Seven (王东)

Objective:
Software engineer

Career planning:
System Architect/CTO

October 13,1994 〉

Chengdu, Sichuan 〉

Tianjin University of Technology 〉

Software engineering 〉

PERSONAL RESUME

Active Site

technical blog: http://www.cnblogs.com/7tiny/
github: https://github.com/seventiny
personal website: www.7tiny.com

Professional Skills

- C# (.net/.net core), Java, Python 5+ years experience
- Relational database skills like MySQL/SQL Server and Redis, Mongodb, Elasticsearch non-relational database skills
- HTML, CSS, Javascript, JQuery, Ajax, Json and other Front-end technology skills, Proficient in Restful API design
- Familiar with AOP, IOC, ORM underlying framework design principles
- Distributed microservice architecture Experience
- Familiar with SaaS, PaaS, FaaS, Experience in developing and designing low-code platforms
- git, jenkins, sonarqube、devops skills
- high standards and strict requirements for code cleanliness and readability
- Happy to develop tools for the team

Work Experience

2017.11 -- now Beisen
Position: Development/assistant leader
Work content: Responsible for the development and maintenance of the team's basic components and core business

PERSONAL RESUME

Personal Project

- Chameleon: Low-code development platform
- Bamboo.Spring: AOP/IOC/DataPipline framework
- Bamboo.Configuration: Distributed Remote Configuration Framework
- SevenTiny.Bantina.Bankinate: ORM framework
- Chameleon.FaaS: Function as a Service Engine

Work Project

- Beisen.Spring AOP/IOC/DataPipline framework (Used across departments and teams)

Personal interest projects transplanted, Pipeline middleware built using IL Emit technology.

- Beisen ORM framework (Multiple company-level sharing, cross-department and multi-team adoption)

The automatic generation and mapping of metadata fields of the enterprise PaaS platform to strongly-typed entities, unified APIs, and significantly improved development experience compared with teams that do not use components.

- Beisen Data query tool (The company's only and important business data query site)

Personal interest and spare time to complete alone, fully research the demands of multi-team developers, after multiple version iterations, has now become the company's only data query site.

- Beisen Learning Progress Calculation Core Module

Participated in and designed courses for online learning of B-side/C-side business, in-depth optimization of the progress calculation model, and improved performance, reaching 800 concurrency for a single machine, a response time of 1.436s, and 897 QPS.

