Feng et al. Journal of Cloud Computing (2022) 11:39
https://doi.org/10.1186/s13677-022-00318-1

Journal of Cloud Computing:
Advances, Systems and Applications

RESEARCH Open Access

A maturity model for AI-empowered cloud-native databases: from the perspective of resource management

Xiaoyue Feng, Chaopeng Guo, Tianzhe Jiao and Jie Song*

Abstract

Cloud-native database systems have started to gain broad support and popularity due to more and more applications and systems moving to the cloud. Various cloud-native databases have been emerging in recent years, but their developments are still in the primary stage. At this stage, database developers are generally confused about improving the performance of the database by applying AI technologies. The maturity model can help database developers formulate the measures and clarify the improvement path during development. However, the current maturity models are unsuitable for cloud-native databases since their architecture and resource management differ from traditional databases. Hence, we propose a maturity model for AI-empowered cloud-native databases from the perspective of resource management. We employ a systematic literature review and expert interviews to conduct the maturity model. Also, we develop an assessment tool based on the maturity model to help developers assess cloud-native databases. And we provide an assessment case to prove our maturity model. The assessment case results show that the database's development direction conforms to the maturity model. It proves the effectiveness of the maturity model.

Keywords: Cloud-native database, Maturity model, AI-empowered, Resource management, Maturity assessment

Introduction

Cloud-native databases (CNDBs) have become increasingly important in cloud computing due to various applications' need for elasticity, scalability, manageability, and on-demand usage [1]. These challenges from cloud applications present new opportunities for CNDBs that traditional on-premise enterprise database systems cannot fully address. CNDBs have the features of multi-tenant, compute and storage disaggregation, logs as the database, etc. These features make the database more elastic and scalable, which addresses several challenges described above. Same as traditional databases, CNDBs also support Artificial Intelligence (AI) technology for database optimization. AI technologies applied in the features of CNDBs dramatically improve the performance of the database. In the background of the constant development of AI technology, AI-empowered CNDBs are the trend. Nowadays, the development of CNDBs is still in its infancy, and the application of AI technologies in CNDBs is immature [2]. Without a maturity model for AI-empowered CNDBs, database developers may be confused about the application of AI technologies, and database users may be confused about selecting CNDBs. Therefore, a maturity model for assessing the AI-empowered CNDBs, which serves as a tool to assess the as-it situation of CNDBs and sheds light on step-by-step improvements, is on demand.

The current maturity models for assessing database service capability are oriented to all databases and cover many dimensions [3]. However, the architecture of CNDBs is different from traditional distributed databases. CNDBs are designed to take advantage of cloud

*Correspondence: songjie@mail.neu.edu.cn

Software College, Northeastern University, Shenyang, China

infrastructure [4–6]. And CNDBs' performance dramatically relies on the cloud resource management strategies compared with traditional databases, while the database maturity model without the resource perspective fails to assess the CNDBs' performance. For example, since CNDBs provide services for multiple tenants with a high resource scheduling capability, the maturity model should assess the comprehensive performance of the database in resource allocation for multiple tenants. Last but not least, as the application of AI technologies can significantly improve the resource utilization [7–11], the maturity model should consider the AI dimension. For example, AI technologies can predict the resources required by tenants and perform efficient resource scheduling. To this end, the existing maturity models are unsuitable for assessing AI-empowered CNDBs. Conducting a systematic synthesis of AI-empowered CNDBs from the perspective of resources is needed to bring out the most important dimensions and indicators for assessing and improving the performance of CNDBs.

Our study aims to develop a maturity model of AI-empowered CNDBs, helping developers assess and improve the capability of CNDBs to apply AI technologies from the resource perspective to better leverage them for applications. Our study adopts a systematic literature review and expert interviews to develop the maturity model. Moreover, we develop an assessment tool based on the model to assess CNDBs and provide an assessment case to prove the maturity model.

The contributions of our study are threefold. First, we propose a theoretical maturity model for assessing CNDBs. Especially, the model has indicators for the capability of CNDBs to apply AI technologies to their resource management. Second, our findings provide the foundation to help researchers assess the development level of AI-empowered CNDBs and formulate measures for step-by-step improvement according to the characteristics of the next higher level of the maturity model. Third, the method of developing the maturity model in our study has a reference role for the research of maturity models in other AI-empowered fields, and further promotes the application of AI technologies in more fields.

The rest of this paper is structured as follows. Section "Related work" reviews the state-of-the-art maturity models. Section "Analysis Methods and Results" presents the research methods that conduct our maturity model. Section "Definition" defines the proposed maturity model. Section "Assessment" introduces the assessment tool and the assessment case. Section "Conclusion and Future work" concludes the paper.

Related work

This section analyzes related studies about maturity models and reveals the limitations of the existing maturity models that lead to our study.

AI technology has been applied in many fields [12–15]. Enterprises usually utilize the maturity model related to technologies to appreciate the capabilities of technologies and improve enterprises' capability to apply technologies [16]. A maturity model has two common components: the measured objects and maturity levels. The former are dimensions or criteria such as application targets of technologies within specific measured indicators. The latter are a set of sequential development stages/degrees for the examined object. Maturity models are proposed for guiding developers in designing databases, managing cloud resources, and employing AI technologies. Table 1 lists relevant maturity models in our study context.

The model in Table 1 cannot be used to assess AI-empowered CNDBs. First, the "service capability maturity model of data center" and "service capability maturity model of database" propose the standard specification to measure the service capacity of data centers and databases. They are generalized ones and cover various dimensions. For example, the "service capability maturity model of data center" has three domains and 11 subdomains with 33 specific capabilities; and the "service capability maturity model of database" has three domains with 27 specific capabilities. However, CNDBs have distinctive features, making it difficult for these maturity models to assess the performance of CNDBs. Second,

Table 1 Relevant maturity models in the context of our study

Name of model	Dimensions and/or measured indicators
Information technology service—Service capability maturity model of data center [3]	3 domains: Strategic development (3 subdomains), Operation guarantee (5 subdomains), Organizational governance (3 subdomains), with 33 specific capabilities
Service capability maturity model of database [17]	3 fields: Planning and design, Operation and maintenance, and Implementation deployment, with 27 specific capabilities
Cloud Application Maturity Model [18]	Cloud utilization
Maturity model to assess the development of industrial AI in smart manufacturing [19]	2 domains: Industry (5 subdomains), Artificial intelligence (7 subdomains), with 35 specific capabilities

the “Cloud Application Maturity Model” focus on cloud resource utilization in cloud applications, not in CNDBs. Third, the “Maturity model to assess the development of industrial AI in smart manufacturing” is a maturity model that helps manufacturing firms assess their performance in the industrial AI journey. It cannot assess the performance of AI technologies applied in CNDBs. Moreover, even if we combine the “Maturity model to assess the development of industrial AI in smart manufacturing” with the “Cloud Application Maturity Model”, it can only assess the performance of AI technologies in cloud applications. To this end, these maturity models cannot assess AI-empowered CNDBs from the resource perspective, either individually or in combination.

Analysis methods and results

Over recent years, the number of published maturity models has increased considerably [20, 21]. However, the methodological rigor regarding model development is weak and flawed, resulting in the quality of the maturity models that do not match the current publication quantity [22]. Hence, Felch et al. suggest using literature review and explorative research methods to develop maturity models. And many studies have adopted the systematic literature review (SLR) method and the expert interview to conduct maturity models [19, 23, 24]. We follow the convention to develop the maturity model of AI-empowered CNDBs.

To develop the maturity model, we first conduct a systematic review of empirical studies to identify the measured indicators. Then, we perform expert interviews to establish the maturity levels and determine the characteristics related to different development stages of CNDBs. This section explains the analysis and results of the systematic review and expert interview.

Literature analysis

We review and analyze the literature on CNDBs and AI for CNDBs to identify the measured indicators of the model. The systematic literature review (SLR) method is a better choice because a comprehensive literature analysis on assessing AI-empowered CNDBs is rare. The SLR method follows a rigorous procedure for searching and selecting the sample studies. It is a methodical process of collecting and organizing the published empirical studies with systematic selection criteria to reduce the deviation.

We adopt the evidenced-based paradigm [25] to perform the SLR. As shown in Fig. 1, the SLR process contains four steps. First, following the ways of determining the keywords [26, 27], we identify three major search terms based on the aim of our study: “cloud-native database,” “artificial intelligence,” and “assessment” to develop the alternative terms for search. These keywords are connected with Boolean operators to serve as search strings. Our search begins with the search strings. In this way, we obtain a comprehensive perspective on literature. Second, we perform the search in the specific eight online databases and filter papers from non-computer industries to identify an initial list of articles ($n = 150$) for selection. Third, we retain 15 papers according to our inclusion and exclusion criteria (shown in Fig. 1). Then, we conduct the forward (finding citations to the papers) and backward (using the reference list to identify new papers) search to include 32 articles further. A total of 47 suitable articles are eventually retrieved. Finally, we review each of the 47 articles thoroughly and identify a list of indicators for assessing AI-empowered CNDBs. In the following, we analyze these articles regarding CNDBs and AI for CNDBs.

