

1. careSOFT Alarmserver

1.1 Idee

Die Anforderungen an ein modernes Rufsystem haben sich verändert. Viele Funktionen, wie Rufweiterleitungen mit komplexen Bedingungen, Abwesenheitsschaltungen, Anbindungen an Systeme von Drittlieferanten sind gefordert.

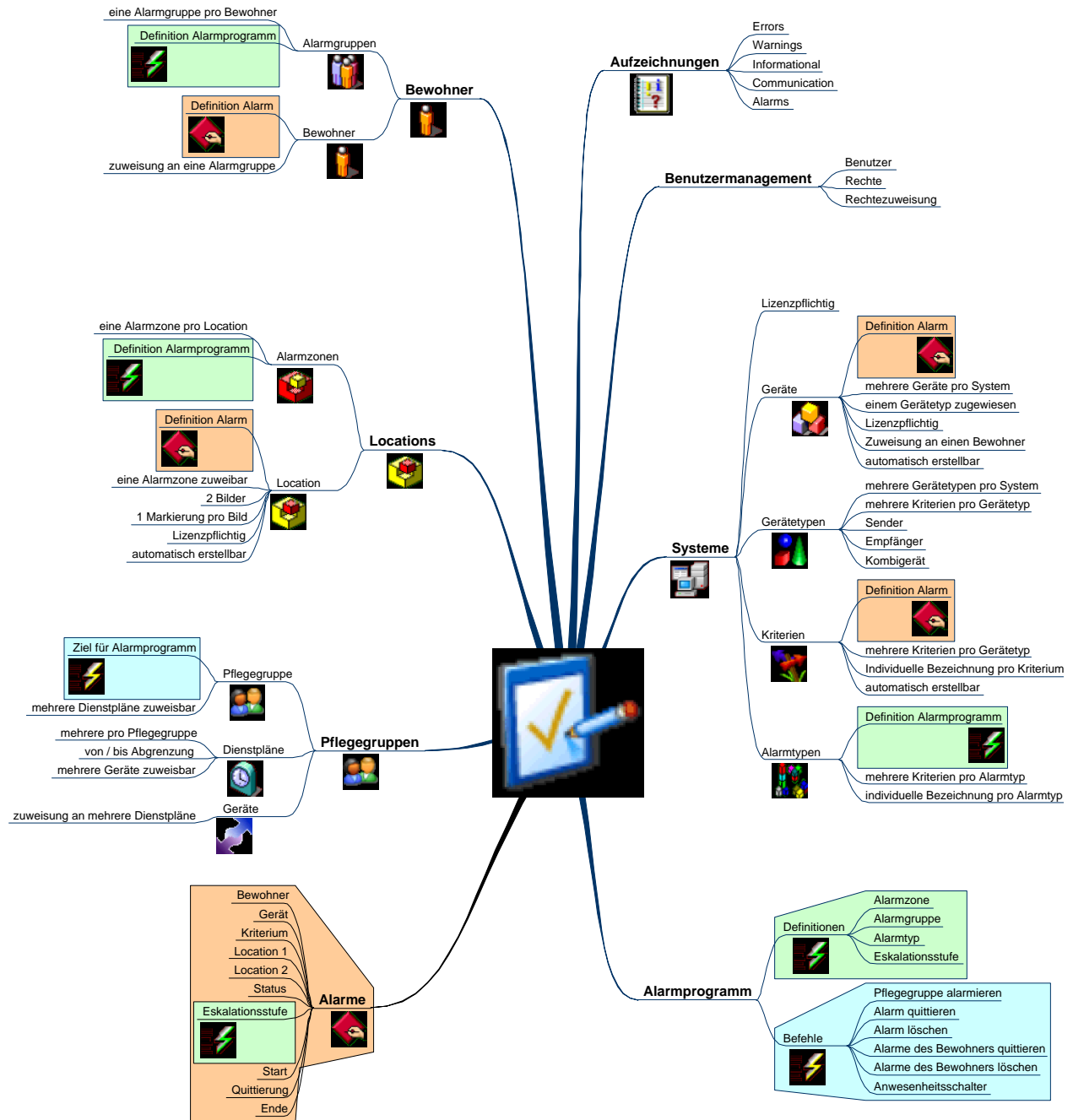
Effektive Pflegeleistungen können bei schrumpfenden Budgets und sinkenden Personalzahlen nur durch eine reibungslose Organisation erzielt werden. Kostenkontrolle im Pflegebereich muss keine Senkung des Pflegeniveaus bedeuten, vielmehr ist dem optimalen Einsatz des Pflegepersonals hohe Beachtung zu schenken.

