY nur das Recht ein bestimmtes Objekt oder ein bestimmtes Schema etc. zu sehen.

Die *AMEV-GA-PLATTFORM* unterstützt eine Vielzahl von firmenspezifischen Protokollen der Gebäude-Automationsstationen und wird federführend von der Landeshauptstadt München (LHM) weiterentwickelt sowie vom AMEV Arbeitskreis Gebäudeautomation (*AMEV-AK-GA*) unterstützt.

Die lizenzkostenfreien IT-Installationen der *AMEV-GA-Plattform* unterstützen Protokolle wie z.B. *BACnet*, *GfR*, *Kieback & Peter (3200 Reihe, P90)*, *LON*, *MODBUS*, *SAIA*, *Sauter (EY2400, EY3600)*, *TREND*, *WAGO*, etc. (*Smart-Home* ist in Vorbereitung).

Die Systeme *GA-Server*, *jLZHview* und *jLZHweb* (für den Zugriff via Web-Browser auf *GA-Server* sowie *ZLT*) wurden von Energie- & Umweltüro e.V. entwickelt.

Das Gesamtsystem kann komplett lizenzkostenfrei installiert werden.

Das *Energie- & Umweltüro e.V.* bietet bei Bedarf Unterstützung.

Einen fetten Dank an die *Open-Source'lerInnen*, und insbesondere an die

- MitarbeiterInnen des Vereins für die Entwicklung dieser Software
- Landeshauptstadt München, Baureferat, für das firmenneutrale Datenübertragungssystem der Zentralen LeitTechnik, eine Software-Entwicklung des vom AMEV spezifizierten FND-Konzeptes
- *Eclipse*-Gemeinde für das tolle Entwicklungswerkzeug
- *PostgreSQL*-Gemeinde für diese wunderbare lizenzkostenfreie Datenbank

Für die vielen Anregungen und für die Unterstützung danken wir allen Mitgliedern aus dem *AMEV-Arbeitskreis-Gebäudeautomation* sowie allen Unterstützern dieses Projektes.

Die verwendeten Gebrauchs- oder Warenzeichen etc. können Marken sein und als solche gesetzlichen Bestimmungen unterliegen.

## B Anhang Linux Nomenklatur

### Allgemeines

Hier werden einige häufig verwendete *Terminal* Befehle gelistet um dem Nichtexperten und auch dem Experten das Leben nicht zusätzlich zu erschweren. Wer mehr wissen will findet alle *Linux* Befehle im Internet.

HINWEIS: Beim Bootvorgang kann es vorkommen, dass das Datum in der Zukunft oder weit in der Vergangenheit liegt, oder es wurden Festplattenfehler (*badblocks*) gefunden.

Beim Datum ist es für den Nichtexperten noch einfach: das angezeigte Datum einfach im BIOS eintragen und neu starten. Dabei den *CMOS* Batteriewechsel nicht vergessen.

Bei Festplattenfehlern ist ein *reboot* gefährlich, denn das System kommt dann meist nicht mehr hoch. *Linux* Systeme laufen oft jahrelang ohne ein *reboot*. Als Folge dessen wird dann beim Booten immer die Festplatte geprüft. Daher ist vor dem *reboot* ein Festplattentest als Benutzer *root* mit dem Befehl *badblock -sv /dev/sda* durchzuführen.

Produziert dieser Test Lesefehler, so ist die Festplatte defekt und zu ersetzen. Alle für den Anwender wichtigen Dateien sind spätestens jetzt so weit wie möglich zu sichern, falls nicht schon regelmäßige Sicherungen durchgeführt wurden.<sup>26</sup>

Das Betriebssystem muss im Fehlerfall sowieso auf neuer Festplatte aufgebaut werden.

Im Fehlerfall stellt sich die Tastatur beim Bootvorgang oft auf eine US Tastatur. Wo befindet sich jetzt auf der deutschen Tastaturbeschriftung das äquivalente US Zeichen?

GER		US
@	→	Shift+2
/	→	-
-	→	ß
y	→	z
=	→	` (Akzent oben rechts neben backspace Taste)

### Terminal öffnen

Gnome Terminal öffnen	<i>Strg+Shift+t</i> (oder <i>Strg+Alt+F1</i> bis <i>F5</i> ) oder via Kontextmenü: <i>Im Terminal öffnen</i> <i>Strg+Alt+F7</i> zurück zur Gnome-Oberfläche
<i>su amev</i>	wechselt zum Objekt <i>amev</i> (Beispiel)
<i>xterm</i>	Eingabe öffnet das X-Terminal
<i>screen -x</i>	öffnet den Bildschirm des GA-Knoten (lila Farbe)

### Die Bash: der Kommandozeileninterpreter

Alles lässt sich mit der *Bash* erledigen, aber das erfordert schon etwas Übung!

<b>history</b>	anzeigen aller eingetippten Kommandos (Befehle)
<i>history -c</i>	löscht die <i>history</i> ( <i>root</i> )
<i>echo \$HISTSIZE</i>	maximale Anzahl der Zeilen
<i>echo \$HISTFILE</i>	Ort und Name der <i>Bash</i> (versteckte Datei)
<i>export HISTSIZE=500</i>	ändert die maximale Anzahl der Zeilen auf 500
<i>Strg +R</i>	Tastatur Kürzel für Suche in der <i>history</i> Anfangsbuchstabe für den gesuchten Befehl tippen

<b>[Tab]</b>	vervollständigt die Eingabe falls bekannt
<i>ls /e + [Tab]</i>	listet das Verzeichnis <i>/etc/</i>
<i>ls /etc/X + [Tab]</i>	listet das Verzeichnis <i>/etc/X11/</i>
<i>ls /etc &gt; etc_inhalt.txt</i>	listet das Verzeichnis in Datei (Umleitung)
<i>ls /etc &gt;&gt; etc_inhalt.txt</i>	dto. jedoch mit anhängen an bestehende Datei

<b>Verkettung</b>	von Befehlen
<i>cd bin; ls &gt; bin_inhalt.txt;</i>	<i>cd ~;</i> 1.Wechselt ins Verzeichnis <i>bin</i> 2.Listet das Verzeichnis in Datei 3.Wechselt zurück

<sup>26</sup> Oft ist es sinnvoll einen täglichen oder wöchentlichen *cron* mit diesem Befehl einzurichten und das Ergebnis in einer rotierbaren *log*-Datei aufzuzeichnen.

## Pipes von Befehlen

Die Übergabe an weiteren Befehl mit dem Pipezeichen |  
ls -l | less Ausgabe ist zu lang für das Terminalfenster  
ps -x | grep kp | less dto. jedoch mit Suche (grep) nach dem Text kp  
Ausgabe beenden mit dem Zeichen q

## Umbenennen

mv <alter name> <neuer name>

## Netz

GA-Server 8-er Netz mit Netzmaske 29 (29 Bits gesetzt)  
GA-Knoten mit lzh 172.31.10.90 (Beispiel)  
Das Netzwerk beginnt mit 172.31.10.88 und endet mit 172.31.10.95  
(Mask 255.255.255.248 oder kurz /29 setzen)  
GA-Server mit lzh virtuell 172.31.10.51 Netz von 100.92 bis 100.95 frei  
Konvention: nächst höhere IP nehmen  
(wird derzeit nicht verwendet)  
ping 172.31.ccc.ddd 5 ping 5 Mal ausführen (count=5)  
ip addr zeigt alle IP's  
ip route Routing  
ip -h Hilfe  
ip link MAC Adresse  
ip neigh MAC Adresse Nachbarschaft  
netstat -npl offene Ports  
zypper install net-tools (falls netstat nicht installiert)  
lsof -i | grep -e LISTEN offene Ports (wie vor)  
nmap -sT 192.168.0.4 Portscanner (hier TCP/IP von BACnet Sauter)  
zypper install nmap (falls nmap nicht installiert)  
rcSuSEfirewall status status stop  
  
/sbin/ip addr show eth0 | awk -F"[ /]+" '/inet / {print \$3}'  
getent hosts "\$(hostname)" | awk '{ print \$1 }'  
Seriell RS232 cat /proc/tty/driver/serial  
für RS485 wird ein Adapter auf RS232 gesteckt.

## Windows

via cmd starten Shell öffnen  
ipconfig /all zeigt alle Netzverbindungen  
ping aaa.bbb.ccc.ddd ping

## Zeiten

sntp 172.31.10.10 synchronisieren z.B. mit EUB etc.  
sntp pool.nt.org ptbtime1.ptb.de, ptbtime2.ptb.de, etc.  
**date -s 'Jan 1 12:00:00 2013'** **Systemzeit setzen (Datum komplett)**  
**hwclock -w** **schreiben von date in die HardwareClock (BIOS)**  
hwclock -s auslesen von date aus der HardwareClock (BIOS)  
date -s 'now -2 min' Kurzform (2 Minuten zurück)  
date -s 'now +2 hours' Kurzform (2 Stunden weiter)

## Sonstiges

localhost:8080/manager/html Tomcat6 Manager aufrufen (im Browser)  
Prioritäten ändern i.a. nicht erforderlich (negative Zahl ist hoch)  
renice -n -12 -u gak Priorität -12 gak.exe (GA-Knoten, viele DP)  
renice -n -15 -u postgres Priorität -15 postgres (Datenbank)  
renice -n -10 -u tomcat Priorität -10 tomcat (Webserver)  
  
df zeigt Dateisystem Info  
free -m zeigt Speicherauslastung  
history zeigt Befehlshistorie (-c löschen)  
sensors zeigt Temperaturen, Volt...  
top zeigt Prozesse an (man top)  
htop zeigt Prozesse an mit vielen Details  
tail /var/log/messages zeigt Fehlerprotokoll (als root ausführen)