CNDBs approximately belong to two branches. One is based on Spanner, such as CockrochDB, TiDB, YugabyteDB, etc. The other is based on Aurora, such as Socrates, PolarDB, CynosDB, ArkDB, TarusDB, etc. These databases have different features, but most of their features are the same. Table 2 compares several CNDBs and shows that their common features are multi-tenant, compute and storage disaggregation, cross Az/Region, near-data processing, logs as the database, and distributed and shared memory.

Furthermore, as a database designed for cloud architecture, CNDBs’ performance is dominated by the effectiveness of cloud resource management. Recent studies apply AI technologies to resource management [34–36].

Table 3 enumerates several studies. We can group them into four categories: resource prediction, resource scheduling, resource control, and resource scaling.

In summary, we identify two dimensions for the maturity model: the cloud-native database and artificial intelligence. Table 2 shows that the cloud-native database dimension has six indicators. And Table 3 shows that the artificial intelligence dimension has four indicators. We finally identify ten indicators for the maturity model based on the results of SLR. These indicators will be described in the section “Definition”.

Expert interviews

Our study adopts the method of the semi-structured expert interview to establish the maturity levels of the proposed maturity model. There are two reasons for it. First, since there are few articles about determining the maturity levels, the SLR method cannot be applied to identify the levels. Second, determining the maturity levels is subjective, and researchers have different opinions on their definitions, so summarizing expert insights can cover the views to the maximum extent.

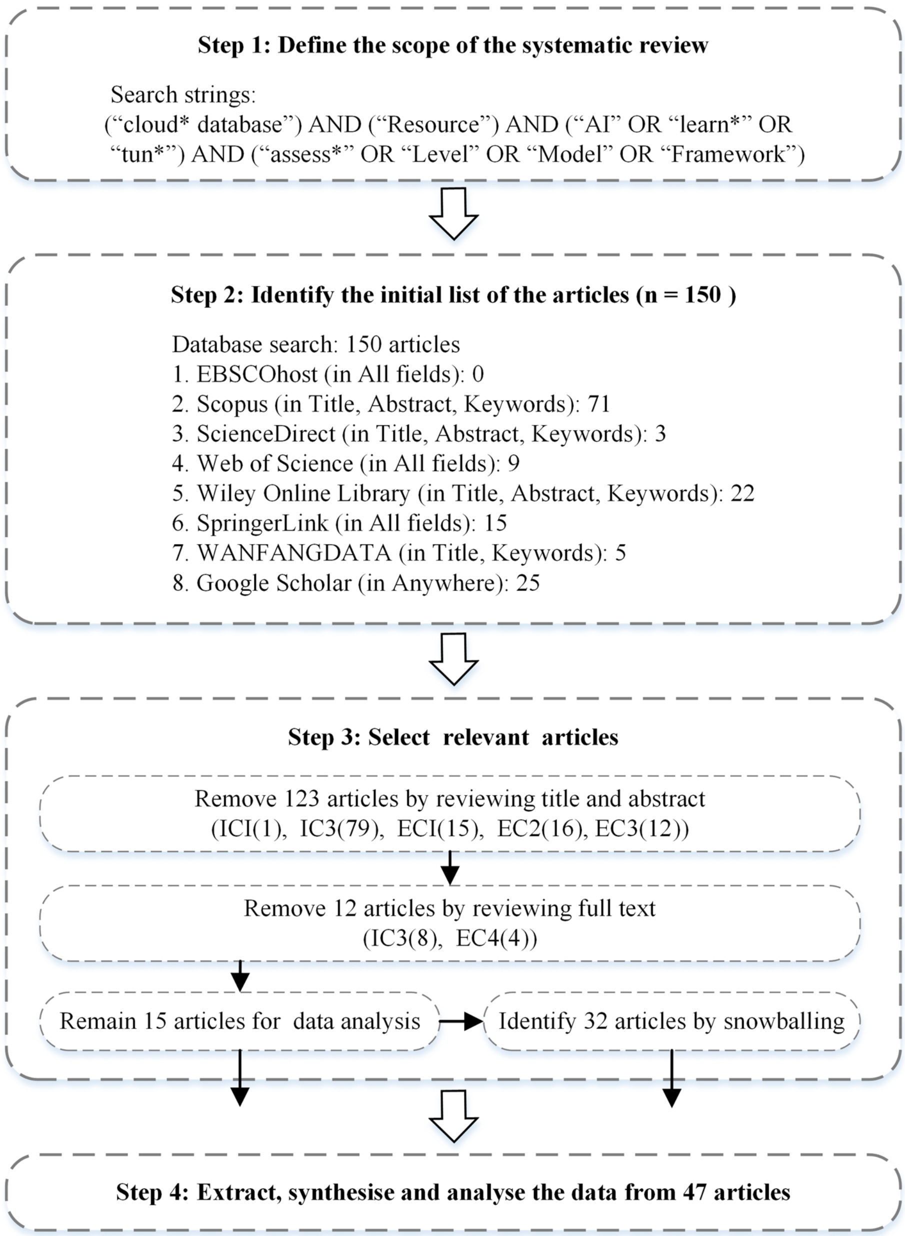

Fig. 1 The SLR process in our study

To determine the list of experts to be interviewed, we find the relevant studies and identify the authors and their affiliations from the studies based on the results of SLR. Since these selected studies have a few industrial reports, it helps to identify the experts with the expertise or work experience in developing AI-empowered CNDBs. Eventually, we invited eight experts to participate in our study. Among them, five are consultants, and three are Product Managers (PM). All participants have a minimum job experience in database and AI technology of 3 years.

To obtain the knowledge and experience of the experts about AI-empowered CNDBs, we conduct a guideline of the interview, which is developed from the literature review. The guideline consists of three parts: start-up, trigger, and follow-up questions. The start-up introduces the purpose of the interview to the experts and helps us understand their job position, background, and related experience. In the trigger part, we give these experts the description of the identified indicators. Then, the experts should provide us with the number of maturity levels and the description of each

Table 2 Comparison of several databases

Database	Feature	Description
Spanner	logs as the database, compute and storage disaggregation, multi-tenant, cross Az/Region, near-data processing, distributed and shared memory	Spanner [28] is a scalable, globally distributed database designed, built, and deployed at Google. It is designed to scale up to millions of machines across hundreds of datacenters and trillions of database rows.
TiDB	HTAP, logs as the database, compute and storage disaggregation, multi-tenant, cross Az/Region, near-data processing, distributed and shared memory	TiDB [29] is a Raft-based HTAP database. It has a multi-Raft storage system which consists of a row store and a column store. It can build a SQL engine to process large-scale distributed transactions and expensive analytical queries.
Amazon Aurora	multi-tenant, compute and storage disaggregation, cross Az/Region, near-data processing, logs as the database, and distributed and shared memory	Amazon [30] Aurora is a relational database service for OLTP workloads. Aurora pushes redo processing to a multi-tenant scale-out storage service, purpose-built for Aurora, which not only reduces network traffic, but also allows for fast crash recovery, failovers to replicas without loss of data, and fault-tolerant, self-healing storage.
POLARDB	multi-tenant, compute and storage disaggregation, cross Az/Region, near-data processing, Hardware-software synergy, logs as the database, distributed and shared memory	POLARDB [31] is a new cloud-native OLTP database designed by Alibaba Cloud. Database computing nodes and storage nodes are connected through a high-speed RDMA network. To ensure high availability, POLARDB uses the Parallel-Raft protocol to write three copies of data across the storage nodes [32].
TaurusDB	multi-tenant, compute and storage disaggregation, cross Az/Region, near-data processing, logs as the database, and distributed and shared memory	TaurusDB [33] is a new multi-tenant cloud database system. It separates the compute and storage layers in a similar manner to Amazon Aurora and Microsoft Socrates and provides similar benefits, such as read replica support, low network utilization, hardware sharing, and scalability.

Table 3 Studies about AI empowered resource management

Type	Reference	Introduction
Resource prediction	[37]	It captures online features while running the job (e.g., job, data, and cluster characters) and tunes the parameters based on estimated resource consumption (e.g., time, CPU, memory) in job-level tuning.
	[38]	It proposes AutoClustC, which estimates the costs of resource provisioning and database repartitioning and chooses the lower cost approach to tune the system in order to re-guarantee the performance SLA when a performance violation occurs.
	[39]	It proposes a rapid KPI trend prediction framework TPC (Trend Prediction based on Clustering) to guide the operation and maintenance team to adjust cloud resources reasonably and timely.
Resource scheduling	[40]	It proposes SmartSLA， a cost-aware resource management system， to intelligently manage the resources in a shared cloud database system.
	[41]	It advocates the cooperation between VM host- and guest-layer schedulers for optimizing resource management and application performance.
	[42]	It presents a Cloud VM scheduling algorithm that considers already running VM resource usage over time by analyzing past VM utilization levels to schedule VMs by optimizing performance.
	[43]	It designs iBTune to automatically orchestrate the buffer pool tuning for the entire database instances.
Resource control	[44]	It focuses on controlling CPU (central processing unit) usage and memory consumption of a virtual database machine in a data center under a time-varying heavy workload.
	[45]	It proposes a Greedy Particle Swarm Optimization (GPSO) search algorithm in the Virtual Design Advisor (VDA) to estimate the cost of database workloads running in virtual machines with varying resource allocation accurately and quickly.
	[46]	It designs ResTune to automatically optimize resource utilization without violating SLA constraints on the throughput and latency requirements.
Resource scaling	[47]	It presents CloudScale, a system that automates fine-grained elastic resource scaling for multi-tenant cloud computing infrastructures.
	[48]	It proposes a model for resource allocation of a data center that includes clusters of hosts. When the utilization of active hosts reaches a predefined threshold value, a new host is added to prevent response time violation, and when host utilization is reduced to a certain threshold, one of the hosts can be deactivated.
	[49]	It proposes a Hybrid Auto-Scaler (HAS) to adjust the required resources automatically to the application in demand. HAS deploys the anticipated resources by computing the required capacity. Further, it scales out the resources in accordance as the provisioned resources are insufficient to deal with the current needs.

level based on the indicators identified from the systematic literature review. The follow-up questions part collects new ideas from the experts.

