Design of a Digitizer-Oriented-Middleware and Applications

Yibing Dong[1, 2,a], Tan Liu[2,b], and Yinjie Zhu[3,c]

[1]University of Science & Technology of China, China

[2]Seismological Bureau of Hebei Province, China

[a]yuehun06@163.com, [b]280613824@qq.com, [c]526252372@qq.com

Keywords: digitizer-oriented-middleware； seismographic instrumentation； state-of-health； web service

Abstract. The network protocols and data formats of several types of seismic digitizer are analyzed, and the development of a middleware is introduced. The middleware provide users with java APIs to implement communication with seismographic digitizers and data receiving, decoding, publishing and storing. It uses a distributed structure with synchronous pipeline, and message delivery mechanism is introduced to implement synergies between components. In practice, with the middleware, a state-of-health monitoring software is developed, which can collect SOH data from digitizers periodically and push them to users with Web interfaces. Using the software, engineers can identify errors quickly.

Introduction

Digitizer is one of the key instruments in the seismic network, which acquires analog signals from the seismographer and converts them into digital signals with AD module[1]. These signals are output via Ethernet in form of TCP/IP flow, for example, in CMG-DM24, there are several flows defined to transmit data with different functions including heartbeat flow, seismic waveform data flow and mass position flow etc[2]. Different from the relational data set, TCP/IP flow is a special kind with characteristics such as its continuous producing, real-time arriving and potential unlimited amounts[3]. Based on these characteristics, there often designs a accessing layer between the data source and application, which continuously and asynchronously collects data from the underlying and sends them to the upper. This hierarchical structure decouples the data producer process and consumer process, which will help to improve the robustness and throughput of the system. This kind of structure has been applied in the seismic network software system, for example, JOPENS system designed a seismic stream service to collect and distribute the seismic waveform data[4]. In practice, because of the limitation of the maximum number of concurrent connections in digitizers (for example, in EDAS24-GN, the number is 4 by default), it's necessary to design a unified data access service to achieve the reception and distribution of all kinds of stream[5].

Requirements

Middleware is a kind of independent system software or service program, locating upper the operating system, responsible for the management of computing and communication resources, with which the distributed applications conduct data exchange between different systems[6]. Our middleware locates between digitizer and computer. In this environment, computer is the client, and the middleware acts as a proxy server. When the system is running, the computers request data from digitizers via the APIs provided by the middleware. This application environment has several features that demand the middleware to support certain function and performance requirements including: (1) Heterogeneity: communication protocols and data formats differ because of manufacturers, leading to the heterogeneity of data source, so the middleware should shield the difference and provide the unified interface. (2) Concurrency: the middleware always faces concurrent access from multiple computers and digitizers so it must support multi-thread concurrency. (3) Timeliness: the Ethernet data flow is instantaneous and perishable, so the middleware must support both high real-time and throughput. (4)

Asynchronous: the digitizer and computer usually work in asynchronous steps, so it must provide a cooperation mechanism such as using a message-oriented-middleware e.g.

Design

The Structure. The producer/consumer model is adopted to to support multi-thread concurrency, which is a classic concurrent model. It decouples the producer and consumer process, and uses a buffer as a shared container to support data exchanging between them, which is proved to be efficient for the implementation of load balancing[7].

With object-oriented-design method, the middleware can be decomposed into five core components including digitizer adapter, data buffer, message publisher, message-oriented-middleware and message subscriber. Each component will provide APIs to perform a particular function. Where, instruments adapter is responsible for building communication with instruments, receiving data and storing them into the data buffer. Data buffer is a thread-safe queue. When there are only producers without consumers, newly created data will be added at the tail of the queue continually; and once a consumer comes, data head of the queue will be consumed firstly. Message publisher acts as a consumer when getting data from buffer and a producer when sending message to the message-oriented-middleware. The message-oriented-middleware is a third-party message service provider, responsible for the management of the message queue, supporting multiple communication models including publish/subscribe, point to point and pushing, supporting message persistence. Message subscriber is the consumer of messages with APIs to require data from instruments. In the structure, each component runs parallel and asynchronously and collaborate with messages, exceptions are suppressed without spreading, help to enhance the robustness and throughput.

