

Contextinformation Routing Networks (CRNs) “Workshop zur Erstellung von Prototypen”

Stefan und Nikita Hans

15.-29. Februar 2016

Inhaltsverzeichnis

1	Einführung	2
2	Testumgebung für Komponenten	2
2.1	CRN-Gateway	2
2.2	CRN-Application	3
2.3	CRN-Shell	3
3	Grundlagen Mobile Devices	3
3.1	Qt Creator und Portierbarkeit	3
3.2	GUI-Design	3
3.3	Kommunikation im Netzwerk (TCP/IP)	3
3.4	Einbinden der Komponenten	3
4	Lokales Netzwerk	4
5	CIC-Rulesets	4
5.1	CIC-Dictionary	4
5.2	GPS-Daten	4
6	Internet	5
6.1	Location-Based Social Networks	5
7	Glossar	6
8	Abkürzungen	6

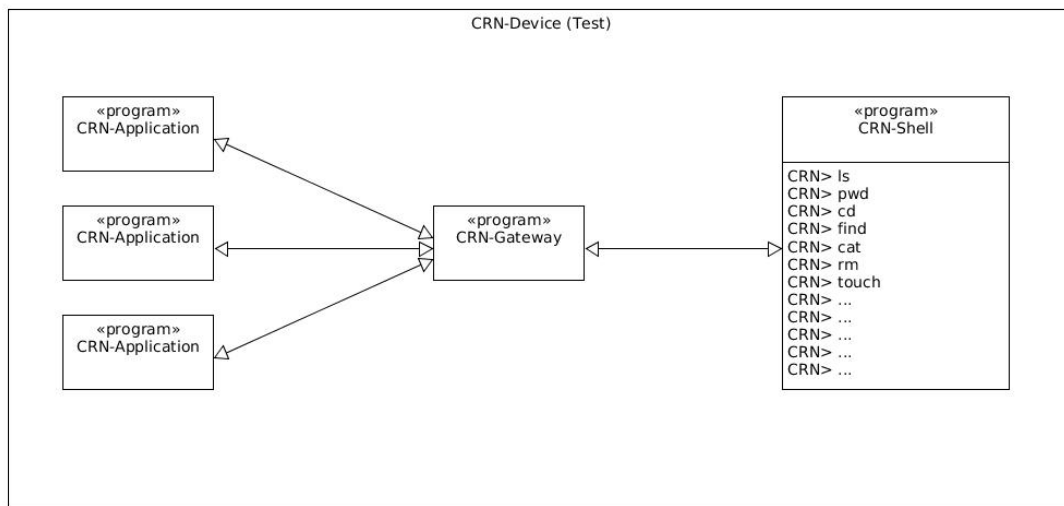
1 Einführung

Nach Fertigstellung der CIP-Spezifikation für CRNs (CIP_Specification.txt) sollen im Rahmen eines zweiwöchigen Workshops die Kernkomponenten und entsprechende Prototypen für Tests implementiert werden.

Programmiersprachen sind C/C++ mit Qt. Als Entwicklungsumgebung wird der Qt Creator verwendet. Als Laufzeitumgebungen sind Linux (Ubuntu), Windows (?) sowie Android und iOS vorgesehen.

Als Versionsverwaltungssystem wird GitHub verwendet (PrototypeWorkshop). Die Dokumentation wird via LaTeX (TeXstudio) im PDF erstellt. Für Modellierungen wird UML (UMLet) verwendet.

2 Testumgebung für Komponenten



Beispielkonstellation als Standalone auf einem Device

Alle Komponenten werden zuerst auf Linux als Laufzeitumgebung implementiert und bei Bedarf portiert.

Ziel: Das Design priorisiert Wiederverwendbarkeit, Portierbarkeit und Erweiterbarkeit. Die Implementierungen sollen einfach und schlank sein. Es sind keine expliziten Test vorgesehen.

2.1 CRN-Gateway

Zentrale Komponente zur Vermittlung und Speicherung von CIP.