After interviewing the experts, we collect the information they provide. In the interviews, we ask the interviewees' opinions on the number of maturity levels. Seven participants answer four levels, and only one gives five levels (the sixth participant). However, it was difficult for the sixth participant to distinguish nuances and describe refined levels. So, we finally exclude the opinion of this participant and determine that the maturity model has four levels. After the experts provide the number of maturity levels and their descriptions of each level, we first identify the similarities of each level based on the experts' descriptions and then develop a definition for each level. Table 4 shows the relevant quotations extracted from interviews to support the concepts for maturity levels.

Table 4 The relevant quotations extracted from interviews

Level 1: AI-ready level

At this level, the developers consider applying AI technologies to improve database performance. The participants most frequently mentioned the terms "consider applying AI technologies" and "optimize" for the first maturity level. "The database developers only establish a preliminary plan for the AI technologies implementation, but it does not apply AI technologies" (PM 1 and PM 2). According to Consultant 5, "at this level, the database has not yet developed intelligence resource management method. Resource management work basically depends on DMA, which is complex and difficult. However, the developers have made some initial plans to apply AI technologies".

Level 2: AI-usage level

At the second maturity level, "according to the preliminary plan in the first maturity level, the developers can apply AI technologies to optimize resource management" (PM 2). Furthermore, "the database only considers the optimization of resource management in a separate part, but it does not consider multiple parts of the database in an integrated manner" (Consultant 4).

Level 3: Semi-automatic level

At this level, the database realizes several intelligent functions in multiple parts of the database with human help. For example, when a database serves multiple tenants on demand, the database can apply AI technologies to complete intelligent resource scheduling. But it needs human assistance to decide whether to operate based on the given data. "At the third level, the database is committed to integrating multiple parts of the database to realize intelligent resource management, further to implement the AI-empowered"(PM 3).

Level 4: Automatic level

This level is the highest level of the maturity model. For this level, 75% of participants considered that CNDBs realize self-optimizing resource management performance in multiple parts. Also, as outlined by both PM 1 and PM 2, "the database embraces various smart functions, such as intelligent resource allocation among tenants, smart resource scheduling between regions, and automatic memory sharing". According to Consultant 3, "the cloud resources are fully utilized at the fourth maturity level".

Definition

In the previous sections, we identify two dimensions with ten indicators and four levels of the proposed maturity model. We will introduce them specifically in the following.

Dimensions and indicators

Dimensions and indicators are the components of the maturity model, and they are identified in the SLR process. Our

Table 5 dimensions and indicators of the proposed maturity model

Dimension	Indicator
Cloud-native database	Multi-tenant, Compute and storage disaggregation, Cross-Az/Region, Near-data processing, Logs as the database, Distributed and shared memory
Artificial intelligence	Smart resource prediction, Smart resource scheduling, Smart resource control, Smart resource scaling

maturity model has two dimensions, including "cloud-native database" and "artificial intelligence", with ten indicators that explain the AI-empowered CNDBs from the resource perspective. Table 5 shows the dimensions and indicators.

Cloud-native database dimension

The CNDB dimension employs the features of CNDBs as indicators, including six indicators in total: multi-tenant, compute and storage disaggregation, cross-Az/Region, near-data processing, logs as the database, and distributed and shared memory. The description of these indicators is shown in Table 6.

Artificial intelligence dimension

The AI dimension depicts the capabilities of AI technologies applied to resource management in CNDBs. It has four indicators:

- Smart resource prediction focuses on applying AI technologies to predict the usage trends of resources (CPU, memory, I/O, and network) [37].
- Smart resource scheduling refers to analyzing the existing resource usage over time and past resource levels to realize automatic and efficient resource scheduling [42].
- Smart resource control emphasizes controlling the resource (CPU, memory) usage and consumption of database servers because resources should not be over-utilized or under-utilized [44].
- Smart resource scaling concerns deciding when and how to expand resources according to the user's resource utilization. In other words, it is to realize the automatic scaling function of resource containers [52].

Interleaving of CNDBs dimension and AI dimension

The above CNDBs and AI indicators depend on each other. In particular, the capabilities of smart resource prediction and smart resource scheduling in the AI dimension can vary significantly according to specific features, such as multi-tenant, distributed and shared memory.

Table 6 The description of the indicators of cloud-native dimension

Indicator	Description
Multi-tenant	Databases provide services for multiple tenants. And tenants share the resources of database servers. The multi-tenant enables cost reduction for the cloud service provider, which can pass on as savings to the tenants [50].
Compute and storage disaggregation	Computing and storage are decoupled from each other. The computing and storage resources are dynamically combined through the network. And the computing process is data-driven to realize on-demand driving better [31]. In this way, independent compute nodes can be flexibly scaled up, and storage nodes can be flexibly scaled-down, improving the cost performance of databases.
Cross-Az/Region	A logical database is divided into multiple shards, each of which is assigned to a node. These nodes can be placed and replicated in different data centers and regions [1].
Near-data processing	Using the processing capacity in the memory to process the data (such as the screening operation of the database) and only transmit the data processing results to the host. The method saves a lot of system resources and reduces time delay and energy consumption. It can be implemented by operator push-down [4].
Logs as the database	The CNDB only writes log files and plays back data at the storage layer to avoid I/O amplification. The method reduces network pressure on the cloud infrastructure [30].
Distributed and shared memory	Memory resources in different nodes are connected through a high-speed network. Databases can share data pages in the remote memory pool, similar to the shared storage pool in the shared storage architecture [51].

And databases should consistently establish the capabilities of smart resource control and smart resource scaling in all features of CNDBs. We show the detailed description in Table 7.

According to the analysis and the results on dimensions, indicators, and their interleaving, we propose the indicator matrix for AI-empowered CNDBs, as shown in Fig. 2. The matrix is an abstract representation of the interleaving of the two dimensions. It provides a conceptual view of intelligent transformation when applying AI technologies to CNDBs in their developing processes. The matrix can help analyze the intelligent development of CNDB indicators. On the one hand, the coarse-grained AI indicators, namely the long bar across all CNDB indicators, indicates that its capability has the same impact on all CNDB indicators. For example, the capabilities of smart resource control and resource scaling should be holistically planned and applied thoroughly across the CNDB. On the other hand, the fine-grained AI indicators, namely the separated bar that across CNDB indicators independently, its capability changes significantly for different CNDB indicators. For example, the database can serve multi-tenant users more efficiently and intelligently if developers apply AI technologies to realize automatic resource prediction and scheduling. The

Table 7 Interleaving of CNDB and AI dimensions in our study

AI dimension	CNDB dimension					
	Multi-tenant	**Compute and storage disaggregation**	**Cross-Az/Region**	**Near-data processing**	**Logs as the database**	**Distributed and shared memory**
Smart resource prediction	Databases provide corresponding resources according to the needs of each tenant, and AI technologies can be applied to predict the resources that each tenant requires.	Databases predict the demand of computing and storage resources under different application backgrounds.	The resource management system can predict the demand of resources in different regions.	Databases predict the computing resources required for the operator push-down in the storage layer.	Databases apply AI technologies to predict the storage resources required to store logs.	Databases predict the shared memory of each machine according to the historical resource usage.
Smart resource scheduling	Databases can apply AI technologies to dynamically schedule resources according to the resource demand of tenants in the use process.	Databases can dynamically adjust the decoupling degree of storage and computing resources according to different application backgrounds.	Databases can dynamically schedule resources based on resource usage in different regions.	Databases can dynamically adjust the computing resources in the storage layer due to the operator push-down	Databases can schedule storage resources in real-time based on the resources required to store logs.	Databases can schedule the memory resources in real-time based on the results of resource prediction to maintain the shared memory pool.
Smart resource control	Databases have a complete resource control system to control the resource usage and consumption in all parts of the database. It can ensure that resources will not be over-utilized or under-utilized.					
Smart resource scaling	Database systems can decide when to add new resources and automatically bring resources into the scope of resource management. It realizes the automatic scaling function of resource containers.					

