eHealth Ontario
www.ehealthontario.on.ca

Service Level Agreement

HIAL Implementation

December 14th, 2014

Dan Birsan
eHealth Ontario

Ontario
eHealth Ontario

Service Level Mission Statement

The SLA is a formal document outlining a service commitment provided by an IT service provider to one or more customers. The Service Level Management mission statement is to "plan, coordinate, negotiate, report and manage the quality of IT services at acceptable cost".

Project Scope

The scope of the SLA project is to implement an enterprise framework that adapts to changing business priorities and service levels, define clear goals to shape the service offered by the provider, and avoid the back and forth associated with service level disagreements.

SLA benefits include open communication and the ability to manage the customers' expectations. IT organizations also benefit from a clearer picture of what the users need, the ability to balance and adjust their resources to meet those expectations, as well as explicitly detail the costs associated with any given level of service.

It is a business requirement that web services are reliable when accessed by health providers or consumers. A service consumer need a possibility to check whether a service meets the reliability requirements, compares it with other offers and ponder the possibilities when he decides for the use of one web service. Service Level Agreements (SLA) should cover this requirement.

Web services became popular in the development of applications. It is comfortable to use them because the transfer is realized by HTTP and the data representation by XML. *A good example is when a health care provider, the hospital, is searching for the patient health record to find out about previous lab results or used medication. The hospital uses web services that are provided by an e-health provider that will store the individual health record or laboratory results. The web services are also used to confirm treatment dates and history.* The availability of these external web services is business critical for the health care provider.

The scope statement is an agreement among the project team, the project sponsor and key stakeholders. It represents a common understanding of the project for the purpose of facilitating communication among the stakeholders and for setting authorities and limits for the project manager and team. The scope statement includes relating the project to business objectives, and defining the boundaries of the project in multiple dimensions including approach, deliverables, milestones, and budget.

SOA Issues that SLA tries to solve

The common and most important problem of enterprise IT architecture is to integrate siloed applications, inflexible data structures and technology-specific platforms. SOA is technology independent. Thus there are several possibilities to implement a service that can be accessed by a service consumer via the service bus. The structure of the legacy systems may also be a problem as they are often siloed applications and separated from other applications.

Unpredictable twists and turns in an ongoing project to suppliers, customers or the environment are not an exception but the norm. These risks are well-known, because they frequently occur in many ongoing projects, as well as in the execution phase. Nevertheless, they often do not get a close attention. Problems are then blandished as challenges, but they are occurred risks. Only if those risks are known, we can handle them.

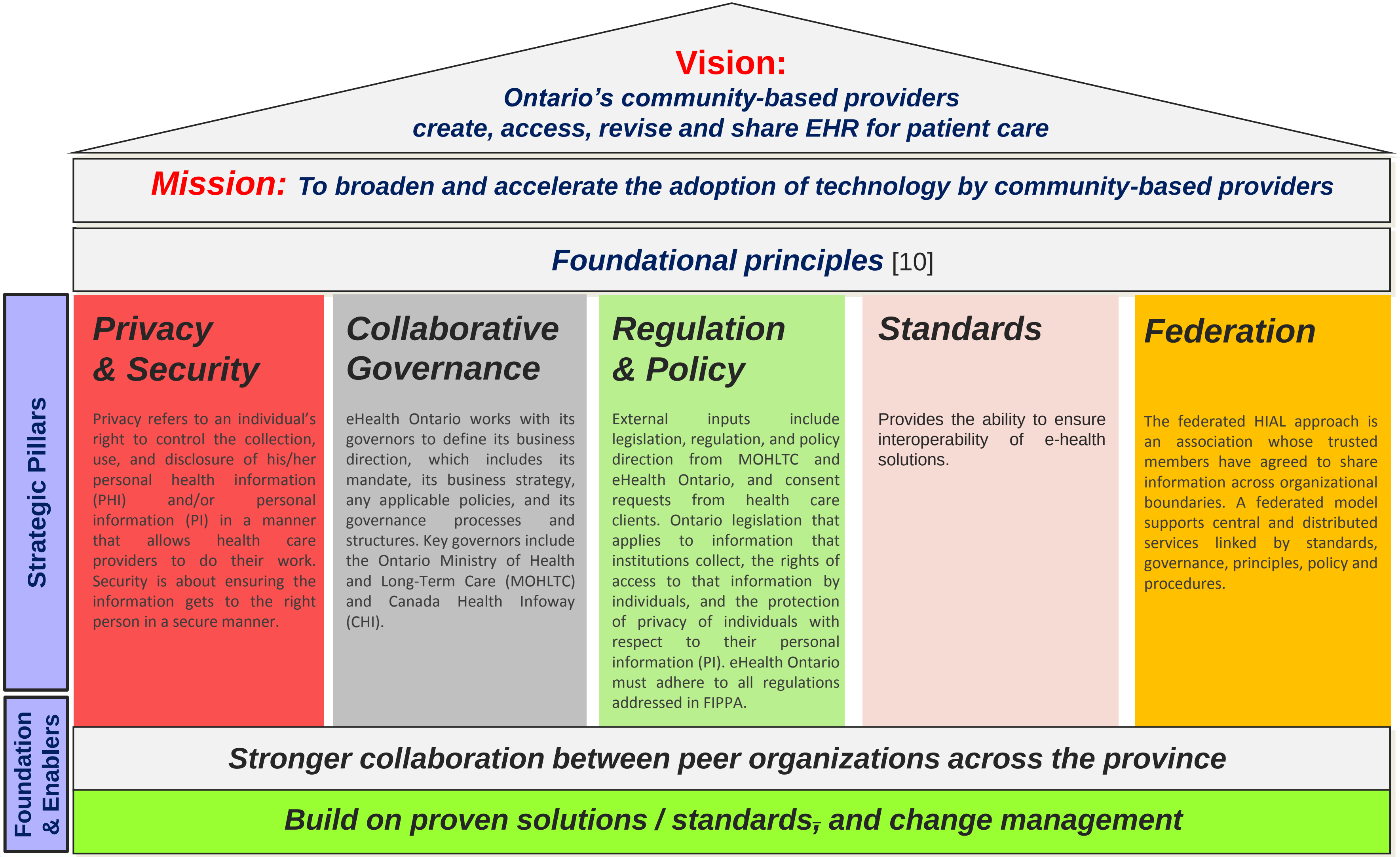
Even though the principle of the separation of concerns is important for creating reusable software and realizing an SOA, it is often violated for many reasons. There are internal and external reasons. An internal reason is the amount of work. It takes time to create software that meets the requirement of separation. Some try to avoid the additional effort and sloppy solutions win over sustainable solutions. The disadvantages of this behavior will appear in the future and many people repress them in the present. An external factor is the behavior of IT vendors that want to lock the business logic of their customers to their proprietary technology. The separation of concerns may reduce their profit, so an organization should not blindly trust on their vendors' technological recommendations. Their greed will prevail over technical advisable solutions. Companies should use open standards to remain independent.

Big IT projects usually fail because too many participants operate over a long period of time on a project and everyone works on his hidden agenda. Starke and Tilkov, SOA experts, suggest determining a strategic direction and a fast implementation of smaller projects. Feedback loops of the projects' experience lead to an adjustment of the strategic direction. They emphasize particularly a solid governance.

Many SOA projects were not as successful as the SOA infrastructure vendors promised. The technical requirements are not the problem. These are solved perfectly. The main problem is a missing methodical approach to identify the business services that exist in a company.

The service oriented approach is not a pure IT issue. It means that the business processes need to be designed service-oriented. That is the main problem of many SOA efforts. The entire company has to get involved in this new way of thinking and abandon established habits. It is clear that this is not a piece of cake. When an SOA project fails, it is most probably not in the technology but in the communication and the lack of willingness to compromise. [1]

eHealth Vision and Strategy to support SLA adoption



Vision:
Ontario's community-based providers create, access, revise and share EHR for patient care

Mission: ***To broaden and accelerate the adoption of technology by community-based providers***

Foundational principles [10]

Strategic Pillars

Privacy & Security

Privacy refers to an individual's right to control the collection, use, and disclosure of his/her personal health information (PHI) and/or personal information (PI) in a manner that allows health care providers to do their work. Security is about ensuring the information gets to the right person in a secure manner.

Collaborative Governance

eHealth Ontario works with its governors to define its business direction, which includes its mandate, its business strategy, any applicable policies, and its governance processes and structures. Key governors include the Ontario Ministry of Health and Long-Term Care (MOHLTC) and Canada Health Infoway (CHI).

Regulation & Policy

External inputs include legislation, regulation, and policy direction from MOHLTC and eHealth Ontario, and consent requests from health care clients. Ontario legislation that applies to information that institutions collect, the rights of access to that information by individuals, and the protection of privacy of individuals with respect to their personal information (PI). eHealth Ontario must adhere to all regulations addressed in FIPPA.

Standards

Provides the ability to ensure interoperability of e-health solutions.

Federation

The federated HIAL approach is an association whose trusted members have agreed to share information across organizational boundaries. A federated model supports central and distributed services linked by standards, governance, principles, policy and procedures.

Foundation & Enablers

Stronger collaboration between peer organizations across the province

Build on proven solutions / standards~~,~~ and change management

Measurement Gap Analysis

TOGAF specifies that a key step in validating an architecture is to consider what may have been forgotten. The architecture must support all of the essential information processing needs of the organization. The most critical source of gaps that should be considered is stakeholder concerns that have not been addressed in prior architectural work.[20]

Measurement gaps:

Target Architecture ---> Baseline Architecture↓	Audit	Monitoring	SLA	Reporting	Eliminated Services↓
Audit	Included				
TPAS					Intentionally eliminated
Logging					Unintentionally excluded - a gap in Target Architecture
Monitoring		Potential match			
Reporting				Potential match	
New --->		Gap: Enhance servies to be developed or produced	Gap: To be developed or produced	Gap: To be developed or produced	

As a result of the GAP analysis we've performed on the existing capabilities we've identified the following services that need to be developed or produced.

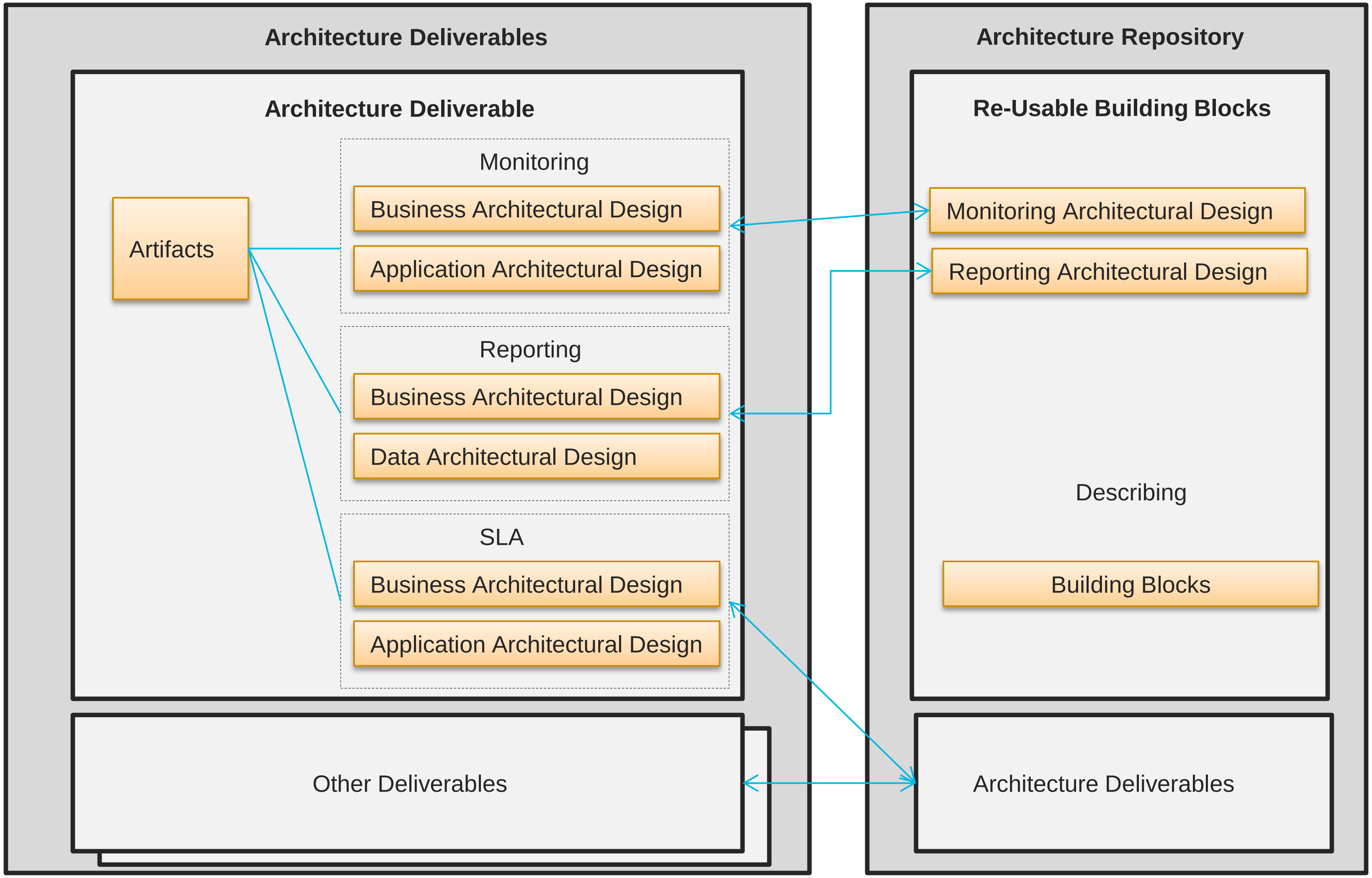
Monitoring capabilities need to be extended to accommodate future SLA and Reporting needs

Reporting capabilities need to be extended as well to support the new KPI and metrics required for implementing SLA