## Passwörter, Rechte, Benutzer

Passwörter der Installation

BN:root	PW:@@@"	System
BN:lzh	PW:@lzh	Benutzername von lzh, ftp
BN:objekt	PW:macho	GA-Knoten BN: amev
BN:postgres	PW:lzh.wks	Postgres Datenbank

### Alle Passwörter lassen sich ändern!

Vorschlag für Postgres-Datenbank Benutzer

BN:postgres	PW:lzh.wks	kann alles
BN:lzh0	PW:lzh.wks	kann alles
BN:lzh1	PW:lzh.wwi	kann schalten und stellen
BN:lzh2	PW:lzh2	kann stellen
BN:msr	PW:msr	Wartung durch MSR
BN:pforte	PW:pforte	verwaltet nur aktuelle Meldungen
BN:gast	PW:gast	kann nur gucken

Diese Benutzer werden automatisch via jLZH erstellt.

pw zeigt alle Passwörter  
passwd ändert Passwort (hier von root)

su -amev switch user wechselt zum Benutzer amev

**chmod** 755 /datei ändert den Zugriff für datei

**chown** lzh:users /datei ändert den Besitzer für datei  
chown postgres:users /backup ändert den Besitzer für backup

logname zeigt den Benutzer  
users zeigt alle Benutzer

**userdel -rf amev** löscht den Benutzer amev  
(-r lösche Verzeichnis, f:force)  
/etc/OBJEKTE der Eintrag amev ist in der Datei zu löschen!  
/etc/HOSTNAME der Eintrag des Host via YaST Netzwerk

who zeigt alle angemeldeten user  
pkill -KILL -u user tötet den Prozess und loggt den user aus  
und ist via YaST Benutzerverwaltung entfernbar.  
ps aux | grep <program> liefert Prozess-ID von <program>  
kill -9 <PID>

## Routinen und Programme

edit crontab einfacher Editor öffnet hier die crontab  
beenden mit :q (Tasten : und q für quit)

find -name dateiname suche dateiname  
ls -la /etc/init.d listet alle Startdateien in init.d  
mkdir /pfad/name Im pfad-Verzeichnis name erstellen  
z.B. mkdir -v -m 700 /lzh-daten/backup

rmdir Verzeichnis löschen  
ps -x listet alle Prozesse  
kill [pid] tötet Prozess mit pid-Nummer  
screen -l öffnet screen (lock)  
screen -x öffnet hier das Fenster kp.exe bzw. gak.exe  
kpstart startet den kp.exe Prozess (Leitzentrale)  
gakstart startet den gak.exe Prozess (GA-Knoten)  
xterm öffnen

## Routinen und Programme und Skripte in Verzeichnissen

/etc/OBJEKTE Datei für aktives Objekt (mehrere sind möglich)  
(nicht aktive Objekte entfernen, nach reboot)

/etc/HOSTNAME Datei für den host

/home/lzh/bin/ Verzeichnis Benutzer lzh

```

gakimport.exe, kp_pg_akga.exe, kpstart, kp.conf,
filter.exe, oem2iso.exe, cfi_pg.exe
import-datenpunkte.sh, import-fenster.sh,...
/home/lzh/bin/jLZHview      jlzhview      Startverzeichnis
/home/lzh/bin/jLZHweb      jlzhweb       Installationsverzeichnis
/home/lzh/bin/postgres/    pg_vacuum.sh  entlüftet die DB
                           pg_vacuum_table-dp.sh entlüftet DATENPUNKTE
                           pg_dumpall.sh  dump der DB (händisch)
/lzh-daten/pgdata01/pg_hba.conf edieren als user postgres

/home/lzh/bin/sensors/    sensor_values.sh
/home/lzh/bin/Transfer_MW/ Transfer_MW.sh
                           synchronisiert DB Tabelle MESSWERTE
                           mit anderen Datenbanken      z.B. ZLT
/home/lzh/bin/Transfer_Table/ Transfer_Table.sh (nur mit java-1_6_0 openjdk)
                           synchronisiert DB Tabellen
                           mit anderen Datenbanken      z.B. ZLT

/usr/bin/                  putty -ssh -l $USER $HOST (-l ist kleines L)

/usr/local/bin/           Verzeichnis für:
clearNXuser               Benutzer löschen (maximal 2 user frei)

erzeuge_gaknoten       erzeugt den GA-Knoten Benutzer xxxx
                           Aufruf im Verzeichnis (z.B. xxxx:=amev)
                           /lzh/gak_import/amev/erzeuge_gaknoten

serial_term               serielle Schnittstelle
set_gak_serial            ttyS0, ttyS1, ttyS2,...

/usr/local/pgsql/bin/     Verzeichnis für postgres, psql, vacuumdb,...
/srv/ftp/                 Repository für Schemata etc.
/user/NX/etc/users.db     Userdatei von NX

```

## SecureShell

Falls keine grafische Oberfläche vorhanden ist.

**ssh remote\_username@remote\_host**

ssh lzh@172.31.10.90 ssh Verbindung herstellen (Beispiel)

**scp <source> <destination>**

Datei von einem Rechner zu einem anderen Rechner  
in das Verzeichnis Downloads

scp firefox.txt lzh@172.31.10.50:/home/lzh/Downloads (Beispiel)

## Tape archive

**tar -zxvf <file.tar.gz>**

tape archive <file> gezippt auspacken  
(z:gz Format, x:extract, v:verbose, f:file)  
oder (j:bz2 Format, J:xz Format)

## RPM Pakete

**rpm -i <file.rpm>**

auspacken (als root)

**rpm -e <file.rpm>**

deinstallieren

## Dienste

/etc/init.d/kp

via YaST auf 3 und 5 setzen

/etc/init.d/kpstart

startet den Screen mit dem KP Prozess

/etc/init.d/gak

via YaST auf 3 und 5 setzen

/etc/init.d/vsftpd status

status|start|stop (s.a. YaST FTPServer Startup)

/etc/init.d/network status

status|start|stop|restart auch rcnetwork

/etc/init.d/ntp

status|start|stop NetworkTimeProtocol

/etc/init.d/nxserver status

status|start|stop|restart

/etc/init.d/postgres status

status|start|stop (pg hat port 5432 standard)

/etc/init.d/oracle

status|start|stop (ora hat port 1521 standard)

/etc/lzh.conf

0:deaktiv 1:aktiv bei Virtueller lzh (lzh-vm)

```

/etc/init.d/cron stop          cron stop bzw. Start
/etc/crontab (-rw-r--r--)    sendet alle 10 min die MESSWERTE zur ZLT

/etc/logrotate.d/           rotiert falls Datei > 256 kB ist (<8 Dateien)
    log_sensor_values       rotiert Sensordateien
    log_pg_vacuum           rotiert Vacuumdateien
    log_Transfer_MW         rotiert Transferdateien
logrotate -f /etc/logrotate.d reload syslog service
logrotate -f datei          erzwingt Rotation der datei
/etc/ntp.conf               NetworkTimeProtokoll
/etc/resolv.conf            falls Netzwerk nicht geht
/etc/resolv.conf.netconfig
/etc/ssh/sshd_config        Konfigurationsdatei für root sperren
                             PermitRootLogin no          (Zeile einfügen)
                             wird aktiviert
/etc/init.d/sshd reload
/etc/init.d/cron start      status|start|stop|restart

/etc/sysconfig/cron         lösche Verzeichnis /tmp bei Neustart
                             YaST->System->      Editor für /etc/sysconfig
                             System->Cron
                             CLEAR_TMP_DIRS_AT_BOOTUP    yes
                             MAX_DAYS_IN_LONG_TMP          1 (1 Tag)

/etc/sysconfig/language    via YaST Sprache (Häkchen setzen für Experten)

```

### Vereinfachte Aufrufe

```

rccron status              status|start|stop|restart
rcnetwork status          status|start|stop|restart
rctomcat6 status          status|start|stop|restart

```

### Systemmeldungen Verzeichnisse

```

/var/log/messages          System Meldungen
/var/log/warn              System Warnung
/var/log/tomcat6/catalina.out Tomcat6
xterm -e tail -f /var/log/messages dto. die anderen (user root)
tail -f -n 5 /var/log/syslog nur die letzten 5 Zeilen

```

### Systemroutinen

```

badblocks -sv /dev/sda     prüft device sda auf Fehler
nut                        USV Anbindungen (bei Bedarf installieren)
pfSense                    Router auf Linux (bei Bedarf installieren)
IPFIRE                     dto.

```

### Prozesse beenden

```

ps aux | grep gak         gibt Prozess-ID zurück von gak.exe
ps -x                     zeigt Prozesse des Benutzers
kill -123 <PID>          killt Prozess 123

```

### PostgreSQL läuft nicht

```

Falls kp.exe oder pgAdminIII nicht startet:
su - postgres             wechselt zum Benutzer postgres
/usr/local/pgsql/bin/postmaster -D /lzh-daten/pgdata01/
Falls Fehlermeldung: den Text genau lesen!
Falls „...or just delete the file "/lzh-daten/pgdata01/postmaster.pid"
dann händisch löschen    >rm /lzh-daten/pgdata01/postmaster.pid
Ausführlich siehe Abschnitt „pgAdmin III bzw. kp.exe arbeitet nicht“

```

### PostgreSQL Datenbank Werkzeuge

Die Datenbank hat hier den Namen **lzh** mit Eigentümer **postgres** und wird ausgeliefert in der Kodierung

- o **LATIN9** (Standard, da mit Sonderzeichen)
- o **UNICODE** (keine Sonderzeichen wie ä, ü, ... z.B. aus Windows)

Eine Änderung der Kodierung in UNICODE wird weiter unten beschrieben.

