

ARSITEKTUR *OBSERVABLE-SOA* UNTUK PENGEMBANGAN PERPUSTAKAAN DIGITAL TERINTEGRASI NASIONAL

Tri A. Kurniawan^{*1}, Johan A. E. Noor², Nurudin Santoso³

^{1,3}Software Engineering Laboratory, Fakultas Ilmu Komputer, Universitas Brawijaya

²Jurusan Fisika, Fakultas Ilmu Pengatahuan Alam (MIPA), Universitas Brawijaya

Email: ¹triak@ub.ac.id, ²jnoor@ub.ac.id, ³nurudin.santoso@ub.ac.id

^{*}Penulis Korespondensi

(Naskah masuk: 26 April 2021, diterima untuk diterbitkan: 19 Juli 2021)

Abstrak

Katalog induk nasional (KIN) memegang peran mendasar dalam pengembangan perpustakaan digital terintegrasi nasional. KIN merupakan hasil konsolidasi katalog dari setiap perpustakaan anggota. Perpustakaan Nasional RI (Perpusnas), yang bertanggung jawab untuk membangun KIN, saat ini menggunakan sebuah platform tunggal dalam konsolidasi tersebut. Semua perpustakaan anggota harus menyediakan sistem yang sama, baik perangkat keras maupun perangkat lunak, untuk bisa berkolaborasi dalam KIN. Arsitektur monolitik seperti ini sangat berpotensi menghalangi perpustakaan yang belum siap dengan sistem yang dibutuhkan untuk berkontribusi dalam pengembangan KIN karena membutuhkan biaya yang tidak sedikit. Artikel ini membahas arsitektur *Observable-SOA* untuk menjadi alternatif arsitektur yang fleksibel sehingga memungkinkan beberapa perpustakaan anggota yang berjalan pada berbagai platformnya masing-masing yang saling berbeda untuk bisa melakukan interoperasi secara efektif dalam mengembangkan KIN. Arsitektur tersebut memanfaatkan konsep yang ada pada SOA (*service-oriented architecture*) dan pola perancangan *Observer*. Arsitektur yang diusulkan, berikut algoritme dari beberapa layanan (*service*) dasar, telah berhasil diuji fungsionalitasnya dalam melakukan konsolidasi KIN dan pencarian katalog pada lingkungan simulasi yang merepresentasikan interoperasi antara Perpusnas dengan setiap perpustakaan anggotanya. Lingkungan uji tersebut melibatkan 4 perpustakaan digital yang diimplementasikan dengan menggunakan 3 sistem perpustakaan terintegrasi yang bersifat *open source*. Arsitektur *Observable-SOA* ini bisa menjadi pengganti arsitektur monolitik yang saat ini digunakan oleh Perpusnas untuk mengembangkan KIN tanpa harus membebani perpustakaan anggota dengan berbagai perangkat tambahan.

Kata kunci: katalog induk nasional, perpustakaan digital, *service-oriented architecture*, *observer design pattern*, interoperasi

OBSERVABLE-SOA ARCHITECTURE FOR DEVELOPING A NATIONAL INTEGRATED DIGITAL LIBRARY

Abstract

The national union catalog (KIN) plays a fundamental role in developing a national integrated digital library (NIDL). KIN is consolidated from the catalogs of its various constituent libraries. The National Library of Indonesia (Perpusnas), which is responsible for building KIN, is currently using a single platform built for such consolidation purposes. All constituent libraries must provide the same system, which includes hardware and software, to collaborate in KIN. This monolithic setting may prevent some libraries, which are not ready with the required system, to contribute in developing such KIN since it costs a lot. This article discusses the *Observable-SOA* architecture to provide a flexible setting allowing some constituent libraries with various different platforms to effectively interoperate in developing such catalog within a NIDL. Such architecture leverages the *Observer* design pattern and SOA (*service-oriented architecture*) concepts. The proposed architecture and some basic services algorithms were successfully tested for its functionalities in consolidating KIN and searching a particular catalog within a simulated environment representing the interoperability between the Perpusnas and its constituents. Such environment involved 4 digital libraries implemented by using 3 open-source integrated library systems (ILSs). This *Observable-SOA* architecture may be used to replace the monolithic architecture currently used by the Perpusnas to develop KIN without burdening the constituent libraries with various additional systems.

Keywords: national union catalog, digital library, *service-oriented architecture*, *observer design pattern*, interoperation

1 PENDAHULUAN

Kolaborasi antar berbagai perpustakaan di suatu negara telah menjadi isu kritis di era informasi yang menuntut pertukaran informasi yang efektif, cepat dan akurat, yang diperlukan dalam rangka pengembangan sumber daya manusia di negara tersebut. Kolaborasi semacam itu saat ini berkembang menjadi kemitraan yang lebih luas termasuk koleksi digital multi-kelembagaan (Buchanan, et al., 2012).

Di Indonesia, Perpustakaan Nasional RI, selanjutnya disebut secara singkat dengan Perpustakaan, memainkan peran utama untuk memfasilitasi terjadinya kolaborasi dari berbagai perpustakaan yang ada dengan kemungkinan beragam platform dan lingkungan yang dimilikinya. Kondisi geografis seperti yang ada di Indonesia dan negara lain yang sejenis bisa menjadi masalah serius dalam rangka pemerataan akses koleksi perpustakaan sehingga mendorong peningkatan kebutuhan akan pengembangan perpustakaan digital untuk memfasilitasi pertukaran informasi yang efektif. Dalam kenyataannya, pengembangan perpustakaan digital terintegrasi nasional sangat bertumpu pada katalog induk nasional (KIN) yang solid yang datanya diperoleh dari berbagai perpustakaan anggotanya secara sistematis (Witten, et al., 2010). Berdasarkan katalog induk yang solid tersebut, beragam layanan interoperasi yang kompleks antar perpustakaan bisa dikembangkan dengan lebih baik dengan mengacu pada proses bisnis yang sudah disepakati bersama.