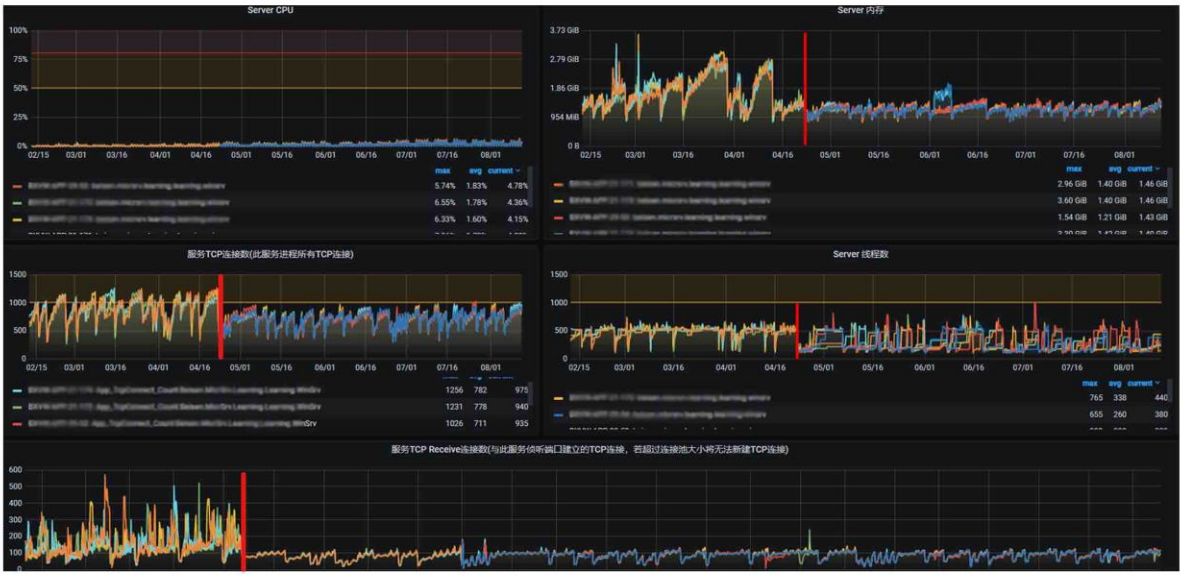
PROJECT INTRODUCTION

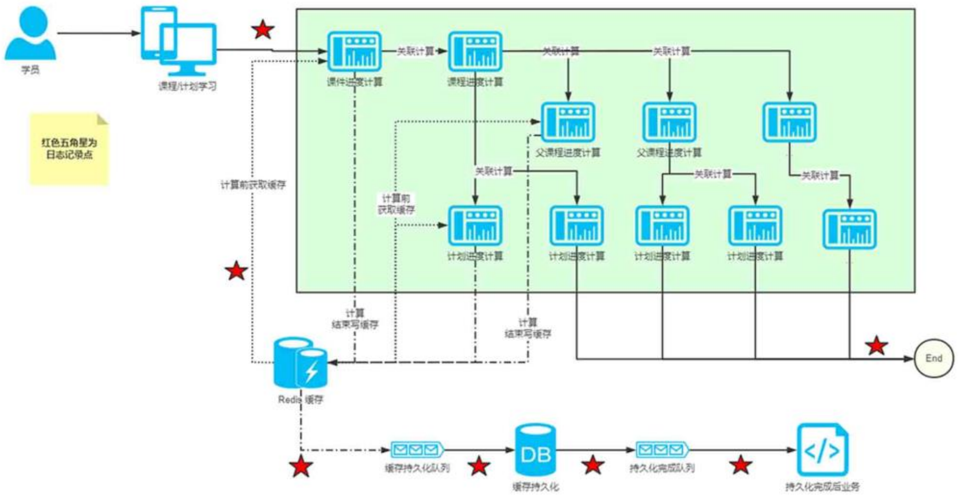
Learning Student Progress Calculation (Core Business Module)

Project Description

This project is the core business module that I am responsible for optimizing in the Beisen learning team. Due to the large number of resources involved in employee learning scenarios and a large number of queries, the basic service Cassandra server is under high pressure, and the computing business is complex and difficult to maintain. Historical logic stacking leads to customer learning computing Error rate is high. In this context, a set of progress calculation modules based on Redis/Kafka/high expansion resource calculation model are redesigned.

After redesign and optimization, 800 concurrent, 10 minutes execution, 90% response time is 1.436s, 897 QPS. The service memory usage is reduced by 60%, the interface response speed is increased by 100%, and the throughput is doubled. (apr. 22, 2022 goes online)





