

BAB II TINJAUAN PUSTAKA

Penelitian-penelitian yang berkaitan pada penelitian. Dasar-dasar penelitian sebelumnya yang menjadi tinjauan pustaka pada penelitian ini dirangkum dalam tabel berikut ini :

Dalam artikel penelitian yang berjudul *Multi-Master Database Replication and e-Learning Theoretical And Practical Evaluation* Mattias Holmgren menjelaskan mengenai kemungkinan menggabungkan teknologi replikasi *database multi-master* yang digabungkan LEMP-stack di *server* untuk meningkatkan ketersediaan *e-learning* di daera terpencil di Tanzania (Kungliga & Högsolan, 2015) . Penelitian ini bertujuan memberikan untuk mengevaluasi replikasi *database multi-master* dan pendeksian konflik replikasi pada sistem *e-learning Moodle*. Dengan melakukan studi literatur tentang teknologi yang relevan diikuti dengan implementasi dan konfigurasi di lokasi maka nantinya peneliti akan mendapatkan hasil apakah layak atau tidak menggunakan replikasi *database multi-master*. Dan hasil setelah pengujian yang didapatkan didapatkan bahwa menggunakan replikasi *database multi-master* akan memakan banyak waktu.

Peneliti (Mulyanto & Ashari, 2016) melakukan penelitian mengenai Implementasi *Highly Available Website Dengan Distributed Replicated Block Device*. Dalam penelitian tersebut dikatakan bahwa *website, email* dan media *E-learning* merupakan sistem yang membutuhkan kehandalan dan tingkat *Availability* yang tinggi. Ketika ukuran data tumbuh menjadi besar karena semakin banyak orang dan sistem yang berinteraksi, mengakibatkan pengelolaan data menjadi tidak efektif. Hal tersebut berimbas terhadap kemampuan sistem untuk menyediakan layanan yang handal. Pada penelitian ini penulis melakukan rancang bangun *website* yang memiliki sifat *highly available*. *Website* yang *high availability* dibangun dengan memanfaatkan *protocol Heartbeat* dan *failover clustering*. *Heartbeat* yang digunakan adalah Linux yaitu *Corosync Cluster Engine* dan replikasi data dengan *Distributed Replicated Block* bertujuan untuk

menyediakan sistem dengan kemampuan *high availability* pada *website* sehingga diharapkan dapat meningkatkan *availability* dalam melayani pertumbuhan dan permintaan data. Penelitian ini juga menguji tingkat *availability* dari sistem dengan menggunakan *multi-node Distributed Replicated Block DeviceDevice (DRBD)*.

Dalam artikel jurnal yang berjudul *Development of High-Availability ATCA/PCIe Data Acquisition Instrumentation* menjelaskan mengenai *setup* sistem IPFN C&DA dengan menggunakan keuntungan performa dari fitur-fitur ATCA dan PCI (Correia et al., 2016). Control and Data Acquisition (C&DA) merupakan fusion diagnostik sistem yang membutuhkan HA yang tinggi. IPFN mengembangkan sebuah C&DA sistem yang diperuntukan sebagai pengontrol alat fusion dengan HA yang tinggi. Platform *hardware* yang digunakan adalah modul instrumentasi-digitalisasi Advanced Telecommunication Computing Architecture (ATCA) dan *switch blade* PCI Express (PCIe). *Switch blade* PCI Express (PCIe) dihubungkan melalui saluran *fabric interface* ACTA yang kemudian disambungkan dengan PCIe *host computer* eksternal. Pada pengaturan *hardware*, sistem arsitektur memanfaatkan karakteristik HA yang berulang pada ATCA seperti pada komponen *hardware*, topologi *backplane*, dan “Hot Swap” *Field Replaceable Unit* (FRU). Pada *software*, PCIe mendukung penyisipan dan pelepasan perangkat “Hot Plug”. Penggunaan fitur-fitur ACTA dan PCI seperti mekanisme *fail-over* berulang dapat digunakan dalam pengembangan sistem *fusion* C&DA.

