A macro-pattern for public sector systems architecture

Jan K. Gravesen

November 05, 2013

The article presents a common systems architecture pattern for public sector organizations that has been derived from a variety of different projects and use cases during the period 2010-2013. The pattern that is presented is sufficiently flexible to express all the use cases and can be used as an accelerator in the design of new systems architectures for public sector organizations.

An adaptable architectural pattern for public sector organizations

Public sector organizations are busy at work improving citizen services despite decreasing budgets, growing service demands and constant policy change. They are preparing their organizations to meet the challenges of health care reform while leveraging the decades of investment they have made in their legacy application portfolio. They are increasingly building online self-service portals for their citizens that provide intuitive and relevant access to relevant services and programs. They are making more and more government services available online and presenting them in a cross-departmental manner in an attempt to make the complexity of government transparent to their citizens.

This article presents a common systems architecture pattern for public sector organizations that has been derived from a variety of different projects and use cases during the period from 2010-2013. The pattern that is presented is sufficiently flexible to express all the use cases and can be used as an accelerator in the design of new systems architectures for public sector organizations.

Architectural Patterns

‘‘Each pattern describes a problem which occurs over and over again in our environment, and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice’’

Christopher Alexander, "A Pattern Language: Towns, Buildings, Constructions", a book on patterns in the design of buildings and cities
Patterns contain best practices and expertise for complex tasks distilled from repeated client and partner engagements. Patterns can help define good solutions in a clear format that addresses reoccurring business objectives. (See Related topics for a more on selecting the best cloud pattern.)

Pattern techniques are generally acknowledged to be helpful in architectural design. Christopher Alexander, a buildings architect, describes this approach in his book *The Timeless Way of Building*. (See Related topics for a link.)

Software and buildings architects have many similar issues to address, and so it was natural for software architects to take an interest in patterns as an architectural tool. Many papers and books have been published on them since Alexander’s 1979 book, perhaps the most renowned being "Design Patterns: Elements of Reusable Object-Oriented Software." See Related topics for more information. This book describes simple and elegant solutions to specific problems in object-oriented software design.

Although patterns have yet to be formalized into a formal language that can be used to describe them or be supported by computer science theory, architectural patterns seem to play a role at many levels of architecture.
## Table 1. Architectural patterns

<table>
<thead>
<tr>
<th>Levels of patterns</th>
<th>Descriptive or Prescriptive</th>
<th>Generic or Specific</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Industry architecture patterns</td>
<td>Prescriptive</td>
<td>Industry-specific</td>
<td>At the level of enterprise architecture, patterns are industry frameworks or common systems architectures such as IBM’s Insurance Applications Architecture. They promote a series of best practices for addressing specific business objectives at the level of the industry and support these best practices with process and component models and conceptual architectures. It is possible to refer to these as industry architecture patterns. Industry architecture patterns are usually prescriptive in how they embed industry best practices into an architectural view.</td>
</tr>
<tr>
<td>Systems architecture patterns</td>
<td>Prescriptive</td>
<td>Problem-specific</td>
<td>At the level of systems architecture, these patterns address specific systems patterns such as IBM’s Integrated Information Core (IIC). Systems patterns are sometimes problem-specific in that they address specific technical situations, such as intelligent event management from SCADA (supervisory control and data acquisition) systems or intelligent parking or water management systems.</td>
</tr>
<tr>
<td>Systems architecture patterns</td>
<td>Prescriptive</td>
<td>Generic</td>
<td>Three-Tier architecture, n-tier architecture, and the ISO OSI 7 layer model are examples of generic and prescriptive patterns that adapt to a variety of different situations in systems or network architecture. They typically embed specific design rules around layer isolation, thus being prescriptive by nature.</td>
</tr>
<tr>
<td>Systems engineering patterns</td>
<td>Descriptive</td>
<td>Generic</td>
<td>At the level of systems engineering there are low-level patterns that address design objectives for object-oriented programming as well as more generic patterns such as the Model-View-Controller (MVC) pattern or IBM’s patterns for e-business. These patterns function at a theoretical level of computer science and are descriptive (as opposed to prescriptive) in how they describe commonly occurring situations in systems engineering.</td>
</tr>
</tbody>
</table>

Over the past three years, it has been possible to identify and extract a generic and descriptive systems architecture pattern from a variety of different client engagements in public sector.
organizations. Although the engagements have been very different in terms of their business scope and use cases (for example, a shared ERP system for 100+ state departments, a federated single view of the citizen (SVC) solution for citizens and patients in a county, a health benefit exchange and so on) at the level of the systems architecture they have been remarkably similar to the point that a common systems architecture for public sector can be identified.