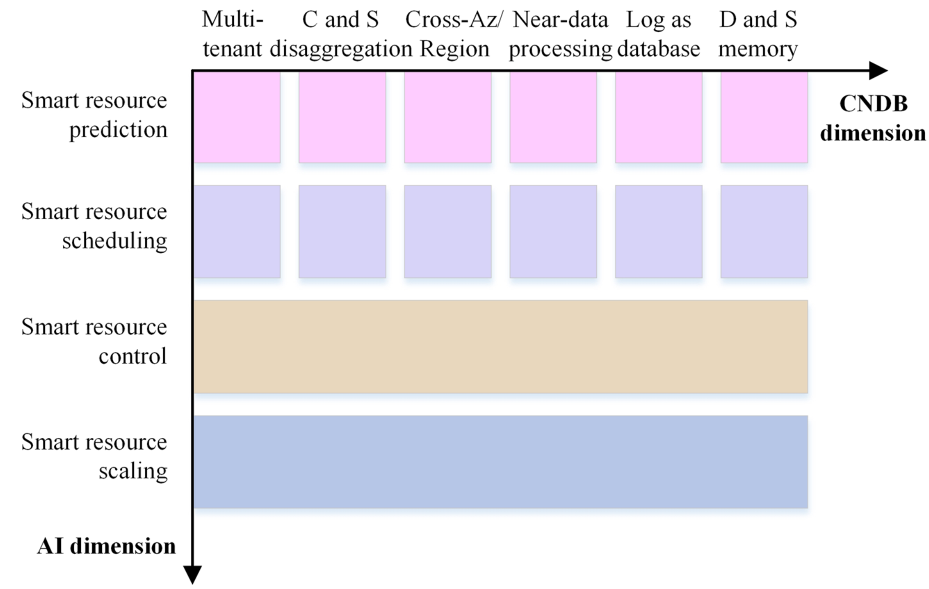

Fig. 2 Indicator matrix for AI-empowered CNDBs

Table 8 The maturity level identified from our study and their concepts

Level	Name of level	Concept of level
Level 1	AI-ready level	Database developers have considered applying AI technologies to optimize the resource management of CNDBs and made a preliminary plan.
Level 2	AI-usage level	Database developers have applied AI technologies to optimize resource management for the separate part of database.
Level 3	Semi-automatic level	Database developers have realized intelligent resource management considering the integration of multiple parts of database, but it still needs human assistance in making decisions.
Level 4	Automatic level	The database can automatically optimize resource management according to the actual needs of the workload and realize the automatic decision-making and optimization of the database.

indicator matrix is useful for understanding the correlation between AI and CNDB indicators.

Maturity levels

Another important component of the maturity model is the maturity level. Our maturity model has four levels covering AI-ready, AI-usage, Semi-automatic, and automatic level. They describe the planning objectives and implement path of AI-empowered CNDBs.

In the section "Expert interviews", we obtain expert opinions by conducting semi-structured interviews to describe the characteristics of maturity levels. As shown in Table 8, our study summarizes the characteristics of AI-empowered CNDBs at different maturity levels and determines the concept of these levels.

Maturity model

We represent dimensions and indicators by the indicator matrix and identify four maturity levels. We integrate the indicator matrix and maturity levels to construct the maturity model. Moreover, we provide supplementary descriptions and complementary examples to understand the maturity model.

Figure 3 shows the maturity model structure constructed by the indicator matrix and maturity level. We accumulate the indicator matrix to each level, namely the superposition of the indicator matrix on the four maturity levels, forming the structure shown in Fig. 3. It intuitively shows a roadmap for achieving AI-empowered CNDBs, from AI-ready to automatic level, and helps database developers to improve the maturity level of CNDBs. However, Fig. 3 lacks semantics and fails to give practical guidance, which poses a challenge to the practical application of maturity models.

To understand how to achieve a higher maturity level of AI-empowered CNDBs, we provide Table 9 as a supplement to Fig. 3. The supplement provides detailed exemplifications about the characteristics of

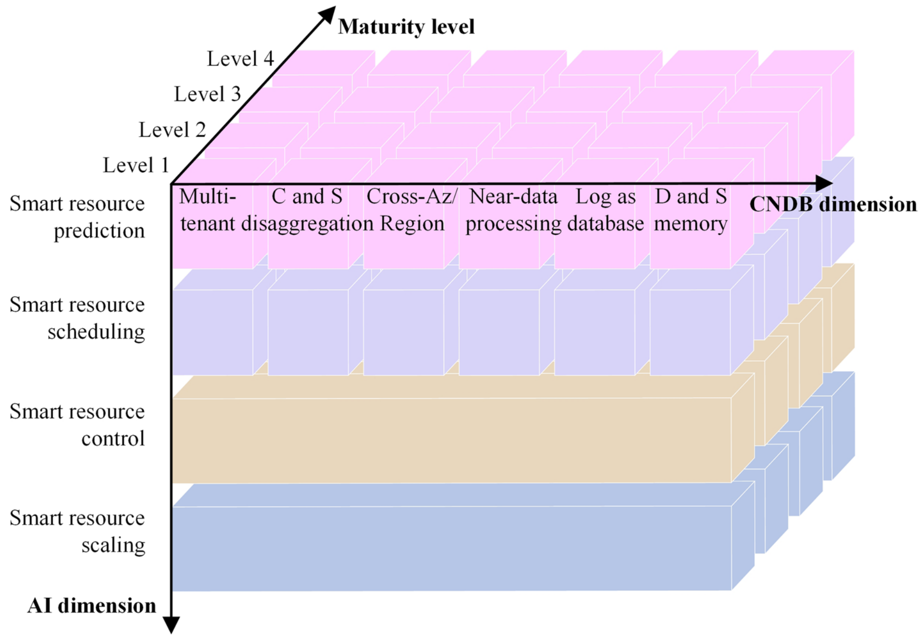

Fig. 3 Integrated maturity levels to indicator matrix

the integrated maturity levels and AI indicators. For example, we could draw the following guidelines from Table 9.

- To achieve a higher level of smart resource prediction, databases should apply AI technologies to predict the usage trend of resources integrating by multiple parts of databases.
- To achieve a higher level of smart resource scheduling, databases should automatically analyze the historical resource usage level and capture the current resource usage in time to optimize resource scheduling.
- To achieve a higher maturity level of smart resource control, databases should monitor the usage of various resources in each part of the database to realize automatic resource control.
- To achieve a higher maturity level of smart resource scaling, databases should perform self-decision for the time and method of expanding resources based on the current resource usage.

To adequately describe the maturity model, we should combine Table 9 and Fig. 3 to give each CNDB indicator a complementary table similar to Table 9. The complement includes the activities related to the characteristics of the integrated three maturity levels with AI indicators and a CNDB indicator since the highest level is introduced in Table 6. But it would generate a large amount of information. For abbreviation, we take the multi-tenant indicator as an example to provide the corresponding complement, as shown in Table 10. This complement helps developers make better use of the maturity model to improve the maturity level of the AI indicators when CNDBs focus on the multi-tenant feature.

Assessment

As mentioned above, our study develops a maturity model of AI-empowered CNDBs. To use the maturity model, we propose an assessment tool based on the model to help developers identify the maturity level of CNDBs. And we provide an assessment case to prove the maturity model.

Assessment tool

We transform the maturity model from a matrix of dimensions to a tool that enables developers to assess their AI-empowered CNDBs. The tool helps developers identify activities and opportunities on the path to achieving their AI-empowered goals. We introduce the tool, explain how to use it, and give an example and some suggestions in the following.

The tool relies on the form of a table to help assess. The table covers all the interleaving of CNDB and AI indicators. It has six child tables, each representing the interleaving of a CNDB indicator and all AI indicators. In each child table, the characteristics of AI indicators at different

Table 9 Characteristics of the integrated maturity levels and AI indicators

AI dimension	Level 1	Level 2	Level 3	Level 4
Smart resource prediction	Make plans to apply AI technologies in resource prediction	Apply AI technologies to predict resource consumption for the separate part of the database	Predict resource consumption semi-automatically considering multiple parts of the database,	Predict resource consumption automatically considering all parts of the database
Smart resource scheduling	Make plans to apply AI technologies in resource scheduling	Apply AI technologies to schedule resource for the separate part of the database	Schedule resources semi- automatically between multiple parts of the database	Schedule resources automatically between all parts of the database
Smart resource control	Make plans to apply AI technologies in resource control	Apply AI technologies to realize resource control for the separate part of the database	Realize semi-automatic resource control considering multiple parts of the database	Realize automatic resource control considering all parts of the database
Smart resource scaling	Make plans to apply AI technologies in resource scaling	Apply AI technologies to scale resources for the separate part of the database	Scale resources semi-automatically considering multiple parts of the database	Scale resources automatically considering all parts of the database

Table 10 The characteristics of the integrated other three maturity levels and the multi-tenant indicator

AI dimension	CNDB dimension: multi-tenant		
	Level 1	**Level 2**	**Level 3**
Smart resource prediction	Database developers plan to apply AI technologies to predict the resource requirements of each tenant.	Database developers have applied AI technologies to predict the resource requirements of each tenant in the separate part of the database.	Database developers have realized smart resource prediction for tenants considering the integration of multiple parts of the database, but it still needs human assistance in making decisions.
Smart resource scheduling	Database developers plan to apply AI technologies to dynamically schedule resources according to the resource requirements of tenants in the application process	Database developers have applied AI technologies to dynamically schedule resources according to the resource requirements of tenants in the separate part of the database.	Database developers have realized smart resource scheduling between tenants considering the integration of multiple parts of the database, but it still needs human assistance in making decisions.
Smart resource control	Database developers plan to apply AI technologies to control the resource usage and consumption of each tenant	Database developers have applied AI technologies to control the resource usage and consumption of each tenant in the separate part of the database	Database developers have realized smart resource control for tenants considering the integration of multiple parts of the database, but it still needs human assistance in making decisions.
Smart resource scaling	Database developers plan to apply AI technologies to help decide when and how to add new resources, and how to expand resources for tenants	Database developers have applied AI technologies to help decide when and how to add new resources for tenants in separate part of the database	Database developers have realized smart resource scaling for tenants considering the integration of multiple parts of the database, but it still needs human assistance in making decisions.