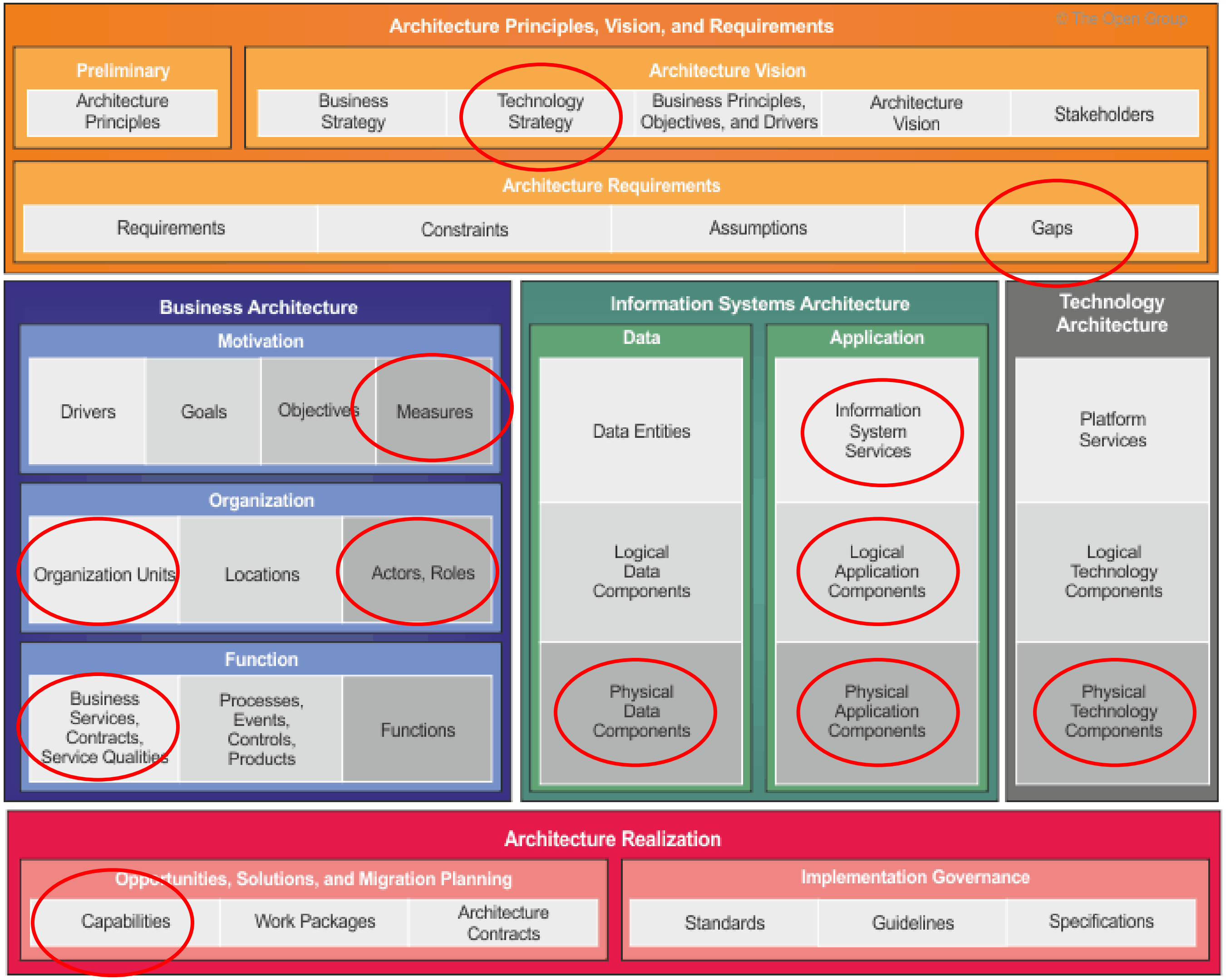
SLA needs to be developed and implemented from scratch

Also we can notice that TPAS is going to be discontinued in the future being replaced by Audit.

Description of Architectural Work Products


Architecture Deliverables
Architecture Deliverable
Artifacts
Monitoring
Business Architectural Design
Application Architectural Design
Reporting
Business Architectural Design
Data Architectural Design
SLA
Business Architectural Design
Application Architectural Design
Other Deliverables
Architecture Repository
Re-Usable Building Blocks
Monitoring Architectural Design
Reporting Architectural Design
Describing
Building Blocks
Architecture Deliverables

Architectural Content Metamodel



Architecture Principles, Vision, and Requirements

© The Open Group

Preliminary: Architecture Principles

Architecture Vision: Business Strategy; Technology Strategy; Business Principles, Objectives, and Drivers; Architecture Vision; Stakeholders

Architecture Requirements: Requirements; Constraints; Assumptions; Gaps

Business Architecture

Motivation: Drivers; Goals; Objectives; Measures

Organization: Organization Units; Locations; Actors, Roles

Function: Business Services, Contracts, Service Qualities; Processes, Events, Controls, Products; Functions

Information Systems Architecture

Data: Data Entities; Logical Data Components; Physical Data Components

Application: Information System Services; Logical Application Components; Physical Application Components

Technology Architecture: Platform Services; Logical Technology Components; Physical Technology Components

Architecture Realization

Opportunities, Solutions, and Migration Planning: Capabilities; Work Packages; Architecture Contracts

Implementation Governance: Standards; Guidelines; Specifications

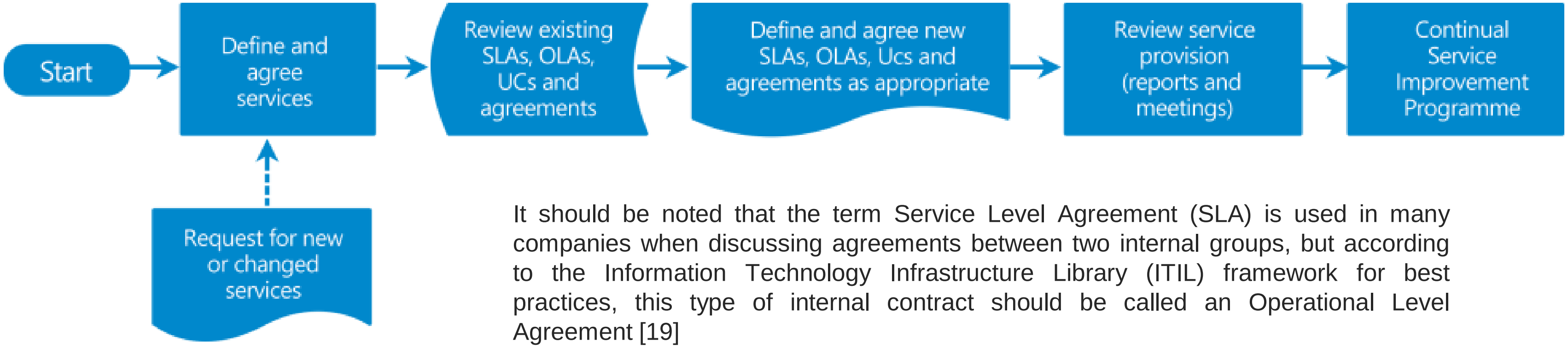
The content metamodel provides a definition of all the types of building blocks that may exist within an architecture, showing how these building blocks can be described and related to one another.

The highlighted areas with a red circle are going to be the main domains where architectural artifacts need to be delivered.

As we can notice from the associated graph, the SLA solution doesn't require an extensive design but we'll have to involve all the stakeholders and inform them about the implementation plan and account for interdependencies.

The content metamodel identifies all of these concerns (i.e., application, data entity, technology, actor, and business service), shows the relationships that are possible between them (e.g., actors consume business services), and finally identifies artifacts that can be used to represent them.[20]

Service Level Agreement Process


Start
Define and agree services
Review existing SLAs, OLAs, UCs and agreements
Define and agree new SLAs, OLAs, Ucs and agreements as appropriate
Review service provision (reports and meetings)
Continual Service Improvement Programme
Request for new or changed services

It should be noted that the term Service Level Agreement (SLA) is used in many companies when discussing agreements between two internal groups, but according to the Information Technology Infrastructure Library (ITIL) framework for best practices, this type of internal contract should be called an Operational Level Agreement [19]

This presentation is focused on SLAs, but there are two additional concepts in the same family that we want to be aware of, OLA and UC.

- Operational Level Agreement (OLA) – an agreement between an IT service provider and another department from the same organization, governing the delivery of infrastructure service.

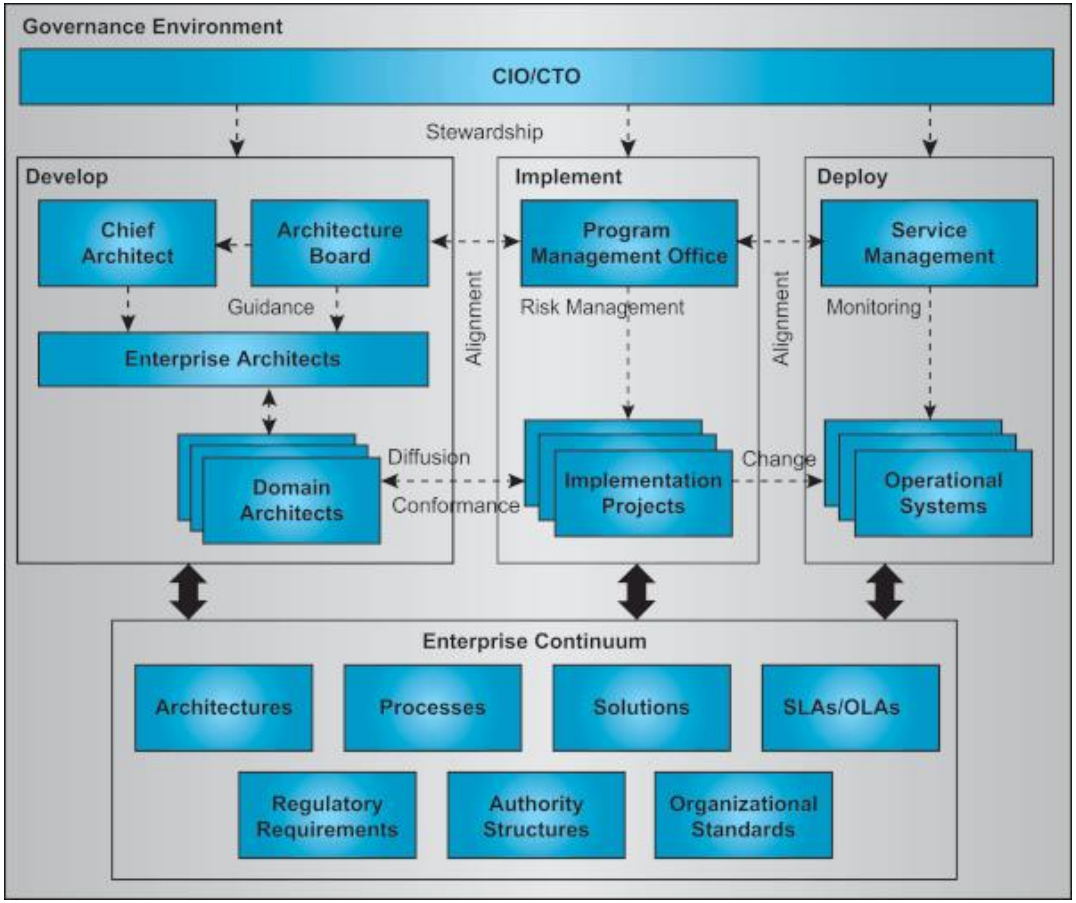
- Underpinning Contract (UC) – a formal contract between an IT service provider and an external provider of an IT or infrastructure service to deliver agreed level of Quality services or goods at specified time.

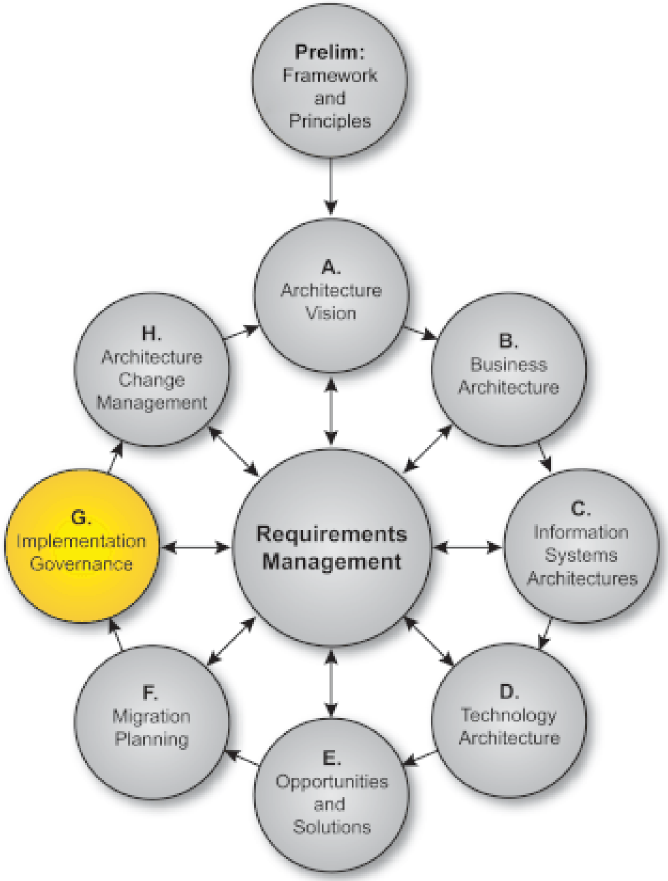
The Service Level Agreement architectural design implies the following:

If we do not outline **WHO** we support, then we support **EVERYONE**.
If we do not outline **WHAT** we support, then we support **EVERYTHING**.
If we do not outline **WHEN** we support, then we support **AROUND THE CLOCK**.
If we do not outline **WHERE** we support, then we support **EVERY LOCATION**.

The SLA clock **STARTS** when the service goes down, **NOT** when a ticket is logged or when customer first reports it!