Es ist ein Wunsch vieler Heimleiter die Auslastung des Pflegepersonals zu kennen um die Dienstpläne optimal gestalten zu können.

Zusammen mit der Optimierung der Dienstpläne soll die Zuweisung der Heimbewohner und der Pflegegruppen variabel gestaltet werden können. Frei definierbare Dienstpläne sind für einen reibungslosen Ablauf im Pflegewesen von grundlegender Bedeutung.

1.2 Funktionale Leistung

1.2.1 Übersicht



1.2.2 Systeme



Die Funktionsgruppe Systeme beinhaltet die Liste der angeschlossenen Systeme, Geräte, Gerätetypen, Kriterien und Alarmtypen. Der Zusammenhang ist folgender:

- Pro System bestehen mehrere Gerätetypen und Geräte
- Jedes Gerät ist an einen Gerätetyp desselben Systems gebunden. Der Gerätetyp definiert, ob es sich um Sendegeräte (können Alarme auslösen), Empfangsgeräte (können Alarme empfangen) oder kombinierte Geräte handelt.
- Jeder Gerätetyp unterstützt mehrere Kriterien.
- Jeder Alarmtyp besteht aus einem oder mehreren Kriterien. Diese können vom selben Gerätetyp, von einem anderen Gerätetyp desselben Systems oder von Gerätetypen aus anderen Systemen bestehen.
- Jedes Gerät wird einem Bewohner zugewiesen.
- Das Gerät und das Kriterium werden in einem Alarm gespeichert

1.2.3 Bewohner



Die Funktionsgruppe Bewohner beinhaltet die Bewohner und Alarmgruppen. Folgende Zusammenhänge sind definiert worden:

- Jeder Bewohner wird einer Alarmgruppe zugewiesen.
- Jedem Bewohner werden ein oder mehrere Geräte zugewiesen.
- Aufgrund der Alarmgruppe werden die Alarmprogramme definiert
- Der Bewohner wird in einem Alarm gespeichert

1.2.4 Positionen



Die Funktionsgruppe Locations besteht aus den Elementen Location und Alarmzone.

Sie arbeiten nach folgendem Prinzip:

- Jede Location wird einer Alarmzone zugewiesen.
- Aufgrund der Alarmzone werden die Alarmprogramme definiert.
- Pro Alarm können 2 Locations, jeweils mit Zeitstempel gespeichert werden. Die Alarmzone wird aufgrund der Location1 ermittelt.
- Location1 und Location2 werden in einem Alarm gespeichert

1.2.5 Pflegegruppen



Für die Alarmierung werden die Pflegegruppen benutzt. Dazu gehören auch die Pflegedienstpläne und die Dienstplan-Geräte-Zuweisung. Der Zusammenhang ist folgendermassen realisiert:

- Eine Pflegegruppe kann als Ziel eines Alarmprogramms definiert werden
- Die Pflegegruppe besteht aus mehreren Dienstplänen.
- Jeder Dienstplan besteht aus einer Startzeit und einer Endzeit.
- Jedem Dienstplan können beliebig viele Geräte zugewiesen werden, die einem Gerätetyp mit Alarmempfängern entsprechen.

1.2.6 Alarmprogramme



Das eigentliche Kernstück der Applikation sind die Alarmprogramme. Sie bestehen aus den Komponenten Alarme, Alarmprogramme, Befehle, Eskalationsstufen und Alarmstatus. Der Zusammenhang dieser Komponenten erfolgt nach folgendem Muster:

- Die eigentlichen Daten werden im Alarm gespeichert. Es sind dies Bewohner, Gerät, Location1, Location2, Alarmstatus, Eskalationsstufe.
- Aufgrund dieser Daten wird ein Alarmprogramm ermittelt. Dieses benötigt jeweils die aktuellen Informationen aus Location1 (→ Alarmzone), Bewohner (→ Alarmgruppe), Kriterium (--> Alarmtyp) und Eskalationsstufe.
- Die Alarmprogramme bestehen aus mehreren Befehlen. Je nach Befehl kann eine Pflegegruppe hinterlegt werden.
- Ist eine Pflegegruppe hinterlegt, wird die aktuelle Information des Alarms an die Geräte der Pflegegruppe weitergereicht.