Perpusnas mengembangkan sebuah platform tunggal untuk membangun KIN sehingga mengharuskan semua perpustakaan anggota untuk menyediakan sistem yang sama, termasuk perangkat keras dan perangkat lunak (Kismiyati, 2018). Arsitektur monolitik semacam ini sangat berpotensi menghalangi beberapa perpustakaan anggota, yang belum siap dengan sistem yang dibutuhkan, untuk bisa berkontribusi secara efektif dalam pengembangan KIN tersebut. Sebagai akibatnya, KIN belum bisa dibangun secara optimal. Dalam artikel ini, arsitektur *Observable-SOA* diusulkan sebagai arsitektur alternatif bagi Perpustakaan dalam pengembangan KIN. Arsitektur ini dibangun berdasarkan konsep SOA (*service-oriented architecture*) dan pola perancangan *Observer*. Dengan mengadopsi SOA, arsitektur ini memungkinkan interoperasi antar perpustakaan dengan beragam platform berbeda untuk membangun KIN tanpa harus dibebani dengan perangkat tambahan. Lebih lanjut, arsitektur alternatif ini juga memungkinkan sebuah perpustakaan bisa bergabung atau keluar kapanpun dari jejaring Perpustakaan tersebut sehingga memberikan fleksibilitas dalam membangun infrastruktur digital yang bisa dikelola secara berkesinambungan (Hustad & Olsen, 2021).

Sistematika pembahasan dibuat sebagai berikut. Bagian 2 membahas penelitian terkait. Bagian 3 menjelaskan konsep dasar perpustakaan digital

terintegrasi nasional, SOA, dan pola perancangan *Observer*. Bagian 4 menjelaskan kebutuhan dasar sistem dari perpustakaan digital terintegrasi nasional sebagai basis pengembangan arsitektur yang diusulkan. Bagian 5 memperkenalkan arsitektur yang diusulkan. Bagian 6 menguraikan eksperimen sebagai pembuktian konsep. Bagian 7 menjelaskan kesimpulan dan penelitian lanjutan.

2 PENELITIAN TERKAIT

Beberapa penelitian tentang konsep SOA yang diadopsi sebagai arsitektur sistem perangkat lunak dengan platform yang beragam telah dilakukan sebelumnya. Ding dan Solvberg mengusulkan model SOA berbasis *peer* (PSOA) untuk mengembangkan sistem perpustakaan digital (Ding & Solvberg, 2004). Arsitektur ini dibangun dengan memperluas infrastruktur SOA dasar sehingga setiap perpustakaan dapat menjadi *service provider*, *service requester*, atau keduanya. Zhao, et.al membahas pendekatan *web service* dalam proyek perpustakaan digital biomedik (Zhao, et al., 2012). Arsitektur ini memungkinkan berbagi data medis dalam beragam format di antara dokter dan peneliti. SOA juga diadopsi dalam penelitian di bidang medis yang lain dalam rangka berbagi data rekam medis antar rumah sakit yang berbeda di Kolombia (Arévalo, et al., 2019), pengelolaan data genomik yang berukuran besar (Gancheva & Borovska, 2019), pengembangan *telecare system* yang melibatkan beragam perangkat yang berbeda (Macis, et al., 2019).

Lebih lanjut, penggunaan SOA dalam pengelolaan informasi yang beragam dan platform yang berbeda juga telah diadopsi secara luas di berbagai bidang yang lain: pengelolaan *big data* di industri gas dan minyak (Azzedin & Ghaleb, 2019), optimasi layanan distribusi makanan di Saudi Arabia (Abdeen, 2020), perancangan platform pengajaran multimedia musik (Liu, 2020), interoperasi antar layanan pada sistem *e-Government* di Indonesia (Fajar & Shofi, 2019), integrasi data biologi yang tersebar dengan format yang beragam (Gancheva, 2019). Elshater, et.al meneliti pola perancangan, yang terinspirasi dari pendekatan berorientasi objek, untuk meningkatkan kinerja *web service* dalam proses pencarian informasi (Elshater, et al., 2015). Ngaogate memanfaatkan pola *Observer* dan *Composite* yang dikombinasikan dengan arsitektur SOA dalam pengembangan sistem *smart city* yang melibatkan beragam perangkat yang berbeda (Ngaogate, 2019).

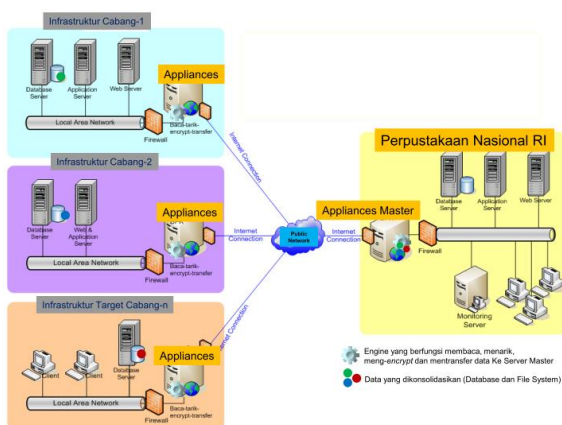
3 KONSEP DASAR

3.1 Perpustakaan digital terintegrasi nasional

Perpustakaan nasional memiliki fungsi kritis dalam memfasilitasi pengembangan jejaring perpustakaan dalam sebuah negara. Poin kritisnya adalah bahwa berbagai pengetahuan yang tersimpan dalam bentuk buku dan literatur lain yang tersebar di berbagai perpustakaan yang ada di negara tersebut

harus bisa dikonsolidasikan untuk membangun jejaring pengetahuan yang efektif yang dibutuhkan dalam rangka pengembangan sumber daya manusianya. Sebagaimana dipahami, kualitas sumber daya manusia yang berpengetahuan dari sebuah bangsa akan sangat berpengaruh pada kemampuan bangsa tersebut untuk berperan di tingkat internasional. Perpustakaan digital menjadi pilihan utama dibandingkan perpustakaan tradisional dalam menjalankan fungsi tersebut karena perpustakaan digital bisa mengoleksi beragam sumber informasi digital yang bisa diakses pengguna dari manapun dengan mudah (Chowdhury & Chowdhury, 2003).

