

Das Icinga Projekt

Nagios Workshop 2009 in Kassel
am 22. und 23.06.2009

The Icinga logo, which consists of a stylized black and white network diagram with three nodes and connecting lines, positioned to the left of the word "icinga" in a lowercase, bold, sans-serif font.

Agenda

- **Agenda**
- **Vorstellung des Teams**
- **Icinga Mission**
- **Warum der Fork ?**
- **Entstehung von Icinga**
- **Konzept Core**
- **Konzept und Ausblick API**
- **Konzept und Ausblick Webinterface**
- **Aktueller Stand der Subprojekte**
- **Roadmap**
- **Fragen**



icinga

Vorstellung des Teams

- Michael Lübben (Webinterface)
- Hendrik Bäcker (Core)
- Matthias Flacke (Quality and Testing)
- Marius Hein (Webinterface)
- Christian Döbler (API)
- Bernd Erk (Skype Supervisor)
- Michael Friedrich (Core)
- Lara Berdelsmann (Dokumentation)
- Wolfgang Nieder (Dokumentation)



Icinga Mission

- Icinga widmet sich lang erwartenden Bugs und Features unter Sicherstellung eines klaren Migrationspfads zwischen Nagios und Icinga
- Icinga ermöglicht eine leichte Erweiterung durch klar definierte Addon-APIs. Dies erlaubt Addons auf zentrale Funktionen wie Authentifizierung und Benutzerverwaltung zurückzugreifen
- Icinga steht für klare Kommunikation innerhalb und außerhalb des Projekts unter Beteiligung aller
- Icinga Release Policy lautet: Early and Often
- Icinga ist ein Open Source Product unter GPL

Warum der Fork?

- **Uns liegt Nagios am Herzen**
- **Seit Nagios Version 3 nahezu Stillstand im Projekt**
 - Bugfixing
 - Neue Features
- **Starke Open Source Alternativen am Markt**
- **Ausbleiben des neuen Webinterfaces**
- **Probleme im Bereich von NDOUtils**
- **Fehlende API**

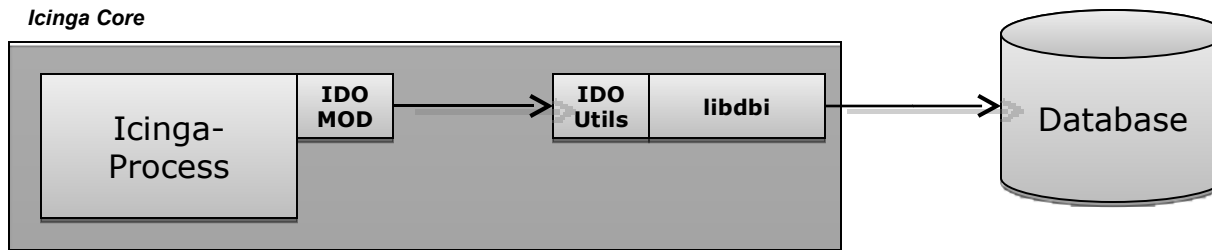
Entstehung von Icinga

- **Entscheidung für Icinga am Osterwochenende 2009**
- **Aufbau der internen Plattformen wie Redmine, Blog und Erstellung CI**
- **regelmässige Telko innerhalb des Core-Teams via Skype**
- **Rename des Nagios-Cores und Include der NDOUtils**
- **Redesign des CGI-Frontends**
- **Erster Release am 20.05.2009 als Rename von Nagios und Include der NDOUtils (V 0.8)**
- **Version mit ersten Community Patches und am 17.06.2009 (V 8.1)**