HINWEIS: der GA-Knoten Prozess *gak.exe* ist in *UNICODE* implementiert, also zwingend auch die symbolische Adresse. (siehe *unicode-table.com/de/*) Die Beschreibung kann jedoch in *LATIN9* implementiert werden: nur der GA-Knoten zeigt falsche Zeichen, was vernachlässigbar ist.

```
/etc/init.d/postgres status status|start|stop
su postgres wechselt zum Benutzer postgres PW:lzh.wks
psql -l listet die Datenbank(en)
createdb lzh erstelle die Datenbank lzh
pg_dump lzh > /x/lzh.out Dump der Datenbank lzh erstellen in /x
psql -d lzh -f /x/lzh.out Dump der Datenbank lzh einspielen aus /x

pg_dumpall > /x/all.db.out Dump aller Datenbanken erstellen in /x
psql -f /x/all.db.out Dump aller Datenbanken einspielen aus /x
```

**Indizes Postgres bearbeiten** (siehe *pgAdmin III: Der PostgreSQL Administrator*)  
*kp Prozess* mit *STOPJ* beenden, falls messwerte und datenpunkte reindiziert werden  
SQL-Editor in pgAdmin öffnen und dort eintippen:  
REINDEX INDEX <name> Name des PostgreSQL Index <name>  
REINDEX INDEX mw\_erb Beispiel <name>:= mw\_erb  
oder  
DROP INDEX mw\_erb; löschen  
CREATE INDEX mw\_erb neu erstellen  
ON messwerte USING btree (erfassungszeit);

**Indizes Oracle bearbeiten (nur als Hinweis)**  
ALTER INDEX <name> REBUILD Name der ORACLE Tabelle <name>  
ALTER INDEX MW\_ERF REBUILD z.B. den Index MW\_ERF in TABLESPACE INDEXES

**Die psql Terminal öffnen** (BN:root)  
psql -h localhost -U postgres lzh  
HOST user Datenbank  
Datei im Terminal einspielen  
\! In welchem Verzeichnis bin ich?  
\o Ausgabedatei (out)  
\i lzh.sql Datei importieren aus Verzeichnis

**Beispiel Dump erstellen** einen dump der Datenbank *lzh* erstellen  
cd /lzh-daten/pgdata01/ in diesem Verzeichnis  
psql -l listet die Datenbank(en)  
pg\_dump lzh > lzh.out ins Verzeichnis /lzh-daten/pgdata01/lzh.out  
Installations-Passwort lzh.wks

**Beispiel Dump einspielen** aus Verzeichnis /lzh-daten/pgdata01/lzh.out  
psql -d lzh -f /lzh-daten/pgdata01/lzh.out

**backup Verzeichnis erstellen**, falls nicht vorhanden  
cd /lzh-daten/ ins Verzeichnis gehen und gucken  
su als Benutzer root anmelden  
mkdir -v -m 700 /lzh-daten/backup Verzeichnis erstellen  
chown postgres:users /lzh-daten/backup/ Besitzer ändern auf postgres  
su postgres als Benutzer postgres anmelden  
psql -l listet die Datenbank(en)

#### **Änderung der Kodierung in UNICODE**

Eine neue Datenbank wird nach folgender Reihenfolge erstellt

- (1) *kp Prozess* stoppen
- (2) **Dump erstellen** und Datenbank Eigenschaften merken/aufschreiben/Foto
- (3) bestehende Datenbank gegebenenfalls via Kontextmenü Löschen/Dropfen
- (4) neue Datenbank erstellen via pgAdmin (Auswahl ist UNICODE)  
Im Baum des pgAdmin die Schaltfläche [Datenbanken] markieren und  
via Kontextmenü den Eintrag [Neue Datenbank...] öffnen  
und die Kodierung wählen z.B. UNICODE
- (5) **Dump einspielen**
- (6) *kp Prozess* starten

#### **SQL-Export der Daten einer Tabelle als user lzh (Schreibrechte)**

Als Benutzer *lzh0* mit Passwort *lzh.wks*

**Variante mit Header: Beispieltabelle schemata**

```
/usr/local/pgsql/bin/pg_dump --table=schemata --data-only --column-inserts
-U lzh0 lzh > /home/lzh/schemata-mitheader.out
```

**Variante ohne Header: Beispieltabelle schemata**

```
/usr/local/pgsql/bin/pg_dump --table=schemata --data-only --inserts
-U lzh0 lzh > /home/lzh/schemata-ohneheader.out
```

In Datenbank einspielen via SQL-Editor.

**PostgreSQL pg\_hba.conf**

Bei einem GA-Server wird in der Datei pg\_hba.conf der Zugang zur Datenbank parametrierd (als BN root oder BN postgres)

Voreingestellt ist in der Datei

```
/lzh-daten/pgdata01/pg_hba.conf
host all all 172.31.0.0 255.255.0.0 md5
```

Bei einem anderen Netzwerk wird der host ersetzt durch (Beispiel)

```
host all all 10.102.0.0 255.255.0.0 md5
```

Bei einem ZLT-Server mit GA-Servern in unterschiedlichen Netzen, wird für jedes Netzwerk die entsprechende Zeile angehängt, z.B. im obigen Fall sind es beide Zeilen.

**Ports**

Ports	default	Geändert...
ftp	21	sperrern (im Router)
ssh, NX	22	65480 (im Router)
nomachine4	4000	NX Protokoll
tomcat6	8080	
PostgreSQL	5432	
Oracle	1521	
ptbtime1	123	
no-ip.com	53	
Router-GA-Server	80	
Router-ZLT	80	9595
BACnet	47808	BAC0 (hex)
ZLT/LZH (kp.exe)	4403	
GA-Knoten (gak.exe)	4404	4405, 4406,... hier für 2 weitere GA-Knoten auf einem GA-Server

**Festplatten bearbeiten**

Lifesystem starten z.B. Ubuntu 12.0 ff und mit gparted prüfen. Welche Platte ist sda bzw. welche ist Platte sdb ?

**Numerical Permissions**

```
+-----+-----+-----+
| rwx | 7 | Read, write and execute |
| rw- | 6 | Read, write |
| r-x | 5 | Read, and execute |
| r-- | 4 | Read |
| -wx | 3 | Write and execute |
| -w- | 2 | Write |
| --x | 1 | Execute |
| --- | 0 | no permissions |
+-----+-----+-----+
```

Beispiel 744 Besitzer darf alles, die beiden anderen nur lesen.

### **Festplattenkopie erstellen**

Kopiert wird im Beispiel von Platte `sda` nach Platte `sdb` mit dem Befehl<sup>27</sup>  
Ausführen als Benutzer `root`

```
dd if=/dev/sda of=/dev/sdb bs=4096          kopieren von sda nach sdb
                                           if: input file
                                           of: output file
                                           bs: blocksize

kill -SIGUSR1 $(pidof dd)    Fortschrittanzeige (mehrfach aufrufbar)
```

### **Kopieren irgendwo hin, verpacken und wieder entpacken**

```
dd if=/dev/sda bs=16065b | pv | gzip -1 >/mnt/irgendwo/clone.gz
gunzip -dc /mnt/irgendwo/clone.gz |pv | dd of=/dev/sdb bs=16065b
```

### **Partition verpacken**

```
dd if=/dev/sda bs=4K | gzip > /PATH/DATEI.IMAGE.gz
```

### **Partition zurückschreiben**

```
zcat /PATH/DATEI.IMAGE.gz | dd of=/dev/sda2 bs=4K
(wird optimal auf der Festplatte gespeichert).
```

---

<sup>27</sup> Die Platte sollte mit Gerätenamen benannt sein z.B. `DISK0`, `Disk1`, ... spart späteren Ärger

## C Anhang: ITX Boards als GA-Server in Betrieb

Eine kleine Zusammenstellung der Hardware für einen GA-Server.  
Auch zum Selbermachen geeignet, da eine Gebrauchsanwendung beiliegt.

### C.1.1 Mainboards für 32Bit mit 2/4 Kernen und maximal 4GB Memory

Folgende lüfterlosen Boards mit RS232 Schnittstelle und LAN wurden getestet und verwendet.

- o D525MW 1xLAN 2xCOM S0,S1 (S0 ist außen)
- o D2500HN 1xLAN 2xCOM S0,S1 (S1 ist außen)
- o D2500CC 2xLAN 4xCOM S0,S1,S2,S3 (S3 ist außen oberer Stecker)  
(S4 ist außen unterer Stecker)
- o D2550B 1xLAN 1xCOM S0 (S0 ist außen)
- o D2550 1xLAN 1xCOM S0 (S0 ist innen, benötigt RS232 Kabel)

#### HINWEIS:

Bei allen Boards ist die Knopfzelle CR2032 zu testen (Lebensdauer ~5 Jahre)

#### Andere Boards:

- o 82574L Gigabit für ZLT
- o HP ProLiant DL380e Gen8-Server für ZLT
- o Virtuelle Maschine für ZLT

Hinweis: Manche Boards können die CPU und/oder die Temperaturen und Spannungen im Panel des Betriebssystems nicht anzeigen: die entsprechende Software nachinstallieren.

### C.1.2 Gehäuse

- o LC-Power LC-1350mi, mit Netzadapter 12V/5A
  - o JX-500 ohne Netzadapter 12V/5A (separat bestellen)
- Man achte auf *USB2* bzw. *USB3* des ITX Boards!

### C.1.3 Memory

- o DDR3 2GB Corsair etc. (4GB nur bei GA-Servern mit > 6000 Datenpunkten)

### C.1.4 Mini-Disk

Diese Laufwerke sind im Dauerbetrieb getestet.