The common systems architecture pattern is interesting because it appears it can adapt to a variety of high-level use cases that address specific public sector objectives ranging from shared financial accounting and procurement practices over coordinated care, to full online availability of public sector services.

Although the common systems architecture pattern functions at the level of the systems architecture, it is descriptive as opposed to prescriptive and it is generic, i.e. adaptable, as opposed to problem-specific.

The advantages of adaptable and common systems architecture are many:

- The pattern can form the basis for a generic public sector industry framework or for a public sector platform that adapts to a multitude of business objectives. (A framework is purely conceptual. A platform can be configured into a working application by embedding specific application logic and data models into a set of integrated and configured middleware components and products.)
- The pattern provides a clear, yet generic format across a large number of public sector objectives and thus belongs between the systems architecture pattern and the industry architecture pattern.
- The pattern can be architecturally modeled using Rational tools such as IBM Rational System Architect, and models can be readily adapted to several different use cases and government and policy objectives. See Related topics for more on Rational System Architect.)
- The pattern is vendor- and technology-agnostic, allowing a public-sector organization to leverage existing technology investments without sacrificing the freedom to rapidly change products now and in the future. This advantage is particularly important in federated organizations where the macro-pattern and the solutions it supports must integrate and interoperate with a range of existing capabilities, such as multiple directory services (one per department), multiple departmental business systems, an enterprise-wide services registry, many departmental content management systems used for integrated case management, and so on.

In this article I show how the pattern is able to adapt to several different yet reoccurring use cases, each of which match a widespread public sector objective:

Table 2. Use cases and their respective architectural challenges

<table>
<thead>
<tr>
<th>Use case</th>
<th>Architectural challenge</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shared business services</td>
<td>Building shared human resources, financial management, and procurement processes across a range of cooperating departments</td>
</tr>
<tr>
<td>Coordinated care</td>
<td>Building an information exchange for coordinated care across social and healthcare services and programs</td>
</tr>
</tbody>
</table>
Healthcare reform  | Building a healthcare insurance exchange  
---|---  
Shared technical services  | Building federated patterns for self-service portals, enterprise content management, or authentication of online users across multiple departments, agencies, or even states  
Full online availability  | Building federated patterns for self-service portals, enterprise content management, or authentication of online users across multiple departments, agencies, or even states  
Single view of the citizen (SVC)  | Creating a single view of the citizen across programs and departments  

The pattern allows multiple, disparate systems to be integrated into a complete solution, combining new and legacy applications for a seamless user experience that remains relevant over time within the same architectural pattern.

Considered in this light, the pattern can be reframed as a framework and if it becomes populated with pre-integrated products, it can be constructed as an integrated public sector platform with well-developed extensibility and adaptability characteristics.

**Trends and use cases in public sector organizations**

The pattern is able to function in a federated governance model because the cases from which it has been harvested typically were federated situations. In other words, it is able to meet the shared needs of multiple cooperating departments or agencies, a strength that yields improved resource efficiency but still allows some amount of modular adaptation to the specific needs of individual agencies and departments. The public sector, whether it is a state, a county, or an entire country, is increasingly interested in improving IT cost efficiency through the use of **shared service platforms**. The pattern is designed to support such platforms.

Another main trend is occasionally referred to as **full online availability**. Public sector organizations increasingly try to offer services through online channels, and full online availability means that electronic channels are used for two-way interaction between citizens or businesses and public sector.