1.2.7 Aufzeichnungen



Über wichtige Ereignisse führt careSOFT ein Ereignis-Log. Hier können Meldungen nach folgenden Kriterien gegliedert aufgezeichnet werden:

- Informative Meldungen
- Warnungen
- Fehler
- Alarme
- Kommunikation
- Syntax-Fehler

1.2.8 Benutzermanagement



Damit die Zugriffe auf wichtige Programmparameter und sensible Benutzerdaten eingeschränkt werden kann existiert ein Benutzermanagement. Dieses besteht aus drei Funktionseinheiten, den Benutzern, den Rechten und der Benutzer – Rechte - Zuweisung

1.3 Konzept

1.3.1 Client-Server Architektur

Der careSOFT AlarmDriver besteht aus zwei Komponenten. Der Server wird in careSOFT integriert, der Client als OCX-Steuererelement kompiliert.

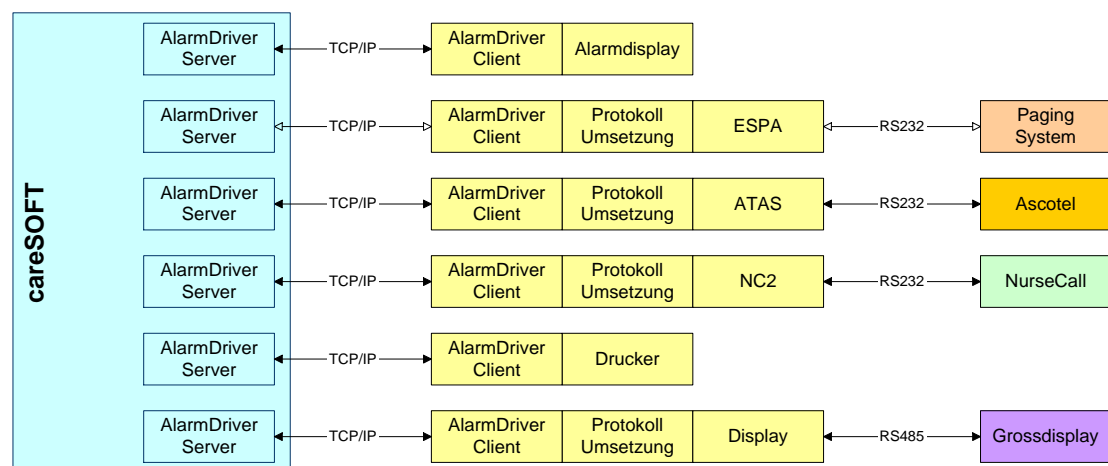
Eine Aufgabe des AlarmDrivers ist es, Informationen von careSOFT zu empfangen und an das Host-Programm in verständlicher Form weiterzureichen. Dieser Informationsfluss besteht aus den Elementen AlarmMessage und InfoMessage.

Ebenfalls soll der Client in der Lage sein, Informationen an careSOFT zu senden. Dieser Informationsfluss besteht aus den Elementen Alarm, AlarmGelesen, AlarmQuittiert, AlarmGelöscht und EventEntry.

Der Client ist für die Verbindungsaufnahme zum Server verantwortlich. Der Server prüft die Anmeldung und akzeptiert oder verwirft eine Verbindung.

Beide Komponenten überwachen die Verbindung laufend, wobei der Server zu diesem Zweck regelmässig Keep-Alive Pakete zum Client sendet.

Hier bildlich dargestellt der Einsatz des AlarmDrivers in einer möglichen Konstellation. Durch die offene Architektur ist eine Erweiterung jederzeit mit einfachen Mitteln zu realisieren.



Der grosse Vorteil dieses Konzepts besteht darin, dass nur noch ein einziger Treiber für die externe Kommunikation notwendig ist. Das vermindert den Entwicklungs- und Verwaltungsaufwand um ein Vielfaches.

Es wird zudem möglich, die Clientkomponente an Lieferanten von Telefon-, Paging- oder anderen Systemen weiterzugeben. Sie können die Protokollumsetzung an ihr System implementieren ohne das Kommunikationsprotokoll bekannt geben zu müssen.