SLA place in the enterprise architecture framework

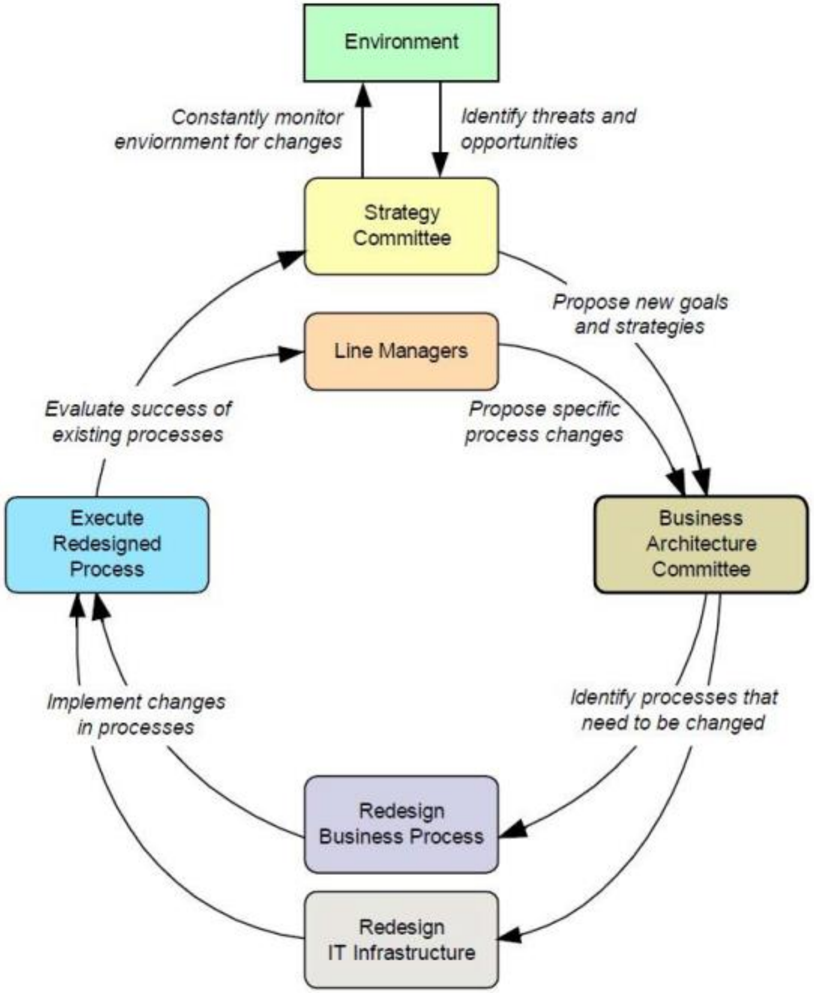




Prelim: Framework and Principles
A. Architecture Vision
B. Business Architecture
C. Information Systems Architectures
D. Technology Architecture
E. Opportunities and Solutions
F. Migration Planning
G. Implementation Governance
H. Architecture Change Management
Requirements Management

As illustrated in the above TOGAF - Architecture Governance Framework - Organizational Structure , the governance of the organization's architectures provides not only direct control and guidance of their development and implementation, but also extends into the operations of the implemented architectures. See also Appendix C for more details. [20]

Adding the SLA in the Architectural Life Cycle

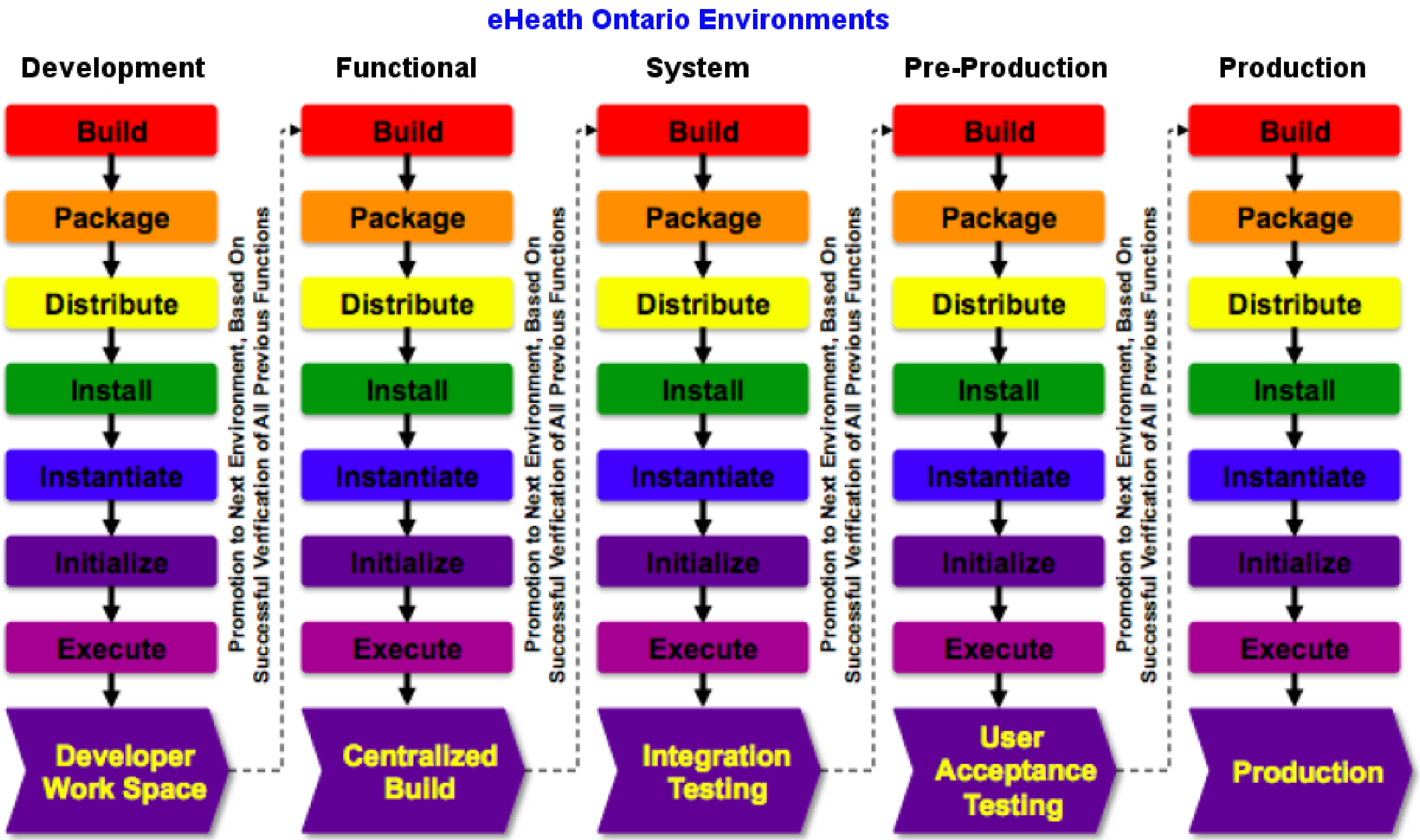

Environment
Constantly monitor envionment for changes
Identify threats and opportunities
Strategy Committee
Propose new goals and strategies
Line Managers
Propose specific process changes
Evaluate success of existing processes
Execute Redesigned Process
Business Architecture Committee
Identify processes that need to be changed
Implement changes in processes
Redesign Business Process
Redesign IT Infrastructure

The cycle displayed on the left side defines a simplified TOGAF process. From a process description point of view TOGAF excels, but we'll use a simplified version of it. It is not in our intend to go through the whole cycle TOGAF process describes, but we'll create a process that best suits the SLA implementation goal. We'll have as our target the integration of the new SLA solution into the existing service to be able to check and measure the overall environment health.

As mentioned before, the implementation of the SLA project relies 100% on the existing implemented services and on the existing eHealth rules and regulations. The existing architectural documents need to be revised to include the specifications related to the SLA, OLA and UC. We are going to implement all these specifications in a step by step approach starting with the business architectural requirement documents and continue with the process described in the graphic.

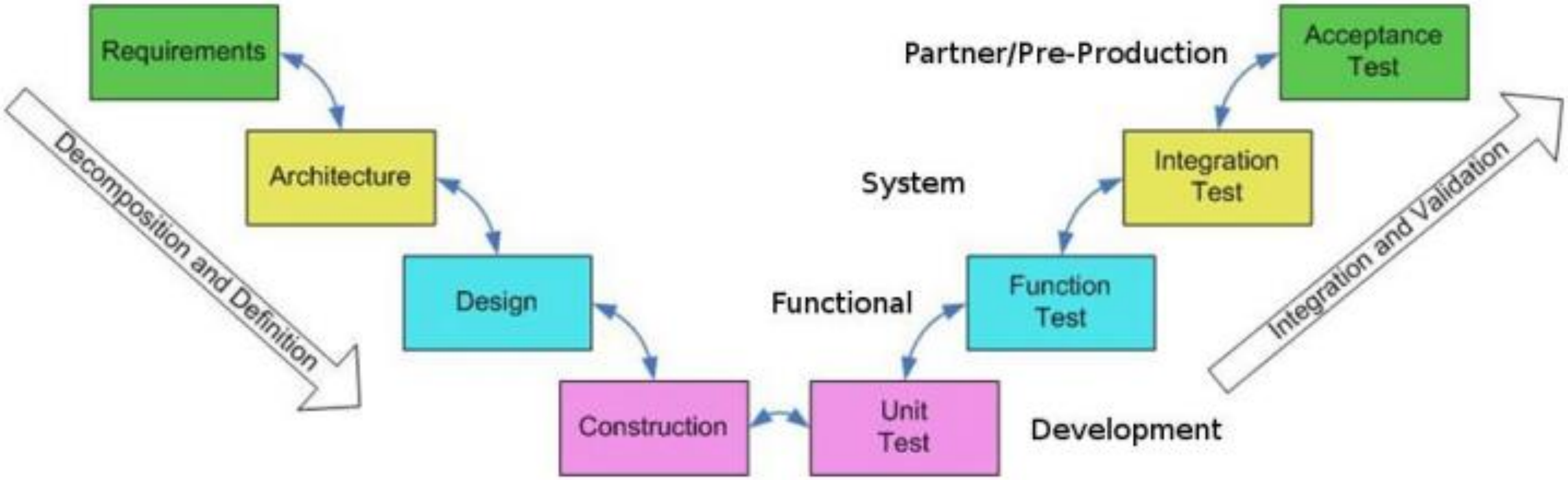
A new system and technology design is required for the SLA solution that will piggyback on the existing services. The what, how, where, who and why of the Zachman framework needs to be answered for each service covered by the SLA. This will result in a matrix that needs to be accounted for and consumed in the technical implementation design.

eHealth IT Environment Framework



Currently lower environments like Development, Functional, System Pre-Production are supported on a best effort basis with the exception of SLAs (which I'm not sure if defined) for environments that are client facing like Partners.

SDLC methodology and environment relationship



Environments are controlled areas where systems developers can build, distribute, install, configure, test, and execute systems that move through the SDLC. Each environment is aligned with different areas of the SDLC and is intended to have specific purposes.

E-Health Ontario defined for its internal use the ETF (Enterprise Test Framework). As it is always better to introduce testing in the early phase of SDLC, as in this model the testing activity gets started from the early phase of the SDLC. Before starting the actual testing, testing team has to work on various activities like preparation of Test Strategy, Test Planning, Creation of Test cases & Test Scripts which is work parallel with the development activity which help to get the test deliverable on time.

In the V Model Software Development Life Cycle, based on same information(requirement specification document) the development & testing activity are both started simultaneous. Based on the requirements document the development team starts working on the design & after completion of the design starts the actual implementation while the testing team starts working on test planning, test case writing and test scripting. Both development and quality assurance activities are performed in parallel.

At the core of the Enterprise Testing Framework is the methodology used to deliver testing functions to verify and validate functionality within a system or solution. The ETF methodology is based on the “V Model’ delivery framework for all testing functions provided at eHealth Ontario, by Quality and Release Management. The V Model, as applied at eHealth Ontario, presents testing activities and functions in alignment with Unified Project Governance Gating phases which in turn support the overall delivery of testing in the form of verification and validation activities.

The eHealth Ontario Enterprise Testing Framework

To align with the eHealth Ontario Project Management Life Cycle (PMLC) as governed by the Unified Project Governance Gating process, the following figure presents a view that will be used in the planning, development and execution of testing functions for both projects and continued enhancements work streams. This view will support the overall project delivery planning efforts that are assumed to be taken when delivering a project and/or a solution enhancement.

Within each of the current project life cycle phases, specific Test Team inputs and deliverables are to be provided to support the project governance and gating process. At a high level, each project phase (gate) and Test Team deliverable or input is described as follows.