A third trend is **citizen-centricity** enabled by operating programs and services with a single view of the citizen (SVC). SVC represents the ability to market and offer programs within social services, public health, healthcare, education, or taxation such that, from the citizen's perspective, the organizational boundaries within the public sector dissolve. Instead of having to know which specific department or agency offers which programs, a citizen has a complete overview of everything that is available, based on what he or she is eligible for in the specific situational context. Citizens are able to interact with the public sector in a uniform manner without having to know about the existence of specific agencies or departments. The citizen is dealing with one organization, typically using an integrated multi-channel approach where field offices are less and less important and where mobile interactions are becoming more and more prevalent (and cost efficient for both parties).

A fourth trend and one that is at the core of the US healthcare reform is **interoperability**. Federal grants for health reform programs typically require compliance with Medicare IT Architecture (MITA). MITA imposes a series of modularity and interoperability criteria and proposed systems
architectures must prove their compliance with MITA design principles. A new health insurance exchange, for instance, will have to expose to its business services so they, at least in principle, can be located and reused by other health insurance exchanges in other states. Interoperability also forms the basis for shared service platforms as well, and for serving these up in a federated, as opposed to centralized, manner.

**Figure 1. Public sector trends**

![Diagram](image)

Each of these trends is important from an architectural perspective and gives cause to a set of new design rules for the macro pattern.

The pattern adheres to several design rules that are hugely important for the challenges faced by the public sector today.

**Designed for federation**

The pattern is able to serve shared business and technical services to a set of cooperating agencies or departments that already run the same category of services locally. This is increasingly important in public sector and a growing body of thinking emphasizes shared services. Examples are the FICAM (Federal Identity Credentialing and Access Management) standard, which uses SAML2 to federate across different department websites and determines how users are identified and authenticated for online services; and the growing interest in a master data management solution that establishes enterprise-wide patient or person indices across programs and departments. The pattern contains sub-patterns -- an enterprise-wide master data management service for citizens or patients, for example -- that must be able to federate with a master data management service in an external service provider such as a Health Information Exchange or by a specific department.

**Designed for interoperability**

Especially in healthcare, federal grants are tied to the ability to demonstrate compliance with the Medicaid Information Technology Architecture (MITA) specification, which emphasizes reusability and interoperability of web services among cooperating agencies and states on healthcare solutions.
Designed for **security**

FIPS 140.2 encryption for messaging with external partners and HIPAA and HITECH Act compliance when ePHI (electronic Personal Health Information) is involved are two ubiquitous requirements in social service and healthcare agencies.

Designed for **local flexibility**

Because the public sector is a federated organization, the pattern must find a balance between a need for shared business and technical services across departments and the need for building local extensions (departmentally specific micro patterns) on top of or adjacent to the pattern.

Designed for **citizen-centric and multi-channel service delivery**

Citizen-centricty increasingly means the ability to integrate programs and citizen service offerings such that boundaries become invisible to the citizen. Citizen-centricty requires two-way communication with across multiple channels: field offices, mobile devices, and web portals. Where private businesses are concerned, channels must also include B2B web services and rudimentary B2B FTP and XML data exchanges using a variety of formats such as NIEM, ebXML, EDI, XML and SOAP.

Designed for **adaptability to an increasing pace of policy legislation**

The pace with which new legislation is introduced and changed is increasing. New legislation manifests as new eligibility rules, new program policy rules, new benefit calculation rules and so on come into effect. Increasingly, public sector organizations want to introduce adaptive patterns for policy rules administration. This requirement means lifting policy rules that previously were embedded in software code into a separate, flexible policy layer, which can be updated without making changes to system code. It also ideally means that the same policy layer can serve more than one system, as the same rules often are used by different social service or public health programs, each of which are automated by different systems.

