

MODEL INTEROPERABILITAS ANTAR APLIKASI E-GOVERNMENT

Jazi Eko Istiyanto¹, Edhy Sutanta²

¹ Jurusan Ilmu Komputer, Universitas Gadjah Mada Yogyakarta

¹ Mahasiswa Program Doktor Ilmu Komputer, Universitas Gadjah Mada Yogyakarta

Masuk: 9 Oktober 2011, revisi masuk : 18 Januari 2012, diterima: 27 Januari 2012

ABSTRACT

Interoperability between information systems is an urgent problem to be solved in the development of e-Gov in Indonesia. This is caused by the need for increasing the data multisectoral policy-making in order to solve problems involving data from the inter-related sectors. While the state government application in the current environment, are still largely sectoral, in isolation, can not communicate with each other, and heterogeneous. Interoperability between e-Gov applications become important things to be sought the solution to the problem of developing e-Gov in Indonesia are not protracted. This paper is a review of the literature reveals the development of e-Gov in Indonesia, the interoperability problems encountered, and how the model of interoperability between e-Gov built to implement a web services models.

Keywords: e-Government, interoperabilitas, model, web services.

INTISARI

Interoperabilitas antar aplikasi sistem informasi menjadi tuntutan mendesak dalam pengembangan e-Gov di Indonesia saat ini. Hal ini disebabkan oleh adanya kebutuhan data multisektoral yang semakin meningkat dalam rangka pengambilan kebijakan untuk mengatasi problem yang melibatkan data dari antar sektor terkait. Sementara kondisi aplikasi di lingkungan pemerintah saat ini, umumnya masih bersifat sektoral, terpisah-pisah, tidak dapat saling berkomunikasi, dan heterogen. Interoperabilitas antar aplikasi e-Gov menjadi hal penting yang perlu segera dicari solusinya agar problem pengembangan e-Gov di Indonesia tidak berlarut-larut. Makalah ini merupakan hasil review pustaka yang mengungkap perkembangan e-Gov di Indonesia, problem interoperabilitas yang dihadapi, dan bagaimana model interoperabilitas antar aplikasi e-Gov dapat dibangun dengan mengimplementasikan model *web services*.

Kata-kata kunci : *e-Government, interoperabilitas, model, web services.*

PENDAHULUAN

Semakin besarnya peran teknologi informasi dan komunikasi (TIK) dalam proses bisnis membuat banyak lembaga berlomba mengimplementasikan TIK untuk proses terintegrasi. Salah satunya adalah melalui pengembangan *e-Government (e-Gov)*, di mana idealnya implementasi *e-Gov* diharapkan dapat membantu meningkatkan interaksi antara pemerintah, masyarakat, dan bisnis sehingga mendorong perkembangan politik dan ekonomi. Inisiatif tentang pengembangan *e-Gov* di Indonesia telah dikenalkan melalui Inpres No. 6 Tahun 2001, namun berdasarkan hasil evaluasi,

pengembangan *e-Gov* di Indonesia masih dijumpai banyak problem yang secara umum berpangkal dari kesalahan pandangan atau paradigma tentang *e-Gov*. Faktor teknis dan non teknis penghambat *e-Gov* juga telah diidentifikasi dalam beberapa kajian.

Interoperabilitas antar aplikasi juga menjadi tuntutan mendesak dalam pengembangan *e-Gov* di Indonesia. Hal ini disebabkan oleh kebutuhan data multisektoral dalam rangka pengambilan kebijakan untuk mengatasi problem yang melibatkan data dari antar sektor terkait. Sementara kondisi aplikasi di lingkungan pemerintah saat ini, umumnya masih

¹jazi@ugm.ac.id,

²edhy_sst@yahoo.com

bersifat sektoral, terpisah, tidak dapat saling berkomunikasi, dan heterogen. Interoperabilitas antar aplikasi *e-Gov* menjadi hal penting yang perlu segera dicari solusinya agar problem pengembangan *e-Gov* di Indonesia tidak berlarut-larut.

e-Government (e-Gov) adalah penggunaan teknologi informasi yang dapat meningkatkan hubungan antara pemerintah dan pihak-pihak lain, didalamnya melibatkan otomisasi dan komputerisasi pada prosedur *paper-based* yang akan mendorong cara baru dalam kepemimpinan, cara baru dalam mendiskusikan dan menetapkan strategi, cara baru dalam transaksi bisnis, cara baru dalam mendengarkan warga dan komunitas, serta cara baru dalam mengorganisasi dan menyampaikan informasi [Pascual, 2003]. Sementara Ahmadjayadi [2006] mengartikan *e-Gov* sebagai kegiatan yang terkait dengan upaya seluruh lembaga pemerintah dalam bekerja bersama-sama memanfaatkan teknologi informasi dan komunikasi (TIK), sehingga dapat menyediakan jasa layanan elektronik dan informasi yang akurat kepada individu masyarakat dan dunia usaha.

Semakin besarnya peran TIK dalam proses bisnis membuat lembaga pemerintah berlomba-lomba untuk mengimplementasikan TIK untuk proses terintegrasi. Salah satunya adalah melalui implementasi *e-Gov*, di mana idealnya implementasi *e-Gov* diharapkan dapat membantu meningkatkan interaksi antara pemerintah, masyarakat, dan bisnis sehingga mampu mendorong perkembangan politik dan ekonomi [Supangkat, 2006]. Mengacu pada Inpres No. 6 Tahun 2001 aparat pemerintah harus menggunakan teknologi telematika untuk mendukung *good governance* dan mempercepat demokrasi. Penerapan *e-Gov* di setiap lembaga pemerintah mengacu pada tahapan pengembangan *e-Gov* nasional sesuai dengan kondisi setiap lembaga pemerintah, meliputi [Inpres No. 3 Tahun 2003]: 1). Tingkat persiapan: pembuatan situs *web* di setiap lembaga pemerintah; pendidikan SDM, penyediaan sarana akses publik, sosialisasi keberadaan layanan informasi

elektronik untuk publik dan internal, pengembangan *e-leadership* dan *awareness building*, serta penyiapan peraturan. 2). Tingkat pematangan yaitu pembuatan situs informasi layanan publik interaktif; dan pembuatan *hyperlink*. 3). Tingkat pemantapan: penyediaan fasilitas transaksi elektronik; dan penyatuan aplikasi dan data dengan lembaga lain (interoperabilitas). 4). Tingkat pemanfaatan: pembuatan program layanan G2G, G2B, dan G2C terintegrasi; pengembangan proses untuk layanan *e-Gov* yang efektif dan efisien, dan penyempurnaan menuju kualitas *best practice*.