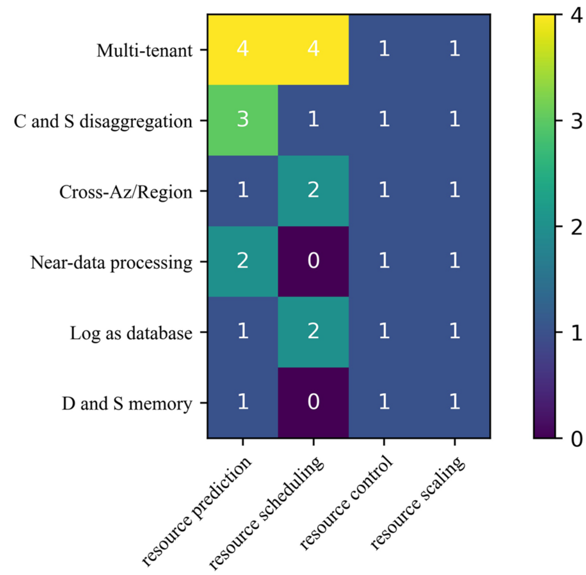

Fig. 4 Maturity level of the interleaving of two dimensions (an example)

maturity levels are restated as yes/no questions. In other words, CNDBs are performing an activity, or it is not. This yes/no format eliminates the ambiguity in assessing the level of compliance for a specific activity. The tool's core data is presented in Appendix 1.

To analyze the maturity level of CNDBs, we give a simplified example of the assessment strategies. First, we assume that the indicators are not weighted during the assessment. Developers can consider the selected AI technologies capabilities of the CNDBs to achieve this level only if the CNDB developer responds "yes" to all questions for the specific integration of CNDB and AI indicators at that maturity level. Second, when the actual situation of the CNDB meets the required characteristics of a certain level (e.g., level 1) but does not respond "yes" to all questions for a higher level (e.g., level 2), the CNDB is then determined as the lower level (i.e., level 1). Third, the CNDB can apply for a higher-level assessment only if it meets the requirements of the lower level. As shown in Fig. 4, we give an example of the maturity levels of the indicators.

We analyze an example of the assessment results shown in Fig. 4 and illustrate the suggestions based on the results. In Fig. 4, the CNDB realizes automatic resource prediction and scheduling for the multi-tenant feature by applying AI technologies, while it only performs simple resource prediction for the distributed and shared memory feature. At this point, the database developer should put more effort into smart resource prediction for the distributed and shared memory feature. Significantly, when implementing AI technologies, developers should improve the capabilities of CNDBs step by step rather than blindly pursue the highest goal. For example, from AI-ready level to AI-usage level, but not to semi-automatic and automatic level

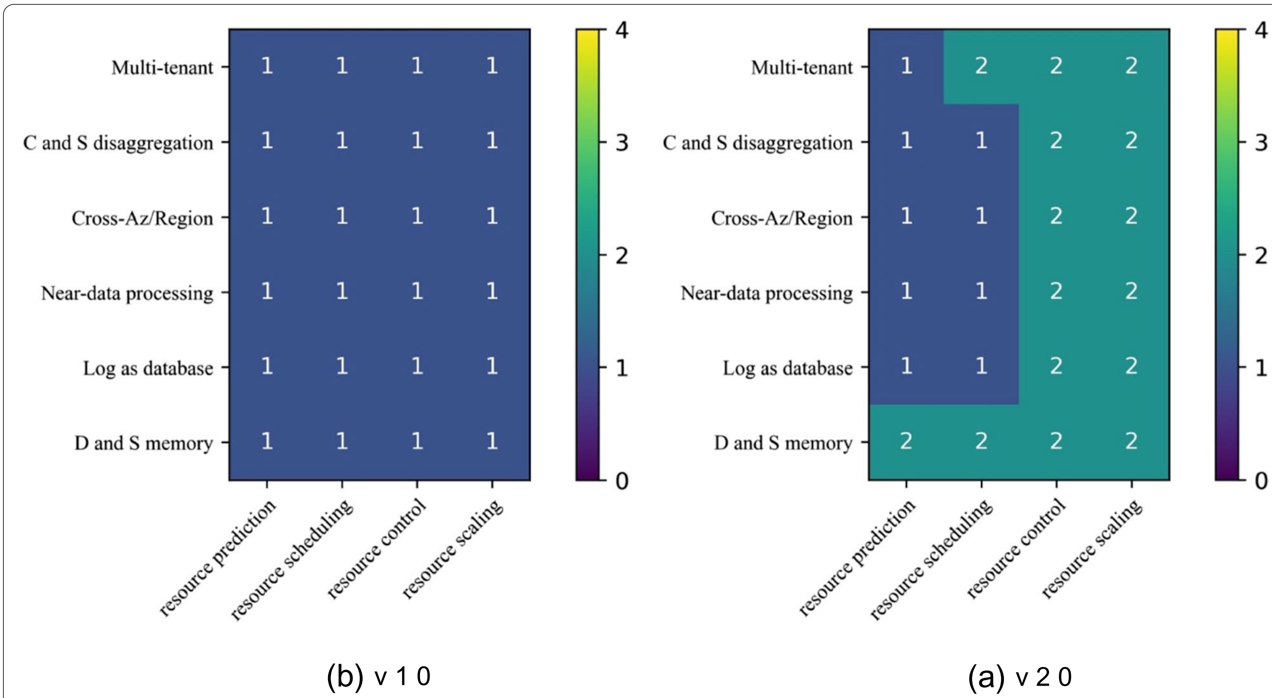

Fig. 5 The assessment results of POLARDB v1.0 and v2.0

directly. This tool helps developers assess the current level of CNDBs and recommend the next level as a target.

Assessment case

Now a large number of CNDBs have emerged, but they are still in the preliminary stage. We take POLARDB as an example and give the assessment results of its two versions.

The assessment relies on the internal materials for each established indicator of CNDBs. To perform the assessment, developers use the assessment tool based on the materials and make their judgments (i.e., giving yes/no answers) to each question from the lowest level (see Appendix 1). However, we cannot obtain these internal materials since they involve trade secrets. In the end, we rely on the published papers [31, 51] and public information to perform assessments. The assessment results obtained in this way are not objective, while the overall development trend of CNDBs reflected in the results is objective. We can verify the maturity model through the overall development trend of CNDBs. We use the maturity model to analyze the maturity levels of POLARDB's two versions. Figure 5 presents the assessment results, and we analyze the development trend through the results.

POLARDB v1.0 implements the basic functions without intelligent optimization. We argue that the maturity levels of the indicators are level 1 (i.e., AI-ready level), as shown in Fig. 5(a). POLARDB v2.0 optimizes the database in many aspects by applying AI technologies, making the database achieve preliminary intelligence. The maturity levels of several indicators increase by a level compared to v1.0, as shown in Fig. 5(b). The improvements are shown as follows.

- First, the maturity level of the interleaving of multi-tenant and smart resource scheduling is raised to level 2. POLARDB v2.0 can intelligently schedule resources in the resource pool by applying AI technologies to meet the needs of multiple tenants.
- Second, for the distributed and shared memory indicator, the capabilities of POLARDB on smart resource prediction and smart resource scheduling meet level 2. POLARDB v2.0 can predict the size of memory resource blocks participating in memory pooling and schedule the resources in the memory pool by applying AI technologies.
- Third, to the entire database, all CNDB indicators realize level 2 of smart resource control and smart resource scaling. POLARDB v2.0 can intelligently allocate different computing nodes for OLAP and OLTP to control the computing resources occupied by OLAP. And it can automatically bring resources into the scope of resource management to scale out resources.

The assessment results show that the maturity levels of POLARDB are low, but the overall trend shows a gradually mature development direction. In other words, although the assessment results are subjective, the overall development direction of the versions is consistent with the proposed maturity model. The results prove that our maturity model is effective.

Conclusions and future work

Our study proposes a maturity model of AI-empowered CNDBs that contains CNDB and AI dimensions with ten indicators and four maturity levels, based on the SLR and expert interviews. The maturity model contributes to understanding and assessing the capabilities of AI technologies applied in CNDBs. The findings of our study help database developers select appropriate targets and formulate improvement measures to realize AI-empowered CNDBs. The analysis of the assessment case shows that although the maturity levels of CNDB are low, its development direction conforms to the maturity model.

Our work can be extended in multiple directions. First, we follow the guidelines of Wolfswinkel et al. [25] to search and select articles, and our search was limited to the eight specific online databases with our keywords. There may still be relevant studies that have not been included in our SLR. Although these are the main sources on assessing AI-empowered CNDBs addressing confidence that our SLR has identified the key literature, some researchers may still question the comprehensiveness of the results. We welcome researchers and practitioners to discuss more key literature to supplement the results of this study. Second, the indicators of the maturity model are identified according to the current development of CNDBs. With the continuous development of CNDB, the features of the CNDBs will change. And the indicators of maturity model will also change accordingly. In the future, researchers may need redefine the indicators with the same method. Third, the assessment tool in our study requires developers to make judgments on the corresponding questions based on relevant materials, resulting in a heavy assessment workload. It is difficult to perform the large-scale operation in actual database assessment applications. Developing an intelligent assessment tool by applying deep learning technologies (e.g., NLP) is an item of future work.