Pada perpustakaan digital, katalog menjadi infrastruktur dasar yang memungkinkan pengguna menemukan, mengidentifikasi, dan memilih sumber bibliografi secara digital dari sebuah koleksi tertentu (Witten, et al., 2010) (Joundrey, et al., 2015). Pada lingkungan perpustakaan *online*, isi katalog harus bisa diakses secara elektronik oleh mesin untuk keperluan proses manipulasi informasi lebih lanjut. Sistem MARC (*M*ACHINE-*R*eadable-*C*ataloging)¹ telah digunakan untuk mengembangkan isi katalog elektronik (Joundrey, et al., 2015). Beberapa negara telah mengadopsi sistem katalog tersebut ke dalam spesifikasi tertentu yang sesuai dengan kebutuhan di negara masing-masing, misalnya INDOMARC (Indonesia), AUSMARC (Australia), JPNMARC (Jepang). Sementara itu, versi terbaru dari MARC yang digunakan di USA adalah MARC 21.



Gambar 1. Arsitektur monolitik dalam pengembangan KIN di Indonesia (Kismiyati, 2018)

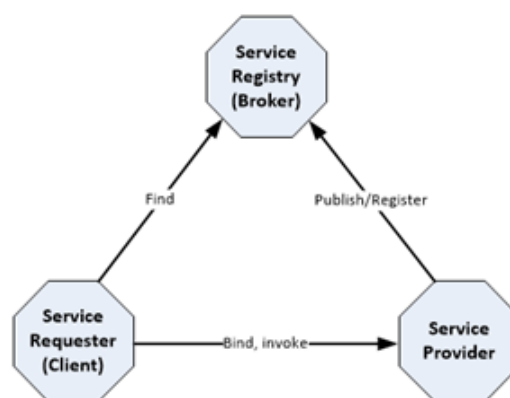
Di Indonesia, Perpustakaan Nasional membangun KIN berdasarkan arsitektur monolitik yang diilustrasikan pada Gambar 1. Setiap perpustakaan anggota (Infrastruktur Cabang) harus dilengkapi dengan perangkat keras dan perangkat lunak yang disediakan oleh Perpustakaan Nasional untuk melakukan proses konsolidasi katalog secara terpusat di server induk yang dikelola oleh Perpustakaan Nasional. Dalam praktiknya, Perpustakaan Nasional memiliki keterbatasan dana untuk menyediakan perangkat tambahan tersebut sehingga distribusinya tidak merata pada setiap perpustakaan anggota.

Sebagai konsekuensinya, cakupan konsolidasi KIN masih belum mampu menjangkau seluruh perpustakaan yang ada di Indonesia, baik yang dikelola oleh instansi pemerintah/swasta, pemerintah daerah, lembaga penelitian, dan lembaga pendidikan.

3.2 Service-Oriented Architecture (SOA)

Arsitektur berorientasi layanan (SOA) adalah pendekatan baru dalam mendeskripsikan arsitektur sistem yang diarahkan untuk mengembangkan sistem terdistribusi yang tidak bergantung pada suatu protokol tertentu dan cukup fleksibel untuk berkembang (Papazoglou & Van Den Heuvel, 2007). Arsitektur ini memungkinkan pengembang sistem untuk mengatasi permasalahan yang dihadapi pada lingkungan komputasi yang bersifat heterogen, baik dari sisi kebijakan integrasi, platform, keamanan, dan manajemen transaksi. Layanan (*service*) menjadi bagian fundamental yang disediakan bagi setiap komponen yang terlibat dalam sebuah sistem. *Web service* adalah implementasi umum dari SOA.

Gambar 2 mengilustrasikan arsitektur dasar dari *web service* yang terdiri dari tiga jenis partisipan, yaitu *service provider*, *service requester*, dan *service registry*. *Service provider* membuat *web service* dan mendaftarkannya ke *service registry*. *Service registry* menyimpan semua layanan yang terdaftar agar siap digunakan oleh setiap *service requester* yang memerlukan. *Service requester* mencari layanan tertentu di *service registry* dan melakukan transaksi (*binding*) ke *service provider* yang relevan untuk menjalankan layanan yang sesuai. Berbagai interaksi tersebut mengacu pada beberapa standar (XML, SOAP, WSDL, dan UDDI) yang digunakan untuk koneksi, komunikasi, deskripsi, dan pencarian, (Singh & Huhns, 2005).



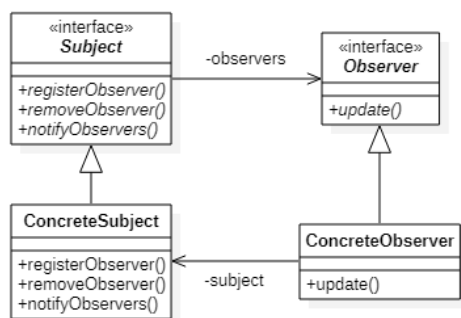
Gambar 2. Arsitektur dasar dari *web service*

3.3 Pola perancangan *Observer*

Pola perancangan *Observer* merepresentasikan ketergantungan satu-ke-banyak di antara sekumpulan objek sehingga jika status satu objek, disebut *Subject*, telah diubah, maka semua objek yang bergantung

¹ <https://www.loc.gov/marc/>

padanya, disebut *Observer*, akan diberikan notifikasi yang sesuai (Freeman, et al., 2004). Pola perancangan ini melibatkan *ConcreteSubject* dan *ConcreteObserver* untuk mengimplementasikan masing-masing *Subject* dan *Observer*, seperti yang ditunjukkan pada Gambar 3. Setiap *Subject* dapat memiliki banyak *Observer* yang harus didaftarkan agar nantinya bisa mendapatkan notifikasi. Pola ini memungkinkan setiap *Observer* untuk kapanpun bisa bergabung dengan *Subject* dan meninggalkannya.