Konzept Core

- **Verwendete Komponenten**
 - Nagios in Version 3.1
 - NDOUtils 1.4b7

- **Vorteile**
 - Single Installation
 - Datenbankabstraktion



Konzept Core

- Build und Installation des Cores

Configure

```
./configure --enable-idoutils
```

Build

```
make all
```

Install

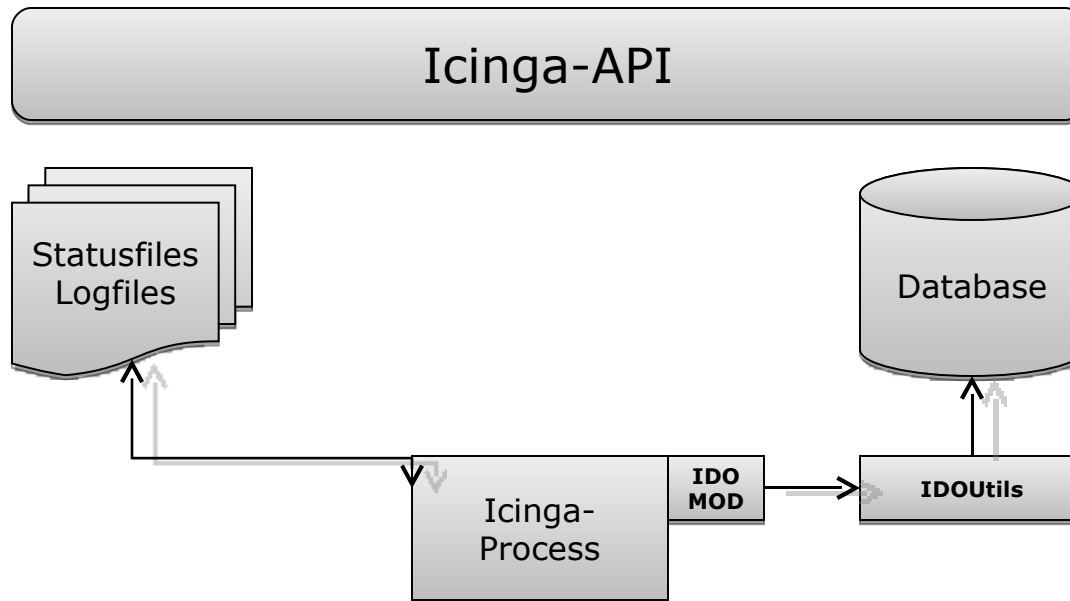
```
make install  
make install-init  
make install-config  
make install-commandmode  
make install-idoutils
```

oder

```
make fullinstall
```

Konzept und Ausblick API

- Verwendete Komponenten
 - PHP 5.2
 - PHP PDO (PHP Data Objects)



Konzept und Ausblick API

- Ein Beispiel

Zugriff konfigurieren

```
$idoConfig = array (
    'type'           => 'mysql',
    'host'           => 'localhost',
    'database'       => 'ido',
    'user'           => 'idouser',
    'password'       => 'idopassword',
    'persistent'     => true,
    'table_prefix'  => 'icinga_',
);
```

API Instanz erzeugen

```
$api = IcingaApi::getConnection(IcingaApi::CONNECTION_IDO, $idoConfig);
```

Suche erzeugen

```
$apiRes = $api->createSearch()
->setSearchTarget(IcingaApi::TARGET_HOST)
->setResultColumns(array('HOST_NAME', 'HOST_CURRENT_STATE'))
->fetch();
```

Ergebnis verarbeiten

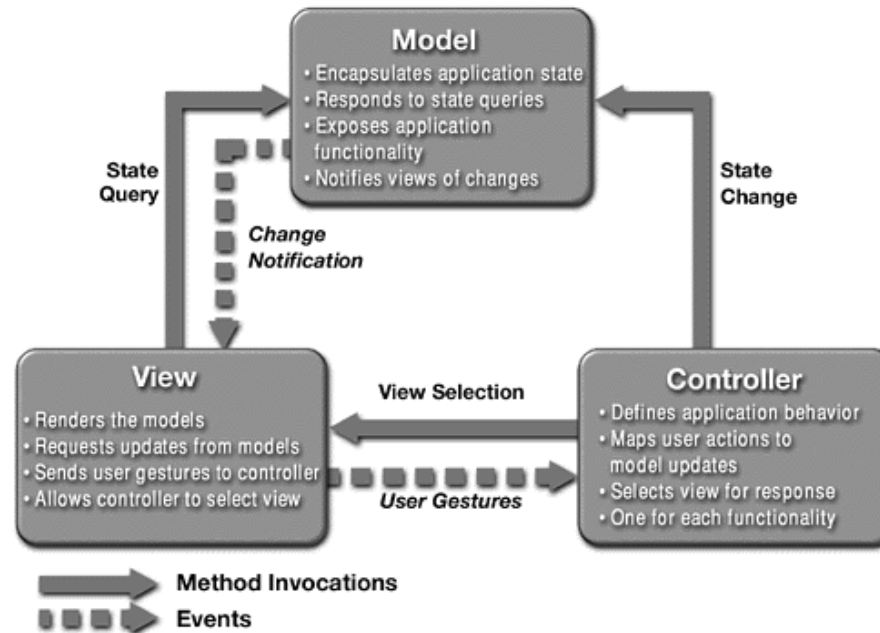
```
foreach($apiRes as $apiHandle){
    echo 'Host ' . $apiHandle->host_name . ' has state ' . $apiHandle->host_current_state . '<br />';
}
```

Konzept und Ausblick Webinterface

- Verwendete Komponenten
 - PHP 5.2
 - Agavi MVC Framework
 - YUI (Yahoo User Interface Library)
 - Doctrine (PHP Object Relational Mapper)
 - Icinga Appkit

Konzept und Ausblick Webinterface

- Agavi (MVC)