Appendix

Please insert Table 11 here.

Table 11 The assessment tool developed in our study for assessing AI technologies capabilities for a CNDB

For assessing the AI capabilities of the multi-tenant				
AI dimension	**Maturity levels**			
	Level 1	Level 2	Level 3	Level 4
Smart resource prediction	(1) Have database developers considered applying AI technologies in resource prediction to serve multiple tenants better? (2) Have the database developers made a preliminary plan for smart resource prediction to serve multiple tenants better?	(1) Have database developers applied AI technologies in resource prediction to serve multiple tenants better? (2) Have the database developers realized smart resource prediction in the separate part of the database to serve multiple tenants better?	(1) Have database developers realized smart resource prediction considering the integration of multiple parts of the database to serve multiple tenants better? (2) Do the database developers need human assistance in making decisions?	Could the database automatically execute resource prediction according to the actual needs of the workload and realize the automatic decision-making and optimization to serve multiple tenants better?
Smart resource scheduling	(1) Have database developers considered applying AI technologies in resource scheduling to serve multiple tenants better? (2) Have database developers made a preliminary plan for smart resource scheduling to serve multiple tenants better?	(1) Have database developers applied AI technologies in resource scheduling to serve multiple tenants better? (2) Have database developers realized smart resource scheduling in the separate part of the database to serve multiple tenants better?	(1) Have database developers realized smart resource scheduling considering the integration of multiple parts of the database to serve multiple tenants better? (2) Do database developers need human assistance in making decisions?	Could the database automatically execute resource scheduling according to the actual needs of the workload and realize the automatic decision-making and optimization to serve multiple tenants better?
Smart resource control	(1) Have database developers considered applying AI technologies in resource control to serve multiple tenants better? (2) Have database developers made a preliminary plan for smart resource control to serve multiple tenants better?	(1) Have database developers applied AI technologies in resource control to serve multiple tenants better? (2) Have database developers realized smart resource control in the separate part of the database to serve multiple tenants better?	(1) Have database developers realized smart resource control considering the integration of multiple parts of the database to serve multiple tenants better? (2) Do database developers need human assistance in making decisions?	Could the database automatically execute resource control according to the actual needs of the workload and realize the automatic decision-making and optimization to serve multiple tenants better?
Smart resource scaling	(1) Have database developers considered applying AI technologies in resource scaling to serve multiple tenants better? (2) Have database developers made a preliminary plan for smart resource scaling to serve multiple tenants better?	(1) Have database developers applied AI technologies in resource scaling to serve multiple tenants better? (2) Have database developers realized smart resource scaling in the separate part of the database to serve multiple tenants better?	(1) Have database developers realized smart resource scaling considering the integration of multiple parts of the database to serve multiple tenants better? (2) Do database developers need human assistance in making decisions?	Could the database automatically execute resource scaling according to the actual needs of the workload and realize the automatic decision-making and optimization to serve multiple tenants better?
For assessing the AI capabilities of the compute and storage disaggregation				
AI dimension	**Maturity levels**			
	Level 1	Level 2	Level 3	Level 4
Smart resource prediction	(1) Have database developers considered applying AI technologies in resource prediction for the compute and storage disaggregation feature? (2) Have database developers made a preliminary plan for smart resource prediction to realize the compute and storage disaggregation feature?	(1) Have database developers applied AI technologies in resource prediction for the compute and storage disaggregation feature? (2) Have database developers realized smart resource prediction in the separate part of the database for the compute and storage disaggregation feature?	(1) Have database developers realized smart resource prediction considering the integration of multiple parts of the database for the compute and storage disaggregation feature? (2) Do database developers need human assistance in making decisions?	Could the database automatically execute resource prediction according to the actual needs of the workload and realize the automatic decision-making and optimization for the compute and storage disaggregation feature?

Table 11 (continued)

Smart resource scheduling	(1) Have database developers considered applying AI technologies in resource scheduling for the compute and storage disaggregation feature? (2) Have database developers made a preliminary plan for smart resource scheduling to realize the compute and storage disaggregation feature?	(1) Have database developers applied AI technologies in resource scheduling for the compute and storage disaggregation feature? (2) Have database developers realized smart resource scheduling in the separate part of the database for the compute and storage disaggregation feature?	(1) Have database developers realized smart resource scheduling considering the integration of multiple parts of the database for the compute and storage disaggregation feature? (2) Do database developers need human assistance in making decisions?	Could the database automatically execute resource scheduling according to the actual needs of the workload, and realize the automatic decision-making and optimization for the compute and storage disaggregation feature?
Smart resource control	(1) Have database developers considered applying AI technologies in resource control for the compute and storage disaggregation feature? (2) Have database developers made a preliminary plan for smart resource control to realize the compute and storage disaggregation feature?	(1) Have database developers applied AI technologies in resource control for the compute and storage disaggregation feature? (2) Have database developers realized smart resource control in the separate part of the database for the compute and storage disaggregation feature?	(1) Have database developers realized smart resource control considering the integration of multiple parts of the database for the compute and storage disaggregation feature? (2) Do database developers need human assistance in making decisions?	Could the database automatically execute resource control according to the actual needs of the workload and realize the automatic decision-making and optimization for the compute and storage disaggregation feature?
Smart resource scaling	(1) Have database developers considered applying AI technologies in resource scaling for the compute and storage disaggregation feature? (2) Have database developers made a preliminary plan for smart resource scaling to realize the compute and storage disaggregation feature?	(1) Have database developers applied AI technologies in resource scaling for the compute and storage disaggregation feature? (2) Have database developers realized smart resource scaling in the separate part of the database for the compute and storage disaggregation feature?	(1) Have database developers realized smart resource scaling considering the integration of multiple parts of the database for the compute and storage disaggregation feature? (2) Do database developers need human assistance in making decisions?	Could the database automatically execute resource scaling according to the actual needs of the workload and realize the automatic decision-making and optimization for the compute and storage disaggregation feature?
For assessing the AI capabilities of the cross-Az/Region				
AI dimension	**Maturity levels**			
	Level 1	Level 2	Level 3	Level 4
Smart resource prediction	(1) Have database developers considered applying AI technologies in resource prediction for the cross-Az/Region feature? (2) Have database developers made a preliminary plan for smart resource prediction to realize the cross-Az/Region feature?	(1) Have database developers applied AI technologies in resource prediction for the cross-Az/Region feature? (2) Have database developers realized smart resource prediction in the separate part of the database for the cross-Az/Region feature?	(1) Have database developers realized smart resource prediction considering the integration of multiple parts of the database for the cross-Az/Region feature? (2) Do database developers need human assistance in making decisions?	Could the database automatically execute resource prediction according to the actual needs of the workload and realize the automatic decision-making and optimization for the cross-Az/Region feature?
Smart resource scheduling	(1) Have database developers considered applying AI technologies in resource scheduling for the cross-Az/Region feature? (2) Have database developers made a preliminary plan for smart resource scheduling to realize the cross-Az/Region feature?	(1) Have database developers applied AI technologies in resource scheduling for the cross-Az/Region feature? (2) Have database developers realized smart resource scheduling in the separate part of the database for the cross-Az/Region feature?	(1) Have database developers realized smart resource scheduling considering the integration of multiple parts of the database for the cross-Az/Region feature? (2) Do database developers need human assistance in making decisions?	Could the database automatically execute resource scheduling according to the actual needs of the workload and realize the automatic decision-making and optimization for the cross-Az/Region feature?

Table 11 (continued)

Smart resource control	(1) Have database developers considered applying AI technologies in resource control for the cross-Az/Region feature? (2) Have database developers made a preliminary plan for smart resource control to realize the cross-Az/Region feature?	(1) Have database developers applied AI technologies in resource control for the cross-Az/Region feature? (2) Have database developers realized smart resource control in the separate part of the database for the cross-Az/Region feature?	(1) Have database developers realized smart resource control considering the integration of multiple parts of the database for the cross-Az/Region feature? (2) Do database developers need human assistance in making decisions?	Could the database automatically execute resource control according to the actual needs of the workload and realize the automatic decision-making and optimization for the cross-Az/Region feature?
Smart resource scaling	(1) Have database developers considered applying AI technologies in resource scaling for the compute and storage disaggregation feature? (2) Have database developers made a preliminary plan for smart resource scaling to realize the compute and storage disaggregation feature?	(1) Have database developers applied AI technologies in resource scaling for the cross-Az/Region feature? (2) Have database developers realized smart resource scaling in the separate part of the database for the cross-Az/Region feature?	(1) Have database developers realized smart resource scaling considering the integration of multiple parts of the database for the cross-Az/Region feature? (2) Do database developers need human assistance in making decisions?	Could the database automatically execute resource scaling according to the actual needs of the workload and realize the automatic decision-making and optimization for the cross-Az/Region feature?
For assessing the AI capabilities of the near-data processing				
AI dimension	**Maturity levels**			
	Level 1	Level 2	Level 3	Level 4
Smart resource prediction	(1) Have database developers considered applying AI technologies in resource prediction for the near-data processing feature? (2) Have database developers made a preliminary plan for smart resource prediction to realize the near-data processing feature?	(1) Have database developers applied AI technologies in resource prediction for the near-data processing feature? (2) Have database developers realized smart resource prediction in the separate part of the database for the near-data processing feature?	(1) Have database developers realized smart resource prediction considering the integration of multiple parts of the database for the near-data processing feature? (2) Do database developers need human assistance in making decisions?	Could the database automatically execute resource prediction according to the actual needs of the workload and realize the automatic decision-making and optimization for the near-data processing feature?
Smart resource scheduling	(1) Have database developers considered applying AI technologies in resource scheduling for the near-data processing feature? (2) Have database developers made a preliminary plan for smart resource scheduling to realize the near-data processing feature?	(1) Have database developers applied AI technologies in resource scheduling for the near-data processing feature? (2) Have database developers realized smart resource scheduling in the separate part of the database for the near-data processing feature?	(1) Have database developers realized smart resource scheduling considering the integration of multiple parts of the database for the near-data processing feature? (2) Do database developers need human assistance in making decisions?	Could the database automatically execute resource scheduling according to the actual needs of the workload and realize the automatic decision-making and optimization for the near-data processing feature?
Smart resource control	(1) Have database developers considered applying AI technologies in resource control for the near-data processing feature? (2) Have database developers made a preliminary plan for smart resource control to realize the near-data processing feature?	(1) Have database developers applied AI technologies in resource control for the near-data processing feature? (2) Have database developers realized smart resource control in the separate part of the database for the near-data processing feature?	(1) Have database developers realized smart resource control considering the integration of multiple parts of the database for the near-data processing feature? (2) Do database developers need human assistance in making decisions?	Could the database automatically execute resource control according to the actual needs of the workload and realize the automatic decision-making and optimization for the near-data processing feature?