Dalam jurnal yang berjudul Implementasi Sistem Interkoneksi Basis Data Terdistribusi Menggunakan *Socket API* membahas pemrograman *socket* menggunakan komponen *Application Programming Interface* (API) socket dalam pembuatan sebuah aplikasi (Yulianto, 2017). Penelitian ini merupakan penelitian studi kasus yang dilakukan di SMA / SMK dibawah naungan Muhamadyah. Konsep API memungkinkan struktur *database* yang berbeda dapat berkomunikasi dengan saling mengirim data yang ditentukan oleh masing-masing sistem. Konsep ini juga memungkinkan implementasi *bigdata* dengan mengumpulkan data – data pada sistem yang berada dalam instansi atau lembaga. *Database* mengenai data

siswa, jumlah kelas, jumlah siswa per angkatan dapat diketahui secara langsung dari data yang terkumpul. Rancangan arsitektur ini kemudian diimplementasikan pada sistem interkoneksi basis data terdistribusi di sekolah Muhammadiyah. Hasil dari implementasi rancangan arsitektur tersebut diketahui bahwa ketika dilakukan edit data pada aplikasi sekolah, maka data yang berada di aplikasi Majelis Pendidikan Dasar dan Menengah Pimpinan Daerah Magelang juga ikut berubah.

Dalam jurnal yang berjudul *Perancangan Virtualisasi Replikasi Database Pada Arsitektur Cloud Computing* mengangkat tema mengenai penerapan teknologi informasi yang terintegrasi dengan *cloud computing* (Prasetyo, 2017). Penggunaan *cloud computing* yang infrastrukturnya dibangun menggunakan proses virtualisasi yang terdiri dari *web server*, *database*, dan sistem operasi akan berdampak pada efisiensi perawatan, komputasi, dan daya listrik. Metode yang dipergunakan adalah pendekatan *library research* dan *field research*. Metode tersebut digunakan untuk mendesain pola replikasi dua arah pada *database* yang terintegrasi pada virtualisasi secara *cloud computing*. Hasil pengujian dan analisis virtualisasi replikasi *database* pada arsitektur *cloud computing* menunjukkan performansi yang optimal. Hal tersebut dibuktikan dengan kondisi *throughput* dengan rata-rata 0,09 *Kbps* dalam melaksanakan seluruh perintah *service* dari *user* tanpa adanya *delay* dan *jitter* dengan kondisi 0 m/s saat mengirimkan data.

Dalam penelitian yang berjudul *Distributed Database and Application Architecture for Big Data Solutions* yang membahas mengembangkan arsitektur *database* terdistribusi dengan melakukan konversi data dan penentuan abnormalitas *multi-bloked* (Misaki et al., 2017). Arsitektur ini dibangun dengan cara memisahkan data aplikasi dan *database*. Aplikasi dan *database* kemudian digabungkan dalam proses distribusinya yang didukung dengan multi proses. Hal tersebut menurut peneliti menghasilkan layanan volume data besar dalam pendistribusiannya dapat terlayani lebih cepat. Kecepatan dalam melakukan proses dengan asitektur ini didapatkan pemrosesan data akan lebih cepat 27%.

Dalam jurnal yang berjudul *Penerapan Sistem Replikasi Dan Integrasi Basis Data Terdistribusi Pada Pangkalan Data Pendidikan Tinggi (PDPT)* yang membahas mengenai membangun sistem informasi dan mereplikasikan data antar

Program Studi dan PDPT di Universitas Muslim Malang (Belluano, 2017). Metode yang digunakan yaitu studi lapangan dan kepustakaan untuk mengetahui data pelaporan EPSBED dari Prodi ke PDPT-Universitas hingga ke PDPT-DIKTI. Perancangan sistem informasi dan basis data menggunakan *Convention Over Configuration*, pemodelan sistem *Unified Modeling Language*, pengelolaan Data Base Management System menggunakan *PostgreSQL*, dan port *TCP-IP* sebagai sarana komunikasi data dan melakukan replikasi data update dari *database* PDPT ke *database* replikasi.