Administrasi publik adalah salah satu area di mana internet dapat digunakan untuk menyediakan akses layanan yang paling mendasar dan menyederhanakan hubungan masyarakat dan pemerintah bagi masyarakat. *e-Gov* dengan memanfaatkan layanan internet dapat dibagi dalam beberapa tingkat yaitu: penyediaan informasi, interaksi satu arah, dan interaksi dua arah dan transaksi (layanan elektronik penuh). Interaksi satu arah bisa berupa fasilitas *download* formulir yang dibutuhkan, pengumpulan formulir secara *online* merupakan contoh interaksi dua arah, sedangkan contoh layanan elektronik penuh adalah pengambilan keputusan dan proses pelayanan pembayaran [<http://dishubkominformo.belitungkab.go.id/>, 08-03-2012].

Hasil evaluasi pengembangan *e-Gov* di Indonesia menunjukkan masih dijumpai banyak problem, namun secara umum problem tersebut berpangkal dari kesalahan pada pandangan atau paradigma tentang *e-Gov* [Supangkat, 2006]. Setidaknya, terdapat delapan persepsi keliru tentang pengembangan *e-Gov* selama ini, yaitu [Nugroho, 2008]: 1). *e-Gov* adalah situs *web* lembaga pemerintah. Persepsi ini akan mereduksi makna *e-Gov*. 2). *e-Gov* adalah adanya ketersediaan infrastruktur. Infrastruktur tidaklah identik dengan *e-Gov*. Tujuan *e-Gov* adalah menumbuhkan kekuatan pemberdayaan, dan infrastruktur harus dimanfaatkan untuk tujuan yang lebih besar. 3). *e-Gov* adalah pembangunan sistem-sistem informasi. Akar

permasalahannya adalah tidak selarasnya antara sistem-sistem yang terus akan dikembangkan dengan proses-proses birokrasi yang dilakukan sehari-hari, masing-masing berjalan sendiri-sendiri, sehingga tujuan dasar sistem-sistem informasi untuk mendukung proses penyelenggaraan pemerintahan tidak optimal. 4). *e-Gov* harus dikembangkan secara bertahap karena alasan keterbatasan biaya. Keterbatasan biaya adalah kekangan, bukan alasan dasar untuk melakukan pentahapan proses pengembangan *e-Gov*. Sedangkan alasan sebenarnya adalah perbedaan tingkat kesiapan masyarakat dalam pemanfaatan adanya TIK dan kesiapan pengembangan *e-Gov*. Ketersediaan rencana jangka panjang/induk, *blueprint*, dan variasinya sangat penting sebagai *guideline* bagi pentahapan proses pengembangan *e-Gov*. 5). *e-Gov* adalah pembangunan sistem informasi di berbagai SKPD. Saat ini, integrasi informasi menjadi syarat penting bagi terselenggaranya kegiatan program pemerintahan. Keberadaan sistem informasi-sistem informasi di SKPD tidak banyak berarti jika tidak diikuti dengan integrasi antar sistem. Integrasi ini mengidentifikasi jalur akses data dan informasi antar sistem untuk memenuhi kebutuhan informasi multisektor. 6). *e-Gov* hanya memerlukan SDM TIK. Fakta menunjukkan bahwa implementasi *e-Gov* tidak akan berhasil tanpa *leadership* dan kemampuan manajerial yang baik. 7). *e-Gov* adalah mahal. *e-Gov* adalah mahal jika investasi (infrastruktur, sistem informasi, dan lainnya) tidak bisa memenuhi sasaran, sebaliknya investasi menjadi tidak berarti jika implementasi TIK mampu menumbuhkan *multiplier effect* dengan *outcome* yang jauh lebih bernilai dibandingkan investasi. 8). Sasaran *e-Gov* adalah tuntasnya implementasi TIK. TIK hanyalah alat bantu untuk mencapai tujuan yang lebih besar, yaitu peningkatan pemberdayaan masyarakat, kualitas layanan publik, transparansi, akuntabilitas, dan efisiensi penyelenggaraan pemerintahan.

Faktor penghambat implementasi *e-Gov* di Indonesia juga berhasil diidentifikasi, antara lain disebabkan oleh

[Raharjo, 2001] :1).Rendahnya komitmen pemerintah dalam integrasi dan juga transparansi publik, 2). Minimnya budaya berbagi informasi. 3). Minimnya budaya tertib dokumentasi. 4). Resistensi perubahan. 5). Kelangkaan SDM yang handal. 6). Infrastruktur belum memadai dan mahal. 7). tempat akses yang terbatas.

Ketidaktelitian pengembangan *e-Gov* di Indonesia disebabkan oleh: 1) *master plan* tidak mempunyai kekuatan formal, karena baru berupa kajian sehingga posisi strategis *e-Gov* sulit terealisasi secara optimal dan bukan merupakan kewajiban unit-unit, 2) organisasi pengambil keputusan tidak cukup kuat memastikan integrasi arsitektur, pengelolaan portofolio, dan eksekusi proyek TIK tahunan, dan 3) lemahnya koordinasi antar unit dalam rencana tahunan proyek TIK [Supangkat et. al, 2007].

Aspek non teknis, yaitu resistensi terhadap perubahan juga menghambat penerapan *e-Gov*. Problem ini terjadi karena [Indrajit, 2006]: 1). Ego sektoral lembaga masih tinggi sehingga menutup kemungkinan diatur atau bekerjasama dengan lembaga lain, 2). Anggapan bahwa sistem informasi di lembaga sendiri adalah terbaik dibanding lainnya, 3). Konteks kepentingan yang berbeda di setiap lembaga sehingga sulit dilakukan integrasi. 4).Keinginan menjadi pemimpin dalam integrasi. 5). Ketidakinginan saling membagi data dan informasi karena mengurangi keunggulan kompetitif. 6). Ketidaktahuan dari mana harus memulai integrasi sehingga kondusif untuk dilakukan sejumlah pihak terkait; dan sebagainya

Persaingan bisnis yang terus meningkat di benua Eropa di tahun 1990-an mengakibatkan banyak perusahaan melakukan integrasi antar perusahaan agar tetap bertahan dalam persaingan. Upaya integrasi ini menghadapi problem dan melahirkan konsep *enterprise interoperability*. Konsep tersebut diyakini lebih sesuai diterapkan dalam lingkungan perusahaan yang tersebar dan lebih fleksibel. Pendekatan *enterprise interoperability* memerlukan biaya relatif yang jauh lebih murah dan dapat

diimplementasikan lebih cepat [Shorter, 1997]. Pertimbangan yang mendukung pernyataan tersebut adalah: 1) integrasi melalui pemodelan perusahaan terpadu, seperti CIMOSA [Shorter, 1997], atau 2) integrasi sebagai pendekatan metodologi untuk mencapai konsistensi pengambilan keputusan pada perusahaan besar, seperti usulan metodologi GRAI [Doumeingts et al.1998].