Gate 0 - Opportunity

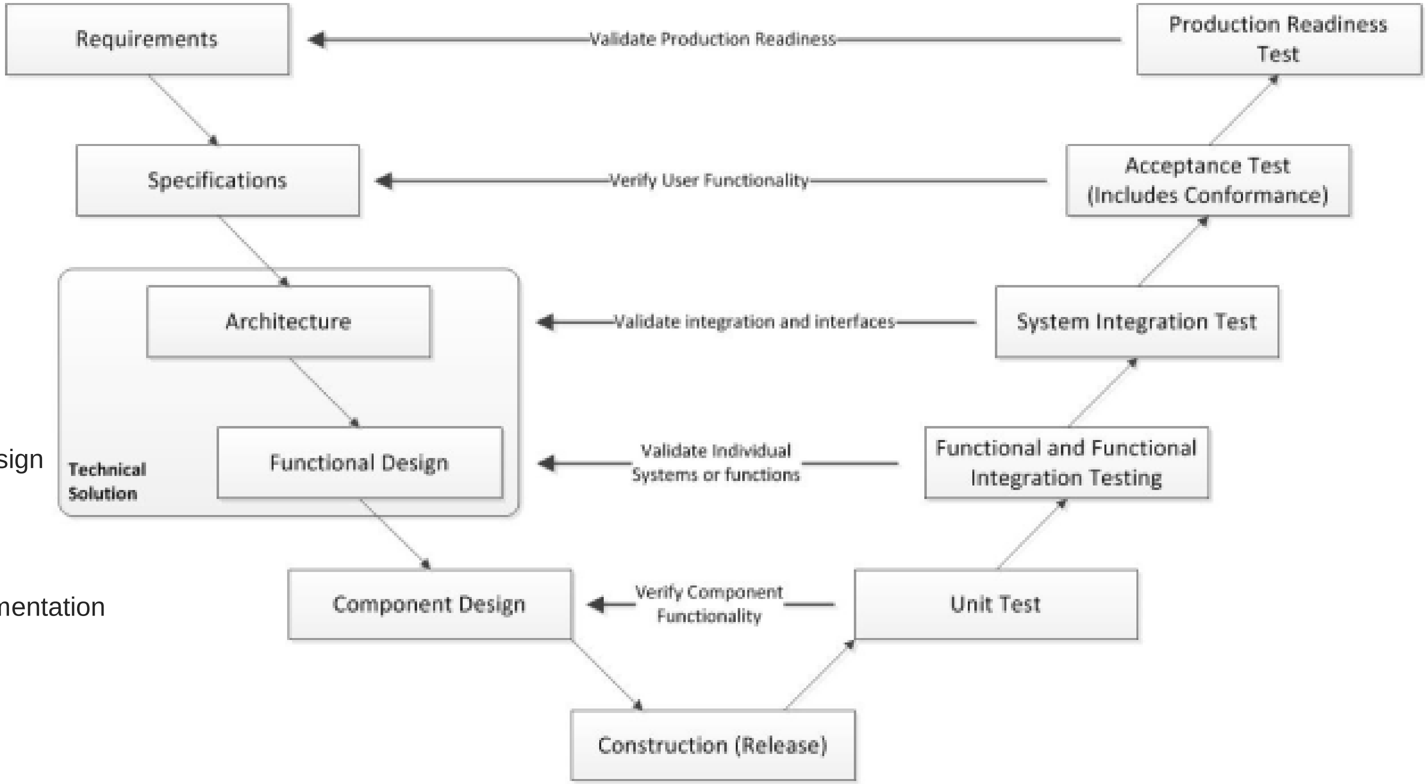
Gate 1 – Approval

Gate 2 - Definition

Gate 3 – Planning and Design

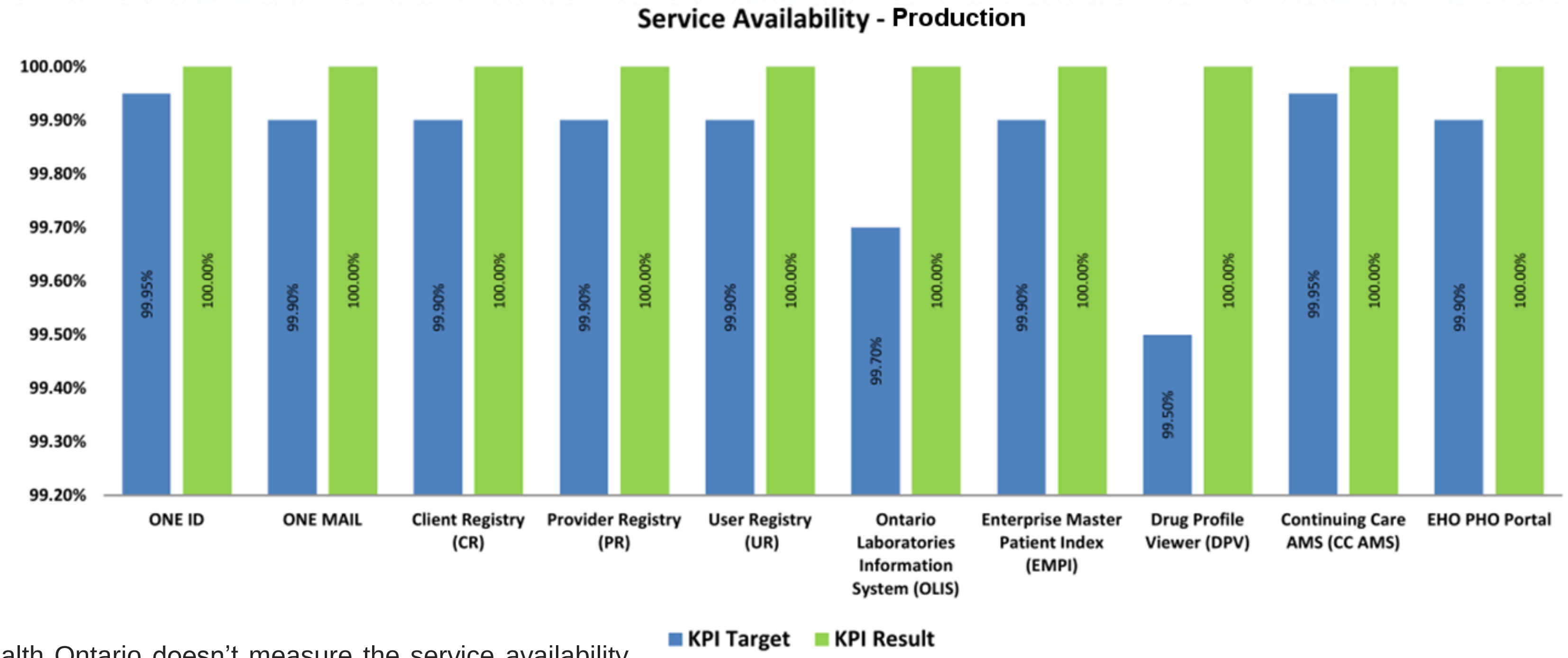
Gate 4 – Build and Implementation

Gate 5 – Close Out



Requirements
Validate Production Readiness
Production Readiness Test
Specifications
Verify User Functionality
Acceptance Test (Includes Conformance)
Architecture
Validate integration and interfaces
System Integration Test
Technical Solution
Functional Design
Validate Individual Systems or functions
Functional and Functional Integration Testing
Component Design
Verify Component Functionality
Unit Test
Construction (Release)

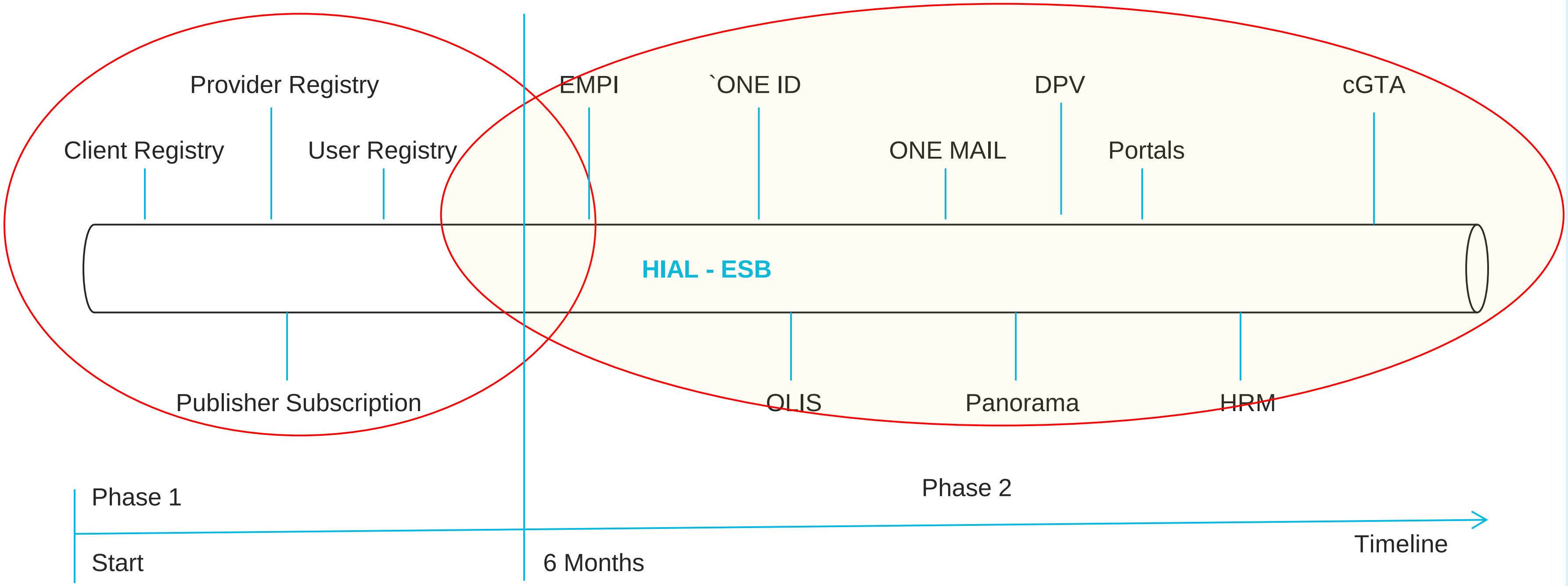
Current state evaluation



eHealth Ontario doesn’t measure the service availability on all components in the lower environments. Therefore we don’t have a picture that might reveal some numbers in support to the current state evaluation. Therefore we will use the existing current capabilities existing in the Production environment to measure the identified principles like: Auditing, Monitoring, Logging and Reporting. Within the Production BI environment, the above KPIs was captured and might be presented by scorecards, dashboards or other simple graphical readouts.

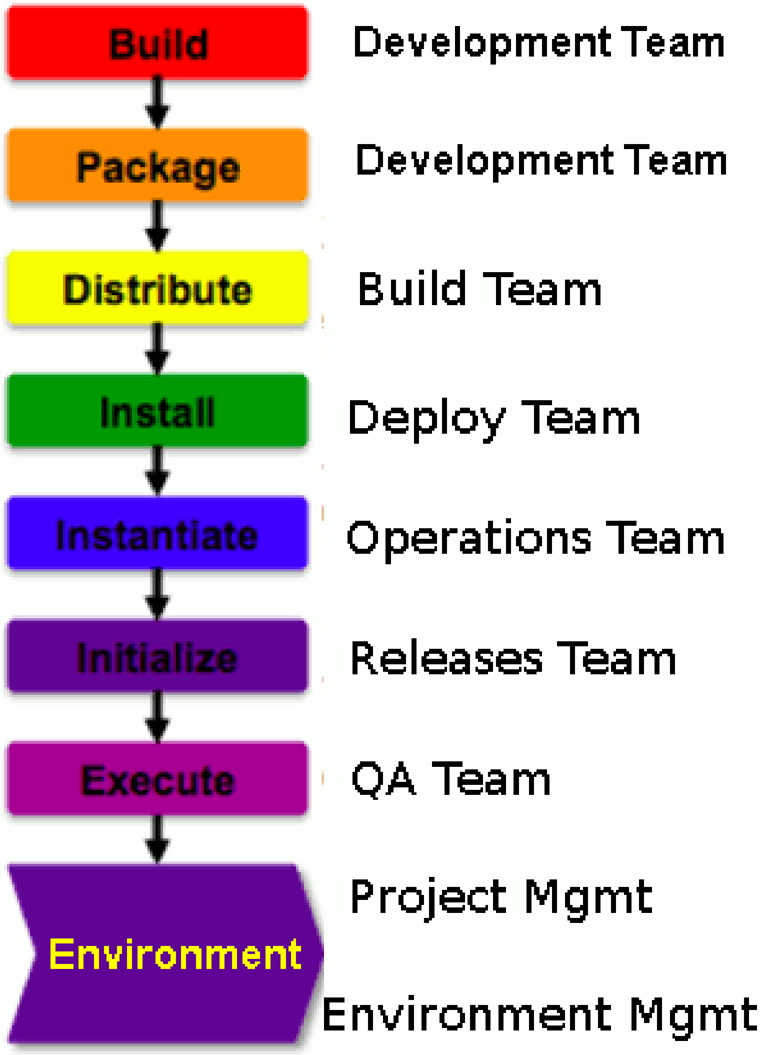
Env/Capability	Auditing	Monitoring	Logging	Reporting	SLA
Development	N	N	N	N	N
Functional	Y	N	N	N	N
System	Y	N	N	N	N
Partners	Y	N	Y	N	N
Pre-Production	Y	Y/N	Y	N	N
Production	Y	Y	Y	Y	N

SLA Implementation Roadmap



The implementation of the SLA for the above mentioned services will be dived into two parts. The first and most important part will be implementing the common shared services. In Phase 2 we'll based on the feedback we've got we'll redefine our strategy. Once the resources have been identified and the project is sponsored the implementation Phase 1 shall not exceed six months.

SLM Environment usage



As seen in the associated chart on the left, there are a number of teams working together towards the same goal. In a SOA environment teams are not always located on the same site. As a matter of fact they don’t even have to know each other, hence the need to closely monitor their activity. What a better tool to achieve this goal than implementing a SLA capability in the Health Information Access Layer (HIAL).

The GTA HIAL links the applications, integration engines and data repositories from across the GTA to form an integrated system. The HIAL will provide a set of communication and integration services, as well as Business Process Orchestration and rules.

As all these services are processed and orchestrated by a DataPower device, we can define the following types of SLM statements to manage a service level agreement (SLA) in our environment:

SLM statements to monitor requests and failures during a predefined interval at a specific level in the tree hierarchy. We can manually create or have the appliance auto-generate the following level-specific statements:

- A global statement monitors all transactions.
- A WSDL-specific statement monitors all Web services in a specific WSDL file. If the Web Service Proxy is based on a single WSDL file, this statement and the global statement are equivalent.
- A service-specific statement monitors a single Web service.
- A port-specific statement monitors a single Web service port.
- An operation-specific statement monitors a single Web service operation.

Custom SLM statements to provide more precise control over monitored transactions

To manage the SLA across multiple appliances, we can define an SLM peer group.

Record the Terms of the Agreement

eHealth Ontario Production SLA outlines the roles and responsibilities for both the customer and the service provider including definitions of terms like contract duration, locations and service times. For instance the duties of the service provider, duties of the customer, responsibilities of service users (e.g. with respect to IT security), IT Security aspects to be observed (if applicable, references to relevant IT Security Policies)

The detail of the SLA document, listing each service in a standard format contains the description, delivery point, availability, quality levels, measurement procedures and escalation procedures.

Currently lower environments are supported on *a best effort basis* with the exception of SLAs (which I'm not sure if defined) for environments that are client facing.

Recovery Point Objective (RPO) stands for maximum tolerable period during which data might be lost from an IT service due to a major incident.

Recovery Time Objective (RTO) is the duration of time and service level within which a business process must be restored after a disaster or disruption, to avoid unacceptable consequences associated with a break in business continuity.

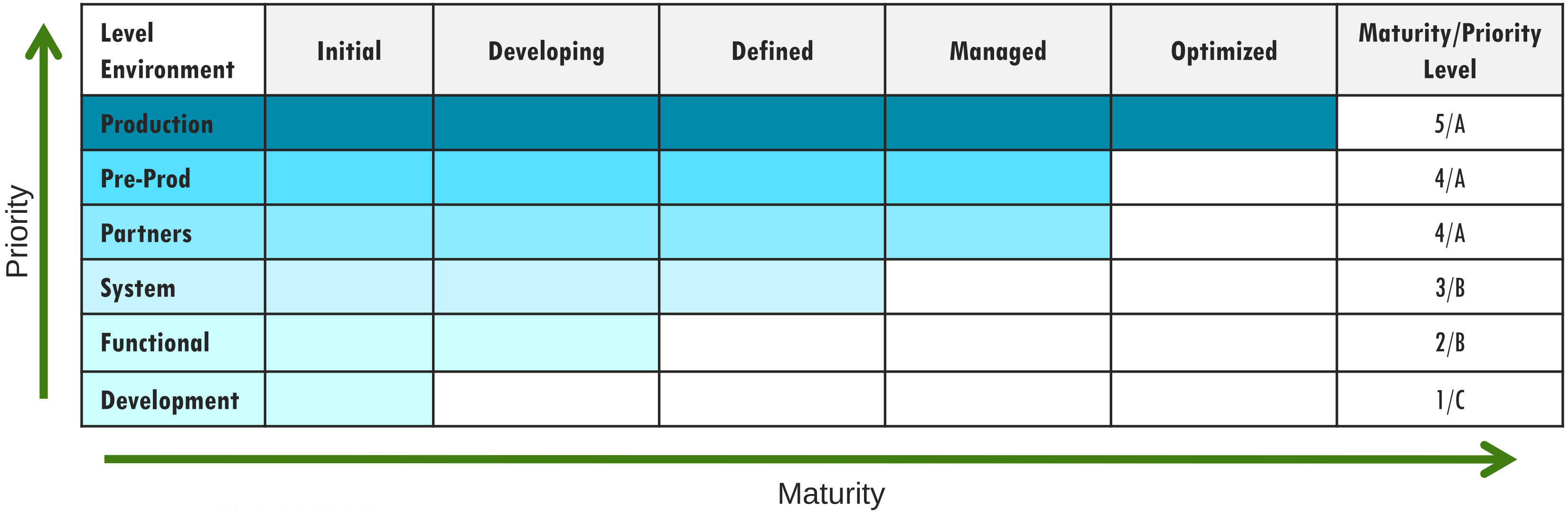
RTO is differentiated by priorities and for PCR everything is high priority. This is covering just the Production environment. Lower environments like Pre-Production, Partners, Functional, System and Development are covered by lower priority indicators.