PROJECT INTRODUCTION

Chameleon Low-code development platform

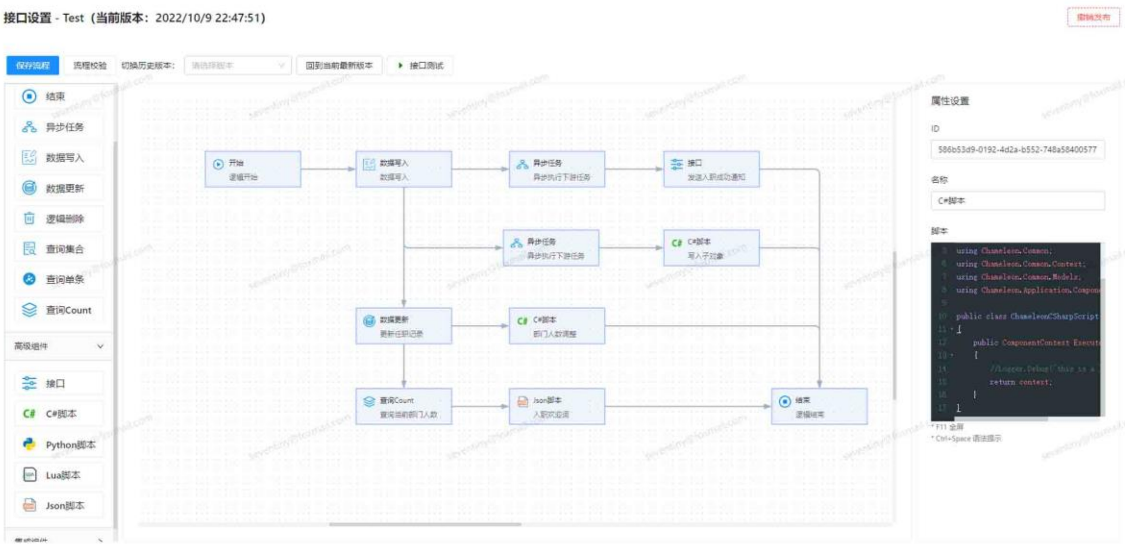
Project Description

open source address：https://github.com/sevenTiny/Chameleon

This project is a metadata-driven low-code PaaS platform developed by myself that supports multi-tenancy and takes business process configuration as the core. The modular combination method makes the platform easy to adapt to various complex business scenarios and easy to reuse.

technology stack：aspnetcore，reactjs/typescript/antd，mongodb

Technical Characteristics



接口设置 - Test（当前版本：2022/10/9 22:47:51）

属性设置

ID: 586b53d9-0192-4d2a-b552-748a58400577

名称: C#脚本

脚本:

```
using Chameleon.Common;
using Chameleon.Common.Context;
using Chameleon.Common.Models;
using Chameleon.Application.Compon

public class ChameleonCSharpScript
{
    public ComponentContext Execute
    {
        //Logger.Debug("this is a
        return context;
    }
}
```

* F11 全屏
* Ctrl+Space 语法提示

1. Preset common components, so that the interface can be completed by simply adding/deleting/modifying/checking services, which is efficient and fast

2. The streamlined configuration method makes the business logic very intuitive

3. Provides highly customizable capabilities through scripting components

4. Configuration versioning for easy restoration of historical versions

5. Support multi-tenancy, database-level isolation (uncommon field-level distinction), safe and reliable

PROJECT INTRODUCTION

Bamboo.Spring lightweight AOP/IOC/DataPipeline framework

Project Description

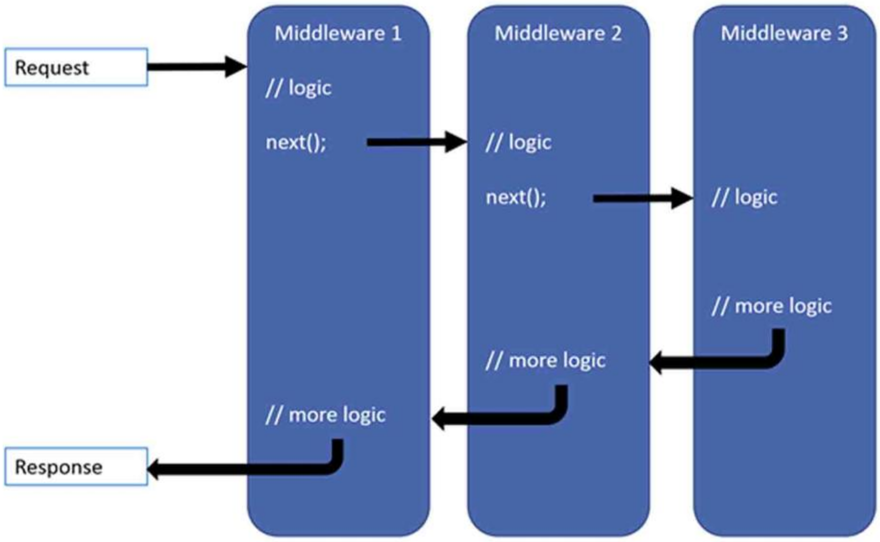
project blog ： https://www.cnblogs.com/7tiny/p/9657451.html