Dalam jurnal yang berjudul *Different Techniques to Ensure High Availability in Cloud Computing* membahas mengenai teknik untuk menyelidiki teknik yang dibuat untuk menghindari kegagalan tertentu didalam sistem (Bajaber et al., 2017). Metode yang dipergunakan untuk melakukan penelitian ini menggunakan *aristektur cloud computing service layer's* dimana didalam arsitektur ini berisi *Infrastructure as a Service (IaaS)*, *Platform as a Service (PaaS)*, *Software as a Service (SaaS)*. Selain itu penulis juga menggunakan *Service level agreement (SLA)* yang dipergunakan untuk menghitung tingkat layanan kepada pelanggan dari sisi ketersediaan serta skalabilitas. Dan metode *Measuring the availability* dipergunakan untuk memperhitungkan ketersediaan layanan yang tinggi, menghitung waktu henti dan waktu kinerja sistem. Data yang diolah dari beberapa penyedia *cloud computing* yaitu *Salseforce.com*, *Google App engine*, *MenuMate*, dan *Aire*. Dengan demikian setelah penulis melakukan serangkaian pengujian didapatkan bahwa ketersediaan tinggi tidak dapat dipastikan, tetapi dapat ditingkatkan dengan menghindari kegagalan sistem dengan teknik yang berbeda.

evaluasi tingkat ketersediaan layanan apakah teryani dengan ketersediaan tinggi atau tidak pada *cloud computing* . *Cloud computing* merupakan layanan untuk mengatasi ketersediaan pengguna sistem. Layanan ini mengacu pada tingkat kegagalan pada layanan *cloud computing*.

Penelitian dari (Wang & Cheng, 2018) yang berjudul *The Distributed Database Design Solution for Internet Application* menjelaskan database terdistribusi dapat memberikan kemampuan pemrosesan yang lebih kuat.

Database terdistribusi dapat desain arsitektur *database* terdistribusi yang lebih baik untuk *platform* aplikasi Internet.

Dalam jurnal yang berjudul *Distributed Database Problems, Approaches and Solutions - A Study* membahas pemrosesan data di *database* terdistribusi cara memprosesnya dengan menggunakan pendekatan pada sistem *database* dan jaringan komputer (Rana et al., 2018). Sistem basis data dan jaringan komputer akan memberikan transparansi data yang didistribusikan dan kemudian data dapat direplikasi. Permasalahan mencapai integrasi data dan transparansi distribusi data, mempelajari dan mengenali masalah dan pendekatan dari sistem basis data terdistribusi merupakan masalah utama pada penelitian ini. Basis data terdistribusi memungkinkan memungkinkan untuk melakukan penyimpanan dan mengambil data di mana saja jaringan komputer .

Pada jurnal penelitian penelitian dari (Pires & Fagotto, 2018) yang berjudul *High Availability: A DNS and Reverse Proxy Approach in Multi-Cloud* yang membahas solusi menerapkan layanan *multi-cloud* dengan layanan ketersediaan yang tinggi melalui akses DNS dan *reverse proxy*. Dengan membandingkan layanan yang single cloud computing menggunakan perhitungan *High Availability* dengan dua (2) kali percobaan didapatkan layanan 99,43% dan 99,49%. Sedangkan dalam menyediakan layanan *High Availability* menjadi *multi-cloud* didapatkan hasil yang memuaskan yaitu 99,9971%. Dengan demikian layanan *High Availability* dari layanan *multi-cloud* dapat menjamin ketersediaan data.