The design rule **designed for federation** breaks into a further set of sub-requirements for the pattern:

- It must be possible to federate a services registry with a state-wide or county-wide services registry to expose web services implemented by the pattern. This is especially important for purposes of MITA compliance within health care.
- It must be possible to federate identity and access management with other applications in different departments or even across a range of external service providers. The federal FICAM standard is adopted by states as a local SICAM standard and requires the ability to exchange SAML2 tokens for security purposes.
- If the pattern offers an enterprise-wide MDM service for patients, citizens, households, or providers (hospitals, clinics, general practitioners, and so on), it must be able to federate in two specific ways: with pre-existing systems in the departments that are storing these metadata already in as non-invasive a manner as possible; and with other departmental or even external MDM services, for instance with an external health information exchange (HIE).
- The enterprise service bus (ESB) must be able to federate with pre-existing ESBs in departments and agencies in a secure way.
• The web portal must be able to federate with pre-existing web portals in departments and agencies using the Web Service for Remote Portlets (WSRP) standard or it must be able to provide a general platform for web content management (WCM) and pre-built portlets to a wide range of departments that intend to use the shared portal platform to build departmental micro-portals.

Not all these requirements will be present in every use case, but increasingly where a state or county has a defined enterprise architecture, a subset of them invariably will apply.

A common systems architecture pattern for public sector

A study of a variety of public sector architectures and proposals reveals that the trends described above have intensified.

Figure 2. A macro-pattern that fits across a wide range of use cases

Key points of the pattern

The pattern is comprised of layers and sets of services:

• Client layer
• Security services
• Business-to-business services
• Master data management services
• Process and component framework
• Portal delivery framework
• Data delivery services
• Data warehouse and data marts
• Data services

Client layer

The client layer comprises a set of external or departmental services that federate with the common systems architecture pattern. Not all of the services in the following list will exist or require federation but increasingly, many will.

• Enterprise-wide, agency-wide, or departmental service registries
• Metadata directories
• Multiple departmental message brokers or service buses
• Multiple departmental LDAPs for employee authentication
• Multiple departmental systems
• External applications such as electronic health records, judicial court systems, pharmacy systems, banks, and certificate authorities
• Public sector systems such as state-wide eligibility systems, federal eligibility systems, and so on
• Departmental or external master data management (MDM) systems such as external health information exchanges (HIEs) or specific departmental MDM instances

The client layer represents an often overlooked component in a public sector foundation architecture and common systems architecture. The client layer must be included to demonstrate how the architecture is interoperable with departmental and external systems and services.

Security services

Due to HIPAA and HITECH Act compliance requirements, Security Incident and Event Monitoring (SIEM) services monitor intrusions and security events on the enterprise message bus and application services.

• HIPAA requires column-level database monitoring to ensure that ePHI (personal health information) is not compromised
• Both data in transit and data at rest must be encrypted. Data in transit encryption /decryption (compliant with FIPS 140-2) occurs in the B2B services layer. Data at rest encryption can be performed by the database itself.
• Multi-tenancy becomes a key design criteria, especially for security and access controls, as common systems architectures increasingly are deployed on cloud computing infrastructures.

B2B services
Mobile web and portal applications need to be designed with separate presentation layers for different clients, such as mobile devices and desktops.

- The *presentation layer* caters to mobile devices, to deliver contents that are tailored for the device browser, and to deliver the appropriate content, layout, and look and feel for the requesting device.
- The *business services layer* contains the core business logic, which must be reused across the different presentation layers.
- *Web applications* deployed in the application server must have features built in to detect access from mobile devices, to determine the properties such as screen resolution of the device, and to serve the appropriate web pages to the mobile device browser.

**Master data management services**

Master data management plays an important role in public sector use cases. The pattern includes a shared enterprise-wide MDM (EW-MDM) capability that can federate across existing MDM systems regardless of whether these are encapsulated in existing applications or operate as MDM-solutions. The EW-MDM operates in registry style to be as non-invasive as possible.

**Process and component framework**

The process and component framework is at the center of the pattern. It provides:

- Process orchestration and specific business logic required by the use case. Typically, these are implemented in a business process management system using WS-BPEL or BPMN.
- The capability to process electronic forms that are interwoven into the process workflows.
- Serve appropriate notifications and task lists for case workers and other public sector employees. Typically, the process and component framework must interact with the portal delivery framework to serve this function.