1.3.2 Kommunikation

Für die Kommunikation der beiden Komponenten wird TCP/IP verwendet. Der Server stellt pro definierten Client eine Port-Adresse zur Verfügung.

1.3.3 Schnittstellen

Beide Komponenten stellen Schnittstellen zu ihren Host-Programmen bereit.

- Der Empfang von Meldungen wird mittels Ereignissen signalisiert.
- Das Senden von Mitteilungen kann über Methoden gestartet werden.
- Für die Definition der Komponenten (Port-Adressen, Systemnamen etc.) stehen entsprechende Eigenschaften zur Verfügung.

1.3.4 Service

Die Client-Komponente stellt zusätzlich noch die notwendigen Methoden bereit, um das Host-Programm als Systemdienst deklarieren zu können. Für den Dienstnamen wird die System-ID verwendet.

1.3.5 Protokoll

1.3.5.1 Verbindungsaufbau

Der Verbindungsaufbau wird vom Client initialisiert. Es muss sichergestellt sein, dass sich nur der richtige Client an einem definierten Kommunikationsport an careSOFT anmelden kann.

Hierzu wird die System-ID sowohl auf dem Server, wie auch auf dem Client implementiert. Beim Anmeldevorgang wird diese Information gegenseitig ausgetauscht. Stimmen die Daten überein, wird die Anmeldung gewährt.

1.3.5.2 Verbindungsüberwachung

Damit sichergestellt ist, dass sowohl Client als auch der Server jederzeit einen Ausfall des Anderen erkennt, wird eine Überwachung mittels KeepAlive implementiert. Die Verantwortung für das KeepAlive liegt beim Server.

1.3.5.3 Datenübermittlung

Ist die Kommunikationsschicht etabliert und die Anmeldung korrekt durchgeführt, können die Systeme gegenseitig Daten austauschen. Dabei unterhalten Sie ein einfaches Handshake.

1.3.5.4 Meldungsaufbau

Jede Meldung beinhaltet eine Vielzahl von Informationen. Damit sowohl der Server als auch der Client diese Informationen ermitteln können, folgt das Protokoll einer definierten Struktur.

Die Struktur ist gemäss dem XML-Protokoll implementiert. XML hat den Vorteil, dass es beliebig aufgebaut werden kann. XML entspricht zudem einem anerkannten Standard.

Auf Wunsch kann der XML-Stream mit einer 128-Bit Verschlüsselung gesichert werden. Die Verschlüsselung wird mit einem geheimen Passwort gesichert. Dieser ist direkt in die Applikation kompiliert.

Der XML-Meldungsaufbau ist in diesem Beispiel illustriert:

Kopf	<?xml version="1.0"?>	
Root	<careSOFT>	
Daten		<ALA_ID>1</ALA_ID>
Daten		<MeldungsTyp>2</MeldungsTyp>
.....		<SYS_ID>Alarmdisplay</SYS_ID>
Abschluss	</careSOFT>	

1.3.5.5 Meldetypen

Alle notwendigen Gesprächsthemen sind als Meldetypen definiert. Jeder Typ hat eine bestimmte Aufgabe, Richtung und eine Anzahl möglicher Parameter (Nutzdaten).