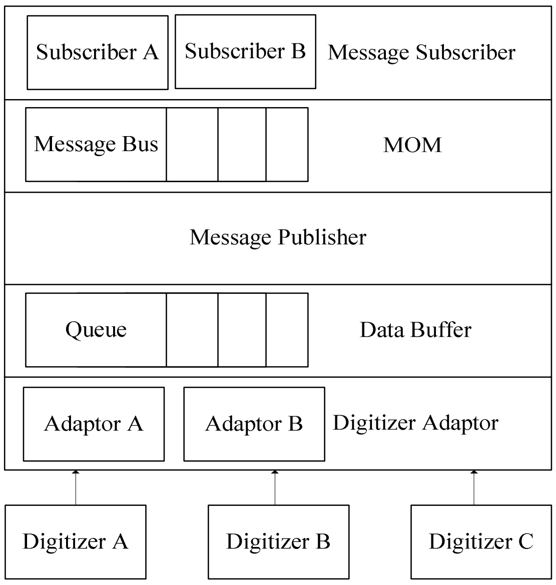


Figure 1. Structure of the Middleware

The Digitizer Adapter. This component is responsible for communicating with digitizers, collecting and storing data from digitizer to buffer. Digitizers use embedded TCP/IP protocols, which has a vertical muli-layer architecture[8]. In the digitizer, data from sensors will be encapsulated into IP packages and broadcast through Ethernet Interface. These packages will be captured by the computer, and after a serial of inverse processes reverted to the original data that can be used by the application processes directly. Socket is an abstract layer, which provides the upper layer with service interface of the transport layer. According to the transport layer protocols, there are 2 kinds of Sockets, stream (TCP used) and datagram (UDP used). TCP is a connection-oriented, reliable protocol, and UDP is a connection-less and unreliable protocol, they are used by different kinds of digitizer[9]. For example, EDAS-24 uses the TCP, and CMG-DM24 uses UDP.

Application

Requirements. The middleware is used in an system to monitor the state-of-health (SOH) of instrumentation at telemetry seismographic stations from both desktop and Android. In an station, the instrumentation mainly includes seismic sensors, digitizers, power supply and communication devices. When running, there may has exceptions taken place that will reduce the station's observation effectiveness, and SOH data created by the digitizer can provide tips and warnings for some exceptions such as the Input voltage, Clock Error, Frequency Error, Free Space and Mass Positions etc. Table 1 includes every SOH data and its normal range in the system. When a detected value exceeds its normal range, there may be an exception taken place that certain instrumentation needs to be checked and debugged. The range can be customized by the users.

Table 1 SOH data dictionary

Id	Name	Description	Normal Range
1	Pwr_input	Input voltage @ V	[-9，18]
2	Pwr_output	Output voltage @ V	[-9，18]
3	Pwr_battery	Battery voltage @ V	[0，9]
4	Dsp_temp	Temperature @ ℃	[-20，60]
5	Clk_diff	Clock Error @ μS	[-100，100]
6	Frq_diff	Frequency Error @ PPB	[-10，10]
7	Drive_free_space	Free Space @ 100%	[20，100]
8	Z_mp_volts	Mass Position of Chanel UD @ V	[-1，1]
9	N_mp_volts	Mass Position of Chanel NS @ V	[-1，1]
10	E_mp_volts	Mass Position of Chanel EW @ V	[-1，1]