Michel [1997] menyatakan bahwa integrasi dapat diperoleh dalam tiga hal, yakni: data (model data), organisasi (model sistem dan proses), dan juga komunikasi (model pada jaringan komputer, misal model OSI). Integrasi total hanya akan terjadi pada perangkat lunak atau sistem itu sendiri. Integrasi dapat diperoleh dengan dua cara, yaitu: 1) penyatuan dengan standarisasi (standarisasi metode, arsitektur, konstruksi, dan bagian model yang dapat digunakan kembali), atau 2) federasi (standarisasi *interface*, model referensi atau *ontologi*). Sementara Chen dan Vernadat [2004] menyatakan bahwa integrasi dapat dilakukan dengan cara berbagai cara disesuaikan dengan kepentingannya, namun pada dasarnya dapat diklasifikasikan dalam tiga tingkatan, yakni: 1) integrasi fisik (interkoneksi perangkat, mesin NC, PLC, melalui jaringan komputer), 2) integrasi aplikasi (berkaitan dengan adanya interoperabilitas *software* aplikasi dan *database* dalam lingkungan komputasi yang heterogen), dan 3) integrasi bisnis (koordinasi fungsi untuk mengelola, mengontrol, dan memonitor proses bisnis).

Terkait dengan perkembangan tentang integrasi dan interoperabilitas, pada tahun 2007 didominasi oleh pemodelan berbasis model, sedangkan tren masa depan lebih mengarah pada interoperabilitas sistem yang *loosely-coupled* dan meninggalkan solusi seragam yang memerlukan biaya tinggi dan keberhasilan integrasi yang rendah. Tantangan utama interoperabilitas sistem yang *loosely-coupled* adalah bagaimana mengembangkan model dan metodologi yang mengarah ke berbagai solusi interoperabilitas antar sistem yang diintegrasikan dengan lebih banyak

mempertimbangkan aspek kebutuhan teknologi dan semantik konsep yang akan dipertukarkan dan dipahami secara bulat oleh semua pihak Molina et.al [2007]. Area riset terkait dengan solusi kebutuhan integrasi antar sistem, terdiri atas: 1). Kolaborasi jaringan organisasi, 2). pemodelan perusahaan dan model referensi, 3). interoperabilitas model perusahaan dan proses, 4). validasi, verifikasi, kualifikasi, dan akreditasi model perusahaan, serta 5). penggunaan kembali model dan repositori [Molina et.al, 2007].

Terkait dengan interoperabilitas antar aplikasi, telah dikembangkan sebuah model integrasi B2B berbasis SOA menggunakan layanan *web services* dengan studi kasus pada aplikasi *e-shop* yang mengintegrasikan situs Amazon, eBay, Yahoo!, dan Paypal. Model ini dikembangkan untuk mengatasi problem ketidakselarasan antara sistem bisnis dan sistem informasi yang berkembang dengan kecepatan yang berbeda-beda. Model yang akan dikembangkan membuka peluang untuk membawa definisi *service* ke level abstraksi yang lebih tinggi, berupa model level tinggi yang dapat ditransformasikan ke implementasi *services* yang bebas *platform*. Dengan pendekatan ini dapat dipisahkan antara *platform* terendah, infrastruktur, dan implementasinya, dengan harapan mampu meningkatkan integrasi. Penggunaan gabungan metode ini memungkinkan penggunaan ulang dan peningkatan integrasi pada level model yang berbeda. Metode baru berupa metode integrasi berbasis SOA yang dikembangkan dari metode SOAD dan mBPDM berhasil ditemukan dan telah diujikan untuk melakukan analisis dan perancangan integrasi berbasis SOA, serta diimplementasikan menjadi 16 proses bisnis, 18 *web service*, serta 6 aplikasi komposit [Utomo, 2011].

Penelitian lain telah berhasil mengembangkan model interoperabilitas sistem informasi layanan publik pada aplikasi *e-Gov* di Indonesia dengan studi kasus pada proses Sistem Informasi Kependudukan dan Perpajakan. Pada penelitian dilakukan untuk mengatasi problem kurangnya perhatian setiap

instansi pemerintah untuk berbagi akses data dan informasi, serta belum adanya model interoperabilitas antar sistem informasi yang secara eksplisit dijelaskan dalam Inpres No. 3 Tahun 2003 tentang Kebijakan dan Strategi Nasional Pengembangan e-Gov, maupun pada *blueprint* aplikasi e-Gov. Model interoperabilitas yang dikembangkan berdasarkan model *web services* dengan metode REST. Kebutuhan akan adanya interoperabilitas antar sistem informasi dipetakan berdasarkan keterkaitan antar skema *database* dengan menggunakan model *web services*, sehingga diperoleh model interoperabilitas antara dua sistem. Perancangan metode REST dilakukan menggunakan ROA yaitu model arsitektur yang berorientasi sumber daya informasi. Implementasi model interoperabilitas antar dua sistem informasi dilakukan menggunakan bahasa PHP dan *database* MySQL. Pengujian dilakukan melalui pengambilan data antar kedua sistem yang memiliki perbedaan *platform database* dan terletak pada lokasi fisik yang berbeda [Sukyadi, 2009].

Web services adalah sebuah *software* sistem yang dirancang untuk mendukung interoperabilitas interaksi mesin-ke-mesin melalui sebuah jaringan [WWW Consorsium, 2004]. *Web services* secara teknis memiliki mekanisme penunjang interoperabilitas antar aplikasi dengan melakukan interaksi antar aplikasi, baik yang berupa agregasi (pengumpulan) maupun adanya sindikasi (penyatuan). *Web services* juga memiliki layanan terbuka untuk kepentingan integrasi data dan kolaborasi informasi yang bisa diakses melalui internet oleh berbagai pihak melalui teknologi yang dimiliki masing-masing pengguna.