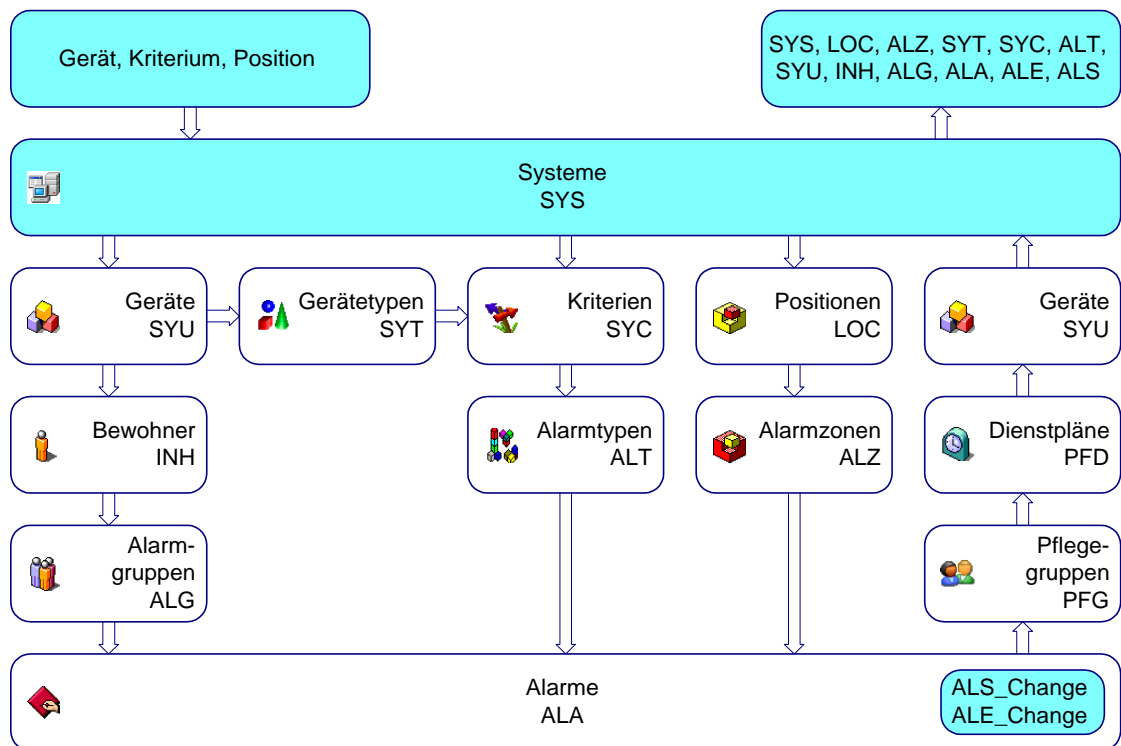
ID	Bezeichnung	Funktion	Richtung
0	Keep Alive	Keep Alive Test durch den Server	S → C
1	Anmeldung	Anmelden des Client Systems	beide
2	AlarmMessage	Informationen über aktuelle Alarmer	S → C
3	Alarm	Auslösen eines Alarmes	C → S
4	AlarmGelesen	AlarmMessage wurde am Empfangsgerät gelesen	C → S
5	AlarmQuittiert	AlarmMessage wurde am Empfangsgerät quittiert	C → S
6	AlarmGelöscht	AlarmMessage wurde am Empfangsgerät gelöscht	C → S
7	Event-Entry	Eintrag ins Event-Log	C → S
8	InfoMessage	Meldungsversand	S → C

1.3.5.6 Nutzdaten

ID	Bezeichnung	Beschrieb	Datentyp	Meldetyp
0	MeldungsTyp	Art der Meldung	Long	0 – 8
1	SYS_ID	Client-System	String 10	0 – 8
2	SYU_Nummer	Nummer des Gerätes	String 10	2 – 8
3	SYU_Name	Bezeichnung des Gerätes	String 50	2 – 8
4	ALA_SYS_ID	System, an welchem der Alarm ausgelöst wurde	String 10	2
5	ALA_SYU_Nummer	Nummer des Gerätes, welches den Alarm ausgelöst hat	String 10	2
6	ALA_SYU_Name	Bezeichnung des Gerätes, welches den Alarm ausgelöst hat	String 50	2
7	SYC_Kriterium	Kriterium	Long	2, 3
8	SYC_Name	Bezeichnung des Kriteriums	String 50	2
9	ALT_Name	Bezeichnung des Alarmtyps	String 50	2
10	INH_Name	Name des Bewohners	String 50	2
11	ALG_Name	Name der Alarmgruppe	String 50	2
12	LOC_Nummer1	Nummer des ersten Location-Codes	Long	2, 3
13	LOC_Name1	Bezeichnung des ersten Location-Codes	String 50	2
14	LOC_TimeStamp1	Zeitstempel des Empfanges des ersten Location-Codes	Date	2, 3
15	ALZ_Name	Bezeichnung der Alarmzone	String 50	2
16	LOC_Nummer2	Nummer des zweiten Location-Codes	Long	2, 3
17	LOC_Name2	Bezeichnung des zweiten Location-Codes	String 50	2
18	LOC_TimeStamp2	Zeitstempel des Empfanges des zweiten Location-Codes	Date	2, 3
19	ALE_Stufe	Eskalationsstufe	Long	2
20	ALE_Name	Bezeichnung der Eskalationsstufe	String 50	2
21	ALS_Status	Alarmstatus	Long	2
22	ALS_Name	Bezeichnung des Alarmstatus	String 50	2
23	ALS_Color	Farbe des Alarmstatus	Long	2
24	ALA_Start	Beginn des Alarmes	Date	2
25	ALA_Quitt	Quittierzeit des Alarmes	Date	2
26	ALA_Ende	Abschlusszeit des Alarmes	Date	2
27	SYS_Reciever	Empfangsinterface	String 10	2, 3
28	SYS_FieldStrenght	Empfangsfeldstärke	String 10	2, 3
29	SYS_Data	Zusatzdaten	String 255	2, 3, 8
30	ALA_ID	ID des Alarmes	Long	2, 4, 5, 6
31	LOG_Typ	Art des Event-Eintrages	Enum	7
32	LOG_Message	Log-Mitteilungstext	String 255	7

