

Marcel Weiher (@mpweiher)

Objective-Smalltalk

Objective-Smalltalk

- Variante von Smalltalk
- Objective-C kompatibel
- Skript- und Programmiersprache
- Architekturorientierte Konzepte
- Frameworks, Sprache, Tools

Objective-Smalltalk

- Motivation
- Die Sprache

Objective-Smalltalk

Motivation

Objective-Smalltalk: Motivation

- Scripting
- Sprachvereinfachung
- Programmvereinfachung

Objective-Smalltalk: Motivation

Scripting

Objective-Smalltalk: Scripting

- Scripted Components / Software-ICs
- Interaktive und Iterative Entwicklung
- Kleine Systeme werden groß
- WebScript

Objective-Smalltalk: Scripting

- Applikationen?
- Shell Scripts?
- Web Server?
- Naked Objects

Objective-Smalltalk: Framework-Scripting

- App = Frameworks + GUI
- Scripted App = Frameworks + Script + GUI
- Web Server = Framework + Script
- Neues Framework = Framework + Script?

Objective-Smalltalk: Motivation

Sprachvereinfachung

Objective-Smalltalk: Sprachvereinfachung

- C ist groß
- Smalltalk is klein

Smalltalk

C

Objective-C: Überlappungen

- Syntax: []
- C Array / Objective-C Array
- C String / NSString
- int,float / NSNumber
- Literale / @Literale
- struct / Klasse
- Funktionen / Methoden

C → Smalltalk

- Kontrollstrukturen: Messages
- Array Syntax: Messages
- Operatoren: Messages
- Nur eine Art von Block

Objective-Smalltalk: Motivation

Programmvereinfachung

Programmvereinfachung

- Wunderlist iOS client: 160 KLOC
- Microsoft Office: 400 MLOC
- Fred Brooks: No Silver Bullet

Programmvereinfachung

- Programmiersprachen sind prozedural
- Unsere Aufgaben sind es nicht
- Lücke zwischen Programmtext und System

Programmvereinfachung

- Programmiersprachen sind prozedural
- Vor allem die Abstraktionsmechanismen
- Unsere Aufgaben sind es nicht

Zitate

- Feynman: "Computers don't compute"
- Guy Steele: "No longer about termination"
- A. Black: "Communication, not computation"