IT Service	Production support group	Priority	RTO	RPO

Service or application provided to clients by eHealth Operations	Group responsible for providing support for this IT Service	A = Critical
		B = Medium
		C = Non-critical
			Priority A = 0 – 6 hrs
			Priority B = 6 - 24 hrs
			Priority C = 24 - 72 hrs
				Priority A = 0 – 6 hrs
				Priority B = 6 - 24 hrs
				Priority C = 24 - 72 hrs

Provincial Client Registry	PCR, Unix and Oracle Technical Support	A	0 – 6 hrs	0 hrs

Environment specific RTO, RPO and Priorities



Level Environment	Initial	Developing	Defined	Managed	Optimized	Maturity/Priority Level
Production	■	■	■	■	■	5/A
Pre-Prod	■	■	■	■		4/A
Partners	■	■	■	■		4/A
System	■	■	■			3/B
Functional	■	■				2/B
Development	■					1/C

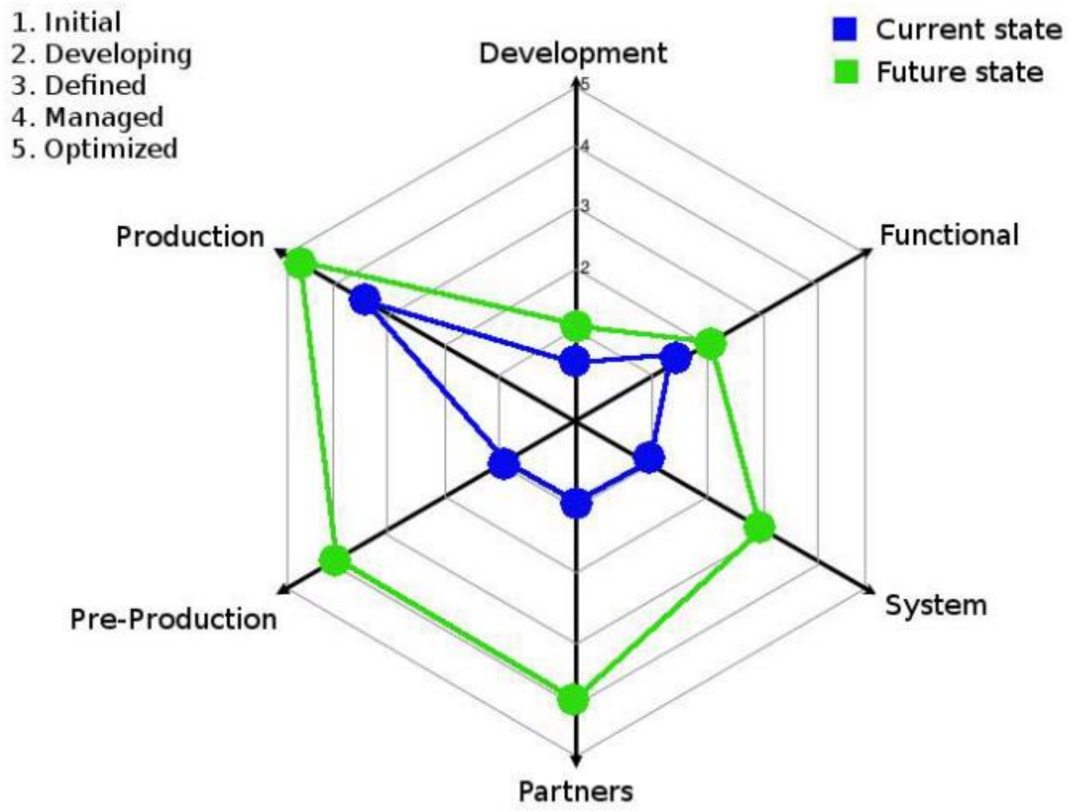
IT Service	Functional support group	Priority	RTO	RPO

Service or application provided to clients by eHealth Operations	Group responsible for providing support for this IT Service	A = Critical
		B = Medium
		C = Non-critical
			Priority A = 0–6 hrs
			Priority B = 6-24 hrs
			Priority C = 24-72 hrs
				Priority A = 0–6 hrs
				Priority B = 6-24 hrs
				Priority C = 24-72 hrs

Provincial Client Registry	PCR, Unix and Oracle Technical Support	B	6-12 hrs	6 hrs

Generally speaking the higher the importance / maturity of an environment the higher the priority level is. Anyway, breaking down the IT Service on components and sub-components we'll see that this rule doesn't apply uniformly on different services like the infrastructure ones or the database ones. There might be different levels of priority that cover the OLA and the UC as well.

Current and Future State of eHealth Environments

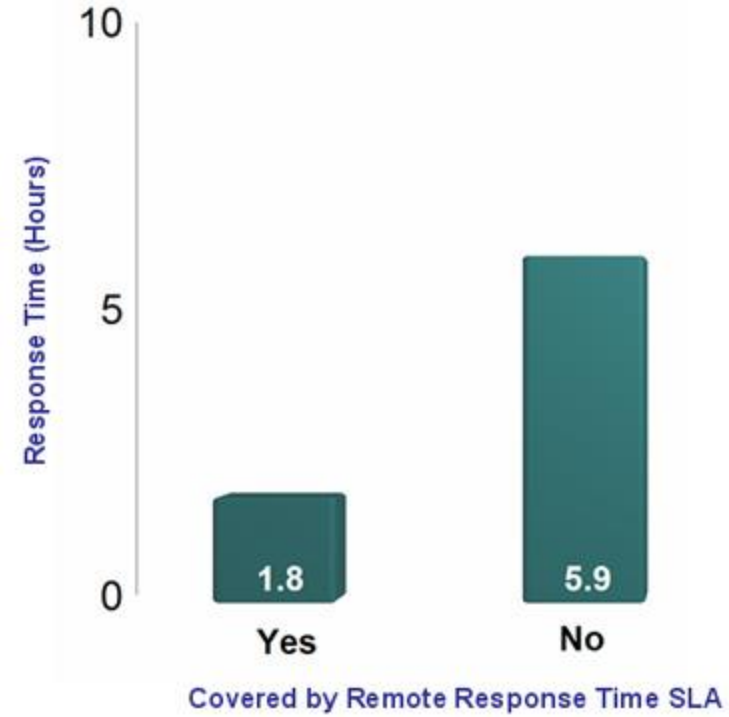


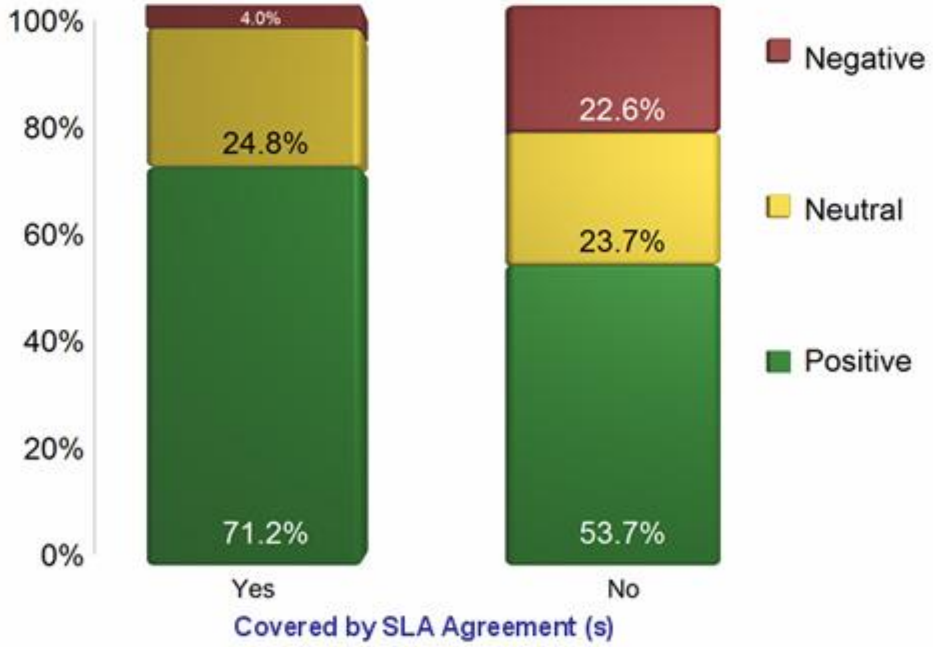
Most of the development, build and testing activities are happening in the lower environments where code deployments are more frequent, hence the need to regulate the engagement between teams and groups.

The current state (blue line) of the ITIL Spider web is measured based on the previously identified capabilities and represents present eHealth environment status.

Future state (green line) represents the desired capabilities implemented in the eHealth environment by measuring these capabilities after introducing the SLA agreement. These capabilities are gradually increasing from a level 1 to a level 5 as required by business and defined in the CMMI and SDLC.

The Benefits of Offering SLAs





The very existence of SLAs significantly helps vendors assure a positive service experience. These positive feelings are further compounded when SLAs are met. Customers are generally content with their vendors' SLA compliance levels, but have very little tolerance for missed performance goals. Conversely, customers that are not covered by SLAs are likely to give their vendors far more latitude in service-level performance. Still, when SLAs are not in place, the chance of a negative service experience increases significantly.

SLAs offer significant benefits to a service provider by helping them set and manage customer expectations and are integral to conveying the full value of services. Moreover, the ability to meet service-level agreements is key to providing a positive service experience. Although it can be a daunting step to introduce SLAs, it is not a commitment to deliver the impossible. A service level agreement can be as informal as a performance target or as rigid as a committed time to restore a system to operation backed by penalties. In either case the SLA serves as a basis for establishing a shared understanding of the service relationship. When properly developed SLAs offer a win-win situation for both the service provider and the customer.[21]

Any Questions, Comments or Queries ?

Next we are talking about the Physical Proposed Solution - DataPower

DataPower - SLA, SLD, and SLM Policies

First, we must define common terminology used when building SOA policy solutions. This section presents different terms, which are often used while dealing with policy and governance of business services. The most important are the following:

- Service Level Agreements (SLA)
- Service Level Definitions (SLD)
- DataPower configuration artifacts (such as Service Level Management policies)

What is a Service Level Agreement?

A Service Level Agreement (SLA) is a negotiated and formally defined agreement between two parties, where one is the (service) consumer and other is the (service) provider. It records a common understanding about the following areas of an agreement:

- Services
- Priorities
- Responsibilities
- Guarantees
- Warrantees

The SLA management process includes the following activities:

- SLA contract definition (basic schema with the quality of service parameters)
- SLA negotiation
- SLA enforcement, according to defined policies and metrics
- SLA monitoring

The concept of SLAs is important to DataPower because SOA appliances are regularly used as a Policy Enforcement Point (PEP) in SOA solutions.

There is no industry-wide standard today for expressing SLA agreements within SOA components (such as Policy Administration Point and Policy Enforcement Point). Therefore, custom solutions are common and can be non-portable between service gateways.

What is a Service Level Definition?

A Service Level Definition (SLD) defines the capabilities of a provider to deliver a service in compliance with conditions that its owner has defined as required for protecting the service endpoint.

An SLD is not negotiated as it is defined solely by the service provider owner. It is the responsibility of a service provider to deliver a level of service that meets all terms and conditions defined in its SLD. DataPower performs the enforcement of the SLD policy requirements and acts as a protector of the backend service endpoint.

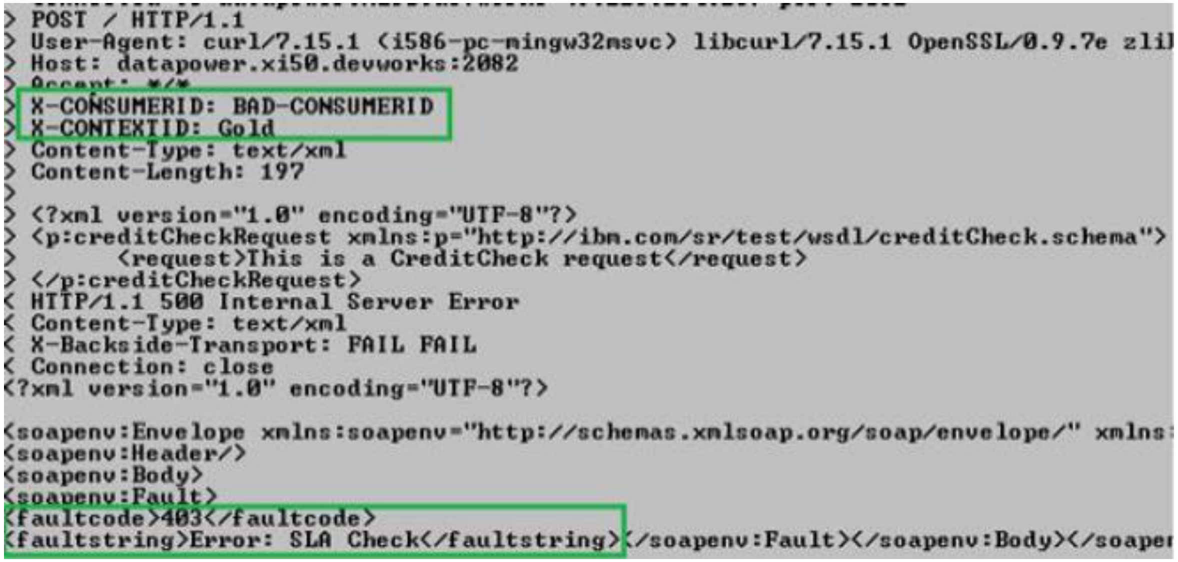
The SLD management process includes the following activities:

- SLD definition (level and quality of service, security, and so on)
- SLD enforcement, according to defined policies
- SLD monitoring

What is a Service Level Management policy?