- o SATA 2.5 z.B. 500GB Seagate BarraCuda
- o SSD 2.5 z.B. 250GB Seagate (inzwischen kostengünstig erhältlich)

### C.1.5 Was kostet ein kompletter GA-Server?

#### Hardware

- o Gehäuse JX-500 mit Netzteil
- o ITX-Board 4 x CPU D525MW 1.85 GHz mit 1 x LAN und 2 x COM S0,S1
- o Memory 2 GB
- o Disk 250 GB SSD 2.5

#### Software

- o AMEV-GA-Software (Aktueller Stand)

#### Zusammenstellung (Tagespreise)

1. Hard- und Software 500 EUR (netto ohne Versand)
2. Datenpunkte 2 bis 4 EUR (je Datenpunkt)  
(abhängig von den bereitgestellten Unterlagen)

## D Anhang: NonRaid zu (software)Raid1 SuSE 10 1

[http://wiki.linux-club.de/opensuse/NonRaid\\_zu\\_%28software%29Raid1\\_SuSE\\_10\\_1](http://wiki.linux-club.de/opensuse/NonRaid_zu_%28software%29Raid1_SuSE_10_1)

Falls mal ein Software Raid1 auf der Wunschliste steht:

Diese Methode kann **nachträglich** von einem laufenden System eine Spiegelplatte herstellen!

Ist Platte `sda` defekt, so übernimmt Platte `sdb`.

Das **Howto** runterladen. Im *Howto* ist jeder einzelne Schritt beschrieben. Alles via *Terminal!*

Die Partitionen sollten identisch gewählt werden, sonst sind viel *Experten* gefordert!

Das lief bei uns auf Anrieb mit der unten gezeigten Partitionierung auf SuSE 11.4 sowie auf SLES 11.3 unter Gnome Desktop.

Ein fettes Lob an die Leute vom `linux-club`!

Auszug aus der Konfiguration des Ausgangssystems (Partitionen):  
vor dem Anlegen des Raids

```
# fdisk -l          (L als Kleinbuchstabe)
```

```
Disk /dev/sda: 36.7 GB, 36703932928 bytes
64 heads, 32 sectors/track, 35003 cylinders
Units = cylinders of 2048 * 512 = 1048576 bytes
```

Device	Boot	Start	End	Blocks	Id	System	
/dev/sda1	*	1	65	66544	83	Linux	/boot
/dev/sda2		66	5186	5243904	83	Linux	/
/dev/sda3		5187	25667	20972544	83	Linux	/home
/dev/sda4		25668	35003	9560064	f	extended	
/dev/sda5		25668	27716	2098160	82	Linux swap	/ Solaris swap
/dev/sda6		27717	35003	7461872	83	Linux	/lzh-daten

```
Disk /dev/sdb: 36.7 GB, 36703932928 bytes
64 heads, 32 sectors/track, 35003 cylinders
Units = cylinders of 2048 * 512 = 1048576 bytes
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		1	35003	35843056	83	Linux

## E Anhang: SQL-Abfragen

Eine kleine Sammlung von SQL-Abfragen. Bei Bedarf ergänzen...

### E1.0 SQL-Import in die PostgreSQL-Datenbank

Via jLZH können alle Nutzerdaten in der Datenbank eingetragen, geändert bzw. gelöscht werden. Falls dies automatisiert werden soll, so sind SQL Skripte zu erstellen. (siehe auch den download *GA-Plattform-parametieren\_nn*).

Folgendes SQL-Skript kann via *pgAdmin* in die Datenbank importiert werden

```
/home/lzh/gak_import/AMEV_STRASSE+OBJEKT+GAKNOTEN+SCHEMA_INSERT.SQL
```

Die Einträge im Beispiel beziehen sich auf das Musterobjekt *AMEV* und sind bereits als Musterbeispiel in der Datenbank eingetragen (also vorher via *jLZHview* löschen).

**STRASSE:** Sind bereits einige Straßen vorinstalliert, so sind diese bei Bedarf zu löschen.

```
INSERT INTO strassen(  
nummer, name)  
VALUES (10, 'AMEV-Str.');
```

### OBJEKT: AMEV

```
INSERT INTO objekte(  
kennung, name, betreuer, status, strasse, hausnummer,  
ansprechpartner, telansprechpartner, bemerkungen,  
betreuer_vor_ort, telbetreuer_vor_ort,  
zentralenbezeichnung, informationsschwerpunkt, telglt, zuordnung,  
wartungsbereich, wartungsmeister, wartungsvertretung, einrichtungsname,  
ersteller, stadtbezirk)  
VALUES ('AMEV', 'AMEV-Str.', 'Fr.Dr.Kurde', 'in Betrieb', 1, '13',  
'EUB', '030 7871 7651', 'EUB AMEV-Str.13',  
'Fr.Pilar Munoz','030 7871 7651',  
'', '', 'Tel.: GLT','',  
'', 'Wartungsmeister:', 'Wartungsvertretung:', 'Einrichtung Büro',  
'EUB', 1);
```

### GAKNOTEN: AMEV\_1

Die Kennung ist per Konvention immer der Objektname mit dem Anhang \_1

```
INSERT INTO gaknoten(  
kennung, gagesperrt, gagestoert, gagestartet, einbauort,  
einbaudatum,  
hersteller, wartungsfirma, verbindungsart, verbindungsprotokoll,  
verbindungsadresse, verbindungsgesperrt, verbindunggestoert, verbindungaufgebaut,  
callingpartynumber, treibergesperrt, treibergestoert, treibergestartet,  
bemerkungen, cfg_kommentare)  
VALUES ('AMEV_1', 0, 0, 0, 'Beispiel AMEV',  
to_timestamp('2016-11-16 13:00:00','YYYY-MM-DD HH24:MI:SS'),  
'Hersteller: EUB', 'Wartung: EUB', 40, 2,  
'127.0.0.1#4404', 0, 0, 0,  
'127.0.0.1#4404', 0, 0, 0,  
'Bemerkungen: IP-Adresse mit Port ist aaa.bbb.ccc.ddd#4404  
IP localhost: 127.0.0.1#4404 ', 'CFG_Kommentar:...');
```

### SCHEMATA: AMEV\_HK1