Penelitian dari (Ezéchiél et al., 2019) yang berjudul *A Systematic Review On Distributed Databases Systems And Their Techniques* yang membahas mengenai teknik yang efisien dalam penggunaan sistem *database* terdistribusi (DDBS) atau sistem *database* yang terdesentralisasi data. Teknik yang efisien untuk melakukan distribusi database dengan cara membangun arsitektur replikasi data untuk mengelola pertukaran data antar database di DDBS. Arsitektur tersebut menggunakan sinkronisasi database melalui arsitektur peer to peer (P2P) yang terdesentralisasi. Arsitektur ini diyakini mampu mengatasi fragmentasi data, alokasi data dan replikasi data.

Dalam jurnal yang berjudul *Implementasi Sistem Database Terdistribusi Dengan Metode Multi Master Database Replication* membahas mengenai replikasi database menggunakan metode *multi master* (Heryanto & Albert, 2019). Penyimpanan data dalam jumlah besar akan meningkatkan kemungkinan data akan mengalami kerusakan, data tidak dapat diakses atau bahkan kemungkinan terburuk yang dapat terjadi adalah data tersebut akan hilang (*loss*). Gangguan tersebut bisa diakibatkan dari berbagai macam hal, seperti gangguan *maintenance*, kerusakan *database*, kerusakan media serta data *corruption* dan bencana yang tak terduga (bencana alam). Dengan demikian perlu adanya ketersediaan data yang baik secara realtime. Replikasi data adalah sebuah proses yang sering dilakukan pada sistem *database* terdistribusi. Penelitian ini menggunakan model arsitektur replikasi model *Galera Cluster*. Model ini merupakan replikasi *Synchronous multi master* menggunakan *Mysq* dengan format data *InnoDB*. Proses replikasi *Synchronous* terjadi secara *real time*. Sinkronisasi data menyediakan *recovery* data yang konsisten. Serta sebuah aplikasi tidak perlu tahu server mana yang bisa mereka tulis (*Master*) dan server mana yang mereka bisa baca datanya (*Slave*). Penelitian ini menghasilkan kesimpulan bahwa sistem *kluster database* mampu mengatasi kekurangan *resource* komputasi pada *engine database*. Aplikasi *master-slave* mampu menyediakan backup terhadap data yang tersimpan didalam *database*.

Dalam jurnal yang berjudul *Dynamic Software Updates to Enhance Security and Privacy in High Availability Energy Management Applications in Smart cities* membahas mengenai pembaharuan perangkat lunak menggunakan konektivitas yang tinggi di internet untuk meningkatkan layanan berbagai informasi, sistem keamanan, perlindungan privasi pengguna *internet* dan *smart cities* yang masih memerlukan *restart* perangkat setelah pembaharuan selesai (Mugarza et al., 2019). Pembaharuan perangkat lunak sangat penting untuk menjaga keamanan aplikasi dan privasi pengguna aplikasi. Solusi yang ditawarkan penulis adalah menyajikan arsitektur dan desain yang dibangun dalam kerangka *Ceratus* dimana teknik ini menggunakan isolasi temporal dan spasial. Dengan demikian

pembaharuan perangkat lunak yang dilakukan secara dinamis selesai, maka perangkat tidak melakukan *restart* layanan untuk *high Availability* dapat tercapai.

Dalam jurnal yang berjudul *High Availability Server Using Raspberry Pi 4 Cluster and Docker Swarm* membahas mengenai perlunya infrastruktur *web server* yang berkinerja tinggi yang mampu melayani segala aktivitas dan transaksi yang dibutuhkan pengguna tanpa kendala apapun. Dengan merancang infrastruktur yang berkinerja tinggi dengan biaya yang rendah dan efisiensi, maka dengan menggunakan teknologi *cluster computing* pada teknologi *Raspberry Pi Single Board Computing* dan *Docker Container*. Sistem cluster dibangun menggunakan lima modul *raspberry Pi tipe 4B* sebagai *node cluster*, dan sistem *Web server* dibangun menggunakan teknologi virtualisasi *docker container*. Sedangkan pengelolaan cluster menggunakan teknologi *Docker Swarm*. Pengujian performansi (*Quality of Service*) sistem *cluster* dilakukan dengan mensimulasikan sejumlah beban (*request*) dan mengukur respon sistem berdasarkan parameter *Throughput* dan *Delay (latency)*. Hasil pengujiannya adalah bahwa sistem *Raspberry Pi Cluster* menggunakan *Docker Swarm* dapat digunakan untuk membangun sistem *High Availability Server* yang mampu menangani request yang sangat tinggi yang mencapai *Throughput = 161.812.298 request/sec* dengan *Error rate = 0%*.