Table 11 (continued)

Smart resource scaling	(1) Have database developers considered applying AI technologies in resource scaling for the near-data processing feature? (2) Have database developers made a preliminary plan for smart resource scaling to realize the near-data processing feature?	(1) Have database developers applied AI technologies in resource scaling for the near-data processing feature? (2) Have database developers realized smart resource scaling in the separate part of the database for the near-data processing feature?	(1) Have database developers realized smart resource scaling considering the integration of multiple parts of the database for the near-data processing feature? (2) Do database developers need human assistance in making decisions?	Could the database automatically execute resource scaling according to the actual needs of the workload and realize the automatic decision-making and optimization for the near-data processing feature?
For assessing the AI capabilities of the logs as the database				
AI dimension	**Maturity levels**			
	Level 1	Level 2	Level 3	Level 4
Smart resource prediction	(1) Have database developers considered applying AI technologies in resource prediction for the logs as the database feature? (2) Have database developers made a preliminary plan for smart resource prediction to realize the logs as the database feature?	(1) Have database developers applied AI technologies in resource prediction for the logs as the database feature? (2) Have database developers realized smart resource prediction in the separate part of the database for the logs as the database feature?	(1) Have database developers realized smart resource prediction considering the integration of multiple parts of database for the logs as the database feature? (2) Do database developers need human assistance in making decisions?	Could the database automatically execute resource prediction according to the actual needs of the workload and realize the automatic decision-making and optimization for the logs as the database feature?
Smart resource scheduling	(1) Have database developers considered applying AI technologies in resource scheduling for the logs as the database feature? (2) Have database developers made a preliminary plan for smart resource scheduling to realize the logs as the database feature?	(1) Have database developers applied AI technologies in resource scheduling for the logs as the database feature? (2) Have database developers realized smart resource scheduling in the separate part of the database for the logs as the database feature?	(1) Have database developers realized smart resource scheduling considering the integration of multiple parts of the database for the logs as the database feature? (2) Do database developers need human assistance in making decisions?	Could the database automatically execute resource scheduling according to the actual needs of the workload and realize the automatic decision-making and optimization for the logs as the database feature?
Smart resource control	(1) Have database developers considered applying AI technologies in resource control for the logs as the database feature? (2) Have database developers made a preliminary plan for smart resource control to realize the logs as the database feature?	(1) Have database developers applied AI technologies in resource control for the logs as the database feature? (2) Have database developers realized smart resource control in the separate part of the database for the logs as the database feature?	(1) Have database developers realized smart resource control considering the integration of multiple parts of the database for the logs as the database feature? (2) Do database developers need human assistance in making decisions?	Could the database automatically execute resource control according to the actual needs of the workload and realize the automatic decision-making and optimization for the logs as the database feature?
Smart resource scaling	(1) Have database developers considered applying AI technologies in resource scaling for the logs as the database feature? (2) Have database developers made a preliminary plan for smart resource scaling to realize the logs as the database feature?	(1) Have database developers applied AI technologies in resource scaling for the logs as the database feature? (2) Have database developers realized smart resource scaling in the separate part of the database for the logs as the database feature?	(1) Have database developers realized smart resource scaling considering the integration of multiple parts of the database for the logs as the database feature? (2) Do database developers need human assistance in making decisions?	Could the database automatically execute resource scaling according to the actual needs of the workload, and realize the automatic decision-making and optimization for the logs as the database feature?

Table 11 (continued)

For assessing the AI capabilities of the distributed and shared memory				
AI dimension	**Maturity levels**			
	Level 1	Level 2	Level 3	Level 4
Smart resource prediction	(1) Have database developers considered applying AI technologies in resource prediction for the distributed and shared memory feature? (2) Have database developers made a preliminary plan for smart resource prediction to realize the distributed and shared memory feature?	(1) Have database developers applied AI technologies in resource prediction for the distributed and shared memory feature? (2) Have database developers realized smart resource prediction in the separate part of the database for the distributed and shared memory feature?	(1) Have database developers realized smart resource prediction considering the integration of multiple parts of the database for the distributed and shared memory feature? (2) Do database developers need human assistance in making decisions?	Could the database automatically execute resource prediction according to the actual needs of the workload and realize the automatic decision-making and optimization for the distributed and shared memory feature?
Smart resource scheduling	(1) Have database developers considered applying AI technologies in resource scheduling for the distributed and shared memory feature? (2) Have database developers made a preliminary plan for smart resource scheduling to realize the distributed and shared memory feature?	(1) Have database developers applied AI technologies in resource scheduling for the distributed and shared memory feature? (2) Have database developers realized smart resource scheduling in the separate part of the database for the distributed and shared memory feature?	(1) Have database developers realized smart resource scheduling considering the integration of multiple parts of the database for the distributed and shared memory feature? (2) Do database developers need human assistance in making decisions?	Could the database automatically execute resource scheduling according to the actual needs of the workload and realize the automatic decision-making and optimization for the distributed and shared memory feature?
Smart resource control	(1) Have database developers considered applying AI technologies in resource control for the distributed and shared memory feature? (2) Have database developers made a preliminary plan for smart resource control to realize the distributed and shared memory feature?	(1) Have database developers applied AI technologies in resource control for the distributed and shared memory feature? (2) Have database developers realized smart resource control in the separate part of the database for the distributed and shared memory feature?	(1) Have database developers realized smart resource control considering the integration of multiple parts of the database for the distributed and shared memory feature? (2) Do database developers need human assistance in making decisions?	Could the database automatically execute resource control according to the actual needs of the workload and realize the automatic decision-making and optimization for the distributed and shared memory feature?
Smart resource scaling	(1) Have database developers considered applying AI technologies in resource scaling for the distributed and shared memory feature? (2) Have database developers made a preliminary plan for smart resource scaling to realize the distributed and shared memory feature?	(1) Have database developers applied AI technologies in resource scaling for the distributed and shared memory feature? (2) Have database developers realized smart resource scaling in the separate part of the database for the distributed and shared memory feature?	(1) Have database developers realized smart resource scaling considering the integration of multiple parts of the database for the distributed and shared memory feature? (2) Do database developers need human assistance in making decisions?	Could the database automatically execute resource scaling according to the actual needs of the workload and realize the automatic decision-making and optimization for the distributed and shared memory feature?

Abbreviations
CNDBs: Cloud-native databases; AI: Artificial Intelligence; SLR: Systematic literature review; PM: Product Managers.

Acknowledgements
The authors would like to thank all anonymous reviewers for their invaluable comments.

Authors' contributions
Jie Song is the corresponding author and contributed to the "Introduction", "Definition" and "Assessment" sections. Xiaoyue Feng contributed to all of the manuscript sections. Tianzhe Jiao contributed to the "Related works" section and the "Analysis Methods and Results" section. Chaopeng Guo contributed to the "Assessment" section. All authors have read and approved the manuscript.

Funding
This paper is supported by the National Natural Science Foundation of China under Grant No. 62162050.

Availability of data and materials
The data used to support the findings of this study are available from the corresponding author upon request.

Declarations

Competing interests
The authors declare that they have no competing interests.