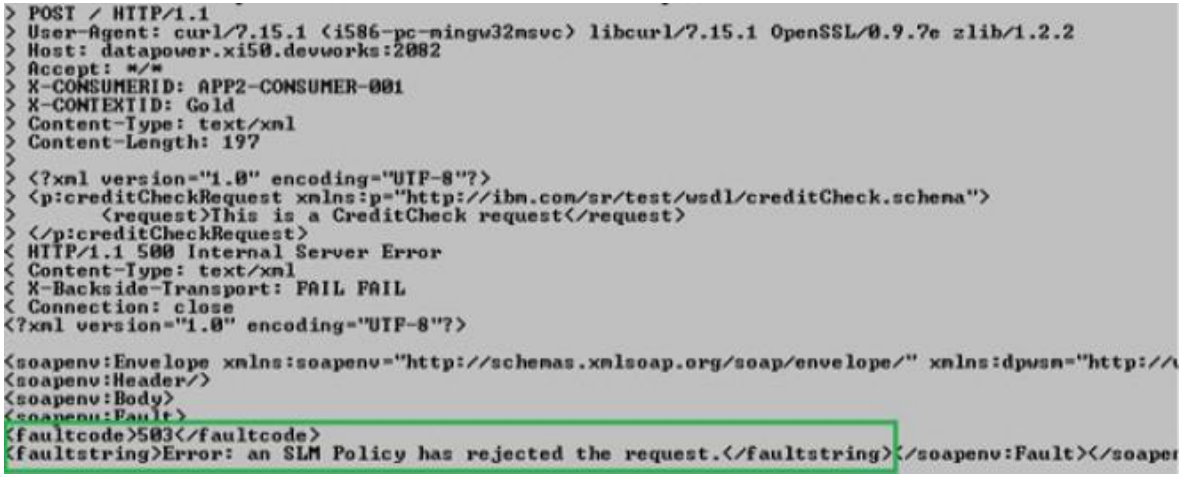
A Service Level Management (SLM) policy is a DataPower configuration artifact that you can configure to enforce an SLA or SLD policy requirement. This DataPower configuration policy is exclusively based on DataPower configuration property details and is usually created by the Policy Developer responsible for translating SLA and SLD policies requirements into operational configuration in DataPower.

SLA, SLM policy throttling



```
> POST / HTTP/1.1
> User-Agent: curl/7.15.1 (i586-pc-mingw32msvc) libcurl/7.15.1 OpenSSL/0.9.7e zlil
> Host: datapower.xi50.devworks:2082
> Accept: */*
> X-CONSUMERID: BAD-CONSUMERID
> X-CONTEXTID: Gold
> Content-Type: text/xml
> Content-Length: 197
>
> <?xml version="1.0" encoding="UTF-8"?>
> <p:creditCheckRequest xmlns:p="http://ibm.com/sr/test/wsdl/creditCheck.schema">
>       <request>This is a CreditCheck request</request>
> </p:creditCheckRequest>
< HTTP/1.1 500 Internal Server Error
< Content-Type: text/xml
< X-Backside-Transport: FAIL FAIL
< Connection: close
<?xml version="1.0" encoding="UTF-8"?>

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:
<soapenv:Header/>
<soapenv:Body>
<soapenv:Fault>
<faultcode>403</faultcode>
<faultstring>Error: SLA Check</faultstring></soapenv:Fault></soapenv:Body></soaper
```

XML response received in the case of an SLA check failure



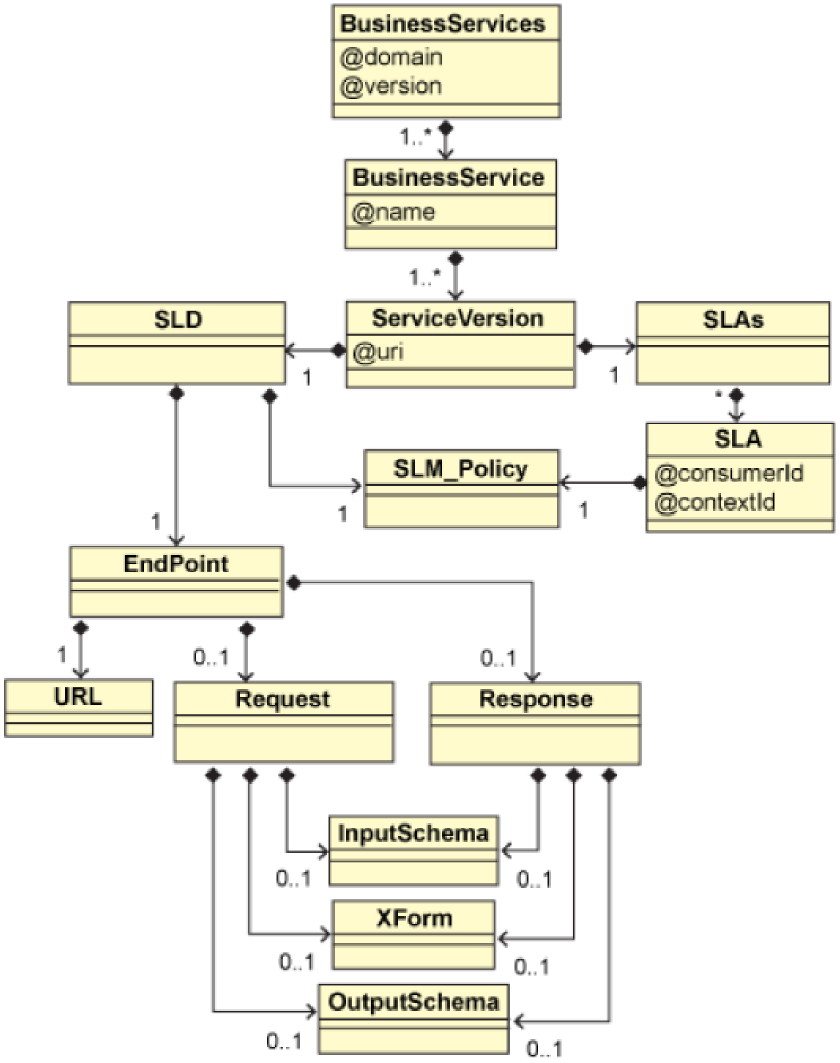
```
> POST / HTTP/1.1
> User-Agent: curl/7.15.1 (i586-pc-mingw32msvc) libcurl/7.15.1 OpenSSL/0.9.7e zlib/1.2.2
> Host: datapower.xi50.devworks:2082
> Accept: */*
> X-CONSUMERID: APP2-CONSUMER-001
> X-CONTEXTID: Gold
> Content-Type: text/xml
> Content-Length: 197
>
> <?xml version="1.0" encoding="UTF-8"?>
> <p:creditCheckRequest xmlns:p="http://ibm.com/sr/test/wsdl/creditCheck.schema">
>       <request>This is a CreditCheck request</request>
> </p:creditCheckRequest>
< HTTP/1.1 500 Internal Server Error
< Content-Type: text/xml
< X-Backside-Transport: FAIL FAIL
< Connection: close
<?xml version="1.0" encoding="UTF-8"?>

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:dpwsm="http://
<soapenv:Header/>
<soapenv:Body>
<soapenv:Fault>
<faultcode>503</faultcode>
<faultstring>Error: an SLM Policy has rejected the request.</faultstring></soapenv:Fault></soaper
```

If an SLM policy specified at the SLA or SLD level is not respected, then the request is rejected, as shown below

In this example, the SOAP faults are returned to the consumer application in case of problems.

Another solution is to return a simple HTTP status code bound to a specific reason phrase.

DataPower - SLA Control File



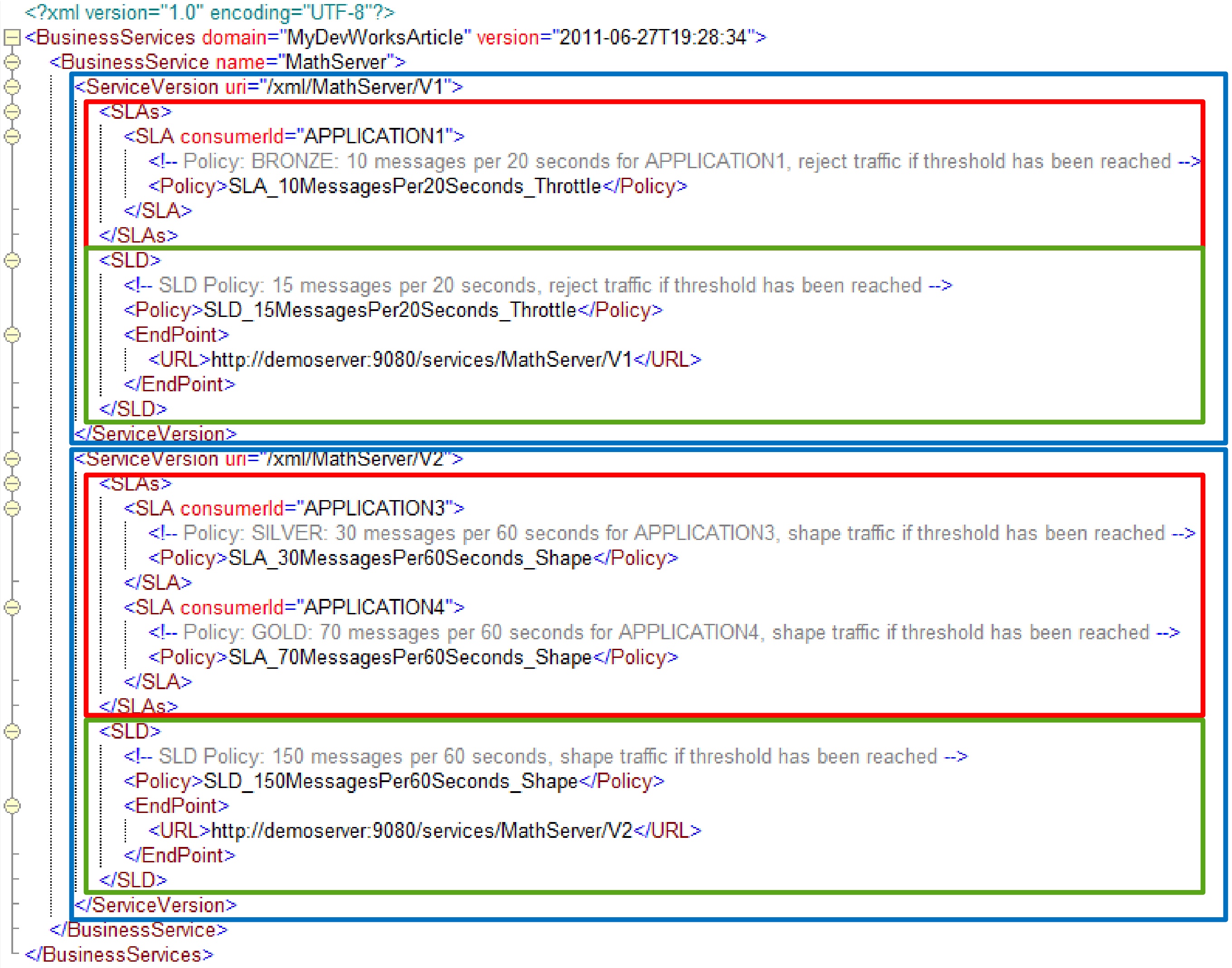
The SLA Control File is composed of a service catalog (/BusinessServices) that can be qualified by the domain and version information. The service catalog contains multiple services (/BusinessServices/BusinessService). Every business service (identified by a unique name) supports one or more service versions (ServiceVersion).

Each service version is identified by a unique URI. Individual service versions contain a single SLD along with any number of SLAs. The SLD defines a service endpoint (EndPoint) and a specific SLM Policy to protect the backend server. A service endpoint must include the required URL, whereas other elements are optional.

Class model description of elements and attributes.

/BusinessServices - The catalog of business services for which SLA and SLD must be enforced.
@domain - The DataPower domain on which the SLA Control File is applied.
@version - The current version of the file or its creation date.
//BusinessService - A business service instance.
//BusinessService/@name - Name (identifier) of a business service.
ServiceVersion - A specific business service version instance.
@uri - Service version URI. Every service version must have a unique URI.
SLAs - Unbounded list of SLAs that must be enforced when the business service is requested.
@consumerId - Required Identifier of the consumer assoc. with SLA
@contextId - Identifier of the context associated with SLA.
//EndPoint - Declaration of the service version endpoint.
//URL - Endpoint URL of the service version backend.