Penelitian diatas dirangkum dalam tabel 2.1 berikut ini :

Tabel 2.1. Tabel Tinjauan Pustaka

No	Nama / Tahun	Judul Penelitian	Data	Kontribusi
1	(Kungliga & Höskolan, 2015)	<i>Multi-Master Database Replication and e-Learning Theoretical And Practical Evaluation Mattias Holmgren</i>	- Data e-learning	Menggunakan replikasi <i>multi-master</i> yang didapatkan dalam penelitiannya dalam proses sinkronisasi datanya memakan waktu yang cukup lama.

Tabel 2.1. Lanjutan Tabel Tinjauan Pustaka

No	Nama / Tahun	Judul Penelitian	Data	Kontribusi
2	(Mulyanto & Ashari, 2016)	Implementasi <i>Highly Available</i> Website Dengan <i>Distributed Replicated Block Device</i>	- Data <i>Webserver</i>	Merancang layanan <i>website</i> yang memiliki tingkat <i>availability</i> tinggi. Serta membandingkan sistem <i>high availability</i> melalui pengujian <i>availability</i> sistem, waktu proses terhadap permintaan data (<i>response time</i>) dan kemampuan sistem terhadap beban kerja
3	(Correia et al., 2016)	Development of High-Availability ATCA/PCIE Data Acquisition Instrumentation	Data Hardware (ATCA) dan Data Software (PCI)	ATCA form-factor tidak hanya terdiri dari fitur intrinsic namun juga menyediakan redundancy resources yang digunakan untuk meningkatkan sistem <i>availability</i> terutama transmisi level pada ATCA backplane
4	(Yulianto, 2017)	Implementasi Sistem Interkoneksi Basis Data Terdistribusi Menggunakan <i>Socket API</i>	- Data Sekolah - <i>Web API Client</i> - Data JSON	Penelitian yang dilakukan dapat mempermudah Majelis Pendidikan Dasar dan Menengah Pimpinan Daerah Magelang dalam collecting data.

Tabel 2.1. Lanjutan Tabel Tinjauan Pustaka

No	Nama / Tahun	Judul Penelitian	Data	Kontribusi
5	(Prasetyo, 2017)	Perancangan Virtualisasi Replikasi Database Pada Arsitektur <i>Cloud Computing</i>	- Data server	Mendesain pola replikasi dua arah pada <i>database</i> yang terintegrasi pada virtualisasi secara <i>cloud computing</i>
6	(Misaki et al., 2017)	<i>Distributed Database and Application Architecture for Big Data Solutions</i>	Data <i>platform</i> dan arsitektur penyimpanan <i>big data</i>	Membangun infrastruktur IT yang mampu memproses pada kecepatan tinggi dan jumlah <i>manufacturing site</i> dengan skala besar
7	(Belluano, 2017)	Penerapan Sistem Replikasi Dan Integrasi Basis Data Terdistribusi Pada Pangkalan Data Pendidikan Tinggi (PDPT)	Database Pangkalan Data Pendidikan Tinggi (PDPT) sample di Univ. Muslim Indonesia	Sistem informasi yang dibangun untuk mengintegrasikan dan mereplikasikan data antar Program Studi dan PDPT Universitas menggunakan teknologi <i>IP-streaming</i> telah bekerja dengan baik, sesuai dengan hasil implementasi yang dipaparkan. Serta data pelaporan yang diterima PDPT-Universitas telah berjalan konsisten dengan data pelaporan yang ada