Ideally, components connect to process orchestration services through the enterprise service bus (ESB). The service component architecture (SCA) has become more or less foundational in more recent systems architectures. Almost 100 percent of all public tenders since about 2008 request compliance with a service-oriented architecture.

The process and component framework also contains a web services registry. Frequently the requirement is for UDDI compliance but the UDDI standards have not been updated since 2003 and provide little basis for meeting an increasingly important requirement that:

- Business and government services exposed as web services (WSDL) be stored in the web services registry
- The internal web services registry can be federated with departmental or enterprise-wide enterprise service registries

Government services and tasks are the component services, the web services, or the application modules, which define and implement the application logic. Also included are Public Policy Management Services, which are the public policy and business rules management system.
that can be called by services or by the processes implemented in the process and component framework.

**Portal delivery framework**

The portal delivery framework delivers traditional portal services under a federated portal model. The framework must support both local portals such as departmental portals (also referred to as vertical portals) or consumer segment portals such as case worker portals, management portals or citizen or provider portals (horizontal portals). The framework must also support portal proxies through web services for remote portlets (WSRP) when an enterprise portal links to departmental portals in the client layer using the WSRP standard. Of special consideration is the support for ADA (American with Disabilities) standard.

The portal provides integration for electronic forms and universal task lists that trigger case flows or government processes orchestrated by the process and component framework. Furthermore, the portal will frequently be used as the entry point to flexible reporting from the data delivery services.

**Data delivery services**

Data delivery services include online analytical processing (OLAP) services, predictive intelligence services, and entity analytics services to detect non-obvious relationships in relational data sets. Also included are information server services for meta-data catalogues, extract/transform/load processes, and data quality processes.

**Data warehouse and data marts**

Data warehouses and data marts contain the individual operational data stores (ODS), data warehouses, or functional data marts needed for financial, population-based, or outcome-measured statistics and analysis.

**Data services**

Data services include the specific data models that are directly accessed by government services and tasks in the process and component framework. In addition, the data services contain:

- The specific master data management hubs used by the MDM services
- The repositories used by the enterprise content management (ECM) services, for example, the ECM Library or Forms repositories
- The public policy rules repository used by the public policy management services in the process and component framework

**Use case: Single view of the citizen**

The following diagram demonstrates a ready adaption of the macro-pattern to a federated master data management solution for citizen, household, patient, and provider master data. The blue boxes highlight which capabilities in the pattern are activated and the gray boxes signify deactivated capabilities.
Figure 3. A ready adaptation of the macro-pattern to a federated master data management solution

Key points of the pattern

The key points of this pattern include:

- **Business objectives**
- **Client layer**
- **Information services**
- **Portal delivery framework**
- **Master data management (MDM) services**
- **Process and component framework**
- **Data delivery services**

**Business Objectives**

A single consistent view of patients, citizens, households and external providers across all departments and programs

**Client Layer**
The client layer comprises several pre-existing MDM solutions, typically regional health information exchanges. It also contains multiple departmental LDAP services that need to authenticate employees from different departments when the look up a global view of citizens or patients.

Multiple departmental systems have master data extracted and loaded into the MDM server using information services for extract/transform/load (ETL) processing. Data from these systems is usually analyzed for inconsistencies prior to load, but typically, data is included in the initial load.

Departmental systems are able to perform create, read, update, delete (CRUD) functions against the MDM server using web service calls that are received by the MDM servers in-bound message broker.

**Information services**

In addition to initial ETL and data quality analysis functions, information services will also be used to verify the format of addresses.

**Portal delivery framework**

The portal may contain a lookup-retrieval function that allows employees to search for citizens and patients using their name, social security number, or an enterprise ID (EID) assigned to the citizen.

The portal also provides a task list for appointed data stewards which can be used to manage the data and orchestrate the data governance workflows.

**MDM services**

The crucial part of the pattern is MDM services. Traditionally, MDM implementations have been mastered in the sense that they stored a master copy of the demographic person, patient, or household data that made it necessary for the departmental or external applications in the client layer (case management systems, booking systems, electronic health journals, portals etc) to use the central master copy when data was to be looked up. This was an invasive approach that frequently required changes to the applications in the client layer and consequently carried significant cost and complexity.