1.4 Funktionale Abläufe

Damit die einzelnen Objekte zu einem Gesamtsystem zusammenwachsen können wird das Zusammenspiel der Komponenten in den funktionalen Abläufen geregelt.



1.4.1 Speichern eines Alarmes

- Das System meldet beim Auslösen eines Alarmes System (SYS_ID), Nummer (SYU_Nummer) und Kriterium (SYC_Kriterium). Optional werden Location1 (LOC_Nummer1), Timestamp zu Location 1 (LOC_TimeStamp1), Location2 (LOC_Nummer2) und Timestamp zu Location2 (LOC_TimeStamp2) mitgeliefert.
- Unter Verwendung des Systems und der Nummer wird in tbl_SYU das Gerät nachgeschlagen (SYU_ID mittels SYU_SYS_ID und SYU_Nummer). Kann das Gerät nicht ermittelt werden, wird es automatisch erstellt.
- Zum Gerät wird der zugeteilte Bewohner ermittelt (SYU_INH_ID). Kann kein Bewohner ermittelt werden, wird der Default-Bewohner eingetragen.
- Zum Gerät wird der Typ in tbl_SYU nachgeschlagen (SYU_SYT_ID). Kann kein Gerätetyp ermittelt werden, wird der Default-Typ des Systems eingetragen.
- Gerätetyp und Kriterium werden verwendet um das interne Kriterium in tbl_SYC nachzuschlagen (SYC_ID mittels SYC_SYT_ID und SYC_Kriterium)
- ALA_ALS_ID wird mit dem niedrigsten Wert aus tbl_ALS eingetragen.
- ALA_ALE_ID wird mit dem niedrigsten Wert aus tbl_ALE eingetragen.
- ALA_LOC1_ID wird durch Location1 in tbl_LOC (LOC_ID mittels LOC_Nummer) nachgeschlagen.
- ALA_LOC1_TS wird mit dem Zeitstempel der Location 1 gefüllt
- ALA_LOC2_ID wird durch Location2 in tbl_LOC (LOC_ID mittels LOC_Nummer) nachgeschlagen.
- ALA_LOC2_TS wird mit dem Zeitstempel der Location2 gefüllt
- ALA_Start erhält das aktuelle Datum/Zeit
- ALA_Quitt und ALA_Ende werden mit Null initialisiert.

1.4.2 Abarbeiten eines Alarmes

- Für den Alarm werden nun die Befehle in tbl_ALP ausgeführt. Hierzu werden nachfolgende Felder ermittelt:
- Der Alarmtyp ALP_ALT_ID wird mittels ALA_SYC_ID → SYC_ID → SYC_ALT_ID → ALT_ID → ALP_ALT_ID definiert
- Die Alarmgruppe ALP_ALG_ID wird mittels ALA_INH_ID → INH_ID → INH_ALG_ID → ALG_ID → ALP_ALP_ID definiert
- Die aktuelle Eskalationsstufe ALP_ALE_ID wird mittels ALA_ALE_ID → ALE_ID → ALP_ALE_ID definiert. Befehle von tieferen Eskalationsstufen werden ebenfalls in diese Liste miteinbezogen. (ALP_ALE_ID <= ALA_ALE_ID)
- Die Alarmzone ALP_ALZ_ID wird mittels ALA_LOC_ID → LOC_ID → LOC_ALZ_ID → ALZ_ID → ALP_ALZ_ID definiert.
- Heraus folgen kein, einer oder mehrere Befehle die einem Eintrag in tbl_ALB entsprechen und die notwendigen Parameter beinhalten. Die Befehle werden entsprechend der Nummerierung abgearbeitet.