Tabel 2.1. Lanjutan Tabel Tinjauan Pustaka

No	Nama / Tahun	Judul Penelitian	Data	Kontribusi
8	(Bajaber et al., 2017)	<i>Different Techniques to Ensure High Availability in Cloud Computing</i>	Data cloud: <i>salseforce.com, google app engine, menumate, aire</i>	Konfigurasi otomatis harus dipasang untuk menambah <i>validation checks</i> guna meningkatkan mekanisme pendeteksi dan <i>service failure recovery</i>
9	(Wang & Cheng, 2018)	<i>The Distributed Database Design Solution for Internet Application Platform</i>	- Data produk pelanggan - Data akun pembayaran pelanggan	Perancangan dan analisis arsitektur sistem basis data pada <i>platform</i> aplikasi internet yang fokus pada teknologi yang aman dan stabil.
10	(Rana et al., 2018)	<i>Distributed Database Problems, Approaches and Solutions — A Study</i>	- Data testing <i>hardware</i> , data aplikasi, data DBMS	Memberikan gambaran untuk rancangan arsitektur database dengan <i>server model</i> untuk perancangan serta implementasi distribusi <i>database</i>
11	(Pires & Fagotto, 2018)	<i>High Availability: A DNS and Reverse Proxy Approach in Multi-Cloud</i>	- Data ujicoba <i>DNS Server</i>	Mengimplementasikan arsitektur <i>High Availability</i> yang menghasilkan data akses <i>DNS Client</i> dengan <i>Service Layer Agreement</i> (SLA) 99,95%

Tabel 2.1. Lanjutan Tabel Tinjauan Pustaka

No	Nama / Tahun	Judul Penelitian	Data	Kontribusi
12	(Ezéchiél et al., 2019)	<i>A systematic review on distributed databases Systems and their techniques</i>	-Alokasi Data <i>Sql</i> - Data Replikasi	Metode replikasi dengan menggunakan arsitektur <i>peer to peer master slave</i> dapat memberikan distribusi <i>database</i> yang baik.
13	(Heryanto & Albert, 2019)	Implementasi Sistem <i>Database</i> Terdistribusi Dengan Metode <i>Multi Master Database Replication</i>	- Data <i>Mysql Server</i>	Solusi untuk ketersediaan data yang tinggi dengan replikasi yang sinkron, <i>failover</i> dan resinkronisasi dan tidak akan ada data yang hilang
14	(Mugarza et al., 2019)	<i>Dynamic Software Updates to Enhance Security and Privacy in High Availability Energy Management Applications in Smart Cities</i>	- Data <i>software architecture based on the Cetratus runtime framework</i> - Data privasi pengguna	Desain <i>software</i> dengan menggunakan <i>Cetratus runtime framework</i> memungkinkan update data pada runtime tanpa shutdown ataupun <i>restart</i> .
15	(Hadiwandura & Candra, 2021)	<i>High Availability Server Using Raspberry Pi 4 Cluster and Docker Swarm</i>	- Data toleransi <i>swarm</i> pada <i>docker</i> , data latensi keterlambatan , <i>data throughput</i>	<i>Raspberry devices</i> dapat menjalankan <i>virtualization technology</i> menggunakan <i>docker</i> . <i>Docker virtualization technology</i> dapat digunakan untuk

Tabel 2.1. Lanjutan Tabel Tinjauan Pustaka

No	Nama / Tahun	Judul Penelitian	Data	Kontribusi
				menciptakan <i>cluster computer</i> untuk menyokong <i>high availability servers</i>
16	Usulan penulis	<i>High Availability</i> Pada Basis Data Terdistribusi: Studi Kasus STP AMPTA Yogyakarta.	-Data Pegawai - Data Client Komputer. - Data Transaksi	Menganalisis data hasil kajian untuk menghasilkan konfigurasi <i>database</i> sehingga layanan <i>High Availability, flexibility, dan scalability</i> dapat tercapai.