open source address： https://github.com/sevenTiny/Bamboo

This project is a lightweight AOP/IOC framework developed by myself. The AOP part uses Emit to generate an intermediate language to realize dynamic proxy, and supports interface/implementation mode and abstraction/inheritance mode. On this basis, the ability of AOP is used to expand the data pipeline, and support various projects to connect with highly scalable pipeline middleware.

Technical Characteristics

1. AOP dynamic proxy (Emit implementation)
2. Support for injecting method aspects in the form of annotations
3. Built-in IOC container, use built-in pipeline to control the life cycle, support construction injection and property injection
4. The pipeline model is built internally to support runtime injection of middleware like netcore, and rich middleware can be provided to the pipeline model in a flexible pluggable manner.



PROJECT INTRODUCTION

Chameleon.FaaS Function as a Service Framework

Project Description

open source address: https://github.com/sevenTiny/Chameleon.FaaS

The function-as-a-service framework is a set of flexible, reliable, dynamically scalable, multi-language environment, and sandboxed scripting service frameworks. With the capabilities of this framework, clients do not need to care about the operating environment of business scripts, machine deployment, service expansion and other infrastructure. Instead, focus on the writing of core function scripts and the trigger timing of function scripts.

The core of the FaaS framework is the "dynamic scripting engine", which provides the ability to compile, load, and execute code scripts at runtime.

Through this capability, script codes stored in file systems such as databases and disks can be acquired in real time, compiled and loaded into memory for execution.

The ability of dynamic compilation can be easily applied in serverless, PaaS, FaaS, sandbox, plug-in and other scenarios to meet the requirements of "dynamic" and "hot swap" to the greatest extent.

Technical Characteristics

Support mode:

1. Nuget SDK local mode
2. Rpc remote execute mode
3. Restful api request mode
4. Timed scheduler mode

Supported languages:

only C# language is supported now, and in the future, it will be implemented in multiple languages such as java, python, javascript, etc.

PROJECT INTRODUCTION

Bamboo.Configuration Distributed Remote Configuration Framework

Project Description

open source address：https://github.com/sevenTiny/Bamboo

This project is a distributed remote configuration component developed by myself, which solves the problem of synchronous remote configuration of multiple service nodes in the microservice scenario. The framework supports registering different data sources as remote configuration centers (mysql, git, interfaces, json files, xml files, etc.). Support remote configuration hot update, local mode and other features.

Technical Characteristics

1. Support database, git service, json, xml file as remote configuration center data source
2. The abstract access layer supports the expansion of various data sources and has strong scalability
3. Supports different configuration formats such as json files, xml files, etc.
4. The configuration file is automatically mapped to the strongly typed entity model, which is convenient for the client to use
5. Synchronize remote configuration changes in a timed pull mode
6. Support local mode, support local configuration of a node, suitable for different scenarios

PROJECT INTRODUCTION

SevenTiny.Bantina.Bankinate ORM framework

Project Description

project blog ： https://www.cnblogs.com/7tiny/p/9575230.html
open source address： https://github.com/sevenTiny/SevenTiny.Bantina.Bankinate

This project develops a lightweight ORM framework for myself, which realizes the lambda API of SQL statements, provides efficient Mapping of strongly typed entities, supports different types of databases, and provides free switching between different databases without changing the code.

Technical Characteristics

1. Dynamically parse SQL statements， Linq API Lambda
2. API database isolation, easy to replace different databases later
3. Query sets are automatically mapped to strongly typed entities
4. Supports multiple databases such as relational/non-relational (abstract database layer)
5. Support local and distributed secondary cache (redis/local)
6. Support for model-bound data validation (extension)
7. Support load balancing in database master-slave mode
8. The project is fully pluggable. Different extension packages are used according to different databases.