Sekalipun mirip dengan proses *Application Programming Interface* (API) berbasis *web*, *web services* memiliki keunggulan karena dapat dipanggil dari jarak jauh melalui internet, pemanggilan *web services* bisa menggunakan bahasa pemrograman apa saja, dan dalam *platform* apa saja, sementara API hanya bisa digunakan untuk *platform* tertentu [Lucky, 2008]. *Web services* dapat dipahami sebagai *Remote Procedure*

Call (RPC) yang mampu memproses fungsi-fungsi program yang didefinisikan pada sebuah aplikasi *web* dan mengekspos sebuah API atau *User Interface* (UI) melalui *web*. Kelebihan penggunaan *web services* adalah: 1) lintas *platform*, 2). *language independent*, 3). jembatan penghubung dengan *database* tanpa perlu *driver database* dan tidak perlu tahu jenis DBMS, 4). mempermudah proses pertukaran data, dan 5). penggunaan kembali komponen aplikasi [Lucky, 2008]. Layanan fungsional *web services* dilakukan dalam empat langkah kegiatan, 3 diantaranya dilakukan oleh *web services*, yaitu: 1). entitas pengguna (*requester entity*), 2). entitas penyedia (*provider entity*), dan 3). entitas perantara (*discovery entity*) [Sukyadi, 2009].

Berdasarkan konsep hubungan dan penyampaian informasi, *web services* dikembangkan melalui 4 model arsitektur, masing-masing memiliki orientasi pada *message*, *action*, *resource*, dan *policy*. Pengembangan model yang diturunkan berdasarkan orientasi pada *action* (*Service Oriented Model/SOM*) menghasilkan *Services Oriented Architecture* (SOA), yaitu model arsitektur berbasis layanan. Sementara pengembangan model yang diturunkan berdasarkan orientasi pada *resource* (*Resource Oriented Model/ROM*) yang menghasilkan adanya *Resource Oriented Architecture* (ROA), yaitu model arsitektur berbasis sumberdaya informasi [Sukyadi, 2009]. Dalam proses perkembangannya, model *web services* memiliki dua metode yang berorientasi pada layanan dan sumberdaya informasi, yaitu: SOAP (*Simple Object Access Protocol*) dan REST (*REpresentational State Transfer*). Implementasi *web services* model SOA telah banyak dilakukan dan dikembangkan oleh banyak *vendor*, seperti Microsoft, Sun dan IBM, melalui dukungan *platform* infrastruktur *dotNet* dan Java. Arsitektur SOAP memiliki tiga komponen utama dalam melakukan proses layanan yaitu: 1) *service provider*, 2) *service requester*, dan 3) *service broker*, serta komponen pendukung yaitu: 1). XML, 2) SOAP-XML (terdiri atas *header* dan *body*), 3). WSDL,

serta 4). UDDI [DSIPLK, 2008]. Metode REST diusulkan oleh Fielding [2000] dengan didasari oleh empat prinsip utama teknologi, yaitu: 1). *Resource identifier through Uniform Resource Identifier* (URI), 2). *uniform interface* (sumberdaya CRUD menggunakan operasi-operasi PUT, GET, POST, dan DELETE), 3). *self-descriptive messages* (sumberdaya tidak terikat sehingga dapat mengakses konten HTML, XML, PDF, JPEG, plain text, meta data, dll), serta 4). *stateful interactions through hyperlinks* (bersifat *stateless*) [Pautasso, 2008]. Metode REST lebih sederhana karena menggunakan format standar (HTTP, HTML, XML, URI, MIME), namun jika diperlukan proses pengambilan data, maka konten berupa teks dari hasil eksekusi *web services* dapat diolah dalam format teks seperti XML atau HTML dengan menggunakan utilitas komunikasi data melalui koneksi *socket* protokol HTTP. Utilitas ini umumnya tersedia dalam pustaka komunikasi pada bahasa pemrograman seperti Java, Visual Basic, Delphi, PHP, ASP, maupun JSP [Sukyadi, 2009].

Perbedaan mekanisme pada metode proses SOAP dan REST adalah [Sukyadi, 2009]: 1). *Protocol layering* dimana metode REST menganggap penggunaan protokol HTTP sebagai *application-level protocol*, sedangkan pada SOAP menganggap penggunaan protokol, khususnya HTTP sebagai *transport-level protocol*. 2). *Dealing with heterogeneity* metode SOAP dan REST memiliki kesamaan dalam penanganan keragaman komponen pada protokol HTTP, namun berbeda dalam dukungan antar *vendor* aplikasi *browser* dan *enterprise computing*. 3). *Loose coupling* dalam aspek *time/availability* dan *location transparency*, metode REST dan SOAP memiliki kecenderungan menjadi *loose coupling* (bebas ketergantungan akses), namun dalam aspek *service evolution*, metode REST memiliki *loose coupling* yang lebih tinggi daripada SOAP. Metode REST menggunakan format URI yang bebas dari bentuk format deskripsi, sedangkan dalam metode SOAP, perubahan yang terjadi dalam struktur fungsi *web services* akan mempengaruhi

deskripsi *web services* di dalam WSDL, sehingga WSDL harus dirubah untuk menyamakan struktur dan tipe datanya. Akibatnya metode SOAP lebih kompleks dalam pengembangannya, sementara REST dapat dilakukan lebih cepat dan sederhana.

Implementasi SOA dalam e-Gov memungkinkan *sharing* informasi dapat dilakukan tanpa memberi hak akses secara langsung ke *database* bagi pihak yang mengaksesnya. Pengakses informasi pada aplikasi e-Gov di instansi pemerintah lain dapat menggunakan *service* yang disediakan oleh aplikasi e-Gov yang mendukung konsep-konsep interoperabilitas. Secara teknis SOA memisahkan antara pesan/*query/call* dengan bagian pengolahan *database*, sehingga wilayah privat dan publik dapat terpisah secara tegas. Bagian privat hanya dapat diakses oleh bagian penanggungjawabnya, sementara bagian publik dapat diakses oleh siapa pun melalui *service* yang disediakan. Agar pesan/*query/call* dapat digunakan oleh pengguna lain, maka harus disusun berdasarkan standar tertentu tanpa bergantung pada produk TIK tertentu [<http://arvantc40s.blogspot.com/2012/02/interoperabilitas-data-dalam-e.html>, 08-03-2012].

Sebagai contoh implementasi SOA, *database* kependudukan di Indonesia merupakan tanggungjawab Dinas Dukcapil dan dapat menyediakan *service* ke publik berupa informasi data *series* jumlah penduduk berdasarkan pendidikan, pekerjaan, dan lainnya. Sedangkan untuk pengguna lembaga pemerintah, *service* dapat diperluas dengan pemberian informasi yang lebih lengkap seperti nama, alamat, tanggal lahir, status dan lainnya, sehingga Dinas Kesehatan misalnya, dapat dengan memanfaatkannya untuk membangun informasi kesehatan bagi berbagai kepentingan. Melalui *sharing* informasi tersebut, data pokok (misal biodata penduduk) dapat dilengkapi dengan berbagai atribut yang dibutuhkan dalam sistem lain. Misal, seseorang dengan Nomor Induk Kependudukan (NIK) tertentu, dapat dilengkapi dengan atribut data kesehatan di Dinas Kesehatan,

dilengkapi dengan atribut data pendidikan di bagian Dinas Pendidikan, dilengkapi dengan atribut data kepemilikan barang di Dinas Pajak. Pada akhirnya, pemerintah akan memiliki data yang sangat lengkap tentang setiap penduduk Indonesia.