2.2 CRN-Application

GUI mit Anfragen an CRN-Gateway (Offer-CIP und Request-CIP).

2.3 CRN-Shell

Shell mit Kommandozeile zur Analyse der CIPs im CRN-Gateway.

3 Grundlagen Mobile Devices

Die benötigten Kenntnisse zur Einbindung mobiler Devices in CRNs werden erworben.

Ziel: Einen Überblick über die verwendeten Technologien zu haben und konkrete Vorgehensweisen zu beherrschen.

3.1 Qt Creator und Portierbarkeit

Erstellung einfacher Anwendungen (Listings) auf allen Laufzeitumgebungen (Windows?). Die weiteren Punkte sollen, soweit praktikabel, auch auf allen Laufzeitumgebungen erstellt werden.

3.2 GUI-Design

Verstehen vorhandener Methodik, Erstellung einfacher Anwendungen (Listings) und Bewerten möglicher Designentscheidungen.

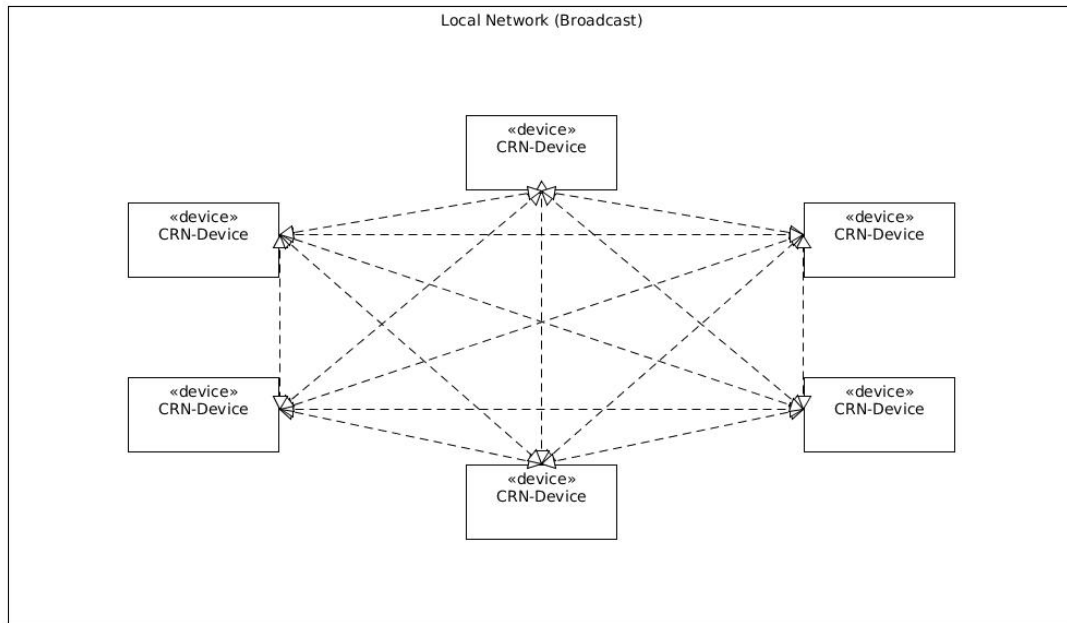
3.3 Kommunikation im Netzwerk (TCP/IP)

Einfache bidirektionale Kommunikation über Sockets mit TCP und IP Paketen. Broadcast in lokalen Netzen.

3.4 Einbinden der Komponenten

Einbinden einfacher C/C++ Komponenten in Anwendungen. Nachladen zur Laufzeit.

4 Lokales Netzwerk



Beispielkonstellation verschiedener CRN-Devices in einem lokalen Netzwerk

Erstellen eines Ad-hoc-Netzes. Anwendungen als Teil eines reaktiven Systems (Das Reaktive Manifest).

Ziel: Aufbau eines ersten stabilen und funktionsfähigen CRNs.

5 CIC-Rulesets

Erarbeiten der grundlegenden Konzeption von CIC-Rulesets und Erstellung erster Designs und Implementierungen.