Gambar 3. Arsitektur dasar dari pola perancangan *Observer*

4 KEBUTUHAN SISTEM

Kebutuhan sistem dalam pengembangan sistem terintegrasi yang kompleks yang memungkinkan interoperasi antar perpustakaan secara nasional dalam sebuah negara harus bisa didefinisikan dengan jelas. Sistem yang dibangun harus mempertimbangkan kenyataan bahwa setiap perpustakaan yang akan terlibat dalam jejaring sangat mungkin beroperasi pada platform yang berbeda-beda. Lingkungan terintegrasi tersebut merepresentasikan sebuah jejaring interoperasi dari berbagai perpustakaan yang melibatkan 1 perpustakaan utama (yaitu perpustakaan nasional) dan sejumlah perpustakaan anggota. Deskripsi berikut menjelaskan kebutuhan dasar sistem perpustakaan digital terintegrasi nasional sebagai basis dari arsitektur yang diusulkan:

R1 – Sistem terintegrasi ini memungkinkan perpustakaan dengan berbagai platform yang berbeda untuk melakukan interoperasi dalam jejaring perpustakaan digital nasional. Perpustakaan anggota dapat bergabung dengan jejaring perpustakaan nasional dengan tetap menggunakan sistemnya sendiri. Setelah terhubung, perpustakaan tersebut dapat melakukan interoperasi dengan perpustakaan yang lain dalam jejaring. Selain itu, sebuah perpustakaan anggota juga dapat dihapus dari jejaring, jika perpustakaan tersebut sudah tidak bergabung lagi dalam jejaring karena beberapa alasan.

R2 – Setiap perpustakaan anggota dan perpustakaan nasional harus dapat mengkonsolidasikan masing-masing katalognya secara berkala. Perpustakaan nasional memainkan peran utama dalam jejaring karena katalog induk nasional (KIN) yang dikonsolidasikan dari semua perpustakaan anggota dikelola di situ. Setiap perpustakaan anggota akan melaporkan setiap perubahan yang dilakukan pada

katalog lokalnya ke perpustakaan nasional untuk dikonsolidasikan. Setiap perpustakaan menggunakan standar katalog yang sama, misalnya MARC 21. Dalam rangka menjaga kinerja sistem khususnya dalam proses pencarian katalog, informasi dalam katalog induk juga harus tersedia di setiap katalog lokal dimana proses pencarian katalog diinisiasi.

R3 – Layanan pencarian informasi berdasarkan katalog yang sudah dikonsolidasikan harus bisa diinisiasi secara efektif dari setiap perpustakaan anggota dan perpustakaan nasional itu sendiri. Setiap perpustakaan dalam jejaring menyediakan layanan pencarian untuk penggunaannya berdasarkan katalognya sendiri yang dikelola secara lokal. Jika, karena kondisi tertentu, katalog lokal gagal menyajikan informasi yang diperlukan, pencarian diteruskan ke katalog induk yang ada di perpustakaan nasional melalui jejaring yang ada.

Kebutuhan dasar ini akan mengarahkan bagaimana arsitektur sistem terintegrasi harus dikembangkan. Setelah arsitektur yang solid sudah dirancang dengan benar, layanan perpustakaan lanjutan yang mungkin kompleks, misalnya peminjaman antar perpustakaan (*inter-library loan*), dapat disiapkan melalui interoperasi antar perpustakaan berdasarkan proses bisnis yang telah disepakati bersama.

5 ARSITEKTUR OBSERVABLE-SOA

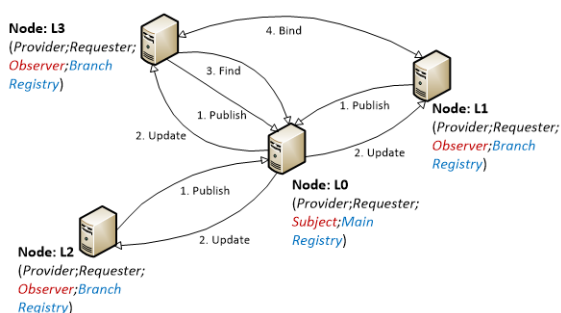
5.1 Struktur arsitektur

Berdasarkan kebutuhan yang telah dibahas sebelumnya, arsitektur dirancang berdasarkan pada arsitektur berorientasi layanan (SOA) yang memungkinkan sejumlah perpustakaan yang berjalan pada platformnya masing-masing bisa saling beroperasi satu sama lain (berdasarkan kebutuhan **R1**). Dalam desain seperti itu, katalog menjadi perhatian utama karena sebagian besar layanan perpustakaan akan mengandalkan informasi yang ada pada katalog tersebut. Setiap metadata koleksi perpustakaan yang dijelaskan dalam katalog harus dipandang sebagai layanan di dalam kerangka SOA. Berdasarkan arsitektur SOA dasar yang digambarkan pada Gambar 2, Perpustakaan akan berperan sebagai *service registry* yang mengelola katalog induk sebagai hasil konsolidasi, yang disebut sebagai *Main Registry*. *Main Registry* juga harus menyimpan lokasi fisik perpustakaan dari setiap metadata koleksi perpustakaan yang dikelola.

Sementara itu, setiap perpustakaan anggota akan dipandang sebagai *service provider* dan *service requester*, yang selanjutnya masing-masing hanya akan disebut sebagai *provider* dan *requester*. Setiap perpustakaan anggota akan memublikasikan seketika perubahan apa pun yang dilakukan pada metadata koleksinya, yaitu katalog lokal, ke dalam *Main Registry*. Hal ini akan membuat *Main Registry* tetap mutakhir. Jika terjadi permintaan pencarian terhadap sebuah koleksi tertentu yang mungkin tidak tersedia

di katalog lokal, *requester* meminta perpustakaan nasional untuk mencari di *Main Registry* dari setiap *provider* mana pun yang memiliki koleksi yang sesuai dengan yang sedang dicari.

Namun demikian, arsitektur semacam ini berpotensi menurunkan kinerja sistem jika sejumlah permintaan terjadi secara bersamaan dari sejumlah *requester* karena terjadinya *bottlenecks* saat mengakses koleksi kritis, yaitu *Main Registry*. Skenario konsolidasi katalog yang digunakan pada arsitektur dasar ini menggunakan pendekatan satu arah, yaitu dari perpustakaan anggota ke Perpustnas.



