

Embedded Betriebssystem

für ARM Cortex-A8

Nicolaj Höss, Marko Petrović, Kevin Wallis



Fachhochschule Vorarlberg
Master Informatik

S1: Softwarelösungen für ressourcenbeschränkte Systeme

14. Mai 2015

1. Was ist YAML
2. Aufbau und Syntax von YAML
3. Design des Kommunikationsprotokolls
4. Bewertung

Inhalt

Was ist YAML

Aufbau und Syntax
von YAML

Design des
Kommunikations-
protokolls

Bewertung

Referenzen

Inhalt

Was ist YAML

Aufbau und Syntax
von YAML

Design des
Kommunikations-
protokolls

Bewertung

Referenzen

1. Was ist YAML

2. Aufbau und Syntax von YAML

3. Design des Kommunikationsprotokolls

4. Bewertung

- ▶ „YAML Ain't Markup Language“
- ▶ vereinfachte Auszeichnungssprache
mensch- und maschinenlesbar
Gliederung und Formatierung von Text und Daten
- ▶ zur Datenserialisierung
- ▶ ursprünglich an XML angelehnt
- ▶ Datenstrukturen von Perl, Python und C
- ▶ entwickelt von Clark Evans (2001)

Die Designziele von YAML sind: [6]

- ▶ leicht von Menschen lesbar
- ▶ Portabilität zwischen Programmiersprachen
- ▶ Übereinstimmung mit natürlichen Datenstrukturen
- ▶ konsistentes Schema für generische Tools
- ▶ aussagekräftig und erweiterbar
- ▶ leicht zu Implementieren und zu Verwenden

C	C++
JavaScript	Java
Perl	PHP
Python	ActionScript
Ruby	.NET Plattform

Und viele weitere mehr.

Inhalt

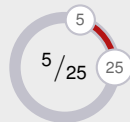
Was ist YAML

Aufbau und Syntax
von YAML

Design des
Kommunikations-
protokolls

Bewertung

Referenzen



1. Was ist YAML

2. Aufbau und Syntax von YAML

2.1 Besonderheiten

2.2 Semantik

2.3 Andere Features

2.4 Anwendungsbeispiel (Ruby)

2.5 Vergleich mit JSON

3. Design des Kommunikationsprotokolls

4. Bewertung

Inhalt

Was ist YAML

**Aufbau und Syntax
von YAML**

Besonderheiten

Semantik

Andere Features

Anwendungsbeispiel
(Ruby)

Vergleich mit JSON

Design des
Kommunikations-
protokolls

Bewertung

Referenzen

Inhalt

Was ist YAML

**Aufbau und Syntax
von YAML**

Besonderheiten

Semantik

Andere Features

Anwendungsbeispiel
(Ruby)

Vergleich mit JSON

Design des
Kommunikations-
protokolls

Bewertung

Referenzen

Jede Datenstruktur darstellbar durch:

- Scalars (Einzelwerte)
 - Sequences (Arrays)
 - Mappings (Assoziativen Listen)
-
- ▶ Leerzeichen zum Kennzeichnen von Strukturen
 - ▶ Tabs sind nicht erlaubt

Unterstützte Typen:

- Strings
- Numerische Werte
- Boolesche Ausdrücke
- Spezielles Format für Datum

```
1 #Strings
2 'String'
3 "String"
4 #Numbers
5 12      #int
6 014     #octal
7 0xC     #hex
8 13.4    #float
9 1.2e+34 #exponential
10 .inf    #infinity
11 #Nulls
12 null
13 ~
14 #Boolean
15 true
16 false
17 #Dates
18 2002-12-14
```

Quellcode 1: Einzelwerte

Quelle: [2]

Erzeugen von Array:

- ▶ Dash+Space
- ▶ oder im Flowstyle

```
1 a_sequence:  
2   - PHP  
3   - Perl  
4   - Python  
5 #or Flowstyle:  
6 a_sequence:[PHP, Perl, Python]
```

Quellcode 2: Arrays

Quelle: [2]

Assoziative Listen (Mappings)

```
1 a_map:  
2   PHP: 5.2  
3   MySQL: 5.1  
4   Apache: 2.2.20  
5 #or Flowstyle:  
6 a_map: {PHP: 5.2, MySQL: 5.1,  
7   Apache: 2.2.20}
```

Quellcode 3: Assoziative

Quelle: [2]