Ziel: Erstellung und Einbindung anwendbarer CIC-Rulesets.

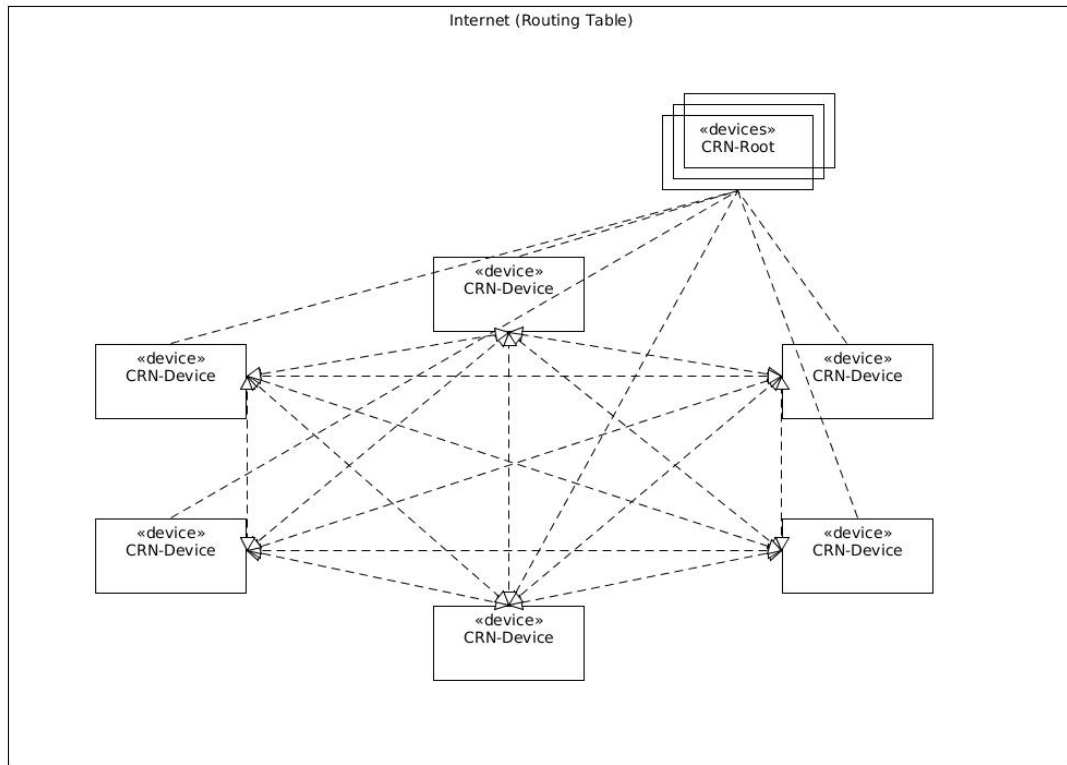
5.1 CIC-Dictionary

Einfache statische Wortliste.

5.2 GPS-Daten

Einfache zweidimensionale GPS-Daten ohne Kartenansicht. Verwendung eigener GPS-Position.

6 Internet



Beispielkonstellation verschiedener CRN-Devices mit CRN-Root als zentrale Vermittlung (Fallback) im Internet

Erstellen eines (reaktiven) Systems (CRN-Root) zur initialen Vermittlung von CRNs im Internet. ()Erstellen dynamischer Routing Tabellen auf CRN-Devices (CRN-Nodes). CRN-Root als Fallback, wenn gepufferte Routings nicht ausreichen.)

Ziel: Erstellung und Verwendung eines einfachen zentralen Routings im Internet.

6.1 Location-Based Social Networks

Abschließende Zusammenstellung aller erarbeiteter Funktionalitäten in einer mobilen Anwendung zur Verwendung im Internet.

Ziel: Erstellung einer ersten Anwendung, die für Ende-zu-Ende Tests bei der Weiterentwicklung von CIR verwendet werden kann.

7 Glossar

8 Abkürzungen

Tabelle Englische Abkürzungen