Interoperabilitas antar aplikasi *e-Gov* lahir akibat meningkatnya kebutuhan informasi multisektor, sementara di sisi lain ada tuntutan independensi dan *loose coupling* antar aplikasi. Secara logika dimungkinkan membangun satu aplikasi *e-Gov* yang mampu menampung semua data negara, namun secara fisik hal ini sulit diterapkan. Struktur pemerintahan Indonesia (pusat dan daerah) disusun atas beberapa sektor untuk tujuan profesionalisme. Untuk itu setiap instansi diharapkan membangun aplikasi *e-Gov* sesuai wilayah kerjanya, namun harus dibarengi dengan semangat *sharing* informasi ke publik dan antar lembaga, sehingga pengembangan *e-Gov* dapat dilaksanakan sesuai dengan konsep yang benar yaitu mampu memberikan layanan elektronik penuh pada tiga ranah, yaitu G2G, G2B, dan G2C.

METODE

Makalah ini merupakan hasil review kritis atas pustaka yang relevan mengungkap perkembangan *e-Gov* di Indonesia, problem interoperabilitas, dan bagaimana model interoperabilitas antar aplikasi *e-Gov* dapat dibangun dengan mengimplementasikan pada model *web services*. Model interoperabilitas antar aplikasi *e-Gov* ditunjukkan dengan memanfaatkan model *web services* yang meliputi: model pemetaan proses pengambilan data antar aplikasi *e-Gov*; proses akses data menggunakan fungsi *remote* untuk pengambilan data antar aplikasi *e-Gov*; model infrastruktur interoperabilitas antar aplikasi *e-Gov*, serta implementasi pada model interoperabilitas antar aplikasi *e-Gov*.

PEMBAHASAN

Kebijakan pemerintah Indonesia tentang *blueprint* aplikasi *e-Gov* yang menggambarkan blok-blok fungsi-fungsi layanan, administrasi, dan kelembagaan yang disusun menjadi sebuah bagan

fungsi yang disebut Kerangka Fungsional Sistem Pemerintahan [Direktorat *e-Gov*, 2004]. Kerangka Fungsi Sistem Pemerintahan tersebut terdiri atas empat kelompok dinas dan lembaga, yaitu: 1). pemerintahan (meliputi pengelolaan barang daerah, katalog barang daerah, pendapatan daerah, perusahaan daerah; 2). kewilayahan (meliputi tata ruang dan lingkungan hidup; potensi daerah; kehutanan; pertanian, peternakan, perkebunan; perikanan dan kelautan; pertambangan dan energi; pariwisata, IKM); 3). kemasyarakatan (meliputi kesehatan, pendidikan, industri dan perdagangan, jaring pengaman sosial); serta 4) sarana dan prasarana (meliputi transportasi, jalan, jembatan, terminal, pelabuhan, sarana umum) [Direktorat *e-Gov*, 2004].

Dalam Kerangka Fungsional Sistem Pemerintahan, aplikasi *e-Gov* disusun dan dikelompokkan berdasarkan fungsi dan layanannya menjadi sebuah sistem kerangka arsitektur yang disebut Peta Solusi Aplikasi *e-Gov*. Aplikasi diklasifikasikan dengan pendekatan matrik orientasi fungsi layanan dan sifat fungsi aplikasi yang meliputi: 1) aplikasi layanan ke pengguna (*front office*), 2) aplikasi untuk pekerjaan administrasi pemerintahan dan fungsi-fungsi kedinasan/kelembagaan (*back office*), dan 3) kelompok aplikasi yang bersifat mendasar/umum (*back office*). Masing-masing kelompok aplikasi tersebut, dibagi ke dalam tiga sub kelompok berdasarkan orientasi pengguna yang dilayaninya, yaitu: 1) aplikasi pada *Government to Citizen* (G2C), 2) *Government to Business* (G2B), dan *Government to Government* (G2G). Sedangkan standar kebutuhan untuk mengembangkan sebuah aplikasi *e-Gov* adalah: 1) *reliable*, 2) *interoperable*, 3) *scalable*, 4) *user friendly*, serta 5) *integrateable* [Direktorat *e-Gov*, 2004].

Interoperabilitas antar aplikasi *e-Gov* merupakan tuntutan yang semakin mendesak sebagai akibat dari adanya: 1) kebutuhan untuk melakukan pertukaran informasi secara cepat dan akurat, 2) kebutuhan untuk *upgrade* dan migrasi *software*, dan 3) kebutuhan data pada multisektoral [Nugroho, 2008]. Proses

pemenuhan kebutuhan data-data pada multisektoral, setidaknya menghadapi tiga masalah, yaitu: 1). Masalah utama pada format data, 2). Masalah mekanisme pertukaran, dan 3). Masalah karena tidak semua instansi bersedia membuka detail internal aplikasinya ke pihak lain, dengan alasan keamanan data [Nugroho, 2008]. Aspek yang terkait dengan problem interoperabilitas adalah Miller [2000]: 1). Teknik, meliputi standar komunikasi, pemindahan, penyimpanan, dan penyajian data; 2). Semantik, yakni standar penggunaan istilah untuk indeks dan temu kembali; 3). Politis/manusia, yakni keputusan untuk berbagi dan bekerjasama; 4). Interkomunitas, yakni kesepakatan berhimpun antar lembaga dan disiplin ilmu; 5). Legal, terkait dengan peraturan akses koleksi digital dan peraturan HAKI; 6). Standar internasional, yaitu standar yang memungkinkan kerjasama internasional.

Kesulitan interoperabilitas antar aplikasi e-Gov juga diakibatkan oleh: 1). tidak dimilikinya dokumentasi sistem, 2). belum tersedianya kamus data (*data dictionary*) yang jelas, 3). Adanya perbedaan persepsi tentang konsep interoperabilitas, 4). Belum dikenalnya interoperabilitas sistem informasi, 5). Belum merasa membutuhkan adanya interoperabilitas sistem informasi, serta 6). belum menyadari perlunya *sharing data* <http://arvantc40s.blogspot.com/2012/02/interoperabilitas-data-dalam-e.html>, [08 Maret 2012].