```
INSERT INTO schemata(  
kennung, typ,  
beschreibung, dateiformat,  
dateiname, breite, hoehe,  
schema_oben, schema_rechts, schema_links, schema_unten,  
handbuch)  
VALUES ('AMEV_HK1', 'Anlagenplan Heizung',  
'WE: Fernwärme, Heizkreise, Heizkurven, Pumpen-Gateways, Justierungen', 10,  
'AMEV_APVO_image003.gif', 2400, 3331, '', '', '', '',  
'AMEV-Schule, Amevstr.13, 12345 Berlin  
AMEV DDC BACnet/Ersteller EUB');
```

### SCHEMATA: (händisch)

Alle Hintergrundbilder werden im *Ressourcen Verzeichnis* /srv/ftp/schema abgelegt.

Im Baum der *jLZHview* wird die Tabelle *SCHEMATA* geöffnet.

Via Kontextmenü *DB: Neueintrag* oder *Neueintrag mit Referenz* wird der Eingabedialog geöffnet.

In folgender Reihenfolge bearbeiten

(1) Dateiname	<i>AMEV_HK1_image001.gif</i>	via [...]	auswählen
(2) Kennung	<i>AMEV_HK1</i>		eintippen oder kopieren
(3) Typ	<i>Anlagenplan Heizung</i>	via [v]	auswählen
(4) Beschreibung	<i>WE: Fernwärme, Heizkreise</i>		eintippen
(5) Erzeugen			

Das Hintergrundbild ist jetzt in der Datenbank-Tabelle *SCHEMATA* gespeichert und kann via Menüleiste (Symbol *Pumpe*) geöffnet werden.

Wie wird das Hintergrundbild belebt?

- Im Baum der *jLZHview* wird die Tabelle *DATENPUNKTE* geöffnet.<sup>28</sup>
- Via Menüleiste (Symbol *Pumpe mit Bleistift*) wird der *Schema-Editor* geöffnet.
- In die Tabelle *DATENPUNKTE* werden die gewünschten Zeilen markiert und kopiert (Strg+c) und in das Hintergrundbild des *Schema-Editors* eingefügt (Strg+v).
- Mit der Maus oder den Pfeiltasten wird positioniert.
- Abschließend wird das Hintergrundbild mit den Datenpunkt-Fenstern gespeichert.

Via Menüleiste (Symbol *Pumpe*) wird das belebte Hintergrundbild geöffnet.

Im Beispiel steht in allen Datenpunkt-Fenstern der Text `NULL`, denn es gibt ja keine Messwerte.

HINWEIS: Ist ein Hintergrundbild bereits belebt, so können mit dem *Schema-Editor* die Datenpunkt-Fenster verschoben oder auch entfernt werden.

---

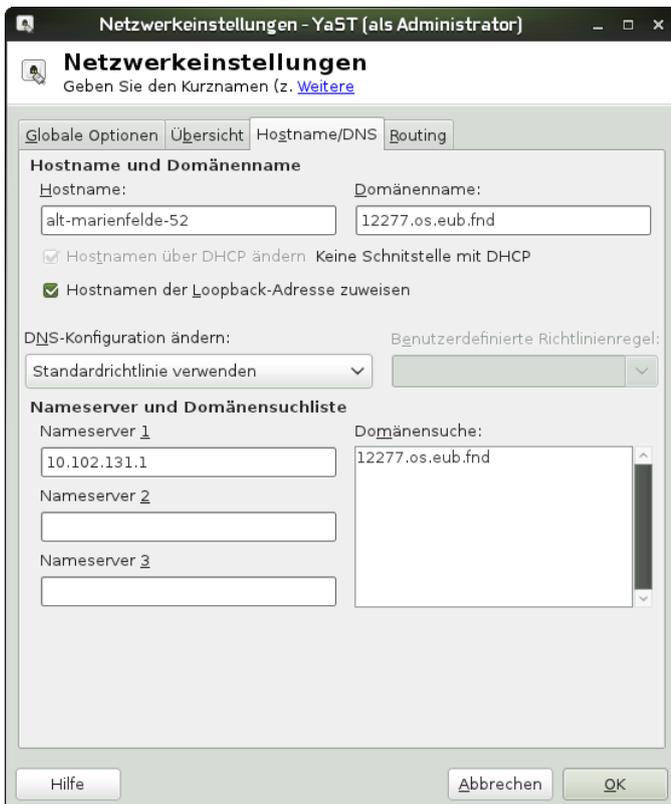
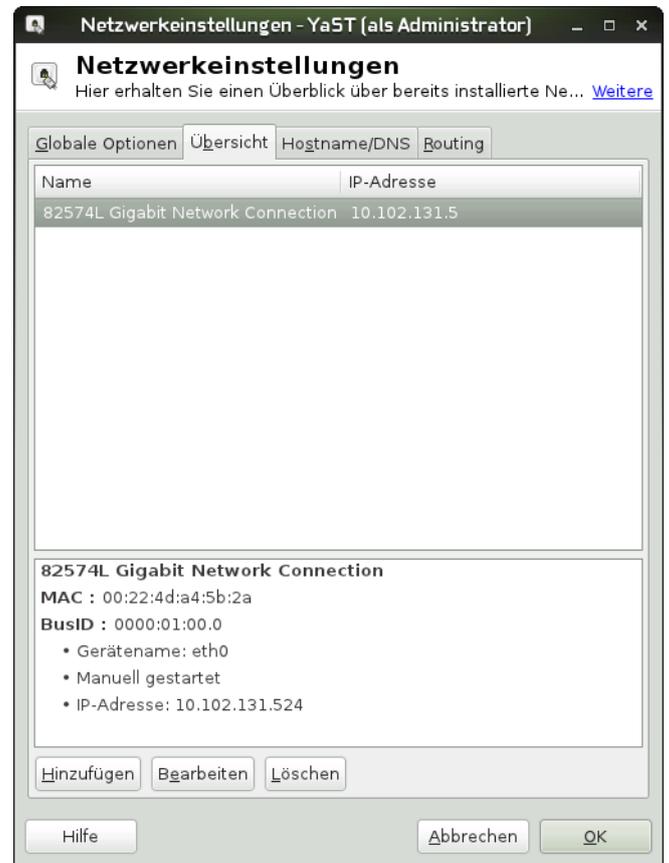
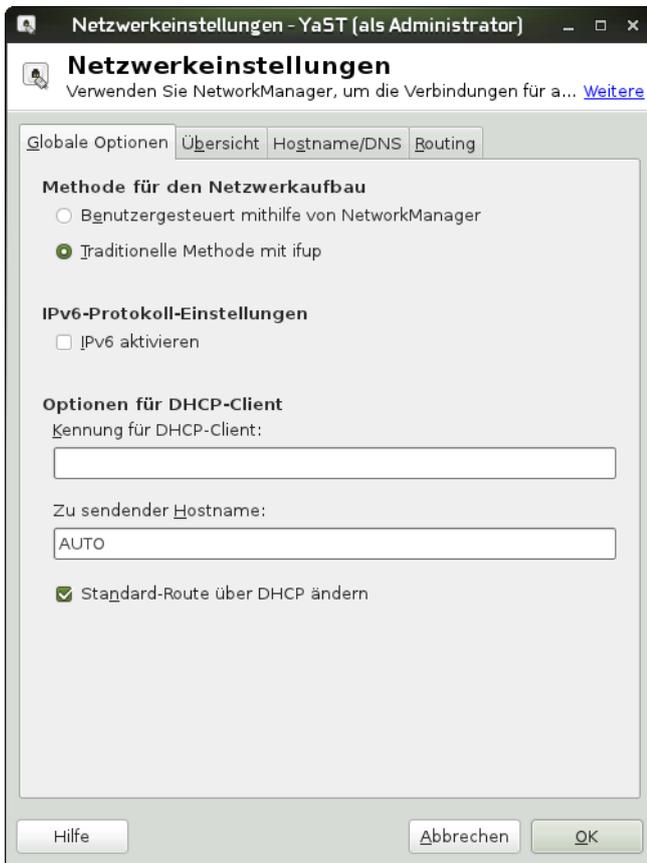
<sup>28</sup> Ist die Tabelle noch leer, so ist via *Terminal* folgende Anweisung auszuführen  
`1_import-datenpunkte.sh amev`

## F Anhang: Netzwerk und Zeitserver

### F1.1 Netzwerk Installation mit 1 Netzwerkkarte

Um 1 Netzwerkkarte zu aktivieren sind folgende Schritte via YaST auszuführen.

Beispiel: 10.102.131.5/24 eth0 im Netz 10.102.131.1 Genmask 255.255.255.0

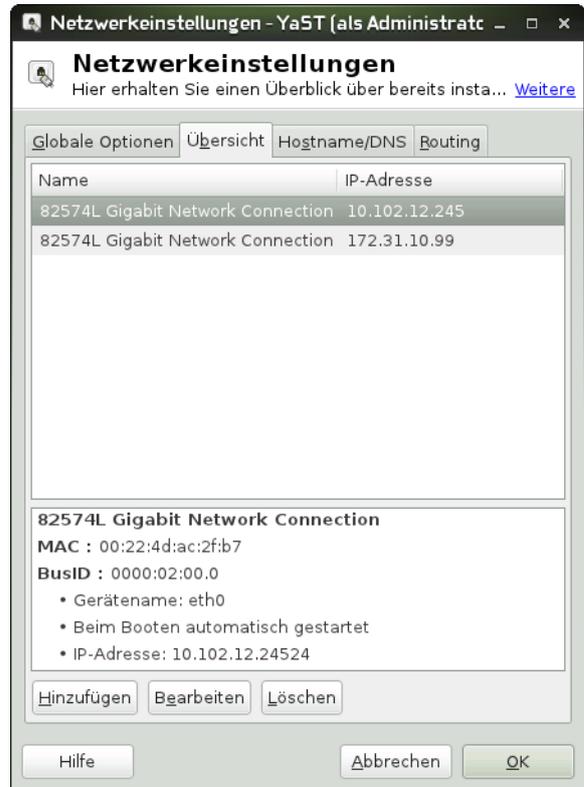
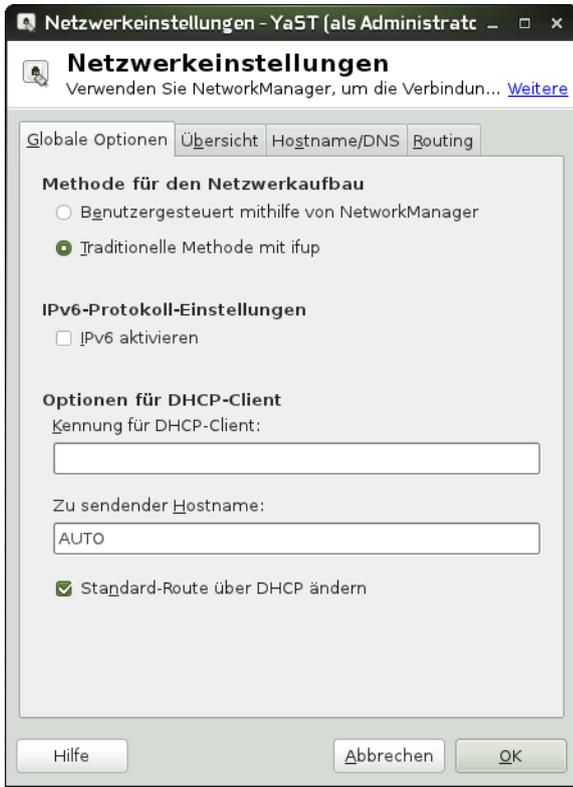


Gerät 'eth0' oder '-':beliebig

## F1.2 Netzwerk Installation mit 2 Netzwerkkarten

Um 2 Netzwerkkarten zu aktivieren sind folgende Schritte via YaST auszuführen.

Beispiel: 10.102.12.245/24 eth0 im Netz 10.102.0.1 Genmask 255.255.0.0  
 172.31.10.99/24 eth1 im Netz 172.31.10.1 Genmask 255.255.255.0

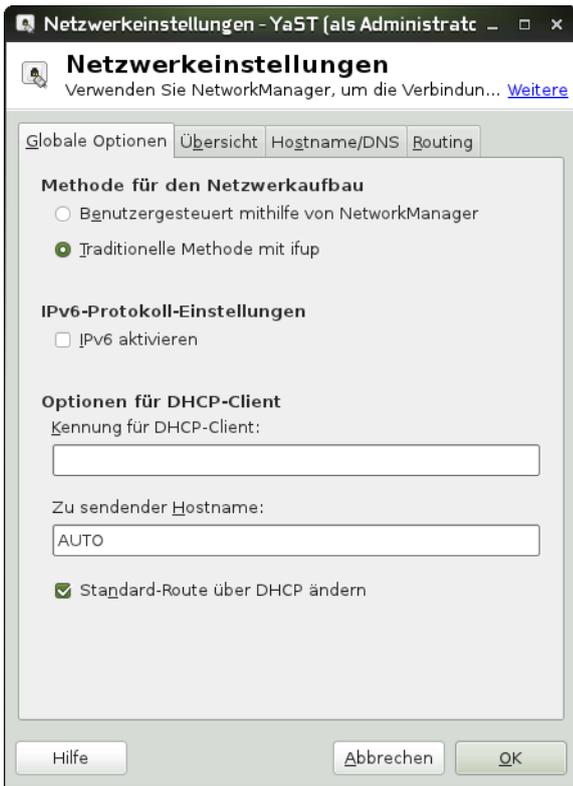


Wichtig ist das Routing mit dem Standard-IPv4-Gateway 172.31.10.1 welches ins Internet kommt. Die Routing-Tabelle zeigt auf einen zentralen Router ohne Internet mit den Subnetzen Ziel 10.102.0.0/24 Gateway 10.102.12.1 Genmask 255.255.0.0

### F1.3 Netzwerk Installation mit 2 Netzwerkkarten (Beispiel AMEV\_BAcnet)

Um 2 Netzwerkkarten zu aktivieren sind folgende Schritte via YaST auszuführen.

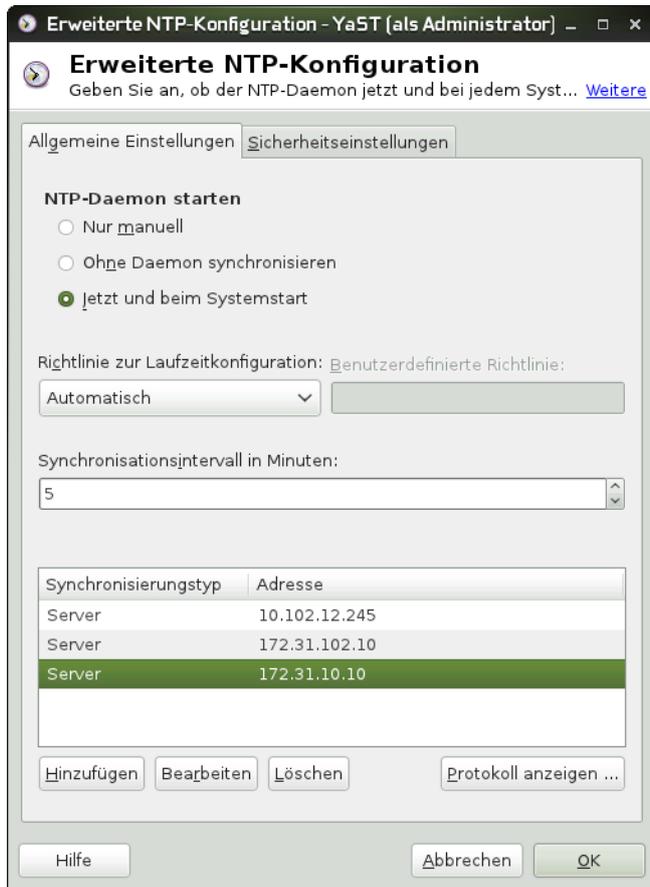
Beispiel: 172.31.140.50/24 eth0 im Netz 172.31.140.1 Genmask 255.255.255.0  
 192.168.0.5/24 eth1 im Netz 192.168.0.1 Genmask 255.255.0.0



## F1.4 Zeitserver NTP

Eine Zeitsynchronisation eines GA-Servers ohne Internet ist via YaST auszuführen  
Beispiele:

ZLT PostgreSQL        via 10.102.12.245  
ZLT-EUB Oracle        via 172.31.102.10    (alternativ)  
ZLT-EUB Oracle        via 172.31.10.10     (alternativ)



Gibt es Internet, so ist vorinstalliert: ptbtime1.ptb.de, ptbtime2.ptb.de

## G Anhang: Tomcat6 Installation

Um die Anwendung jLZHweb in Browser Technologie betreiben zu können, kann die nachfolgend beschriebene Installationsanleitung befolgt werden.

Im Router ist eine *Port Redirection* TOMCAT6, TCP, Port 8080 einzurichten.

Verwendet wird hier der Web-Server tomcat6.

Es wird diese Installation verwendet

- openSUSE-11.4
- GA-Server 2.6ff
- Tomcat6

Bereits diese Basisinstallation erstellt die Basis für den Web-Server automatisch.

Die Installation selbst wird via Skript ab G1.0 bzw. händisch ab G2.0 beschrieben.

### G1.0 tomcat6 Installation via Skript

Es gibt ein Installations-Skript, welches als **Benutzer root** gestartet wird.

Vor der Installation ist die Schrift einzustellen (Umlautproblematik):

YaST → System → Sprache → [Details] UTF-8-Kodierung → Häkchen entfernen

Via SSH, Desktop Terminal oder Doppelklick im Browser

