

# BioRem als FESA3-Klasse

## BioRem-Ansteuerung mit einer FESA3-Klasse

Sinn&Zweck  
Funktionalität  
Implementierung

# Was ist ein BioRem?

- Ein BioRem ist ein Detektor für Neutronen- und Gamma-Strahlung radioaktiver Aktivitäten
- An bestimmten Punkten rund um den Beschleuniger sind BioRems zum Messen aufgestellt
- Die Messwerte werden in regelmässigen Abständen aufgenommen und intern im Gerät abgespeichert

# Status Quo

- Momentan werden die abgespeicherten Messwerte von Neutronen- und Gamma-Strahlung bei Bedarf bei jedem BioRem einzeln ausgelesen
- Dazu existiert in der Abteilung Sicherheit und Strahlenschutz ein alter Laptop, auf dem eine spezielle DOS-Software läuft
- → Mühsamer und (zeit-)aufwändiger Prozess!

# Idee

- Vereinfachung und Beschleunigung des Auslesens durch Anschluss der BioRems an das Netzwerk
- Auslesen der Messwert-History mit Hilfe von neuer Software
- Übertragung der Daten über das Netzwerk ermöglicht zeitnahes Auslesen bei Bedarf



# Erste Umsetzung

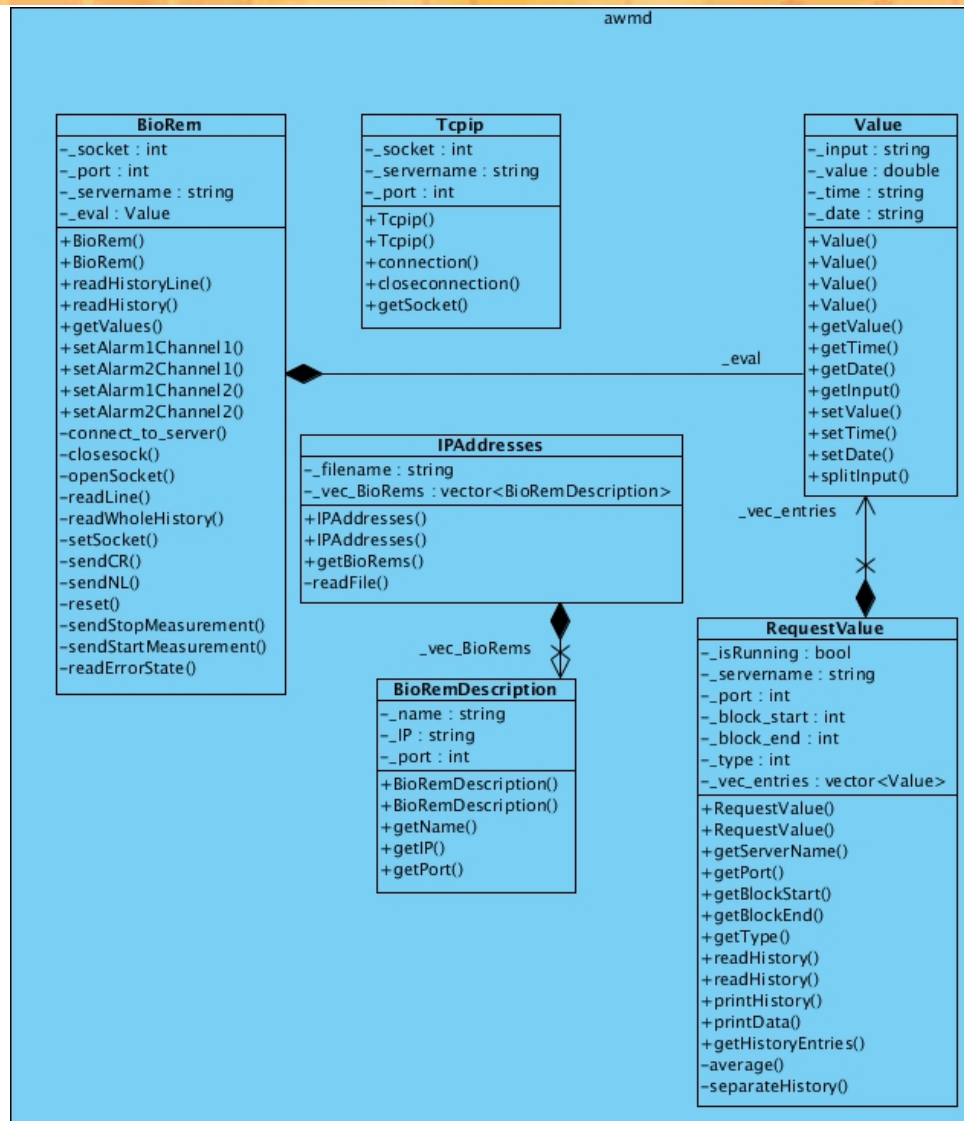
- Entwicklung einer Hardware-Platine zum Umsetzen der seriellen RS232-Daten in das TCP/IP-Protokoll im Rahmen eines Abschlussprojektes von Auszubildenden
- Realisierung einer simplen Java-Test-Software zum Ansteuern der ans Netzwerk angeschlossenen BioRems mit einer Test-Anbindung an eine lokale Datenbank
- Software-basiertes Auslesen der Messwerte der Neutronen- und Gamma-Detektoren



# Realisierung der Software (neu)

- Erstellung einer objektorientierten Bibliothek in C++, die die gesamte Funktionalität der TCP/IP-Kommunikation sowie die verwendbaren Befehle zur Steuerung eines BioRems enthält
- Anbindung an das (zukünftige) Kontrollsystem durch Verwendung von FESA3 für die Gerätesoftware; diese verwendet o.g. C++-Bibliothek
- Liste von BioRem's samt IP wird direkt ausgelesen
- Implementierung von Test-Clients in C++ und Java

# UML Klassen-Diagramm BioRem



# Zusammenfassung

- Momentan existiert lediglich ein Prototyp der neuen Platine
- Gerätesoftware zur Ansteuerung mehrerer BioRem's ist verfügbar in FESA3
- Auslesen mit verschiedenen Test-Clients in C++ und Java funktioniert erfolgreich