In government and health care, a different and more non-invasive approach is often used. Instead of changing the applications and data models in the client layer to fit to a mastered copy, the MDM services offers a subscription model that the client layer applications can use. The MDM services receive data updates from all applications in the client layer through an inbound broker and uses probabilistic algorithms to resolve which of the potentially many versions of the data is the most correct and up to date based on a complete history of the data. The benefits of this model are described in some detail in *MDM: Realizing the Same Benefits through Different Implementations.* (See Related topics for a link.)

In addition to probabilistic algorithms to determine which of several potential versions is the one with the highest likelihood of being correct, MDM Services may be able to resolve names of
citizens across different language groups and nick names as an integral part of the resolution process.

**Process and component framework**

Process orchestration comes into the picture because master data must be managed. In government, this means assigning data stewards from different departments to the task. A portal can be used to automate the workflow needed to ensure that all data stewards have been engaged in the data governance process of eliminating duplicates, adding new data fields to demographic data, resolving conflicts, and so on.

**Data delivery services**

Data delivery services can perform functions related to data quality that rarely can be performed by the MDM services themselves. One typical function is address verification, involving the verification of zip codes, street names, street numbers, cities, and so on to verify that an address is in fact correct.

**Use case: Delivering coordinated care**

The following diagram demonstrates a ready adaption of the macro-pattern to a health information exchange. The blue boxes highlight which capabilities in the pattern are activated. The gray boxes signify deactivated capabilities.
Figure 4. Ready adaptation of the macro-pattern to a health information exchange

Key points of the pattern

Coordinated care represents a complex pattern and lights up large component areas of the macro-pattern. The key elements are:

- Business objectives
- Client layer
- Process and component framework
- Portal delivery framework

Business objectives

The main business objectives are to:

- Provide an integrated, consistent view of the citizens that are served by disparate programs and social and health services
- Provide case workers in different departments the ability to share information about their clients and coordinate their care and rehabilitation plans
• Eliminate handovers among disparate programs and departments and deliver a seamlessly integrated service experience to the citizen

Client layer

Multiple departmental systems are integrated into the pattern to establish a single view of the citizen (SVC) across both public care delivery programs and external private providers.

Case workers with multiple departments need access to the care delivery platform and must be authenticated by means of their departmental LDAP servers. Frequently there exists no super-LDAP directory on an enterprise wide basis in government.

Coordinate care platforms often need to integrate with external regional health information exchanges, some of which are public and some of which will belong to private care delivery organizations. The coordinated care pattern must provide the means to coordinate the identity resolution of citizens across these disparate information exchanges.

Process and component framework

Unsurprisingly, it is the process and component framework that provides the collaborative glue that allows case workers and external providers to coordinate their care delivery. The orchestration is relatively simply and may consist in routing observations and notes made by one case worker to all others who serve the same client, provided that the client has given his or her consent.

More sophisticated is the fundamental need to build a state machine that tracks the network of different states that a citizen may pass through over time and for each new state trigger an event notification to the network of case workers assigned to the citizen. These notifications show up as events or tasks in a universal task list in the employee and provider sub-portals as reminders of workflow activities that the case workers and providers must attend to.

The main business services performed within the component framework are typically:

• Referral management: The ability to refer citizens and patients to specific providers or care delivery organizations so that referrals made by one case worker become visible to the other case workers assigned to the citizen. This shared knowledge enables an integrated outcome plan to be created and managed across formerly disparate programs and departments.
• Collaborative service delivery: Secure logging and exchange of observations of notes pertaining to a citizen or patient. If all case workers and care delivery personnel can share the same view of notes and observations, they are able to provide care with a higher level of awareness of the citizen's unique situation.
• Notifications: The state machine that keeps track of a citizens' network of state-changes and his or her current state. It usually involves a workflow that triggers specific tasks to specific case workers when a new state is encountered and these tasks show up in the task list in the case worker and provider portals.