1.4.3 Änderung der Eskalationsstufe

- Nach der in ALE_Zeit definierten Zeit ändert sich die Eskalationsstufe (ALA_ALE_ID) auf den nächsten definierten Wert. Der Alarm wird von neuem Abgearbeitet
- Befindet sich die Eskalationsstufe auf dem maximalen Wert wird sie nicht erhöht, der Alarm wird dennoch neu Abgearbeitet.

1.4.4 Änderung des Alarmstatus

- Durch Methoden kann der Alarmstatus geändert werden. Der Alarmstatus kann nur vom Wert 1 (Alarm) auf den Wert 2 (Quittiert) sowie vom Wert 2 (Quittiert) auf den Wert 3 (Beendet) gesetzt werden.
- Erreicht der Alarmstatus den Wert 2 (Quittiert) wird das aktuelle Datum/Zeit in das Feld ALA_Quitt geschrieben und der Alarm von neuem Abgearbeitet.
- Erreicht der Alarmstatus den Wert 3 (Beendet) wird das aktuelle Datum/Zeit in das Feld ALA_Ende geschrieben und der Alarm von neuem Abgearbeitet. Danach ist der Alarm erledigt.

1.4.5 Ausführen der programmierten Befehle

Durch das Eintragen eines Alarmes, das Ändern dessen Status oder dessen Eskalationsstufe werden die programmierten Befehle abgearbeitet. Aus den Informationen des Alarms werden folgende Felder ausgelesen:

- ALP_ALG_ID mittels ALA_INH_ID → INH_ID → INH_ALG_ID
- ALP_ALZ_ID mittels ALA_LOC_ID → LOC_ID → LOC_ALZ_ID
- ALP_ALT_ID mittels ALA_SYC_ID → SYC_ID → SYC_ALT_ID
- ALP_ALE_ID mittels ALA_ALE_ID (ALP_ALE_ID <= ALA_ALE_ID)

Aufgrund dieser Angaben kann in der Tabelle tbl_ALP die Liste der Befehle nachgeschlagen werden. Diese wird entsprechend der Programmnummer sortiert.

1.4.5.1 Alarm quittieren

Mittels des Methodenaufrufs in ALA wird der Alarm quittiert. War er schon quittiert, wird der Aufruf ignoriert. Beim Quittieren wird auch ein Eintrag in ALD (Alarm-Details) gemacht.

1.4.5.2 Alarm löschen

Mittels Methodenaufruf in ALA wird der Alarm gelöscht. War er nicht schon Quittiert, wird der Aufruf ignoriert. Beim Löschen wird auch ein Eintrag in die Detail-Tabelle (ALD) gemacht.

1.4.5.3 Alarme des Bewohners quittieren

Mittels Methodenaufruf in ALA wird die Alarmliste des Bewohners quittiert. Es werden nur offene Alarme berücksichtigt.

1.4.5.4 Alarme des Bewohners löschen

Mittels Methodenaufruf in ALA wird die Alarmliste des Bewohners gelöscht. Es werden nur quittierte Alarme berücksichtigt.

1.4.5.5 Pflegegruppe alarmieren

Das Feld ALP_PFG_ID enthält einen Verweis auf eine Pflegegruppe. Unter dieser Pflegegruppe und der aktuellen Zeit wird der zuständige Pflegedienstplan (PFD_ID) ermittelt. Diesem Dienstplan sind in tbl_PFU ein oder mehrere Geräte zugewiesen. Diesen Geräten wird nun eine Aktualisierung des Alarmes gesendet. Dieser Aufruf wird in der Tabelle ALD protokolliert.

1.4.5.6 Anwesenheitsschalter

Eine spezielle Funktion ist der Anwesenheitsschalter. Er ermittelt die Anzahl offener und quittierter Alarme des auslösenden Bewohners. Ist genau 1 Alarm offen (dieser Alarm) und noch keiner quittiert wird der offene Alarm quittiert. Sind mindestens zwei offene Alarme vorhanden, werden diese quittiert und der aktuelle Alarm noch gelöscht. Ist mindestens ein Alarm quittiert, wird dieser gelöscht und der aktuelle Alarm sowohl quittiert als auch gelöscht.