Sedangkan Setyantana [2009] menyatakan bahwa adanya problem interoperabilitas antar aplikasi e-Gov di Indonesia adalah terdapat banyak aplikasi yang dikembangkan secara terpisah dan tidak terintegrasi, bahkan banyak yang akan dikembangkan tidak menggunakan metode sistem *database* (RDBMS), sehingga sulit diintegrasikan dengan sistem lain. Sistem juga dikembangkan dengan teknologi tertutup yang berbeda, seperti FoxPro, dBase, Visual Basic, Delphi, PowerBuilder Terhadap problem tersebut, Setyantana [2009] menyampaikan usulan solusi sebagai berikut: 1). Perlu solusi untuk pertukaran data dan informasi antar sistem. 2). Solusi berarsitektur terbuka

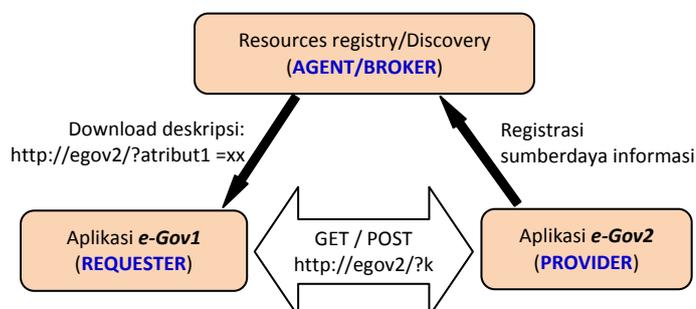
dan memungkinkan interoperabilitas. 3). Lebih diutamakan memakai OSS. 4). Setiap wali data menyediakan data yang dibutuhkan lembaga lain. 5). Perlu ada kebijakan agar aplikasi yang dibangun oleh satu lembaga yang siap berinteroperabilitas dengan aplikasi di lembaga lain (dengan menyediakan *service*).

Interoperabilitas antar aplikasi e-Gov tidak sekedar untuk dipahami sebagai persiapan oleh suatu lembaga sentral yang bekerja memfasilitasi, mendiktekan spesifikasi, dan proses baku. Namun, interoperabilitas harus merupakan konsensus implementasi bersama kerangka kerja (*framework*) yang telah ditetapkan, dan setiap lembaga tetap berwenang membuat keputusan terkait pemilihan *hardware* dan *software* yang digunakan. Tujuan akhir yang ingin dicapai dari solusi interoperabilitas antar aplikasi e-Gov adalah terbentuknya Sistem Informasi yang terintegrasi dan dapat saling berkomunikasi.

Mengacu pada konsep-konsep interoperabilitas menggunakan *web services*, maka aplikasi-aplikasi yang akan dikomunikasikan dapat dipetakan berdasarkan fungsi dan perannya dalam proses pertukaran data. Pertukaran data ini akan melibatkan tiga entitas, yaitu: 1). *Provider entity* sebagai penyedia sumber daya data dan informasi, 2). *requester entity* sebagai pengakses sumber daya informasi. 3). *Agent/broker* yang akan bertugas mengelola dan menyediakan fasilitas untuk registrasi, publikasi, dan penemuan sumber daya informasi. *Agent/broker* dapat disediakan oleh *provider* atau pihak lain yang memiliki fasilitas layanan publik yang sudah tersosialisasi dan diketahui oleh publik. Dengan mengacu pada model yang dikembangkan oleh Sukyadi [2009], pemetaan *web services* yang akan menggambarkan model interoperabilitas pada proses pengambilan data dari dua aplikasi e-Gov, yaitu e-Gov1 ke e-Gov2 ditunjukkan pada Gambar 1. Dalam Gambar 1, mekanisme akses data dari e-Gov1 ke e-Gov2 dilakukan melalui fungsi *web services* yang dipublikasikan, jadi tidak langsung mengakses ke *database*

dalam aplikasi yang diakses, sehingga proses pertukaran data aman dilakukan. Mekanisme tersebut juga tidak

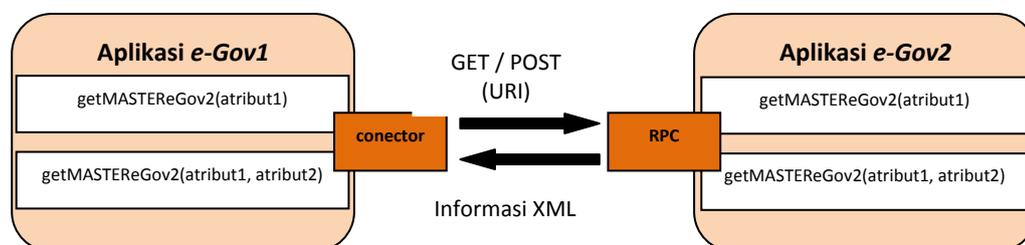
mengharuskan pengguna mengetahui *platform (engine driver)* pada *database* yang diakses.



Gambar 1: Pemetaan *web services* proses pengambilan data dari *e-Gov1* ke *e-Gov2*

Keunggulan REST yang telah menggunakan format URI sehingga bebas dari bentuk format deskripsi, dapat mempercepat proses pengembangan dan lebih sederhana dibandingkan dengan metode SOAP. Karena itu, metode REST akan digunakan sebagai metode proses dalam *web service* untuk interoperabilitas antar aplikasi *e-Gov*. Kasus kebutuhan akses data dibedakan menjadi 2, yaitu proses mengambil satu data dan proses mengambil sekelompok

data secara kolektif. Dengan menggunakan model pada Gambar 1, kedua proses akses data dapat dilakukan melalui *web services* pada aplikasi *e-Gov1* dengan menyediakan fungsi *remote* untuk mengambil satu data pada *atribut1* disebut **getMASTEReGov2** dan fungsi *remote* untuk mengambil sekelompok data pada *atribut1* dan *atribut2* disebut **getMASTERSGov2**. Mekanisme untuk masing-masing proses diilustrasikan pada Gambar 2.