Gambar 4. Arsitektur *Observable-SOA*

Untuk menghindari risiko penurunan kinerja sistem, arsitektur dasar pada jejaring Perpustnas tersebut diperbaiki dengan mengadopsi konsep pola perancangan *Observer*. Gambar 4 mengilustrasikan arsitektur sistem yang sudah mengadopsi kedua konsep SOA dan pola *Observer* yang disebut arsitektur *Observable-SOA*. Pada arsitektur ini, elemen baru yang dimasukkan adalah *Subject* dan *Observer*, yang masing-masing merepresentasikan Perpustnas dan perpustakaan anggota. Setiap *node* merepresentasikan sebuah perpustakaan yang dapat berfungsi sebagai *provider* sekaligus *requester*. *Node L0* merepresentasikan Perpustnas, sedangkan *node L1*, *L2*, dan *L3* merepresentasikan perpustakaan anggota. Pada prinsipnya, jumlah *node* dapat disesuaikan dengan lingkungan yang sebenarnya. Setiap perpustakaan anggota akan terdaftar sebagai *Observer* pada *Subject*, sebagaimana konsep yang diilustrasikan pada Gambar 3.

Perpustnas mengelola *Main Registry*, sedangkan setiap perpustakaan anggota mengelola *Branch Registry* masing-masing. *Branch Registry* merepresentasikan katalog lokal di sebuah perpustakaan anggota. Mirip dengan arsitektur dasar jejaring Perpustnas sebelumnya, setiap perubahan yang dibuat pada *Branch Registry* tertentu akan dipublikasikan seketika ke dalam *Main Registry* oleh perpustakaan anggota yang sesuai (lihat garis panah #1 pada Gambar 4). Berdasarkan pola *Observer*, setiap perubahan yang dibuat di *Main Registry* pada *Subject*, sehubungan dengan perubahan yang dilakukan pada sejumlah *Branch Registry*, akan diperbarui secara berkala ke setiap *Branch Registry* pada semua *Observer* yang terdaftar (lihat garis panah #2 pada Gambar 4). Dengan demikian, setiap *Branch*

Registry idealnya menjadi salinan dari *Main Registry* (berdasarkan kebutuhan **R2**). Namun demikian, pembatasan perlu dilakukan agar kinerja sistem tetap terjaga saat konsolidasi katalog, yaitu hanya metadata koleksi yang diubah saja yang akan dikonsolidasikan di *Main Registry* dan setiap *Branch Registry*, yang mencakup penambahan koleksi baru, perubahan koleksi yang sudah ada, dan penghapusan koleksi.

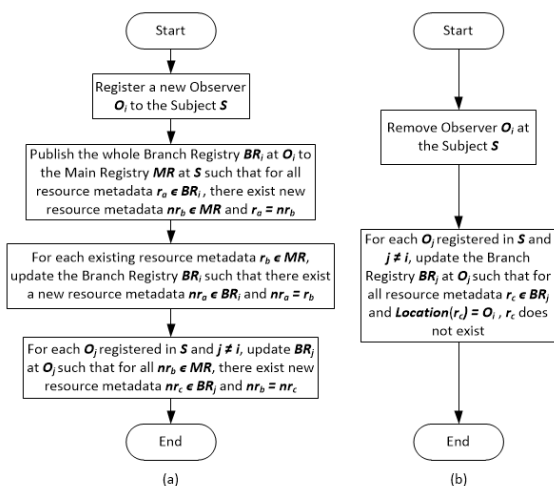
Berkenaan dengan pencarian metadata koleksi (berdasarkan kebutuhan **R3**), setelah permintaan diinisiasi pada *requester* tertentu (misalnya *node L2*), permintaan tersebut akan diproses secara lokal di *Branch Registry* pada *node L2*. Jika permintaan terpenuhi (misalnya *node L3* menyediakan koleksi yang diminta), *requester* melakukan transaksi dengan *provider* (lihat garis panah #4 pada Gambar 4). Jika tidak, permintaan akan diteruskan ke *Main Registry* untuk mencari *provider* yang relevan, yaitu *node L3* (lihat garis panah #3 pada Gambar 4). Kondisi terakhir ini dapat terjadi jika konten terakhir dari *Main Registry* belum dimutakhirkan ke setiap *Branch Registry* pada semua *Observer* yang terdaftar karena proses pemutakhiran semacam itu dilakukan dalam periode waktu tertentu yang ditentukan pengguna. Namun demikian, perbedaan antara *Main Registry* dan setiap *Branch Registry* tidak akan signifikan, mengingat perubahan katalog lokal dalam kenyataannya tidak akan sering terjadi.

5.2 Algoritme layanan

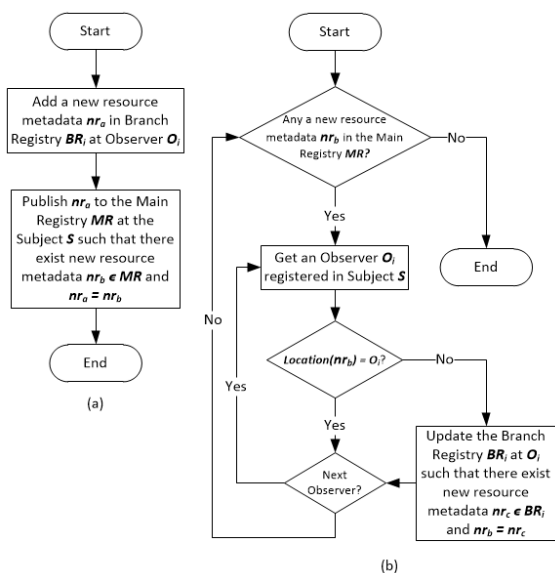
Berdasarkan gambaran umum arsitektur yang diusulkan, algoritme dari beberapa layanan dasar yang diperlukan oleh *Subject* dan *Observer* akan direpresentasikan dalam beberapa diagram alur sehingga bisa dijelaskan secara ringkas dan tepat. Gambar 5 menjelaskan proses penambahan sebuah perpustakaan anggota baru ke dalam jejaring, dan proses penghapusan sebuah perpustakaan yang ada dari jejaring. Kedua proses tersebut sesuai dengan kebutuhan **R1** yang melibatkan beberapa layanan. Misalnya, sebuah layanan *Register* di *Subject* bisa dibangun untuk merealisasikan aktivitas pendaftaran sebuah *Observer* baru ke *Subject*, seperti ditunjukkan pada Gambar 5(a).