```
cd /home/lzh/bin/jLZHweb/          ins Verzeichnis wechseln
sh Installiere-jlzhweb_war_via.sh Skript starten
```

Als Benutzer lzh kann auch das Skript gestartet werden

```
sh starte-jLZHweb_via.sh          öffnet die jLZHweb im Firefox
```

Es öffnet sich der Browser *Firefox*<sup>29</sup> und die Webseite *Gebäude-Automations-Server*

Ein Klick auf den Link *jLZHweb* verlangt einen Benutzer (BN) und sein Passwort (PW)

```
BN: AMEV                          in Großbuchstaben
PW: AMEV                          in Großbuchstaben
```

Das ist die Anmeldung beim Webserver.

Die Anmeldung bei der Datenbank ist bereits vordefiniert (analog der Desktop Anwendung).

**HINWEIS:**

Falls in den Hilfetexten die Umlaute fehlerhaft dargestellt werden:

```
YaST->System->Sprache->Details
[✓] UTF8 Kodierung verwenden
    Häkchen entfernen
```

Anschließend nochmal

```
sh Installiere-jlzhweb_war_via.sh
```

### Mehr ist nicht zu tun!

Der folgende Rest ist Lesestoff... z.B. für „Nichtexperten“ die sich schlauer machen wollen.

---

<sup>29</sup> *Firefox* sollte vorab konfiguriert werden. Menü Edit → Preferences ...

## G2.0 tomcat6 Installation händisch

Eine händische (Nach)Installation beginnt bei Punkt (G2.1).  
Fehlt Tomcat6, so ist vorher mit Punkt (G6.0) zu beginnen

### G2.1 tomcat6 Installation zu Fuß

tomcat-users.xml: Benutzer für den tomcat6 Manager einrichten  
Hier werden die Rollen gesetzt (jlzhview, fotos, enev)

### G2.2 tomcat-users.xml: vorbereitete Datei kopieren

Diese Datei regelt den Zugriff (sonst erscheint das Standardfenster)

Shell öffnen als root

tomcat-users.xml aus dem Importverzeichnis kopieren

von

nach

**cp /home/lzh/bin/jLZHweb/tomcat-users.xml /usr/share/tomcat6/conf/**

### G2.3 tomcat-users.xml: Datei manuell bearbeiten

Eine Shell öffnen als root und mit Editor gedit bearbeiten

gedit /usr/share/tomcat6/conf/tomcat-users.xml

<tomcat-users> diese Stelle suchen und folgende Zeile dahinter einfügen

In openSUSE Version 11.4 (GA-Server)

```
<user name="eub" password="eub" roles="admin-gui,manager-gui"/>
```

In SuseLinuxEnterpriseServer SLES 11 (ZLT-Server)

```
<user name="eub" password="eub" roles="admin-gui,manager-gui,manager"/>
```

Datei tomcat-users.xml Auszug mit roles und password

```
<tomcat-users>
```

```
<user name="eub" password="eub3x3JK" roles="admin-gui,manager-gui"/>
```

```
<role rolename="jlzhweb"/>
```

```
<role rolename="fotos"/>
```

```
<role rolename="enev"/>
```

```
<user username="gast" password="gast" roles="jlzhweb"/>
```

```
<user username="AMEV" password="AMEV" roles="jlzhweb, fotos, enev"/>
```

Weitere Beispiele:

```
<user username="BA-TK" password="BA-TK" roles="jlzhweb, fotos, enev"/>
```

```
<user username="BA-TS" password="BA-TS" roles="jlzhweb, fotos, enev"/>
```

```
<user username="BA-NKN" password="BA-NKN" roles="jlzhweb, fotos, enev"/>
```

```
<user username="BA-RDF" password="BA-RDF" roles="jlzhweb, fotos, enev"/>
```

HINWEIS: username und password sind beliebig, roles ist fest vorgegeben.

### G2.4 tomcat6 starten

YaST

runlevel 3, 5 sollte aktiv sein

Shell öffnen als root

rctomcat6 status

wenn ok, sonst start, stop, restart

rctomcat6 restart

als Test

/usr/sbin/tomcat

ist das Verzeichnis

### G2.5 Tomcat Manager via Firefox öffnen

**localhost:8080/manager/html**

Benutzername: eub

Passwort: eub3x3JK

öffnet: Tomcat Web Application Manager

/docs

/host-manager

/manager

Falls bereits eine Datei jLZHweb.war existiert: mit UNDEPLOY entfernen!

Falls Manager nicht startet, prüfe ob in Software Installation

--> tomcat6-admin-webapps installiert wurde.

Siehe hierfür auch Anhang Tomcat6 Installation.

## G3.0 jLZHweb Installation der .war Datei

Dateiname: jLZHweb\_yyyy-mm-dd.war Versionsdatum entfernen  
Importverzeichnis: /home/lzh/bin/jLZHweb/jLZHweb.war

Tomcat Web Application Manager unter 'war file to deploy'  
Durchsuchen /home/lzh/bin/jLZHweb/jLZHweb.war  
öffnen und |Deploy| Schaltflächen klicken  
unter Applications erscheint die Anwendung 'jLZHweb'  
unter Message: ok wenn nicht, dann... *is wat falsch*

### G3.1 jLZHweb Startseite einrichten

Die Startseite enthält Schaltflächen mit Links zu

- o jLZHweb zwingend erforderlich
- o Doku bei Bedarf
- o TGA bei Bedarf
- o ENEV bei Bedarf

#### G3.1.1 jLZHweb einrichten

Shell öffnen als root

Das Verzeichnis kopieren aus dem Importverzeichnis

von nach  
**cp -rf /home/lzh/bin/jLZHweb/ROOT/ /usr/share/tomcat6/webapps/**

Falls bereits vorhanden, vorher löschen

```
rm -rf /usr/share/tomcat6/webapps/ROOT
```

HINWEIS: Befehl hier rauskopieren und in Shell einfügen mit Shift+Einfüg

#### G3.1.2 Doku einrichten

Shell öffnen als root

Das Verzeichnis kopieren aus dem Importverzeichnis

von nach  
**cp -rf /home/lzh/bin/jLZHweb/Doku/ /usr/share/tomcat6/webapps/**

Falls bereits vorhanden, vorher löschen