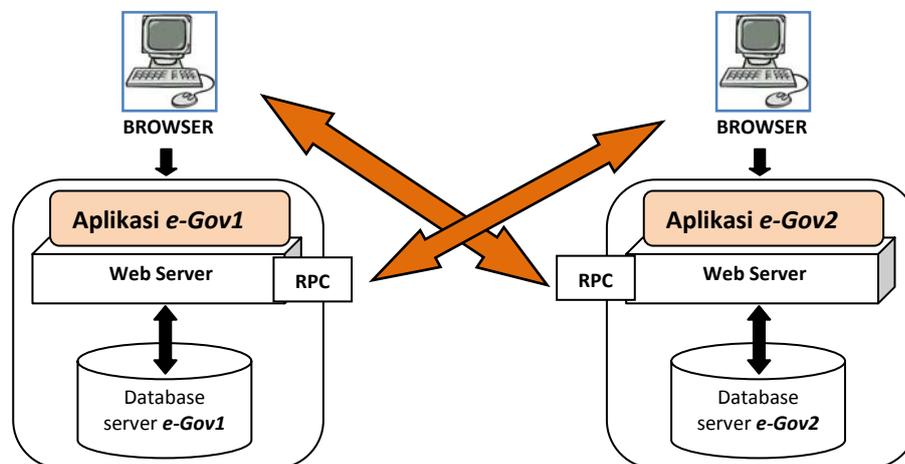
Gambar 2: Mekanisme pengambilan data dari *e-Gov1* ke *e-Gov2* dengan metode REST

Hubungan secara fisik mekanisme proses pertukaran antara aplikasi *e-Gov1* dan *e-Gov2* melalui *web services* dapat diperjelas menggunakan model infrastruktur *web services*. Model ini menjelaskan bahwa setiap aplikasi memiliki sumberdaya informasi yang bersifat publik dan dapat diakses oleh aplikasi lain yang membutuhkan. Setiap ada *request* data dari aplikasi lain melalui *web services*, maka *web services* dalam aplikasi *provider* akan melakukan pengolahan ke *database* internal. Fungsi-fungsi layanan tersebut dapat berada dalam aplikasi atau diletakkan dalam

modul lain sehingga dapat dipakai secara bersama. Pendekatan yang dilakukan bisa melalui proses eksekusi fungsi secara langsung yang dilewatkan melalui parameter URI (dengan mencantumkan nama fungsi yang akan diproses). Alternatif lain adalah proses eksekusi fungsi secara tidak langsung melalui proses interpretasi fungsi ke dalam nama umum, sehingga pada saat nama umum fungsi dilewatkan melalui parameter URI akan diterjemahkan dahulu ke dalam daftar nama fungsi yang bersifat internal. Cara kedua ini biasa digunakan untuk menjaga keamanan data dari akses

pihak luar yang ingin mengakses secara langsung ke dalam fungsi-fungsi internal. Pada Gambar 3 menampilkan model

infrastruktur interoperabilitas antara aplikasi e-Gov1 dan e-Gov2.



Gambar 3: Model infrastruktur interoperabilitas antara aplikasi e-Gov1 dan e-Gov2

Rancangan model interoperabilitas antar aplikasi e-Gov yang menggunakan model arsitektur *web services* menggunakan metode REST, terdiri dari tiga rancangan yaitu Rancangan *provider*, Rancangan *agent/broker* dan Rancangan *requester*.

Rancangan *provider*, meliputi:

- 1). Menentukan fungsi sumberdaya informasi publik yang meliputi: Membuka koneksi *database*. Jika koneksi berhasil, lakukan akses data sesuai kriteria dan kembalikan nilai fungsi dalam *array*. Jika koneksi gagal, kembalikan nilai fungsi dengan nilai *false*.
- 2). Menentukan nama sumberdaya informasi publik pada *web services*, yaitu struktur format parameter URI acuan sebagai deskripsi *web application description language (WADL)*, dengan parameter: a) *resource*, b) format input (XML atau TXT), c) format output (XML, TXT, atau HTML), dan d) data input.
- 3). Membuat *script* program aplikasi *web services* (sebagai RPC) untuk mengolah *request* parameter URI (WADL) yang dikirim oleh *requester* agar dapat melakukan validasi dan memberikan hasil sesuai kriteria yang diminta melalui pemetaan parameter *resource* terhadap nama fungsi internal. Langkahnya sebagai berikut: a). Menentukan *absolute path* berkas *web services* sebagai basis *path*. b).

Menggabungkan modul fungsi yang terkait ke dalam program. c). Mengecek kelengkapan struktur parameter URI. d). Jika parameter lengkap, petakan nama *resource* ke dalam nama fungsi, konversi format data input ke dalam variabel untuk parameter fungsi, jika fungsi ditemukan jalankan fungsi dengan input parameter, kembalikan hasil fungsi ke dalam variabel hasil, konversikan nilai variabel hasil jika berupa format XML atau HTML e.) Jika parameter tidak lengkap, yaitu nama *resource* tidak dapat dipetakan ke dalam fungsi, fungsi tidak ditemukan, atau tidak ada hasil proses fungsi, maka isi variabel hasil dengan pesan kesalahan. f). Tampilkan nilai variabel hasil sebagai *script* yang akan diolah oleh *web server*

Rancangan *agent/broker*, yaitu menyediakan aplikasi *web services* untuk proses registrasi (*registry*) dan penemuan kembali (*discovery*) untuk memudahkan pengelolaan dan pencarian layanan dengan cara melakukan pencatatan dalam *database*, langkahnya:

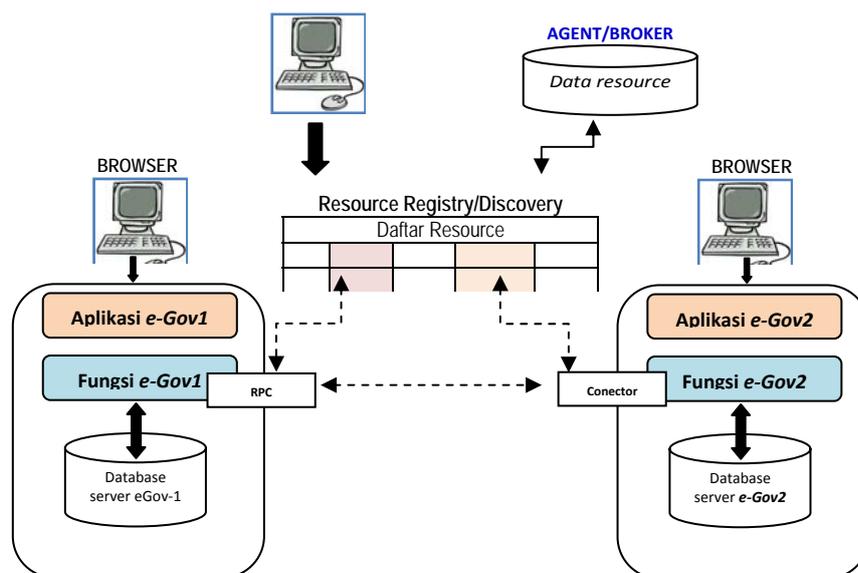
- 1). Membuat struktur *database* untuk *registry/discovery* layanan publik.
- 2). Membuat aplikasi *web server* untuk registrasi *provider* dan *service*.
- 3). Membuat aplikasi *web server* yang bersifat publik dan modul registrasi untuk

mendapatkan kunci akses publik bagi *requester*

Rancangan *requester*; *requester* dapat menggunakan fungsi layanan setelah memperoleh perintah URL pada *browser* di *client*. Jika hasil dari fungsi layanan akan digunakan sebagai sumber data hasil pengolahan, maka perintah

tersebut perlu dimasukkan ke dalam kode program *requester* melalui fungsi komunikasi dalam bahasa pemrograman (*socket*) atau pengolahan file jarak jauh (PHP, ASP, JSP).