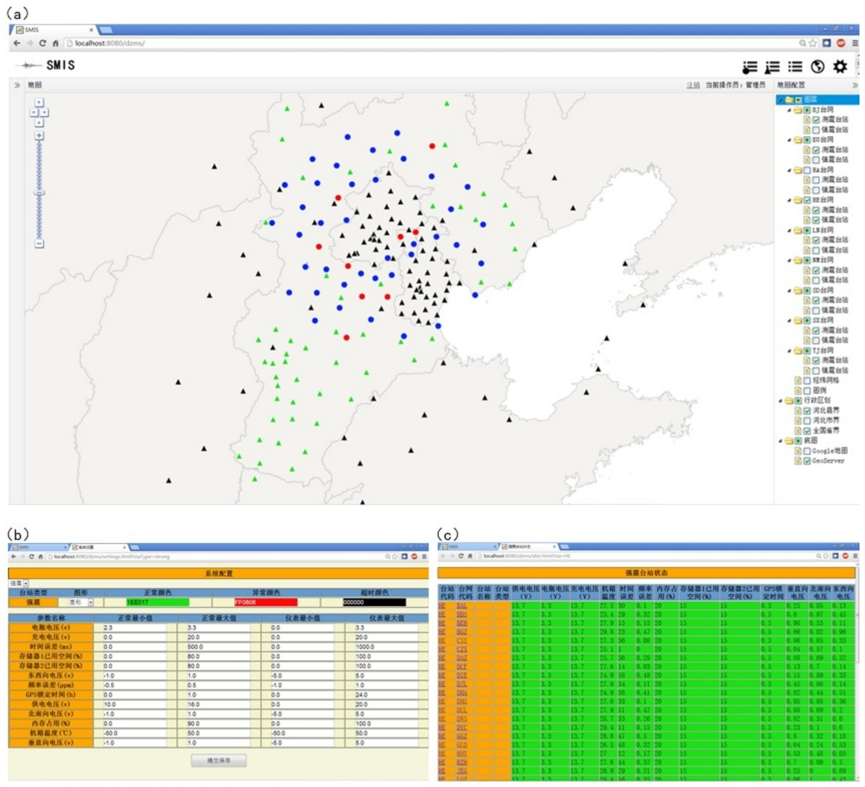
Implementation. Figure 2 provides a group of screen-shot of the interfaces at client end. Figure 2(a) is the main page, it shows the real-time status of instrumentation of the regional seismological network in North China. The status will refresh automatically at a default period, where triangle and circle stands for stations with different functions, green and blue stands for running well, red for error, and black for time out. Figure 2 (b) is the configuration page, where users can change the shape, color of station and SOH range according to themselves' interests. Figure 2 (c) is a list including all the stations and their SOHs.

Conclusions

In this paper, the network protocols and data formats of several types of seismic digitizer are analyzed, and the development of a middleware is introduced. The middleware provide users with java APIs to implement communication with seismographic digitizers and data receiving, decoding, publishing and storing. It uses a distributed structure with synchronous pipeline, and message delivery mechanism is introduced to implement synergies between components. In practice, with the middleware, a state-of-health monitoring software is developed, which can collect SOH data from digitizers periodically and push them to users with Web interfaces. Using the software, engineers can identify errors quickly.

Acknowledgments

This work was financially supported by the Science and Technology Support Program of Hebei Province (15275402D).



系统配置

台站类型	图形	正常颜色	异常颜色	超时颜色
强震	图形	1EE017	FF0808	000000

参数名称	正常最小值	正常最大值	仪表最小值	仪表最大值
电瓶电压(v)	2.3	3.3	0.0	3.3
充电电压(v)	0.0	20.0	0.0	20.0
时间误差(ms)	0.0	500.0	0.0	1000.0
存储器1已用空间(%)	0.0	80.0	0.0	100.0
存储器2已用空间(%)	0.0	80.0	0.0	100.0
东西向电压(v)	-1.0	1.0	-5.0	5.0
频率误差(ppm)	-0.5	0.5	-1.0	1.0
GPS锁定时间(h)	0.0	1.0	0.0	24.0
供电电压(v)	10.0	16.0	0.0	20.0
北南向电压(v)	-1.0	1.0	-5.0	5.0
内存占用(%)	0.0	90.0	0.0	100.0
机箱温度(℃)	-50.0	50.0	-50.0	50.0
垂直向电压(v)	-1.0	1.0	-5.0	5.0