Received: 15 July 2022 Accepted: 21 August 2022
Published online: 07 September 2022

References
1. Li F (2019) Cloud-native database systems at Alibaba: opportunities and challenges. Proc VLDB Endowment 12(12):2263–2272
2. Ton That DH, Wagner J, Rasin A, Malik T (2019) PLI$^+$: efficient clustering of cloud databases. Distributed Parallel Databases 37(1):177–208
3. GB/T 33136–2016 Information technology service—Service capability maturity model of data center. Available at: https://openstd.samr.gov.cn/bzgk/gb/newGbInfo?hcno=F7A2242CAA62FD4466E8BAB0F92661D8. Accessed 20 June 2022
4. Zhan C, Su M, Wei C et al (2019) AnalyticDB: real-time OLAP database system at alibaba cloud. Proc VLDB Endowment 12(12):2059–2070
5. Antonopoulos P, Budovski A, Diaconu C et al (2019) Socrates: the new sql server in the cloud. In: Proceedings of the 2019 International Conference on Management of Data, pp 1743–1756
6. Pang Z, Lu Q, Chen S et al (2021) ArkDB: a key-value engine for scalable cloud storage services. In: Proceedings of the 2021 International Conference on Management of Data, pp 2570–2583
7. Chen Y, Zhao F, Lu Y, Chen X (2021) Dynamic task offloading for mobile edge computing with hybrid energy supply. Tsinghua Sci Technol. https://doi.org/10.26599/TST.2021.9010050
8. Xu J, Li D, Gu W, Chen Y (2022) UAV-assisted task offloading for IoT in smart buildings and environment via deep reinforcement learning. Build Environ 222:109218
9. Chen Y, Zhao F, Chen X, Wu Y (2021) Efficient multi-vehicle task offloading for mobile edge computing in 6G networks. IEEE Trans Veh Technol 71(5):4584–4596
10. Xu X, Jiang Q, Zhang P, Cao X et al (2022) Game theory for distributed IoV task offloading with fuzzy neural network in edge computing. IEEE Trans Fuzzy Syst
11. Chen Y, Gu W, Li K (2022) Dynamic task offloading for internet of things in mobile edge computing via deep reinforcement learning. Int J Commun Syst 2022:e5154
12. Huang J, Tong Z, Feng Z (2022) Geographical POI recommendation for internet of things: a federated learning approach using matrix factorization. Int J Commun Syst 2022:e5161
13. Xu X, Tian H, Zhang X, Qi L, He Q, Dou W (2022) DisCOV: distributed COVID-19 detection on X-ray images with edge-cloud collaboration. IEEE Trans Serv Comput 15(3):1206–1219
14. Sandhu AK (2021) Big data with cloud computing: discussions and challenges. Big Data Mining Analytics 5(1):32–40
15. Zhang Y, Zhang H, Cosmas J, Jawad N et al (2020) Internet of radio and light: 5G building network radio and edge architecture. Intell Converged Netw 1(1):37–57
16. Comuzzi M, Patel A (2016) How organisations leverage big data: a maturity model. Ind Manag Data Syst 116(8):1468–1492
17. Guoqiang GAI, Tingkun Y, Jun XIE, Chenning HUANG (2021) Database service ecology and system in China. Inform Commun Technol Policy 47(1):57–62
18. Spillner J, Bogado Y, Benítez W, López Pires F (2018) Co-transformation to cloud-native applications: development experiences and experimental evaluation. In: 8th International Conference on Cloud Computing and Services Science (CLOSER). SciTePress, Funchal, pp 19–21
19. Chen W, Liu C, Xing F, Peng G, Yang X (2021) Establishment of a maturity model to assess the development of industrial AI in smart manufacturing. J Enterp Inf Manag 35(3):701–728
20. Tarhan A, Turetken O, Reijers HA (2016) Business process maturity models: a systematic literature review. Inf Softw Technol 75:122–134
21. Sadiq RB, Safie N, Abd Rahman AH et al (2021) Artificial intelligence maturity model: a systematic literature review. PeerJ Comput Sci 7:e661
22. Felch V, Asdecker B (2020) Quo Vadis, business process maturity model? Learning from the past to envision the future. In: International conference on business process management. Springer, Cham, pp 368–383
23. Dutta A, Roy R, Seetharaman P (2022) An assimilation maturity model for IT governance and auditing. Inf Manag 59(1):103569
24. Qi L, Hu C, Zhang X, Khosravi MR, Sharma S, Pang S, Wang T (2020) Privacy-aware data fusion and prediction with spatial-temporal context for smart city industrial environment. IEEE Transact Industr Inform 17(6):4159–4167
25. Wolfswinkel JF, Furtmueller E, Wilderom CP (2013) Using grounded theory as a method for rigorously reviewing literature. Eur J Inf Syst 22(1):45–55
26. Kitchenham BA, Budgen D, Brereton P (2015) Evidence-based software engineering and systematic reviews. CRC Press, Boca Raton
27. Chen Y, Xing H, Ma Z, Chen X, Huang J (2022) Cost-efficient edge caching for NOMA-enabled IoT services. Chin Commun
28. Corbett JC, Dean J, Epstein M et al (2013) Spanner: Google's globally distributed database. ACM Transact Comput Syst (TOCS) 31(3):1–22
29. Huang D, Liu Q, Cui Q et al (2020) TiDB: a raft-based HTAP database. Proc VLDB Endowment 13(12):3072–3084
30. Verbitski A, Gupta A, Saha D et al (2017) Amazon aurora: design considerations for high throughput cloud-native relational databases. In: Proceedings of the 2017 ACM International Conference on Management of Data, pp 1041–1052
31. Cao W, Liu Y, Cheng Z et al (2020) POLARDB meets computational storage: efficiently support analytical workloads in cloud-native relational database. In: 18th USENIX conference on file and storage technologies (FAST 20), pp 29–41
32. Cao W, Liu Z, Wang P et al (2018) PolarFS: an ultra-low latency and failure resilient distributed file system for shared storage cloud database. Proc VLDB Endowment 11(12):1849–1862
33. Depoutovitch A, Chen C, Chen J et al (2020) Taurus database: how to be fast, available, and frugal in the cloud. In: Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data, pp 1463–1478
34. Huang J, Lv B, Wu Y, Chen Y, Shen X (2021) Dynamic admission control and resource allocation for mobile edge computing enabled small cell network. IEEE Trans Veh Technol 71(2):1964–1973
35. Chen Y, Liu Z, Zhang Y, Wu Y, Chen X, Zhao L (2020) Deep reinforcement learning-based dynamic resource management for mobile edge computing in industrial internet of things. IEEE Transact Industr Inform 17(7):4925–4934
36. Nath S, Wu J (2020) Deep reinforcement learning for dynamic computation offloading and resource allocation in cache-assisted mobile edge computing systems. Intell Converged Netw 1(2):181–198
37. Herodotou H, Lim H, Luo G et al (2011) Starfish: a self-tuning system for big data analytics. 5th Biennial Conf Innovative Data Syst Res (CIDR'11) 11(2011):261–272. Asilomar

38. Li L, Gruenwald L (2016) An SLA and operation cost aware performance re-tuning algorithm for cloud databases. 2016 IEEE 9th Int Conf Cloud Comput (CLOUD) 2016:966–969
39. Wang X, Li N, Zhang L, Zhang X, Zhao Q (2021) Rapid trend prediction for large-scale cloud database KPIs by clustering. 2021 IEEE/ACM Int Workshop Cloud Intell (CloudIntelligence) 2021:1–6
40. Xiong P, Chi Y, Zhu S, Moon HJ, Pu C, Hacgümüş H (2014) SmartSLA: cost-sensitive management of virtualized resources for CPU-bound database services. IEEE Transact Parallel Distribut Syst 26(5):1441–1451
41. Wang L, Xu J, Zhao M (2012) Application-aware cross-layer virtual machine resource management. In: Proceedings of the 9th international conference on Autonomic computing, pp 13–22
42. Sotiriadis S, Bessis N, Buyya R (2018) Self managed virtual machine scheduling in cloud systems. Inf Sci 433:381–400
43. Tan J, Zhang T, Li F et al (2019) iBTune: individualized buffer tuning for large-scale cloud databases. Proc VLDB Endowment 12(10):1221–1234
44. Armağan Ö, Gören-Sümer L (2014) Feedback control for multi-resource usage of virtualised database server. Comput Electr Eng 40(5):1683–1702
45. Omara FA, Khattab SM, Sahal R (2014) Optimum resource allocation of database in cloud computing. Egypt Inform J 15(1):1–12
46. Zhang X, Wu H, Chang Z et al (2021) ResTune: resource oriented tuning boosted by meta-learning for cloud databases. In: Proceedings of the 2021 International Conference on Management of Data, pp 2102–2114
47. Shen Z, Subbiah S, Gu X, Wilkes J (2011) CloudScale: elastic resource scaling for multi-tenant cloud systems. In: Proceedings of the 2nd ACM Symposium on Cloud Computing, pp 1–14
48. Salmanian Z, Izadkhah H, Isazadeh A (2022) Auto-scale resource provisioning in IaaS clouds. Comput J 65(2):297–309
49. JV BB, Dharma D (2018) HAS: hybrid auto-scaler for resource scaling in cloud environment. J Parallel Distribut Comput 120:1–15
50. Narasayya V, Menache I, Singh M et al (2015) Sharing buffer pool memory in multi-tenant relational database-as-a-service. Proc VLDB Endowment 8(7):726–737
51. Cao W, Zhang Y, Yang X et al (2021) Polardb serverless: a cloud native database for disaggregated data centers. In: Proceedings of the 2021 International Conference on Management of Data, pp 2477–2489
52. Das S, Li F, Narasayya VR, König AC (2016) Automated demand-driven resource scaling in relational database-as-a-service. In: Proceedings of the 2016 International Conference on Management of Data, pp 1923–1934

Publisher's Note