Portal delivery framework
The portal delivery framework typically contains a citizen portal, a case worker portal, a provider portal, and potentially a management portal for population-based analytics and the management and reporting of outcomes for the coordinate care initiative.

The portal usually also provides a very simple lookup and retrieval function that allows a case worker, health clerk, or clinician to look up a citizen (using his or her name, last known address, phone number, or an enterprise identifier such as social service number) and derive a probabilistic match of the identity that best matches the inputs derived from a variety of source systems that all contain demographic data on the citizen.

**Use case: Providing full online availability**

Full online availability of government services is a common objective across most government organizations in the western world.

The pattern adapts readily to the IBM Business Process Accelerator Pattern for building a web portal with the ability to invoke tasks and process flows and fill in e-forms. (See Related topics for a link to the pattern.)

**Figure 5. Ready adaptation of the macro-pattern to online government services**
Key points of the pattern

The key elements of the pattern include:

- Business objectives
- Client layer
- Identity, credentials, and access management (ICAM) layer
- Portal delivery framework
- Enterprise content management (ECM) services

Business objectives

It's a requirement to offer full online availability of government services in a bi-directional manner. Citizens can request or apply for services online and expect to receive replies to their requests online.

Client layer

The client layer comprises the departmental systems that are connected to the pattern to serve citizen's requests. Many of these systems are case management systems that operate on a departmental or per program basis.

The client layer may also contain one or more departmental service buses that will be federated to the pattern's message broker to provide straight through processing of requests that originate in the portal.

If departmental systems are outside of acceptable security controls, a data in transit encryption requirement may be imposed on the pattern.

ICAM services

Citizens are usually authenticated and authorized to request or apply for services, lookup the status of their requests or receive replies, access notifications, or engage in appeals if denied service or eligibility.

A federated identity manager may serve the pattern itself or may be federated to other departmental identity managers for purposes of cross-departmental credentialing.

Portal delivery framework

A web portal provides a personalized task list, notifications and access to e-forms that can be used by citizens to request online access to services. It interacts with other services:

- The portal's universal or personalized task list interacts with a Business Process Manager that orchestrates the case workflow and uses the message broker to access custom built business services embedded in the pattern or business services in departmental systems.
• The portal also interacts with an electronic form viewer and electronic forms list that are served up by a electronic form (e-form) server.

ECM services

Enterprise Content Management (ECM) services may be used to store filled out electronic forms (typically applications) and the forms templates that are used to materialize the specific electronic forms in the portal framework or through mobile devices.

One of the major challenges with full online availability in many countries is the lack of a shared identity and certificate authority scheme that can be used to sign forms electronically. In the absence of such a scheme, local government agencies and departments must frequently resort to building an extra credentialing process around the online portal or they have to build their own X.509 standardized signing technology and purchase their own certificates from a trusted third party certificate authority.

A significant opportunity for public sector and an often encountered roadblock on the way to full online availability is the ability to operate a CA and X.509 signing scheme on as high a level as possible, ideally at the level of entire states or countries. This approach has been practiced in a few countries around the world and yielded tremendous scale advantages reducing the cost per citizen and signature significantly and spreading these costs across all departments, agencies, municipalities and counties. However in order to make it easy to for citizens and businesses to use digital signatures such as the OCES1 standard through a highly shared PKI solution, ideally the private and public key must be managed by a shared service entity that is used by the public web applications in the client layer. This administration of private and public keys by a single organizational entity introduces a significant trust issue and will not be an acceptable solution in most jurisdictions.

Use case: Sharing government services and data

The ability to share government services and data, especially as they pertain to back-end processes such as finance and controlling, accounting, human resources, and procurement-supplier management is becoming more widespread in the public sector.

The pattern exemplifies a SAP FI/CO solution with SAP's Public Budget Finance (PBF) data warehouse and SAP's Public Provider Service (PPS) procurement portal but it can be adapted to other ERP systems such as PeopleSoft and Curam with relative ease.
Figure 6. Ready adaptation of the macro-pattern to a shared ERP platform

The figure shows a SAP ERP and SRM (Supplier Relationship Management) system encapsulated into the inner ring (green boxes) while the outer ring now contains components (blue) that can be used to combine SAP business and data services into composite applications.