Ilustrasi implementasi pada model interoperabilitas antar aplikasi e-Gov tampak di Gambar 4.



Gambar 4: Implementasi rancangan model interoperabilitas antara e-Gov1 dan e-Gov2

KESIMPULAN

Pemodelan interoperabilitas antar aplikasi e-Gov yang berbeda dapat dikembangkan melalui cara pemanfaatan teknologi *web services*. Pemanfaatan teknologi *web services* dalam pertukaran data antar dua aplikasi e-Gov mampu menunjukkan prinsip interoperabilitas.

UCAPAN TERIMAKASIH

Ucapan terima kasih disampaikan kepada Prof. Drs. Jazi Eko Istiyanto, M.Sc, Ph.D. yang secara inspiratif mampu memotivasi penulis untuk terus belajar tentang "interoperabilitas".

DAFTAR PUSTAKA

Ahmadjayadi, C., 2006, *Standarisasi Menuju Interoperabilitas e-Government*, makalah keynote speech pada Workshop Standarisasi Menuju Interoperabilitas e-Government, Jakarta.

Chen, D., dan Vernadat, F., 2004, *Standards on Enterprise Integration and Engineering-A State of the Art*, International Journal of Computer Integrated Manufacturing, 17(3), pp.235-253.

Direktorat e-Government, Depkominfo, 2004, *Blueprint Sistem Aplikasi e-Government*, Jakarta.

Doumeings, G., Vallespir, B., and Chen, D., 1998, *GRAI GridDecisional Modelling*, in *Handbook on Architectures of Information Systems*, Second Edition, Peter Bernus, Kai Mertins and Günter Schmidt (Editors), pp. 321-346, Springer Berlin Heidelberg,.

DSIPLK (Direktorat Sistem Informasi, Perangkat Lunak & Konten), Depkominfo, 2008, *Kerangka Acuan & Pedoman Interoperabilitas Sistem Informasi Instansi Pemerintahan*, Jakarta.

- Fielding R.T., 2000, *Architectureal Style & Design of Network-Based Software Architectures*, Ph.D. Thesis, Department of Information & Computer Science, University of California, Irvine.
- Indrajit, R.E., 2006, *Evolusi Strategi Integrasi Sistem Informasi Ragam Institusi, Kiat Memecahkan Permasalahan Politis dalam Kerangka Manajemen Perubahan*, Prosiding KNTIK untuk Indonesia dipresentasikan tanggal 3-4 Mei 2006, ITB, Bandung.
- Inpres No. 3 Tahun 2003, *Kebijakan & Strategi Nasional Pengembangan E-Government*, Jakarta.
- Inpres No. 6 Tahun 2001, *Telematika (Telekomunikasi, Media dan Informatika)*, Jakarta.
- Lucky, 2008, *XML Web services: Aplikasi Desktop, Internet & Handphone*, Jasakom.
- Michel, J.J., 1997, *Manufacturing, Modelling and Integration*, Presentation at a meeting of the Computer Department at CETIM (slides).
- Miller, P., 2000, *Interoperability: What is it and Why should I want it?*, <http://www.ariadne.ac.uk/issue24/interoperability/>, diakses: 05-03-2012.
- Molina, A., Panetto, H., Chen, D., Whitman, L., Chapurlat, V., Vernadat, F.B., 2007, *Enterprise Integration and Networking: Challenges and Trends*, Studies in Informatics and Control, 16/4, Informatics and Control Publications, December 2007.
- Nugroho, L.E., 2008, *Interoperabilitas*, Modul Kuliah MTI-UGM, Yogyakarta.
- Pascual, P.J., 2003, *e-Government*, e-Asean Task Force UNDP- APDIP, May 2003.
- Pautasso, 2008, C., 2008, *REST vs SOAP Making the Right Architectural Decision*, SOA Symposium, Amsterdam
- Raharjo, B., 2001, *Membangun e-Government*, ITB, Bandung.
- Setyantana, P., 2009, *Interoperabilitas Sistem Informasi*, Makalah dipresentasikan dalam Pelatihan oleh Direktorat Sistem Informasi Perangkat Lunak & Konten, Direktorat Jenderal Aplikasi Telematika, Depkominfo RI, tanggal: 27/28-05-2009, Sragen.
- Shorter, D.N., 1997, *Requirements for Enterprise Model Execution and Integration Services*, in: Kosanke, K. & Nell, J.G. (Eds.) *Enterprise Engineering & Integration: Building International Consensus*, pp. 235-243, Springer-Verlag, Berlin.
- Sukyadi, D., 2009, *Model Interoperabilitas Sistem Informasi Layanan Publik Studi Kasus: e-Government*, Karya Akhir, Prodi Magister Teknologi Informasi, Fasilkom, UI, Jakarta.
- Supangkat, S.H., 2006, *Framework Strategi Implementasi E-Government*, Prosiding Konferensi Nasional Teknologi Informasi & Komunikasi untuk Indonesia, ITB, 3-4 Mei 2006.
- Supangkat, S.H., Sembiring, J., dan Rahmad, B., 2007, *IT Governance Nasional: Urgensi dan Kerangka Konstruksi*, makalah Pertemuan Dewan TIK Nasional, 8-01-2007.
- Utomo, W.H., 2011, *Integrasi B2B Berbasis SOA Menggunakan Web services*, Disertasi Program Doktor Ilmu Komputer, UGM, Yogyakarta.
- WWW Consortium, 2004, *Web services Architectures*, <http://www.w3.org/TR/ws-arch/#whatis>, diakses: 08-03-2012.
-, 2012, *Interoperabilitas Data Dalam e-Government* <http://arvantc40s.blogspot.com/2012/02/interoperabilitas-data-dalam-e.html>, diakses: 08-03-2012.
-, <http://dishubkominfo.belitungkab.go.id/>, diakses: 08-03-2012.