强震台站状态

台站代码	台网代码	台站名称	台站类型	供电电压(V)	电瓶电压(V)	充电电压(V)	机箱温度	时间误差	频率误差	内存占用(%)	存储器1已用空间(%)	存储器2已用空间(%)	GPS锁定时间	垂直向电压	北南向电压	东西向电压
HE	BAL			13.7	3.3	13.7	27.5	30	0.1	20	15	15	0.5	0.25	0.05	0.13
HE	BBS			13.7	3.3	13.7	25.4	29	0.32	20	15	15	0.5	0.8	0.97	0.45
HE	BEH			13.7	3.3	13.7	27.9	15	0.15	20	15	15	0.5	0.98	0.53	0.11
HE	BGZ			13.7	3.3	13.7	29.8	23	0.47	20	15	15	0.5	0.66	0.02	0.96
HE	CSY			13.7	3.3	13.7	27.3	36	0.08	20	15	15	0.5	0.86	0.85	0.33
HE	CZX			13.7	3.3	13.7	25.1	1	0	20	15	15	0.5	0.04	0.57	0.1
HE	DAZ			13.7	3.3	13.7	25.7	36	0.28	20	15	15	0.5	0.88	0.69	0.52
HE	DCF			13.7	3.3	13.7	27.6	14	0.03	20	15	15	0.5	0.13	0.7	0.14
HE	DJE			13.7	3.3	13.7	24.9	58	0.48	20	15	15	0.5	0.15	0.89	0.33
HE	DJL			13.7	3.3	13.7	27.9	34	0.11	20	15	15	0.5	0.45	0.86	0.14
HE	DMA			13.7	3.3	13.7	24.9	38	0.41	20	15	15	0.5	0.82	0.44	0.51
HE	DMG			13.7	3.3	13.7	27.6	33	0.1	20	15	15	0.5	0.95	0.88	0.36
HE	DUL			13.7	3.3	13.7	27.9	51	0.47	20	15	15	0.5	0.88	0.69	0.2
HE	DWS			13.7	3.3	13.7	28.7	33	0.26	20	15	15	0.5	0.82	0.31	0.6
HE	DYC			13.7	3.3	13.7	29.4	11	0.15	20	15	15	0.5	0.23	0.1	0.6
HE	GGZ			13.7	3.3	13.7	26.6	47	0.5	20	15	15	0.5	0.8	0.32	0.18
HE	GLD			13.7	3.3	13.7	26.5	48	0.32	20	15	15	0.5	0.64	0.24	0.53
HE	HNY			13.7	3.3	13.7	27	12	0.17	20	15	15	0.5	0.53	0.48	0.05
HE	HZH			13.7	3.3	13.7	27.6	44	0.37	20	15	15	0.5	0.7	0.09	0.5
HE	JEG			13.7	3.3	13.7	26.9	29	0.21	20	15	15	0.5	0.25	0	0.69
HE	LGZ			13.7	3.3	13.7	28.4	16	0.35	20	15	15	0.5	0.06	1	0.43

Figure 2 Interfaces of the Client

References

[1] Monitoring and Forecasting Department of China Earthquake Administration, Digital Seismic Observation Technology, Beijing:Earthquake Press, 2003.

[2] Guralp Systems, Inc. CMG-DM24 Operator’s Guide, 2009.

[3] Gehrke J, Data Stream Processing, IEEE Computer of Technical Common Data Engg., 2003, 26(1): 111-122.

[4] WU Yong-quan, HUANG Wen-hui, SU Zhu-ji, Real-time Data Transmission and Service of Chinese Seismic Network, South China Journal of Seismology, 33(3):77-84.

[5] DONG Yi-bing, GAO Jing-chun, LIU Sheng-guo, et al. Design and Implementation of Seismometric Instrument Health State Data Exchange Platform, Journal of Seismological Research, 36(3):379-383.

[6] ZHOU Yuan-chun, LI Miao, ZHANG Jian, et al. 2002, Overview on Middleware Technology, Computer Engineer and Applications, 38(15):80-82.

[7] Goetz B, T Peierls, J Bloch, et al. Java Concurrency in Practice , Texas: Pearson Education, Inc. 2012.

[8] Fall K R, W R Stevens, TCP/IP Illustrated Volume 1 (Second Edition), New Jersey: Addison-Wesley Educational Publishers Inc. 2012.

[9] DONG Yi-bing, HE Yong-bo, LIU Qiang, et al. 2015, Design and Application of an Seismic Instrument Stream Accessing Framework, Journal of Seismological Research, 38(2):326-331.