The pattern is slightly simplified in that many ERP systems such as SAP and PeopleSoft require an inner ring middleware stack and a message bus to synchronize their different data base instances.

Key points of the pattern

In this adaptation, the *inner ring and outer ring architecture* is demonstrated. SAP retains its use of the middleware standards that are needed to use the SAP Enterprise Service Registry and synchronize its databases under the SAP Process Integration (PI) architecture, while SAP Enterprise services still fits into the pattern.

IBM has built this pattern with a variety of different ERP systems. Just over the past three years, I have been involved in versions involving SAP, PeopleSoft and Curam but other ERP solutions can certainly be used.
Typically, sharing of government services and data through a shared ERP system follows a specific architectural pattern, which sometimes is referred to as an inner ring and outer ring pattern. This same pattern is also used frequently in private sector to encapsulate an ERP system into a layered architecture where MDM services, data delivery services, portal framework, ECM services, ICAM services and the upper middleware parts of the process and component framework are used to encapsulate the business services and data services (data model) in the ERP system.

The ERP system's business and data services and the necessary ERP middleware form the inner ring while MDM, Portal, ECM, Data Delivery, ICM and Process Services form the outer ring. The benefit of this separation between the two rings is that the outer ring establishes the foundation for enterprise architecture that is greater than the ERP system itself and allows an agency or cooperating group of agencies or counties to build composite applications that include business and data services from the ERP systems according to their own technology standards as opposed to being restricted to the technology stack of the ERP system.

MDM services are rarely used directly in a shared business services pattern because one of the fundamental advantages of using an ERP system for business and data services is the use of standardized meta-, and master-data models.

**Conclusion**

The description of the pattern and the demonstration of how it adapts to a variety of different public sector objectives such as shared business services, full online availability, single view of the citizen, coordinated care, and healthcare reform are examples of how the pattern can be reused to accelerate the build-out of new and composite application architectures for the public sector.

The federated nature of the pattern allows it to readily integrate new services and extend business capabilities in a manner that takes advantage of the investments already made in capabilities within public sector departments such as LDAP servers, enterprise systems, content management solutions, master data representations, and so on.

Taking this a short step further, it becomes possible to build a specific public sector platform for a public sector organization that is interested in a repeatable model for application modernization. In many public sector organizations, the financial crisis has led to a backlog of modernization initiatives that eventually must be handled. The pattern provides a starting point for using shared technical services across modernized systems and provides an architectural pattern and a loosely-coupled SOA compliant approach to modernization, where each individual project can leverage the same MDM, ECM, portal and ESB services instead of re-inventing them and re-investing in them for each project.

For IBM, the pattern provides a highly adaptable and reusable industry framework for building public sector applications in a way that leverages past investments in departmental systems and IT-capabilities and is able to interoperate with these using commonly available standards from OASIS, W3C and The OpenGroup as well as sector specific standards such as FIPS 140-2 (Federal Information Processing Standard), FICAM (Federal Identity, Credential, and Access
Management) design and implementation guidelines, and FISMA (Federal Information Security Management Act).
Related topics

- Find out how to uncover the best adoption pattern to integrate new services and extend business capabilities in the article Select the correct cloud adoption pattern (IBM developerWorks, May 2013)
- For more on simple and elegant solutions to specific problems in object-oriented software design, see Design Patterns: Elements of Reusable Object-Oriented Software. (Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides. Addison Wesley, October 1994, ISBN 0-201-63361-2)
- Read about the benefits of a new MDM model in MDM: Realizing the Same Benefits through Different Implementations. (Dr. Scott Schumacher, October 2008) http://mdmbook.com/?p=40
- Refer to the IBM Business Process Accelerator Pattern for building a web portal with the ability to invoke tasks and process flows and fill in e-forms.
- Check out this related on article, What defines success with public sector enterprise architecture (developerWorks, Dec 2012)
- Download a free, fully enabled trial version of Rational System Architect.