Programmtext vs. System

- FORTRAN

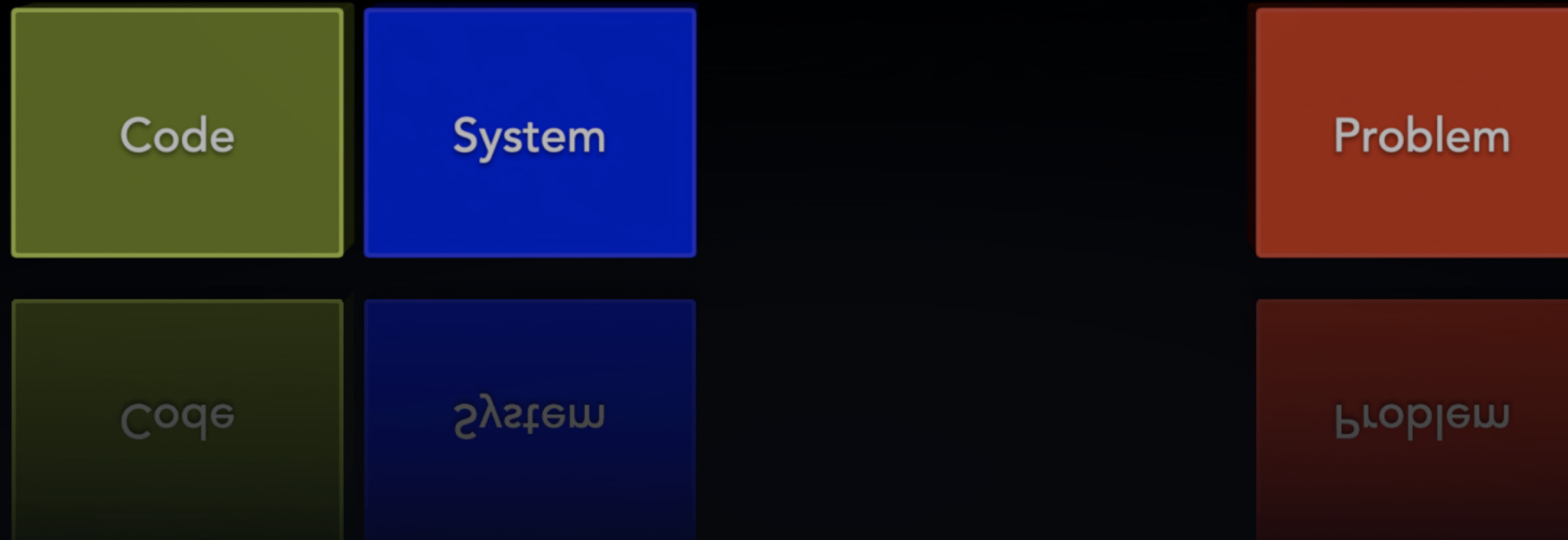
Code

System

Problem

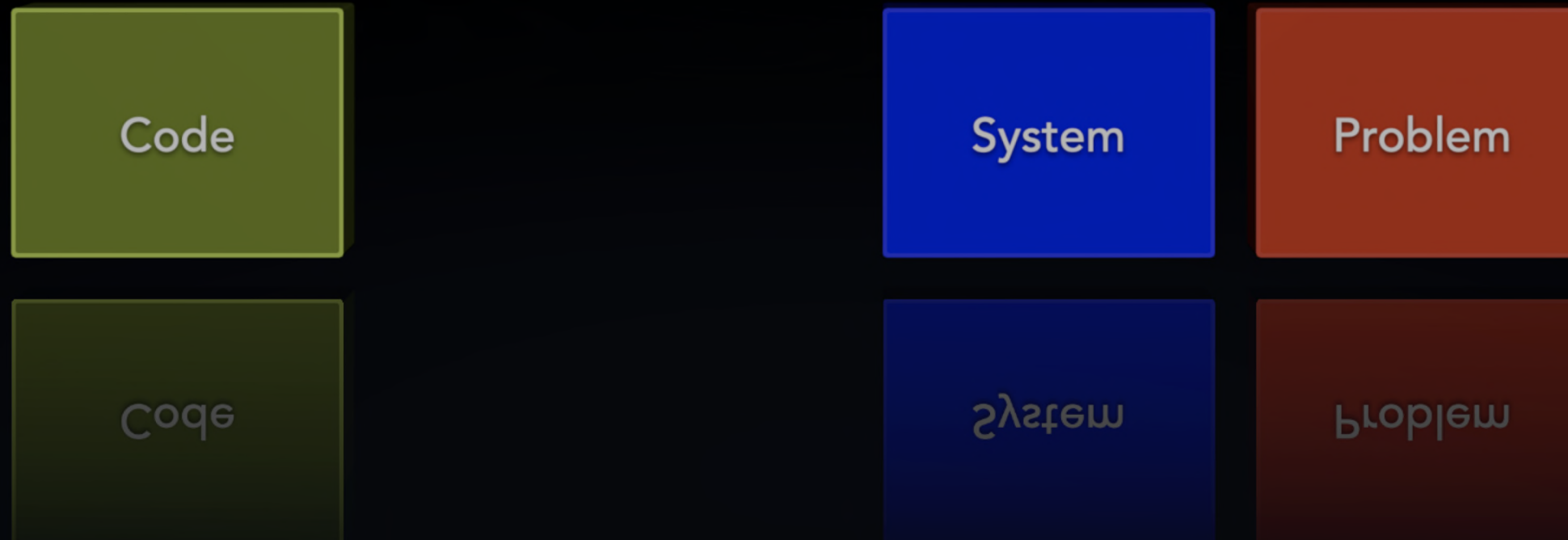
Programmtext vs. System

- FORTRAN
- Non-Computation



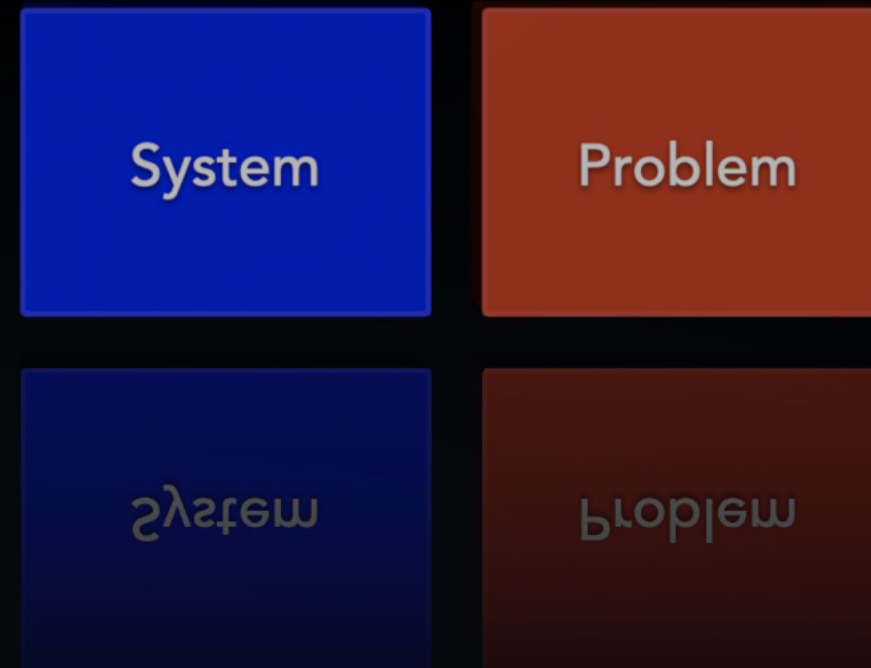
Programmtext vs. System

- FORTRAN
- Non-Computation
- OOP



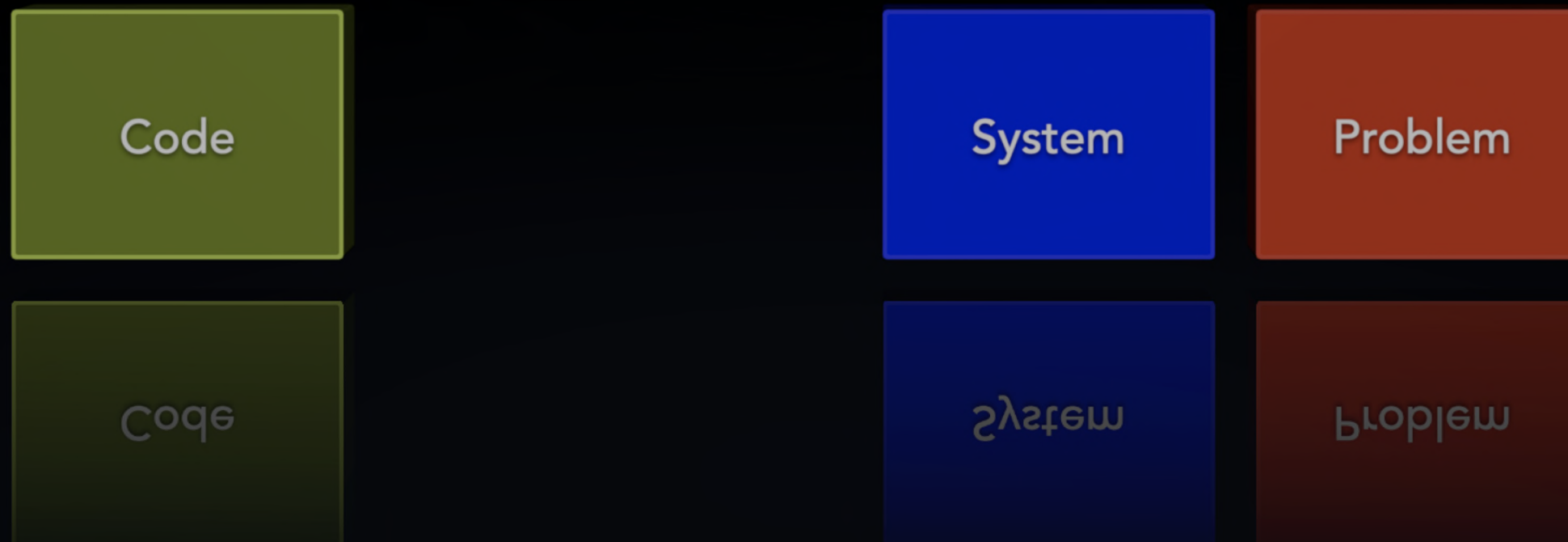
Programmtext vs. System

- FORTRAN
- Non-Computation
- OOP
- Ruby



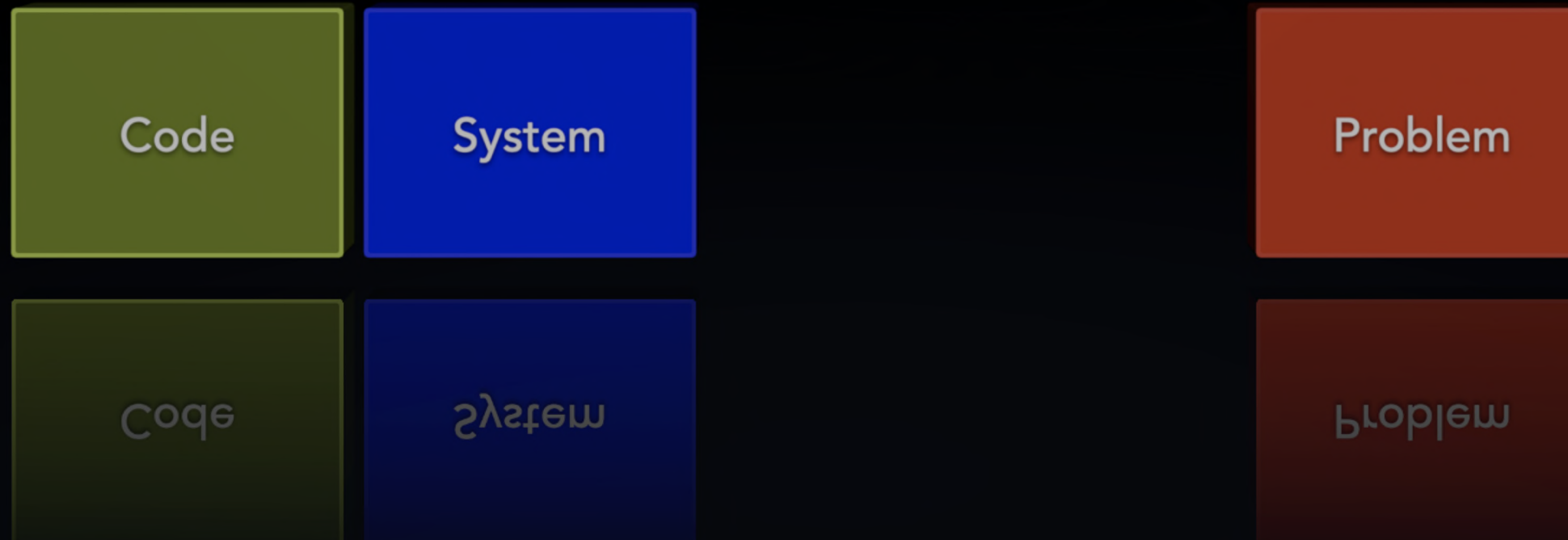
Programmtext vs. System

- FORTRAN
- Non-Computation
- OOP
- Ruby



Programmtext vs. System

- FORTRAN
- Non-Computation
- OOP
- Ruby
- FP



Programmtext vs. System

- FORTRAN
- Non-Computation
- OOP
- Ruby
- FP

Code

Code

System

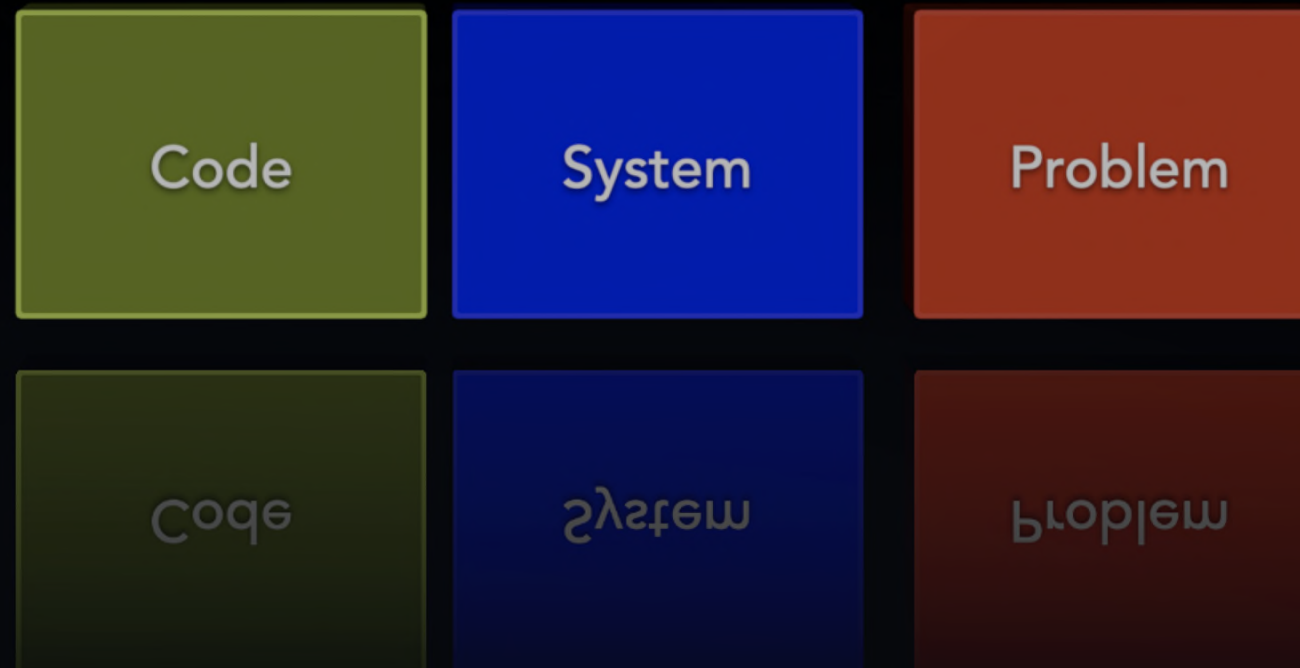
System

Problem

Problem