Berkenaan dengan kebutuhan **R2** dan **R3**, algoritme layanan untuk proses konsolidasi dan pencarian, masing-masing diilustrasikan pada Gambar 6 dan Gambar 7. Kedua proses tersebut juga membutuhkan beberapa layanan untuk mewujudkan setiap aktivitas yang direpresentasikan dalam diagram. Perlu diperhatikan bahwa pemutakhiran setiap *Branch Registry* berdasarkan status terakhir *Main Registry*, seperti yang ditunjukkan pada Gambar 6(b), akan dilakukan secara berkala/periodik, misalnya sekali sebulan. Sebaliknya, publikasi metadata sebuah koleksi yang baru dari *Branch Registry* tertentu ke *Main Registry* akan dilakukan seketika, seperti yang ditunjukkan pada Gambar 6(a). Lebih lanjut, proses pencarian metadata sebuah koleksi yang diilustrasikan pada Gambar 7

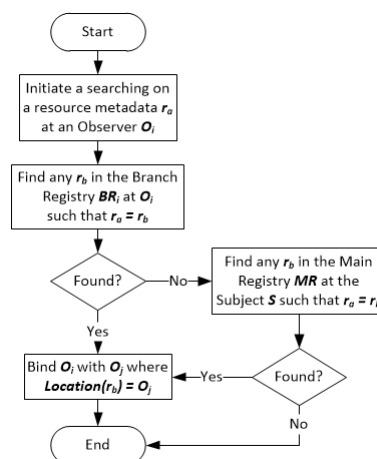
memungkinkan dilakukannya proses lanjutan berdasarkan metadata yang ditemukan, misalnya peminjaman antar perpustakaan. Skenario tersebut dirancang dalam rangka menjaga kinerja sistem tetap baik, khususnya yang berkaitan dengan proses pencarian informasi.



Gambar 5. Proses terkait eksistensi perpustakaan anggota dalam jejaring: (a) penambahan *node* baru; (b) penghapusan sebuah *node*



Gambar 6. Proses konsolidasi katalog: (a) publikasi seketika metadata sebuah koleksi yang baru ke *Main Registry*; (b) perbaruan periodik semua *Branch Registry*



Gambar 7. Proses pencarian metadata dari sebuah koleksi

6 PEMBUKTIAN KONSEP

6.1 Pengaturan eksperimen

Dalam rangka pembuktian konsep yang diajukan, arsitektur *Observable-SOA* telah diimplementasikan dalam sebuah lingkungan simulasi arsitektur perpustakaan digital terintegrasi nasional yang melibatkan beberapa perpustakaan anggota. Setiap perpustakaan diimplementasikan menggunakan sistem perpustakaan terintegrasi *open source* (ILS) tertentu, yaitu Koha², SLiMS³, dan OpenBiblio⁴, untuk merepresentasikan interoperasi dari beberapa perpustakaan dengan platform yang berbeda. Simulasi dilakukan dengan menggunakan 4 *node*, seperti ditunjukkan pada Gambar 4, yaitu 1 *node* merepresentasikan Perpustakaan Nasional dan 3 *node* merepresentasikan perpustakaan anggota.

Konfigurasi detail dari lingkungan eksperimen dijelaskan pada Tabel 1. Koleksi lokal awal mendeskripsikan semua koleksi yang dikelola secara fisik oleh setiap perpustakaan yang terlibat. Sementara, koleksi lokal baru menyatakan koleksi yang baru ditambahkan ke koleksi awal. Selanjutnya, semua kode koleksi yang terkait dengan judul koleksi dijelaskan secara detail pada Tabel 2.

Tabel 1. Konfigurasi detail dari lingkungan simulasi

Deskripsi	Nodes			
	L0	L1	L2	L3
ILS	Koha	SliMS	OpenBiblio	Koha
ILS Platform	Perl	PHP	PHP	Perl
OS	Debian 10	Centos7	Ubuntu 18.04	Ubuntu 20.04
Standar Katalog	MARC 21	MARC 21	MARC 21	MARC 21
Koleksi lokal awal (dengan kode)	5 koleksi: C-01, C-02, C-03, C-04, C-05	6 koleksi: C-11, C-12, C-13, C-14, C-15, C-16	4 koleksi: C-21, C-22, C-23, C-24	5 koleksi: C-31, C-32, C-33, C-34, C-35
Koleksi lokal baru (dengan kode)	1 koleksi: C-06	-	-	1 koleksi: C-36

² <https://koha.org/>

³ <https://slims.web.id/web/>

⁴ <http://obiblio.sourceforge.net/>

Skenario uji dibuat dalam 2 kasus uji (*TC*) berikut:

TC1 – Layanan pencarian berdasarkan data di Branch Registry. Berdasarkan Gambar 7, skenario ini terjadi ketika metadata koleksi yang dicari sudah tersedia di *Branch Registry* dari perpustakaan tempat pencarian diinisiasi. Metadata tersebut telah dimutakhirkan oleh *Subject* pada semua *Observer* yang terdaftar pada *Subject*, sehingga konten dari *Main Registry* serupa dengan konten dari setiap *Branch Registry*. Setiap perpustakaan telah diatur sehingga memiliki semua koleksi lokal awal sebagaimana dijelaskan dalam Tabel 1. Titik pencarian ditentukan pada *node L2*. Sedangkan metadata koleksi yang akan dicari merupakan koleksi dengan judul “*Concurrent programming*”.