- ▶ bestehend aus Key und Value
- ▶ Key: Value
- ▶ auch im Flowstyle

```
1 #Mix Mappings und Sequences:  
2 'Chapter1': [Introduction, Event Types]  
3 "Symfony1": {PHP: 5.0, Propel: 1.2}
```

Quellcode 4: Mischung

Quelle: [2]

```
1 #Anchors:
2 anchored_content: &anchor_name
3 other_anchor: *anchor_name
4
5 #Declare Types:
6 !!str 0.5
7
8 #Set type:
9 set:
10   ? item1
11   ? item2
12   ? item3
13 #entspricht
14 set:
15   item1: null
16   item2: null
17   item3: null
```

Quellcode 5: Features

Quelle: [3]

- ▶ Anchor: ein Value für zwei Keys (Inhalt duplizieren)
- ▶ mit !! wird ein Typ explizit angegeben (tags)
- ▶ !!binary: base64-encoded representation of a binary blob
- ▶ Set entspricht Map mit Null Values

```
1 #require 'yaml'
2 # Set.new([1,2]).to_yaml
3 --- !ruby/object:Set
4 hash:
5   1: true
6   2: true
```

Quellcode 6: Features
Quelle: [4]

- ▶ Kommentar: Zeile in Ruby
- ▶ Der Rest zeigt das Ergebnis in YAML
- ▶ —: Tag der von Ruby erzeugt wurde
- ▶ YAML erlaubt das Hinzufügen von Metadaten

Inhalt

Was ist YAML

Aufbau und Syntax von YAML

Besonderheiten

Semantik

Andere Features

Anwendungsbeispiel (Ruby)

Vergleich mit JSON

Design des Kommunikationsprotokolls

Bewertung

Referenzen

Inhalt

Was ist YAML

Aufbau und Syntax von YAML

Besonderheiten

Semantik

Andere Features

Anwendungsbeispiel (Ruby)

Vergleich mit JSON

Design des Kommunikationsprotokolls

Bewertung

Referenzen

- JSON ist Untermenge von YAML
- JSON Dokument ist gleichzeitig YAML 1.2 Dokument (V1.1 nicht!)
- JSON Designziele: Einfachheit, Universalität
- YAML Designziele: Lesbarkeit durch Mensch, Unterstützt serielle Übertragung von Datenstrukturen
- Was YAML erlaubt und JSON nicht: [7]
 - Kommentare
 - Tags für Metadaten
 - Anchors
 - Dokument Start- und Endmarkierungen

1. Was ist YAML

2. Aufbau und Syntax von YAML

3. Design des Kommunikationsprotokolls

3.1 Message Typen

3.2 Designkriterien des Protokolls

3.3 Aufbau Protokoll

3.4 Messages

3.5 Beschreibung und Abhängigkeiten

4. Bewertung

Inhalt

Was ist YAML

Aufbau und Syntax
von YAML

**Design des
Kommunikations-
protokolls**

Message Typen
Designkriterien des
Protokolls
Aufbau Protokoll
Messages
Beschreibung und
Abhängigkeiten

Bewertung

Referenzen

- ▶ Start
- ▶ Stop
- ▶ Set Default Values
- ▶ Get Default Values
- ▶ Report -Default Values
- ▶ Report -Status
- ▶ Request -Status
- ▶ Report -Error

Inhalt

Was ist YAML

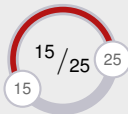
Aufbau und Syntax
von YAML

**Design des
Kommunikations-
protokolls**

Message Typen
Designkriterien des
Protokolls
Aufbau Protokoll
Messages
Beschreibung und
Abhängigkeiten

Bewertung

Referenzen



Inhalt

Was ist YAML

Aufbau und Syntax
von YAML

**Design des
Kommunikations-
protokolls**

Message Typen

**Designkriterien des
Protokolls**

Aufbau Protokoll

Messages

Beschreibung und
Abhängigkeiten

Bewertung

Referenzen

Designziele (nach Priorität):

- Lesbarkeit für den Menschen
- Defaultparameter sind vorhanden
- Reihenfolge der Parameter beliebig
- Wenig Overhead (Geschwindigkeit)