Programmtext vs. System

- FORTRAN
- Non-Computation
- OOP
- Ruby
- FP
- Objective-Smalltalk



Weniger kompliziert durch Architektur

Objective-Smalltalk

objective-smalltalk

Objective-Smalltalk

- Sprache (Ansatz)
- Frameworks
- Werkzeuge

Objective-Smalltalk

Sprache

Objective-Smalltalk: Sprache

- Smalltalk
- + Syntax Verbesserungen
- + Architekturkonzepte

Objective-Smalltalk: Syntax

- Klassendefinition
- Extensions
- Methodendefinition
- {} für Blöcke, statt []
- Objektliterale
- <types>
- Demo (Workspace)

Objective-Smalltalk

Architekturkonzepte

Objective-Smalltalk: Architekturkonzepte

- HOM (Sam Adams talk)
- Objektfilter
- ->
- Stores und Polymorphic Identifier

Objektfilter

- Wie Unix P/F
- Im Prozess, mit Objekten
- Ähnlichkeit zu Rx

Call/Return vs. Objektfilter

Objektfiler

- Eine Objekt das -writeObject: implementiert
- Ein target der wieder in Filter ist
- Demo

Polymorphic Identifier

- Alle Namen sind URIs
- `var:hello := 'World!'`
- Pfadnavigation, Rekursion
- `file:{env:HOME}/Downloads/site := http://objective.st`
- Demo: stsh, file, env, defaultd, db

Komposition mit ->

- Verbindet Default IN mit Default OUT
- Bei Pipes wie | mit stdin/stdout
- Unterscheid zu setXYZ: ?
- Statische Beziehung

Stores

- Wie in-process REST Server
- Oder Dictionaries
- Lokale Variablen, Environmentvariablen
- Dateisystem, defaults, HTTP
- Datenbanken: SQL/SQLite

Was ist ein Store?

- Objekt, implementiert Storage Protokoll
- -objectForReference:aRefence
- -setObjet:object forReference:aRefence
- Demo: default:, env:

Storage Combinators

- Komponierbare Stores
- Durchgehende Abstraktion
- Wunderlist, To-Do
- 2x - 10x Produktivität
- Onward! '19
- Demo: HTTP Server, Browser

Storage Combinator: Platte und Speicher

Storage Combinator: *Asynchrones Schreiben*

Property Paths

- Sugar für -objectForReference:
- Header: /path/to/:parameter
- getter: != { }
- setter: =| { }
- Demo: SiteBuilder, MethodStore, SQLite

Objective-Smalltalk

Swift

Swift und ObjST

- Frameworks von Swift nutzbar
- 80-90% der Funktionalität
- Interop zu Swift...

Zusammenfassung

Objective-Smalltalk

Objective-Smalltalk

- Vereinfachung von Sprache und Programmen
- Nicht-prozedurale Modellierung
- Iterative Entwicklung, Skripte
- Inkrementelle Adoption