```
rm -rf /usr/share/tomcat6/webapps/Doku
```

#### G3.1.3 TGA einrichten siehe Beispiel

Es kann eine 'TGA-Galerie' angezeigt werden. Hier ein Beispiel:

Shell öffnen als root

Das Verzeichnis kopieren aus dem Importverzeichnis

von nach  
**cp -rf /home/lzh/bin/jLZHweb/TGA/ /usr/share/tomcat6/webapps/**

Benutzer wird verwaltet in /TGA/WEB-INF/web.xml

Falls bereits vorhanden, vorher löschen

```
rm -rf /usr/share/tomcat6/webapps/TGA
```

#### G3.1.4 EnEV einrichten siehe Beispiel

Es kann eine 'EnEV-Galerie' angezeigt werden. Hier ein Beispiel:

Shell öffnen als root

Das Verzeichnis kopieren aus dem Importverzeichnis

von nach  
**cp -rf /home/lzh/bin/jLZHweb/ENEV/ /usr/share/tomcat6/webapps/**

Benutzer wird verwaltet in /ENEV/WEB-INF/web.xml

Falls bereits vorhanden, vorher löschen

```
rm -rf /usr/share/tomcat6/webapps/ENEV
```

#### G3.1.5 Rechte setzen

```
chmod -R 755 /usr/share/tomcat6/webapps/jLZHweb (macht tomcat6)
```

```
chmod -R 755 /usr/share/tomcat6/webapps/ROOT/
```

```
chmod -R 755 /usr/share/tomcat6/webapps/Doku/
```

```
chmod -R 755 /usr/share/tomcat6/webapps/TGA/
```

```
chmod -R 755 /usr/share/tomcat6/webapps/ENEV/
```

#### G3.1.6 Besitzer setzen

```
chown -R lzh:users /usr/share/tomcat6/webapps/ROOT/
```

```
chown -R lzh:users /usr/share/tomcat6/webapps/Doku/
```

```
chown -R lzh:users /usr/share/tomcat6/webapps/TGA/
```

```
chown -R lzh:users /usr/share/tomcat6/webapps/ENEV/
```

### G3.1.7 rctomcat6 restart

### G3.1.8 Tomcat Manager öffnen via Firefox

```
localhost:8080/manager/html
Benutzername      eub                (siehe 1.2)
Passwort          eub3x3JK
öffnet            Tomcat Web Application Manager Verzeichnisse
/
/docs
/host-manager
/jLZHweb
/manager
/Doku
/ENEV              (falls vorhanden)
/TGA               (falls vorhanden)
```

### G3.2 jLZHweb Anwendung starten

Folgende URL im Browserfenster eintippen (Varianten)

```
localhost:8080      lokal      auf die Startseite
aaa.bbb.ccc.ddd:8080  via IP    auf die Startseite
ts-amev.loginto.me  via no-ip auf die Startseite (Beispiel)
                    ts:       der Ort TS
                    amev:      das Objekt
aaa.bbb.ccc.ddd:8080/jLZHweb/web  öffnet direkt die jLZHweb.
```

```
jLZHweb:           wird geöffnet und es erscheint der jLZH-Baum
GA-Server:         GA-Server localhost mit der lokalen Datenbank
ZLT-Server:        ZLT-Server mit der übergeordneten Zentralen LeitTechnik
Alle anderen Einträge können entfernt werden.
Neustart via Web-Seite neu laden
```

HINWEIS: Falls gewünscht, so ist aus dem GA-Server heraus auch die Zentrale LeitTechnik (ZLT) erreichbar.

## G4.0 tomcat6 Bedienung und Einstellungen

```
rctomcat6 status      zur Überprüfung
rctomcat6 restart     wenn ok, sonst: start, stop, restart
```

### G4.1 tomcat6 Einstellung via YaST

```
Tomcat6 soll beim Booten starten
YaST öffnen          System | Systemdienste (Runlevel)
tomcat6              auswählen, aktivieren und OK klicken (Expertenmodus)
YaST beenden
reboot
Danach sollte tomcat laufen... das war's schon
```

## G5.0 Update jLZHweb.war

Falls ein Update der Anwendung jLZHweb installiert werden soll:

```
Tomcat Manager      via Firefox öffnen
jLZHweb.war         entfernen falls vorhanden (Undeploy)
localhost:8080/manager/html
Zeile jLZHweb:      die Schaltfläche Undeploy anklicken
```

Ist UNDEPLOY erfolgreich, nachgucken im Verzeichnis nach

```
/usr/share/tomcat6/webapps/      jLZHweb noch vorhanden? ja:löschen
/usr/share/tomcat6/work/catalina/  jLZHweb noch vorhanden? ja:löschen
jLZHweb.war                       neu installieren (siehe oben)
```

Die Datei LEITZENTRALEN.\* enthält die Einstellungen zu welchen Servern eine Verbindung eingerichtet werden soll und liegt im Verzeichnis /usr/share/tomcat6/work/Catalina/localhost/jLZHweb/eclipse/plugins/eub

.j1zhztools\_1.0.0/Verwaltung/Preferences

Diese Dateien sollten, falls vorhanden, vorher gelöscht werden!

Um sich weitere händische Änderungen zu ersparen, so können, falls bereits die jLZHview installiert ist, diese auch kopiert werden (Muster)

```
von
cp /home/lzh/bin/jLZHview/jLZHview-2.5_20yymmdd/plugins/eub
.j1zhztools_1.0.0/ Verwaltung/Preferences/LEITZENTRALEN*
nach
/usr/share/tomcat6/work/Catalina/localhost/jLZHweb/eclipse/plugins/eub
.j1zhztools_1.0.0/Verwaltung/Preferences/
```

## G6.0 Nachinstallation von Tomcat6

Nur erforderlich falls Tomcat6 nicht im Software Repository.

Punkt (6.0) bis (6.2) nur ausführen, falls tomcat6 noch nicht installiert ist.

Falls das Netz noch undefiniert ist: einfach DHCP verwenden!

Netzwerk anschließen und YaST öffnen.

### G6.1 Software Repository

Alle Häkchen sollten im Produktionsbetrieb deaktiviert sein!

Ein Update ist bei openSuse 11.4 nicht erforderlich.

Falls doch etwas zerschossen wurde:

```
Häkchen setzen (Beispiel)           |Kategorie
|v | openSUSE-11.3-NON-OSS           |YaST
|v | openSUSE-11.3-OSS               |YaST
|v | openSUSE-11.3-update            |YUM
|v | Heruntergeladene Pakete nicht löschen (für jedes Paket!)
alle aktivierten jetzt aktualisieren und mit OK beenden.
```

Falls die Kategorie eine andere ist, so kann diese nicht bearbeitet werden!

- URL kopieren und neues Repository mit dieser URL anlegen (Name erweitern)

- altes Repository kann danach gelöscht werden

Eine Neuinstallation ist für den Ungeübten schneller, einfacher und sicherer!

### G6.2 Tomcat6 installieren oder löschen

tomcat6 in Suchfenster eintippen

Häkchen setzen:

```
|v | tomcat6           (mehrere andere haben automatisch ein Häkchen)
```

Häkchen setzen bei:

```
|v | tomcat6-admin-webapps
```

```
|v | tomcat6-docs-webapps
```

tomcat6 in Suchfenster eintippen und alles dazugehörige wird angezeigt.

Anwenden beendet diesen Schritt.

Es müssen noch einige Lizenzen bestätigt werden...

Die Installation dauert... etwa 10 bis 15 Minuten (Netz) Geduld!

Ausschalten mit `shutdown -h now`

Neustart mit `reboot` (aus der Ferne immer nur mit `reboot!`)

## H Anhang: GA-Plattform-parametrieren

Der GA-Server enthält auch einen oder mehrere GA-Knoten, welcher noch zu aktivieren ist. Ein GA-Knoten kann auch separat auf einem handelsüblichen PC installiert werden. Der GA-Knoten ist das Bindeglied zwischen der MSR (Schaltschrank) und dem Datenbank-Server der Zentralen LeitTechnik (ZLT).

### H7.0 GA-Knoten parametrieren

Das GA-Knoten Objekt (z.B. AMEV) ist das Bindeglied zwischen der MSR und dem Datenbank-Server der ZentraleLeitTechnik (ZLT) und ist i.A. im Schaltschrank vor Ort untergebracht.

Hierfür sind folgende textbasierte Dateien erforderlich

- |                             |  |
|-----------------------------|--|
| 1. /gak/cfg/gak.cfg         | Ablaufdatei erstellt aus Datei AMEV.ref  |
| 2. /gak/cfg/anrufer.chk     | Einwählerlaubnis zum GA-Knoten           |
| 3. /gak/cfg/db_atb.txt      | Definition der Attribute (Standard)      |
| 4. /gak/cfg/db_dim.txt      | Definition der Dimensionen (Standard)    |
| 5. /gak/iz/iz04ddev.bn      | Definition der Adresse zum BACnet-Device |
| 6. /gak/lz/vbp/vbp_AMEV.001 | Definition der Adresse zur Datenbank ZLT |

Die Dateien Punkt (3)-(4) sind weitestgehend vordefiniert und ändern sich nur falls neue Dimensionen bzw. Attribute erforderlich werden, wobei die führende Rolle der ZLT in den Datenbanktabellen `ATTRIBUTE` und `AUSWAHLPOSITIONEN|DIMENSION` zu beachten ist.

Da es hier um textbasierte Dateien handelt, ist es sehr mühsam und fehleranfällig die Einträge positionsgenau in einem Text-Editor zu erstellen.

Daher wurde eine Anwendung unter einer lizenzkostenfreien Tabellenkalkulation entwickelt unter Verwendung der verfügbaren Makroprogrammierung.

#### H7.1 Die Anwendung *GA-Plattform\_nn.ods*

Mit dieser auf der Website **AMEV-GA-Plattform.de** bereitgestellten Anwendung können alle erforderlichen Dateien zur Parametrierung eines GA-Knotens und auch der Datenbank erstellt werden.

Diese Anwendung enthält Makros und wurde entwickelt und getestet unter den Betriebssystemen Linux bzw. eingeschränkt auch Windows 7 mit der lizenzkostenfreien Tabellenkalkulation

- OpenOffice 4.1 ff
- LibreOffice 3.3.2 ff

Beim Öffnen der Datei **GA-Plattform\_nn.ods**

müssen Makros zugelassen werden, sonst geht nichts!

Makros werden aktiviert via [Extras](#) | [Optionen](#) | [Open/LibreOffice](#) | [Sicherheit...](#)

Die wesentliche Arbeit ist die Erstellung der sogenannten Referenz-Datei, d.h. die Zuordnung zwischen den

- Hardware-Adressen der MSR firmenspezifisch
- Datenpunkt-Adressen firmenneutral<sup>30</sup>

und die Zuordnung der Eigenschaften der

- Attribute für Melden und Schalten (`AUS-EIN`, `NORMAL-STÖRUNG` etc.)
- Dimensionen für Messen, Stellen, Zählen (`°C`, `K`, `kWh` etc.)
- Beschreibung als möglichst sortier- und filterbarer Klartext

Diese Arbeit vereinfacht sich mittels Tabellenkalkulation und der programmierten Funktionen.

---

<sup>30</sup> Fehleranfällig ist für einen Anwender meist das Erstellen eines zwingend eindeutigen „sprechender Schlüssel“ als Datenpunkt-Adresse. Dies kann in einem ersten Schritt umgangen werden, indem die Datenpunkt-Adresse einfach durchnummeriert wird und man diese so erzeugte `gak.cfg` im GA-Knoten testweise ablaufen lässt: So können die Hardware-Adressen, die Attribute, die Dimensionen, die Beschreibung (Klartext) etc. getestet werden. Danach kann sich der Mensch an den symbolischen Datenpunkt-Adressen 'vergnügen'...

## H7.2 Eine kurze Einführung als Überblick

### Die Datei *GA-Plattform\_nn.ods*

Einfach mal ausprobieren, es sollte weitestgehend selbsterklärend sein.

Download und Aktualisierungen siehe *AMEV\_GA\_Plattform.de*

Die Anwendung *GA-Plattform\_nn.ods* enthält die beiden Menü ‚**Wo**‘ und ‚**Wie**‘.

Aus der Tabelle in dieser Datei vom Typ Referenz\_AMEV (Beispiel)  
wird eine textbasierte Referenzdatei vom Typ AMEV.ref (AMEV:= Objekt)  
erstellt.

Anschließend wird diese Referenzdatei umgewandelt in eine Datei vom Typ *gak.cfg*.

Wird die Anwendung *GA-Plattform\_nn.ods* auf den Desktop der AMEV-GA-Plattform kopiert, ausgepackt und bearbeitet, so ist die Parametrierung via Menü Steuerung automatisch, sowohl der Datenbank, als auch des GA-Knotens.

Alle erforderlichen Parametrierdateien werden erstellt und sind in einer Verzeichnisstruktur analog dem GA-Knoten abgelegt.

Die Dateien sind händisch in das Verzeichnis der Leitzentrale zu kopieren (Beispiel AMEV).

### Leitzentrale:

Verzeichnisstruktur

```
/lzh/gak_import/amev.VORSCHAU_0/cfg
                                /_gencfg
                                /iz
                                /linux
                                /_logfiles
                                /lz
                                /schemata
                                /sql
```

- **Import in die Datenbank (Datenpunkte einfügen)**

*Terminal* öffnen

```
>l_import-datenpunkte.sh amev.VORSCHAU_0 (eintippen)
erforderlich ist nur die /cfg/gak.cfg
```

- **Import in den GA-Knoten (es wird eine GA-Knoten AMEV erzeugt)**

*Terminal* öffnen

```
>cd /gakimport/amev.VORSCHAU_0
```

und anmelden als Benutzer *root*

```
>erzeuge_gaknoten (eintippen)
```

erforderlich ist /cfg/gak.cfg

anrufer.chk (evtl. anpassen)

dp\_atb.txt

dp\_dim.txt

/lz/vbp/vbp\_AMEV.001

/linux/cfg\_net0

gakconf

hostname (evtl. anpassen)

/iz/... falls BACnet

Der Import von Schemata, deren Fenster und Fenstersymbole sowie der Import der SQL-Dateien zur Erweiterung der Datenpunkte kann dem Beispiel entnommen werden.

INSERT\_2-FENSTER.SQL

INSERT\_3-FENSTERSYMBOLER.SQL

UPDATE\_4-DATENPUNKTE\_BESCHREIBUNG.SQL

UPDATE\_5-DATENPUNKTE\_BEMERKUNG.SQL

## I Anhang: Schemata

Schemata sind Anlagenbilder, welche mit einem Grafik- oder Tabellenprogramm<sup>31</sup> hergestellt werden und dienen als Hintergrundbild für die darauf als Fenster positionierten Datenpunkte. Wird die *Schema-Ansicht*<sup>32</sup> geöffnet, so wird der aktuelle Zustand der Datenpunkte angezeigt. Das Hintergrundbild, seine Fenster und Fenstersymbole sind in der Datenbank abgelegt.

Mit dem *Schema-Editor*<sup>33</sup> der *jLZHview* oder *jLZHweb* können jetzt die Fenster der Datenpunkte und deren Positionen auf dem Hintergrundbild an die gewünschte Position geschoben und dauerhaft in der Datenbank<sup>34</sup> gespeichert werden.

### 18.0 Schemata erstellen

Das Hintergrundbild der Anlage liegt im Verzeichnis `/srv/ftp/schemata/`<sup>35</sup> und wird via *jLZHview* oder *jLZHweb* in die Datenbanktabelle `SCHEMATA` importiert.<sup>36</sup>

Mit dem *Schema-Editor* werden die gewünschten Datenpunkte markiert, via *copy/paste*<sup>37</sup> in das Schema eingefügt und via Maus oder Pfeiltasten an der entsprechenden Stelle pixelgenau platziert<sup>38</sup>.

Die einzelnen Schritte sind

1. die Tabelle `DATENPUNKTE` öffnen, die gewünschten Datenpunkte markieren und in den Zwischenspeicher kopieren,
2. das gewünschte Schema mit dem *Schema-Editor* öffnen, den Inhalt des Zwischenspeichers einfügen,
3. via Maus oder Pfeiltasten an der entsprechenden Stelle pixelgenau platzieren
4. und abschließend den *Schema-Editor* schließen und speichern.
5. Die *Schema-Ansicht* öffnen. Datenpunkte außer Betrieb zeigen `NULL` Werte.

Alle erforderlichen Werte der eingefügten Datenpunkte sind in den Datenbanktabellen `FENSTER` und `FENSTERSYMOLE` hinterlegt.

Hinweis: Es sollten mehrere Schemata abgestuft erstellt werden, z.B. ein einfaches Schema für einen Benutzer *gast*, der nur die Hauptdaten sehen soll, z.B. die Vorlauf- und Rücklauftemperaturen und Pumpenmeldungen, oder ein komplexes Schema für einen Benutzer, der eine Vielzahl von Parameter sichten und bearbeiten kann.

## X Anhang: Melden Sie sich, falls...

Das *Energie- & Umweltbüro e.V.* unterstützt Sie beim Aufbau und Betrieb...

Auf Anfrage bieten wir an

- o Schulungen
- o GA-Server Konfigurationen
- o Umwandeln einer MSR-Datei in eindeutige Referenz-Datei (XXXX.ref) bzw. Konfigurations-Datei (gak.cfg)

Erforderliche MSR-Dateien:

- BACnet CSV-Dateien (EDE, StateText, Units)
- GfR (cex-Datei)
- MODBUS (Tabellenkalkulation)
- etc.

---

<sup>31</sup> Schemata können auch mit einem Programm der *Tabellenkalkulation* erstellt, mit dem Werkzeug *Zeichnen* gestaltet und beschriftet werden und ist einfach z.B. via Bildschirmfoto zu exportieren.

<sup>32</sup> Die *Schema-Ansicht* befindet sich in der *jLZH*-Menüleiste als Schaltfläche [*Pumpensymbol*].

<sup>33</sup> Der *Schema-Editor* befindet sich in der *jLZH*-Menüleiste als Schaltfläche [*Pumpensymbol mit Bleistift*].

<sup>34</sup> Weitere Details siehe *jLZH\_Handbuch.pdf* in der *jLZH*-Menüleiste die Schaltfläche [?].

<sup>35</sup> Alternativ kann das Schema aus dem Verzeichnis `/home/lzh/gak_import/amev/schemata` via Terminal `2_import-schemata.sh` `amev` in das Verzeichnis `/srv/ftp/schemata/` kopiert werden.

<sup>36</sup> In der *jLZHview* oder *jLZHweb* die Tabelle `SCHEMATA` öffnen und via Kontextmenü *Neueintrag* den Dialog öffnen.

<sup>37</sup> `Strg+Einfg/Shift+Einfg/Shift+Entf` (*copy/paste/cut*) funktioniert im Webbrowser!  
`Strg+c/Strg+v/Strg+x` (*copy/paste/cut*) funktioniert NICHT im Webbrowser!!!  
In der *jLZH*-Menüleiste gibt es alternativ die Schaltflächen für einen *Zwischenspeicher*.

<sup>38</sup> In der *jLZH*-Menüleiste zeigt die Schaltfläche *Schema* weitere Positionshilfen.