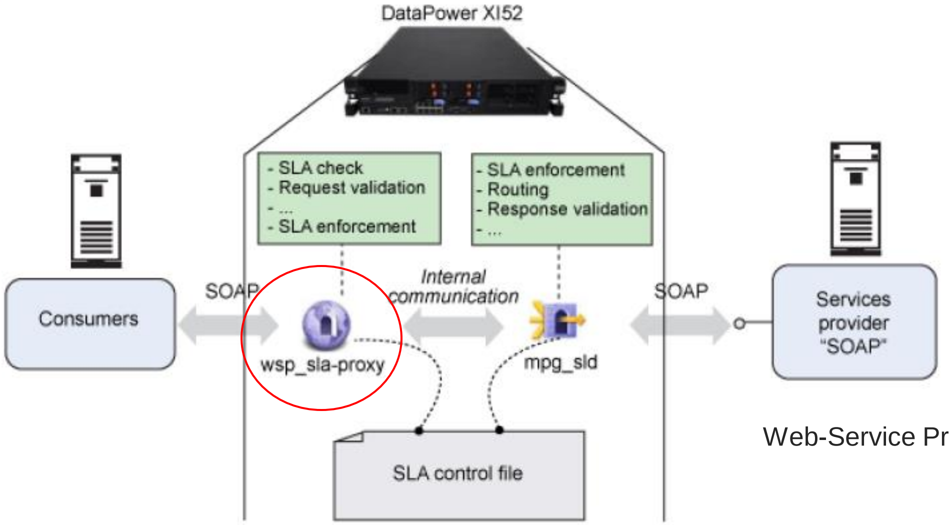
DataPower - SLA Control File Sections



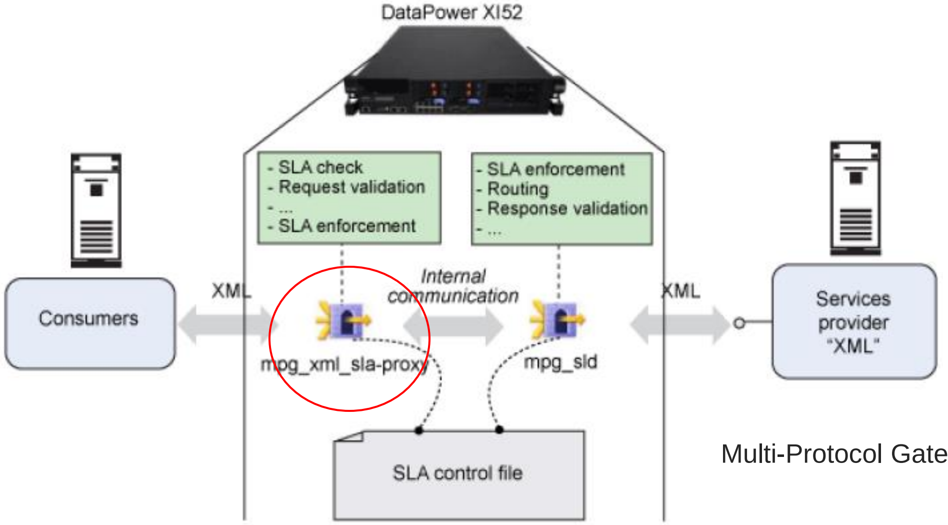
```xml
<?xml version="1.0" encoding="UTF-8"?>
<BusinessServices domain="MyDevWorksArticle" version="2011-06-27T19:28:34">
  <BusinessService name="MathServer">
    <ServiceVersion uri="/xml/MathServer/V1">
      <SLAs>
        <SLA consumerId="APPLICATION1">
          <!-- Policy: BRONZE: 10 messages per 20 seconds for APPLICATION1, reject traffic if threshold has been reached -->
          <Policy>SLA_10MessagesPer20Seconds_Throttle</Policy>
        </SLA>
      </SLAs>
      <SLD>
        <!-- SLD Policy: 15 messages per 20 seconds, reject traffic if threshold has been reached -->
        <Policy>SLD_15MessagesPer20Seconds_Throttle</Policy>
        <EndPoint>
          <URL>http://demoserver:9080/services/MathServer/V1</URL>
        </EndPoint>
      </SLD>
    </ServiceVersion>
    <ServiceVersion uri="/xml/MathServer/V2">
      <SLAs>
        <SLA consumerId="APPLICATION3">
          <!-- Policy: SILVER: 30 messages per 60 seconds for APPLICATION3, shape traffic if threshold has been reached -->
          <Policy>SLA_30MessagesPer60Seconds_Shape</Policy>
        </SLA>
        <SLA consumerId="APPLICATION4">
          <!-- Policy: GOLD: 70 messages per 60 seconds for APPLICATION4, shape traffic if threshold has been reached -->
          <Policy>SLA_70MessagesPer60Seconds_Shape</Policy>
        </SLA>
      </SLAs>
      <SLD>
        <!-- SLD Policy: 150 messages per 60 seconds, shape traffic if threshold has been reached -->
        <Policy>SLD_150MessagesPer60Seconds_Shape</Policy>
        <EndPoint>
          <URL>http://demoserver:9080/services/MathServer/V2</URL>
        </EndPoint>
      </SLD>
    </ServiceVersion>
  </BusinessService>
</BusinessServices>
```

/SLAs/SLA - Declaration of an SLA instance.
/ServiceVersion/SLD - SLD instance declaration, which must be enforced for service version.
/SLD/Policy - Name of the SLD SLM Policy, which must be enforced when service version is requested.
/SLD/EndPoint - Declaration of the service version endpoint.
/SLD/EndPoint/URL - Declaration of the service version endpoint.
/InputSchemaURL – This optional element references a schema or WSDL file (in case of a SOAP message validation), which is used to validate the incoming message during response processing.
/OutputSchemaURL - This optional element references a schema or WSDL file (in case of a SOAP message validation), which is used to validate the outgoing message during response processing.

DataPower - Services to consume the SLA Control File



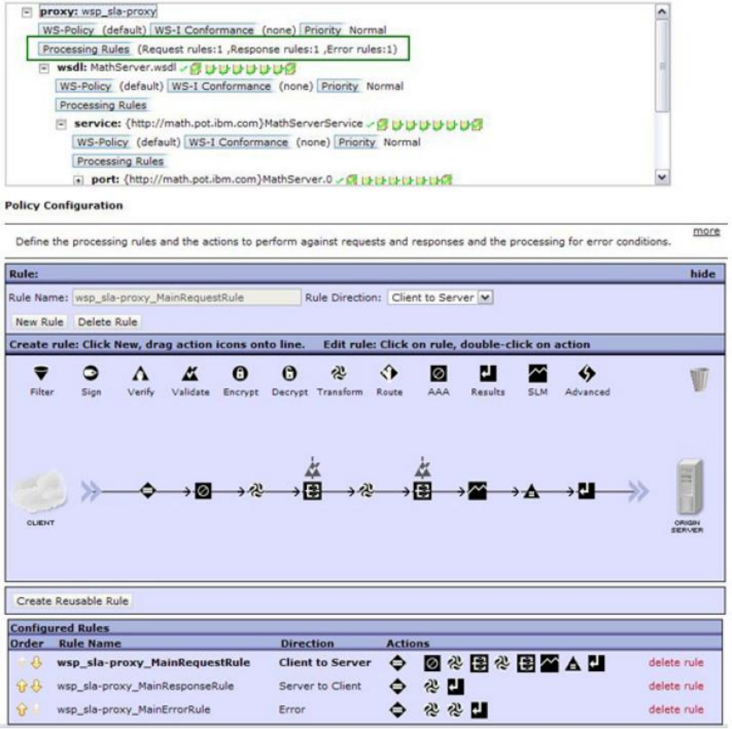
Web-Service Proxy



Multi-Protocol Gateway

Processing rules of the Web-Service Proxy based solution

When dealing with SOAP Web-Services in the SLA Control File, you must reference WSDL files and not XSD schemas, using <inputSchemaURL> and <outputSchemaURL> elements.



proxy: wsp_sla-proxy
WS-Policy (default) WS-I Conformance (none) Priority Normal
Processing Rules (Request rules:1 ,Response rules:1 ,Error rules:1)
wsdl: MathServer.wsdl
WS-Policy (default) WS-I Conformance (none) Priority Normal
Processing Rules
service: {http://math.pot.ibm.com}MathServerService
WS-Policy (default) WS-I Conformance (none) Priority Normal
Processing Rules
port: {http://math.pot.ibm.com}MathServer.0

Policy Configuration

Define the processing rules and the actions to perform against requests and responses and the processing for error conditions.

more

Rule: hide

Rule Name: wsp_sla-proxy_MainRequestRule Rule Direction: Client to Server

New Rule Delete Rule

Create rule: Click New, drag action icons onto line. Edit rule: Click on rule, double-click on action

Create Reusable Rule

Configured Rules

Order	Rule Name	Direction	Actions	
	wsp_sla-proxy_MainRequestRule	**Client to Server**		delete rule
	wsp_sla-proxy_MainResponseRule	Server to Client		delete rule
	wsp_sla-proxy_MainErrorRule	Error		delete rule

Dan Birsan, dan.birsan@gmail.com, 647-886-0864 - qq094401

Appendix A - Principles

Information and Key Architectural Principles and Decisions

- Information needs are business-driven
- Information is a public asset
- Information is shared
- Information is accessible (for those authorized to use it)
- Information is protected
- Information is managed using a life cycle approach
- Information is managed in an integrated manner
- Information needs to be integrated to support better decision making
- Information management is everyone's business objectives.

Integration, interoperability and reusability: systems will be constructed with methods that substantially improve interoperability and the reusability of components.

Standards and open systems: Design choices should be prioritized toward open systems and the creation of adaptable, flexible, and interoperable, vendor-neutral solutions.

Availability, scalability, reliability, and maintainability: to ensure high availability of EHR information and services, reliability and availability must be part of their design.

Mainstream solutions: production IT solutions used in the EHR should use industry-proven, mainstream technologies except in those areas where advanced higher-risk solutions provide a substantial benefit.

Privacy and security: EHR components will be built and/or procured to comply with the privacy and security requirements defined in Ontario law, and will employ adequate safeguards to protect the information they contain and the services they provide and to defend against the broadest possible range of vulnerabilities.

Service-oriented architecture: EHR solutions should be designed using service-oriented architecture principles, enabling reusability so that they can be leveraged or extended.

Technological and operational convergence: EHR solutions should be designed with lower operational complexity in terms of technology, process, systems, and operations, to ensure higher stability, reduced cost, and enhanced delivery and operational capabilities.

Leverage centres of expertise and build from success: promote the use of best practices and reuse of available artifacts, components, services, and processes.

Provide multiple ways to interact with services: for example, interaction with EHR services may be provided through portlets in a portal, web services, and other means including mobile devices.

Support versioning and migration: service interfaces are based on standards and long-standing business processes; however, even the most solid standards change over time. By supporting versions of service interfaces, the HIAL allows for new systems to be brought online to consume new features in the EHR without breaking legacy functionality, while legacy features can be phased out in a predictable manner.

Direct path: when traversing the integrated components of the EHR structure, services should be consumed through the most direct path possible.

Appendix B - Standards and Patterns

The eHealth Standards Program is responsible for the interoperability standards used in the various EHR solutions. The standards fall into these three main categories.

- Messaging Standards such as HL7 v2 and v3, DICOM, and SOAP
- Content Standards such HL7 Clinical Document Architecture (CDA), RESTful approaches, and DICOM
- Terminology Standards such as SNOMED CT, LOINC, pCLOCD, ICD, and CCI

Transaction patterns establish a common set of interoperability processes and behaviours that can be applied to line of business applications invoked within the EHR. This creates predictable system behaviours by relying on a limited number of ways to connect to and use EHR services. Patterns also inform the interoperability requirements for point of service applications and EHR components, defining the responsibilities of the sending and receiving components, and how they should interact.

The patterns are founded on three common types of interactions between point of service applications and the Ontario HIAL Solution (and the services offered through it), including data, portlet, and publish-and-subscribe interactions.

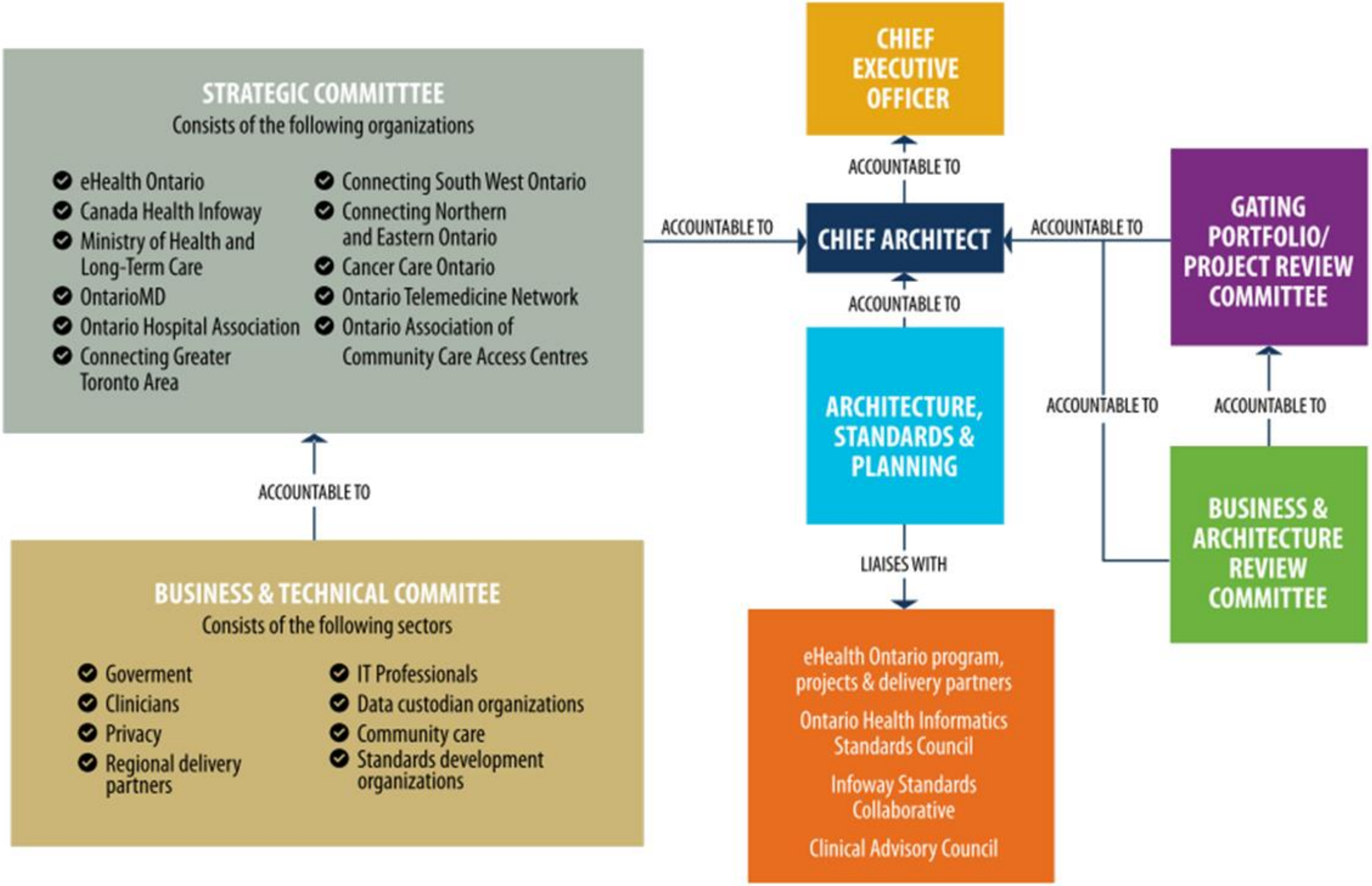
- Data interactions: the exchange of EHR data between point of service applications and the Ontario HIAL Solution through exposed data services
- Portlet interactions: build on the data interactions with the use of portals at the point of service by exchanging EHR data through portlet services
- Publish and Subscribe interactions: a pattern in which a person or application publishes information, triggering an event notification (with or without payload) to all authorized subscribers. An example would be where a line of business application uses this pattern to publish data to other line of business or point of service applications.

These common patterns are divided into processing phases, including the external and internal (regional and provincial) activities to satisfy external requests:

- Point of service patterns: the sequence of activities that point of service applications perform to interact with and consume EHR services
- EHR services patterns: the flow of activities to fulfill service requests invoked by point of service applications or other internal components
- Federated Health Information Access Layer patterns: build on EHR services patterns by defining processing behaviours for the provincial and regional HIAL segments

Appendix C - Governance

The following diagram illustrates the EHR architecture and standards governance committee structure.