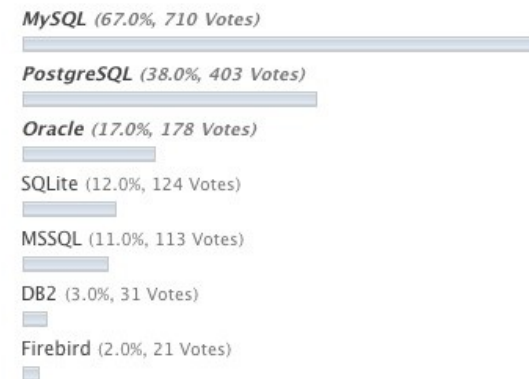
Konzept und Ausblick Webinterface

- Warum ein MVC Framework
 - MVC Paradigma definiert Eckpunkte der Entwicklung
 - erleichterte Maintenance durch zentrale Konfiguration des Frameworks
 - leichte Erweiterbarkeit durch Modulstruktur
- Warum Agavi
 - sehr strikte MVC Paradigmen erfordern einen sauberen Code
 - Unabhängigkeit von verwendeter Datenbankabstraktion
 - leichte Erweiterbarkeit durch Modulstruktur
 - Integration des Buildsystems Phing (**Phing is not Gnu make**)

Aktueller Stand

- Core
 - Core mit aktuellen Patches ist Stable
 - Integration der verfügbaren Patches
 - Hauptaugenmerk bei Unterstützung anderer Datenbankplattformen
 - MySQL
 - PostgreSQL
 - Oracle
 - Umstellung des Quickstart-Guides ins Docbook-Format

Database implementation needed for Icinga?



Total Voters: 1,062



Aktueller Stand

- Dokumentation
 - Migration der Nagios-Dokumentation in das Docbook-Format und Erweiterung um Icinga-Funktionen

```

<?xml version="1.0" encoding="UTF-8"?>
<section version="5.0" xml:id="wb_quickstart-icinga"
  xmlns="http://docbook.org/ns/docbook"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:xi="http://www.w3.org/2001/XInclude"
  xmlns:svg="http://www.w3.org/2000/svg"
  xmlns:m="http://www.w3.org/1998/Math/MathML"
  xmlns:html="http://www.w3.org/1999/xhtml"
  xmlns:db="http://docbook.org/ns/docbook">
<title>Icinga Quickstart</title>

<para><citetitle>Introduction</citetitle></para>

<para>This guide is intended to provide you with simple instructions on how
to install Icinga from source (code) and have it monitoring your local
machine within 20 minutes.</para>

<para>No advanced installation options are discussed here - just the basics
that will work for 95% of users who want to get started.</para>

<para>This guide will give you examples for currently three different Linux
distributions: Fedora, Ubuntu and openSUSE. Similar distributions may work
as well. That should include RedHat, CentOS, Debian and SLES.</para>

<para>Other distributions may inherit from these examples.</para>

```

Aktueller Stand

- Warum Docbook
 - Erstellen verschiedenster Dokumente
 - Hilfssysteme, Webseiten, Bücher, FAQs, Artikel
 - Flexibilität der Veröffentlichung bzw. Präsentation
 - Umwandlung der Dokumente mithilfe von XSLT und XSL-FO in
 - HTML
 - XHTML
 - PDF
 - PostScript
 - RTF
 - Nutzen verschiedener Stylesheets wie
 - XSL
 - FOSIs
 - DSSSL
 - CSS
 - Modularität von Dokumenten durch Includes
 - Weite Verbreitung und gute Dokumentation von DocBook
 - Von Grund auf mit dem Ziel der Erweiterbarkeit und Anpassbarkeit entwickelt

Aktueller Stand

- API
 - Datenbankzugriff zu 50% umgesetzt
 - Filezugriff zu ca. 50% umgesetzt
- Webinterface
 - Basis Appkit fertiggestellt
 - Module Sample lauffähig
 - erste Screenshots Ende Juli



Roadmap

- 0.8.2 (12.08.2009)
 - Optimierung von IDOUTils
 - Entfernung der Timed Events aus Standardkonfiguration
 - Optimierung der Indizes für performantes Housekeeping
 - Erweiterung der Datenbankunterstützung auf Basis von libdbi
 - Erste Verwendung der Icinga API für Ajax-Suche von Host, Services und
- 0.8.3 (02.09.2009)
 - Bereitstellung der gesamten Icinga-Dokumentation im Docbook Format
 - Finalisierung der API Integration in den Installprozess
- V1 Alpha (28.10.2009)
 - Bereitstellung der Icinga API auf Basis von Files und DB
 - Bereitstellung des neuen Webinterfaces

Roadmap

- Icinga im Web
 - Website und Blog (www.icinga.org)
 - Developmentplattform (dev.icinga.org)
 - Sourceforge (sourceforge.net/projects/icinga/)
- Mailinglisten
 - icinga-users
 - icinga-devel
- Others
 - Twitter (www.twitter.com/icinga)
 - Xing (www.xing.com/net/icinga/)
 - LinkedIn

FEEL FREE TO CONTRIBUTE

Fragen & Diskussion