Tabel 2. Deskripsi detail dari kode koleksi

Kode Koleksi	Judul Koleksi
C-01	Practical support for CMMI-SW software
C-02, C-15	Software engineering : practitioner's approach
C-03, C-24	Orchids and algebra
C-04, C-11, C-33	Concurrent programming
C-05, C-14, C-35	Programming Java
C-12, C-32	Java programming for the Internet
C-13, C-34	Java programming with JBuilder
C-16, C-31	The Java programming language
C-21	Elements of linear algebra
C-22	College algebra and trigonometry
C-23	Report of student performance
C-06, C-36	Karl Schwarzschild Meeting on Gravitational Physics

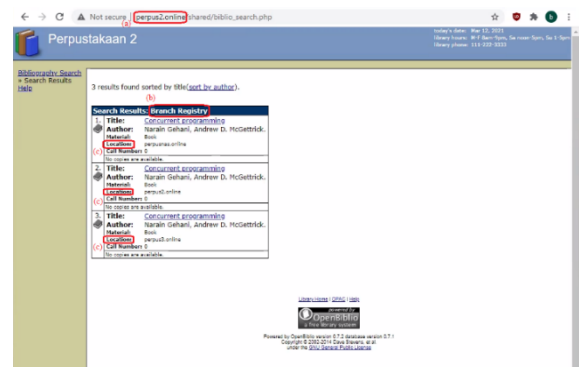
TC2 – Layanan pencarian berdasarkan data di Main Registry. Skenario ini mencakup beberapa kasus di mana metadata koleksi yang dicari belum tersedia di *Branch Registry* dari perpustakaan dimana pencarian diinisiasi. Sehingga, proses pencarian harus diteruskan ke *Main Registry*, berdasarkan algoritme pada Gambar 7. Metadata koleksi yang dicari tersebut belum dimutakhirkan ke setiap *Observer* yang terdaftar pada *Subject*, karena pendekatan pemutakhiran berkala/periodik yang digunakan. Koleksi baru yang berjudul “*Karl Schwarzschild Meeting on Gravitational Physics*” ditambahkan di *Main Registry* pada *node L0* (Perpusnas) dan di *Branch Registry* pada *node L3* (perpustakaan anggota). Sedangkan *node* lainnya tetap seperti dalam skenario pengujian *TC1*. Dalam skenario ini, metadata koleksi yang baru tersebut hanya akan tersedia di *Branch Registry* pada *node L3* dan di *Main Registry* pada *node L0*. Titik pencarian diatur pada *node L2*. Sedangkan metadata koleksi yang dicari adalah sebuah koleksi yang berjudul “*Karl Schwarzschild Meeting on Gravitational Physics*”.

6.2 Hasil dan pembahasan

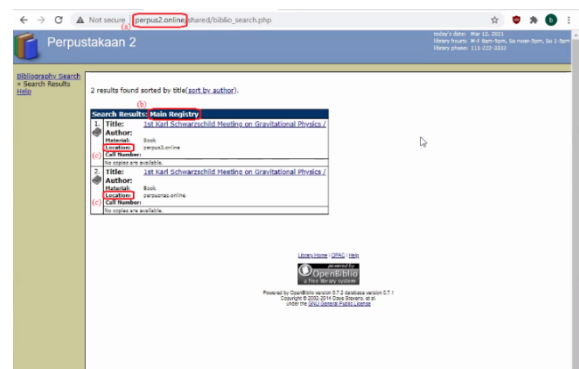
Gambar 8 merupakan tangkapan layar dari hasil eksperimen saat menjalankan skenario pengujian *TC1*, berdasarkan titik pencarian di perpustakaan *L2* (dinotasikan dengan huruf '(a)' – nama 'perpus2').

Karena metadata koleksi yang dicari telah diperbarui ke semua *node* termasuk *Branch Registry* di *node L2*, layanan pencarian berhasil dipenuhi di perpustakaan tempat pencarian diinisiasi *L2*, yaitu *Branch Registry* (dinotasikan dengan huruf '(b)'). Selengkapnya, metadata koleksi yang dicari bisa diketahui berada di *node L0, L1* dan *L3* (dinotasikan dengan huruf '(c)' – masing-masing dengan nama 'perpusnas', 'perpus2', dan 'perpus3'). Hasil ini sesuai dengan data yang dijelaskan pada Tabel 1 dan Tabel 2, yaitu koleksi dengan kode judul *C-04, C-11* dan *C-33*.

Hasil eksperimen untuk mengeksekusi skenario pengujian *TC2* ditunjukkan pada Gambar 9, berdasarkan titik pencarian di *node L2* (dinotasikan dengan huruf '(a)' – nama 'perpus2'). Pada skenario ini, metadata koleksi yang dicari belum dimutakhirkan ke semua *node* termasuk *Branch Registry* pada *node L2*, sehingga metadata yang dicari tidak berhasil ditemukan pada perpustakaan titik pencarian. Dengan demikian, pencarian harus diteruskan ke *Main Registry* pada *node L0*. Sebagai hasilnya, metadata yang dicari berhasil ditemukan di *Main Registry* (dinotasikan dengan huruf '(b)') beserta lokasinya, yaitu *node L0* dan *L3* (dinotasikan dengan huruf '(c)' – masing-masing dengan nama 'perpusnas' dan 'perpus3'). Hasil ini sesuai dengan data yang dijelaskan pada Tabel 1 dan Tabel 2, yaitu koleksi dengan kode judul *C-06* dan *C-36*.