- Inhalt
- Was ist YAML
- Aufbau und Syntax von YAML
- Design des Kommunikationsprotokolls**
 - Message Typen
 - Designkriterien des Protokolls
- Aufbau Protokoll**
 - Messages
 - Beschreibung und Abhängigkeiten
- Bewertung
- Referenzen

Tabelle 1: Protokollaufbau

"MessageType"	:	{	Parametername	:	Wert	,	usw	}
---------------	---	---	---------------	---	------	---	-----	---

YAML Ergebnis:

MessageType:

Parametername1: Wert1

Parametername2: Wert2

Parametername3: Wert3

Inhalt

Was ist YAML

Aufbau und Syntax
von YAML

**Design des
Kommunikations-
protokolls**

Message Typen
Designkriterien des
Protokolls
Aufbau Protokoll
Messages

**Beschreibung und
Abhängigkeiten**

Bewertung

Referenzen

Parametername hätte auch weggelassen werden können.
(Verwendung von Array statt Assoziativen Listen)

Wir haben uns dagegen entschieden weil:

- Lesbarkeit verschlechtert wird
- Reihenfolge der Parameter nicht mehr beliebig wäre

Nachteile dieser Entscheidung:

- Overhead (Geschwindigkeit)

1. Was ist YAML
2. Aufbau und Syntax von YAML
3. Design des Kommunikationsprotokolls
4. Bewertung
 - 4.1 Vorteile
 - 4.2 Nachteile

Inhalt

Was ist YAML

Aufbau und Syntax
von YAML

Design des
Kommunikations-
protokolls

Bewertung

Vorteile

Nachteile

Referenzen

Inhalt

Was ist YAML

Aufbau und Syntax
von YAML

Design des
Kommunikations-
protokolls

Bewertung
Vorteile

Nachteile

Referenzen

YAML:

- ++ Leicht vom Mensch Schreiben, Lesen (vgl. XML)
- ++ Weiterverarbeitung der Daten in anderen Sprachen
 - + Hinzufügen von Tags erlaubt
 - + Sprache ist leicht zu Verwenden

Entwickeltes Protokoll:

- ++ Leicht vom Mensch Schreiben, Lesen (vgl. XML)
 - + Defaultparameter sind vorhanden (nicht alle Senden)
 - + Reihenfolge der Parameter ist beliebig



Inhalt

Was ist YAML

Aufbau und Syntax
von YAML

Design des
Kommunikations-
protokolls

Bewertung

Vorteile

Nachteile

Referenzen

YAML:

- – Geschwindigkeit (vgl. JSON)
- – Verbreitung und Bekanntheit
 - Wird immer noch weiterentwickelt (Ausgereiftheit)

Entwickeltes Protokoll:

- Vermeidbarer Overhead (Siehe Abhängigkeiten)
- Sender und Empfänger brauchen Befehlsliste

- [1] Wikipedia Community
YAML.
<http://en.wikipedia.org/wiki/YAML>, 2015.
- [2] Symfony
The YAML Format.
http://symfony.com/doc/current/components/yaml/yaml_format.html, 2015.
- [3] Adam Brenecki
Learn X in Y minutes. Where X=yaml.
<http://learnxinyminutes.com/docs/yaml/>, 2015.
- [4] Xavier Shay
Robot Has No Heart. YAML Tutorial.
<http://rnhh.net/2011/01/31/yaml-tutorial>, 2015.

Inhalt

Was ist YAML

Aufbau und Syntax
von YAML

Design des
Kommunikations-
protokolls

Bewertung

Referenzen

- [5] yaml-cpp
HowToEmitYAML.
<https://code.google.com/p/yaml-cpp/wiki/HowToEmitYAML>,
2012.
- [6] Clark Evans et al.
YAML Ain't Markup Language Version 1.2.
<http://yaml.org/spec/1.2/spec.html>, 2009.
- [7] CowTalk
What me like YAML?.
http://www.cowtowncoder.com/blog/archives/2012/04/entry_473.html, 2009.

- [8] Malin Eriksson und Viktor Hallberg
*Comparison between JSON and YAML for data
serialization.*
[http://www.csc.kth.se/utbildning/kth/kurser/
DD143X/dkand11/Group2Mads/victor.hallberg.malin.
eriksson.report.pdf](http://www.csc.kth.se/utbildning/kth/kurser/DD143X/dkand11/Group2Mads/victor.hallberg.malin.eriksson.report.pdf), 2011.



The End

Danke für Ihre Aufmerksamkeit.