STRATEGIC COMMITTTEE

Consists of the following organizations

- eHealth Ontario
- Canada Health Infoway
- Ministry of Health and Long-Term Care
- OntarioMD
- Ontario Hospital Association
- Connecting Greater Toronto Area
- Connecting South West Ontario
- Connecting Northern and Eastern Ontario
- Cancer Care Ontario
- Ontario Telemedicine Network
- Ontario Association of Community Care Access Centres

BUSINESS & TECHNICAL COMMITEE

Consists of the following sectors

- Goverment
- Clinicians
- Privacy
- Regional delivery partners
- IT Professionals
- Data custodian organizations
- Community care
- Standards development organizations

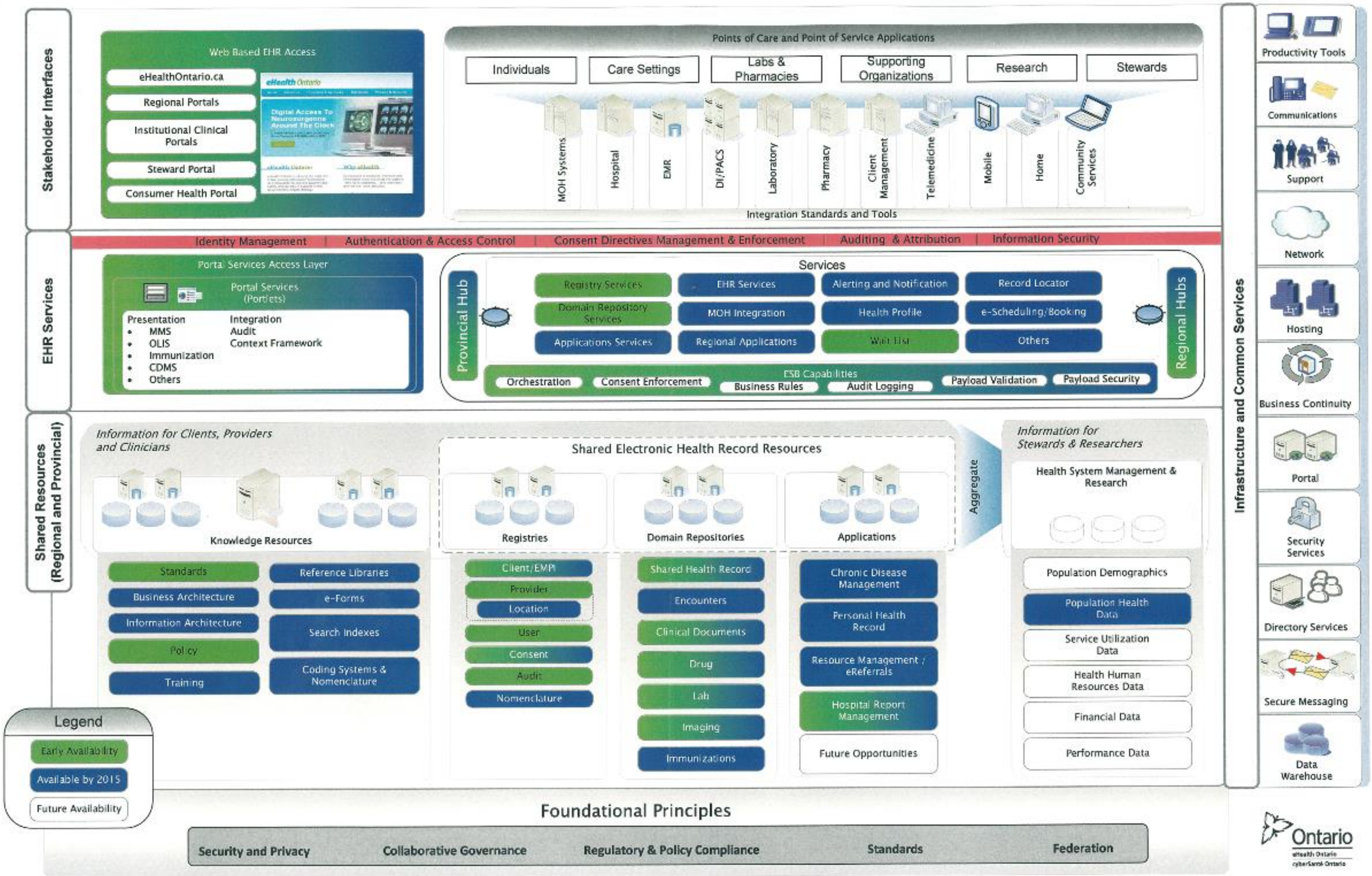
eHealth Ontario program, projects & delivery partners

Ontario Health Informatics Standards Council

Infoway Standards Collaborative

Clinical Advisory Council

Appendix D – 2015 eHealth Blueprint



Stakeholder Interfaces

Web Based EHR Access

- eHealthOntario.ca
- Regional Portals
- Institutional Clinical Portals
- Steward Portal
- Consumer Health Portal

Points of Care and Point of Service Applications

Individuals | Care Settings | Labs & Pharmacies | Supporting Organizations | Research | Stewards

MOH Systems, Hospital, EMR, DI/PACS, Laboratory, Pharmacy, Client Management, Telemedicine, Mobile, Home, Community Services

Integration Standards and Tools

Identity Management | Authentication & Access Control | Consent Directives Management & Enforcement | Auditing & Attribution | Information Security

EHR Services

Portal Services Access Layer

Portal Services (Portlets)

Presentation
- MMS
- OLIS
- Immunization
- CDMS
- Others

Integration
Audit
Context Framework

Provincial Hub

Services

- Registry Services
- Domain Repository Services
- Applications Services
- EHR Services
- MOH Integration
- Regional Applications
- Alerting and Notification
- Health Profile
- Wait List
- Record Locator
- e-Scheduling/Booking
- Others

Regional Hubs

ESB Capabilities: Orchestration, Consent Enforcement, Business Rules, Audit Logging, Payload Validation, Payload Security

Shared Resources (Regional and Provincial)

Information for Clients, Providers and Clinicians

Knowledge Resources

- Standards
- Business Architecture
- Information Architecture
- Policy
- Training
- Reference Libraries
- e-Forms
- Search Indexes
- Coding Systems & Nomenclature

Shared Electronic Health Record Resources

Registries

- Client/EMPI
- Provider
- Location
- User
- Consent
- Audit
- Nomenclature

Domain Repositories

- Shared Health Record
- Encounters
- Clinical Documents
- Drug
- Lab
- Imaging
- Immunizations

Applications

- Chronic Disease Management
- Personal Health Record
- Resource Management / eReferrals
- Hospital Report Management
- Future Opportunities

Aggregate

Information for Stewards & Researchers

Health System Management & Research

- Population Demographics
- Population Health Data
- Service Utilization Data
- Health Human Resources Data
- Financial Data
- Performance Data

Infrastructure and Common Services

- Productivity Tools
- Communications
- Support
- Network
- Hosting
- Business Continuity
- Portal
- Security Services
- Directory Services
- Secure Messaging
- Data Warehouse

Legend

- 🟩 Early Availability
- 🟦 Available by 2015
- □ Future Availability

Foundational Principles

Security and Privacy | Collaborative Governance | Regulatory & Policy Compliance | Standards | Federation

Ontario

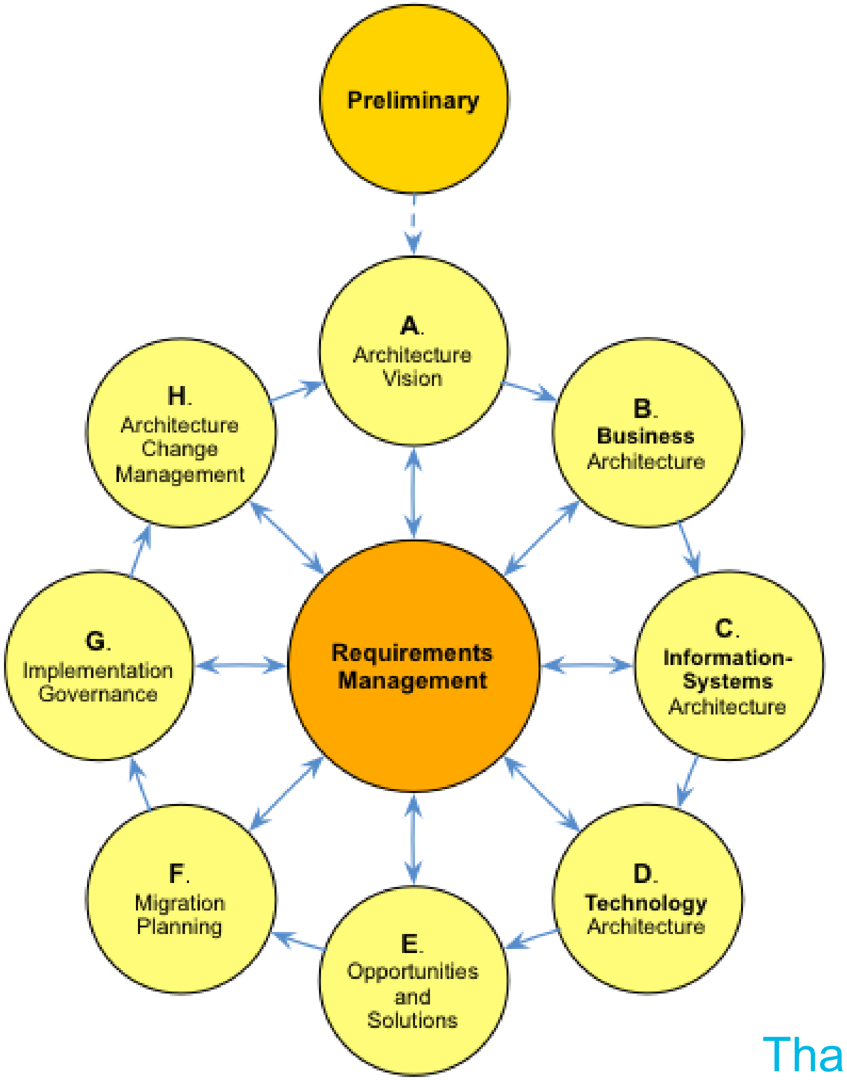
eHealth Ontario

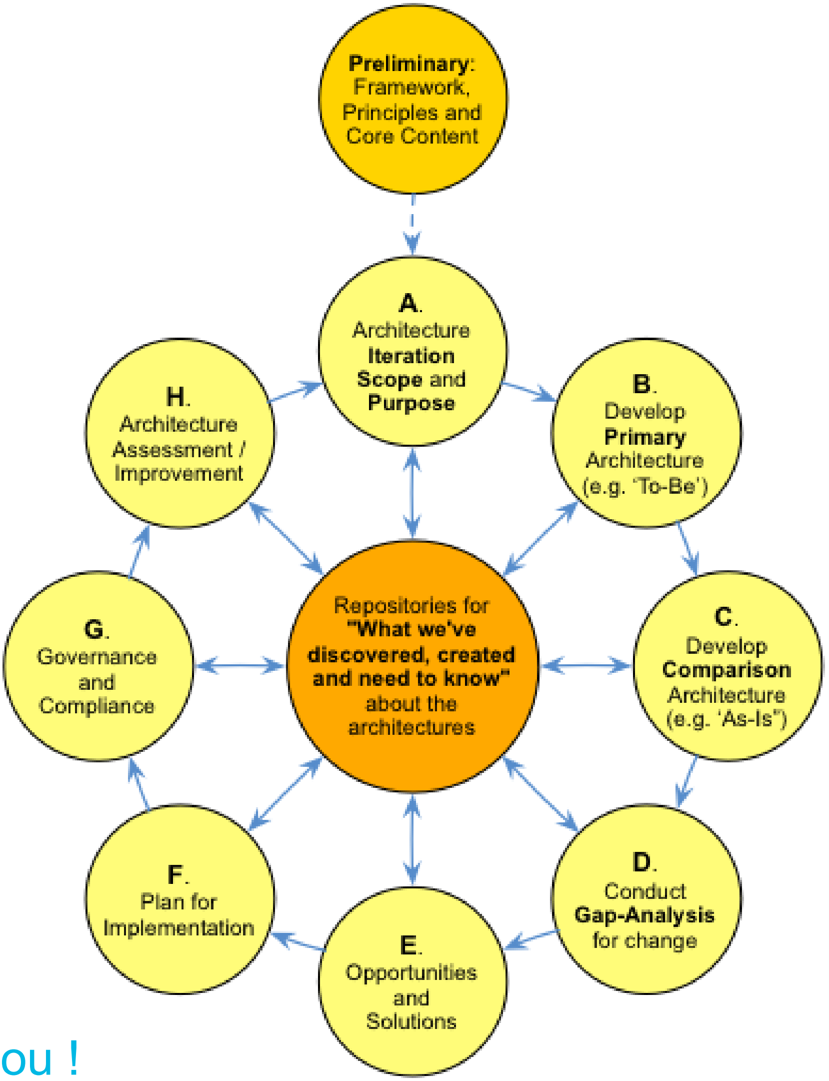
cyberSanté Ontario

References

- [1] - 1851-FOM_ILD_Arbeitspapier_Bd27_Online_12_12_18-01_01.pdf
- [2] - http://www.businessdictionary.com/definition/technological-environment.html#ixzz3EqyzJ63i
- [3] - http://en.wikipedia.org/wiki/Service-oriented_architecture
- [4] - http://www.sei.cmu.edu/reports/08tn021.pdf
- [5] - Service Level Agreements in SOA Environments - Software Engineering Institute
- [6] - Developing an Enterprise Architecture - Author: Paul Harmon
- [7] - Team Quest - Introducing a Capacity Management Maturity Model
- [8] - http://en.wikipedia.org/wiki/Capability_Maturity_Model_Integration
- [9] - EA Frameworks: Pros and Cons – Inventory and Insights: http://www.eadirections.com/papcdl/5/
- [10] – eHealth Ontario Blueprint Book
- [11] – Jean Jamieson - Future of Healthcare Trends: Impact on Postgraduate Medical Education
- [12] - Industry Trends and Enabling Technologies - Appendix I
- [13] - EHealth in Canada - Current Trends and Future Challenges - http://www.ictc-ctic.ca
- [14] - A Guide to SLAs - Barclay Rae
- [15] - http://en.wikipedia.org/wiki/Audit
- [16] - Identity and access management - Beyond compliance
- [17] – IBM - Enforcing Service Level Agreements using WebSphere DataPower
- [18] - eHealth Ontario – Enterprise Testing Framework
- [19] – IT Manager's guide – How to make an SLA - TechExcel
- [20] - TOGAF® Version 9.1 - http://pubs.opengroup.org/architecture/togaf9-doc/arch/index.html
- [21] - The Benefits of Offering SLAs by Tom Sweeny, ServiceXRG - http://www.supportindustry.com/asktheexpert/benefits_slas.htm

Architects spot the Difference


Preliminary
A. Architecture Vision
B. Business Architecture
C. Information-Systems Architecture
D. Technology Architecture
E. Opportunities and Solutions
F. Migration Planning
G. Implementation Governance
H. Architecture Change Management
Requirements Management


Preliminary: Framework, Principles and Core Content
A. Architecture Iteration Scope and Purpose
B. Develop Primary Architecture (e.g. 'To-Be')
C. Develop Comparison Architecture (e.g. 'As-Is")
D. Conduct Gap-Analysis for change
E. Opportunities and Solutions
F. Plan for Implementation
G. Governance and Compliance
H. Architecture Assessment / Improvement
Repositories for "What we've discovered, created and need to know" about the architectures

Thank you !