Gambar 8. Hasil pencarian *TC1*



Gambar 9. Hasil pencarian *TC2*

Berdasarkan pengujian tersebut, algoritme yang dijelaskan pada Gambar 7 telah terbukti dapat digunakan secara efektif dalam arsitektur *Observable-SOA*. Arsitektur ini terbukti mampu

mengakomodasi perbedaan platform yang digunakan oleh setiap perpustakaan anggota sehingga bisa menjadi alternatif solusi arsitektur monolitik yang digunakan Perpustakaan saat ini tanpa harus menambahkan perangkat khusus lainnya. Namun demikian, beberapa isu perlu diperhatikan sehubungan dengan skenario pengujian *TC2* ketika titik pencarian diubah, misalnya menjadi *node L3*. *Node* ini merupakan lokasi dari metadata koleksi yang baru ditambahkan *C-36*. Jika skenario pengujian dijalankan dengan titik pencarian di *node L3*, hasil yang didapatkan hanya satu lokasi, yaitu *node L3* itu sendiri dengan nama '*perpus3*'. Hal ini terjadi karena layanan pencarian hanya cukup dilakukan di *Branch Registry* pada *node L3*, yaitu pada titik pencarian itu sendiri. Dengan hasil seperti itu, eksperimen bisa mendapatkan hasil yang berbeda pada skenario pengujian yang sama tetapi dengan titik pencarian yang berbeda. Situasi ini terjadi ketika periode pemutakhiran *Branch Registry* di setiap *Observer* yang terdaftar di *Subject* memiliki rentang yang cukup lama. Untuk itu, periode pemutakhiran yang tepat perlu dipertimbangkan dengan baik.

7 KESIMPULAN DAN PENELITIAN LANJUTAN

Arsitektur *Observable-SOA*, yang mengadopsi konsep SOA dan pola *Observer*, telah dibahas dan diuji dalam lingkungan simulasi untuk pengembangan jejaring Perpustakaan yang bertumpu pada KIN yang solid sebagai alternatif arsitektur monolitik yang digunakan Perpustakaan saat ini. Arsitektur ini mampu mengakomodasi perpustakaan dengan beragam platform berbeda untuk bisa berkolaborasi dalam jejaring Perpustakaan. Beberapa kebutuhan dasar sistem sudah didefinisikan sebagai dasar pertimbangan dalam mengembangkan arsitektur alternatif. Konsep arsitektur yang diusulkan, termasuk algoritme dari beberapa layanan dasar, telah berhasil diimplementasikan dan diuji dalam lingkungan simulasi menggunakan beberapa sistem perpustakaan terintegrasi *open source*.

Penelitian lanjutan akan dilakukan untuk mengimplementasikan arsitektur yang diusulkan pada lingkungan yang nyata yang melibatkan beberapa perpustakaan dengan beragam platform.

DAFTAR PUSTAKA

- ABDEEN, M. A., 2020. A Service-Oriented Architecture for Optimal Service. *International Journal of Advanced Computer Science and Applications*, 11(1), pp. 556-561.
- ARÉVALO, J., GONZALES, A. & ARÉVALO, L., 2019. Model for the interoperability between health service providers (IPS), based on the electronic clinical information standard (HL7) and architecture to design and develop distributed systems (SOA). In: *IOP Conference Series: Materials Science and Engineering*. IOP Publishing, pp. 012014.
- AZZEDIN, F. & GHALEB, M., 2019. Towards an Architecture for Handling Big Data in Oil and Gas Industries: Service-Oriented Approach. *International Journal of Advanced Computer Science and Applications*, 10(2), pp. 554-562.
- BUCHANAN, S., GIBB, F., SIMMONS, S. & MCMENEMY, D., 2012. Digital library collaboration: a service-oriented perspective. *The Library Quarterly*, 82(3), pp. 337-359.
- CHOWDHURY, G. G. & CHOWDHURY, S., 2003. *Introduction to digital libraries*. Facet publishing.
- DING, H. & SOLVBERG, I., 2004. *Exploiting extended service-oriented architecture for federated digital libraries*.
- ELSHATER, Y., MARTIN, P. & HASSANEIN, E., 2015. *Using design patterns to improve web service performance*.
- FAJAR, A. & SHOFI, I., 2019. Service Oriented Design for Indonesian E-Government System Using SOA. In: *IOP Conference Series: Materials Science and Engineering*. IOP Publishing, pp. 012106.
- FREEMAN, E., ROBSON, E., BATES, B. & SIERRA, K., 2004. *Head First Design Patterns*. O'Reilly Media, Inc..
- GANCHEVA, V., 2019. SOA Based Multi-Agent Approach for Biological Data Searching and Integration. *International Journal of Biology and Biomedical Engineering*, Volume 13, pp. 32-37.
- GANCHEVA, V. & BOROVSKA, P., 2019. SOA based system for big genomic data analytics and knowledge discovery. In: *2019 10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*. IEEE, pp. 536-541.
- HUSTAD, E. & OLSEN, D. H., 2021. Creating a sustainable digital infrastructure: The role of service-oriented architecture. *Procedia Computer Science*, Volume 181, p. 597-604.
- JOUNDREY, D. N., TAYLOR, A. G. & MILLER, D. P., 2015. *Introduction to Cataloging and Classification*. California: ABC-CLIO, LLC.
- KISMIYATI, T., 2018. *Perpustakaan sebagai Pusat Jejaring Perpustakaan di Indonesia*, Perpustakaan Nasional RI.
- LIU, Y., 2020. Design and implementation of multimedia teaching platform based on SOA architecture. *Multimedia Tools and Applications*, pp. 1-16.

- MACIS, S. et al., 2019. Design and usability assessment of a multi-device SOA-based telecare framework for the elderly. *IEEE Journal of Biomedical and Health Informatics*, 24(1), pp. 268-279.
- NGAOGATE, W., 2019. GoF Design Patterns in a Smart City System. *Journal of Software*, 14(5), pp. 220-226.
- PAPAZOGLU, M. P. & VAN DEN HEUVEL, W.-J., 2007. Service oriented architectures: approaches, technologies and research issues. *The VLDB journal*, 16(3), pp. 389-415.
- SINGH, M. P. & HUHNS, M. N., 2005. *Service-Oriented Computing*. West Sussex: John Wiley & Sons Ltd..
- WITTEN, I. H., BAINBRIDGE, D. & NICHOLS, D. M., 2010. *How to build a digital library*. Morgan Kaufmann.
- ZHAO, X. et al., 2012. SOA-based digital library services and composition in biomedical applications. *Computer methods and programs in biomedicine*, 106(3), pp. 219-233.

Halaman ini sengaja dikosongkan