

# Automating SLA negotiation: a policy-based approach

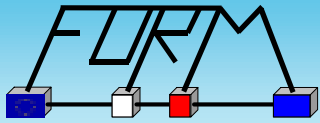


Thanassis Tiropanis

University College London, Computer Science

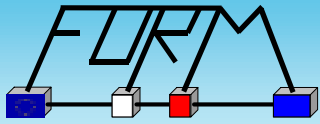
*t.tiropanis@cs.ucl.ac.uk*



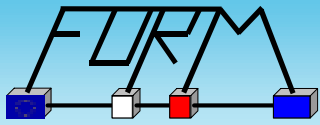


# Presentation Outline

- The FORM project environment
- Current policy issues
- Towards a generic policy model
- Automating SLA negotiation
  - The SLA negotiation process
  - A SLA negotiation engine
  - SLA negotiation policies
- Technical issues
- Further work



# FORM



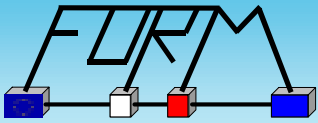
# FORM: Main Goal and Approach

**GOAL**

*To enable FORM's industrial partners to exploit services, software systems and software components for the management of an outsourced Inter-Enterprise Service (IES) supporting B2B communication requirements*

**Approach**

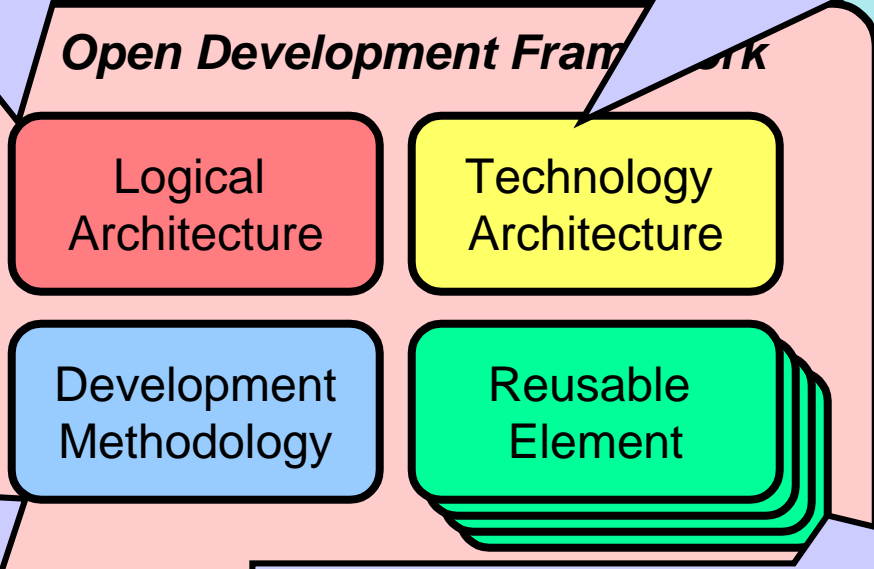
*Definition of an Open Development Framework addressing operational needs for IES management and development of management systems based on software components (Building Blocks)*



# FORM: Anatomy of an Open Development Framework

Business Roles + RPs (TINA)  
Business Processes (TMF)  
BBs in 3 tiers  
BB platform services  
BB contracts

Map BB to specific technologies:  
• BBs to EJB  
• BB platform to container services  
• Contract specs to interface implementation (IDL, HTTP/XML..)  
Mediation - multi-protocol BBs

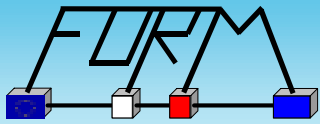


Business models  
Business process  
OOAD  
Contract/BB definition  
RUP  
Notation:  
UML + XML?

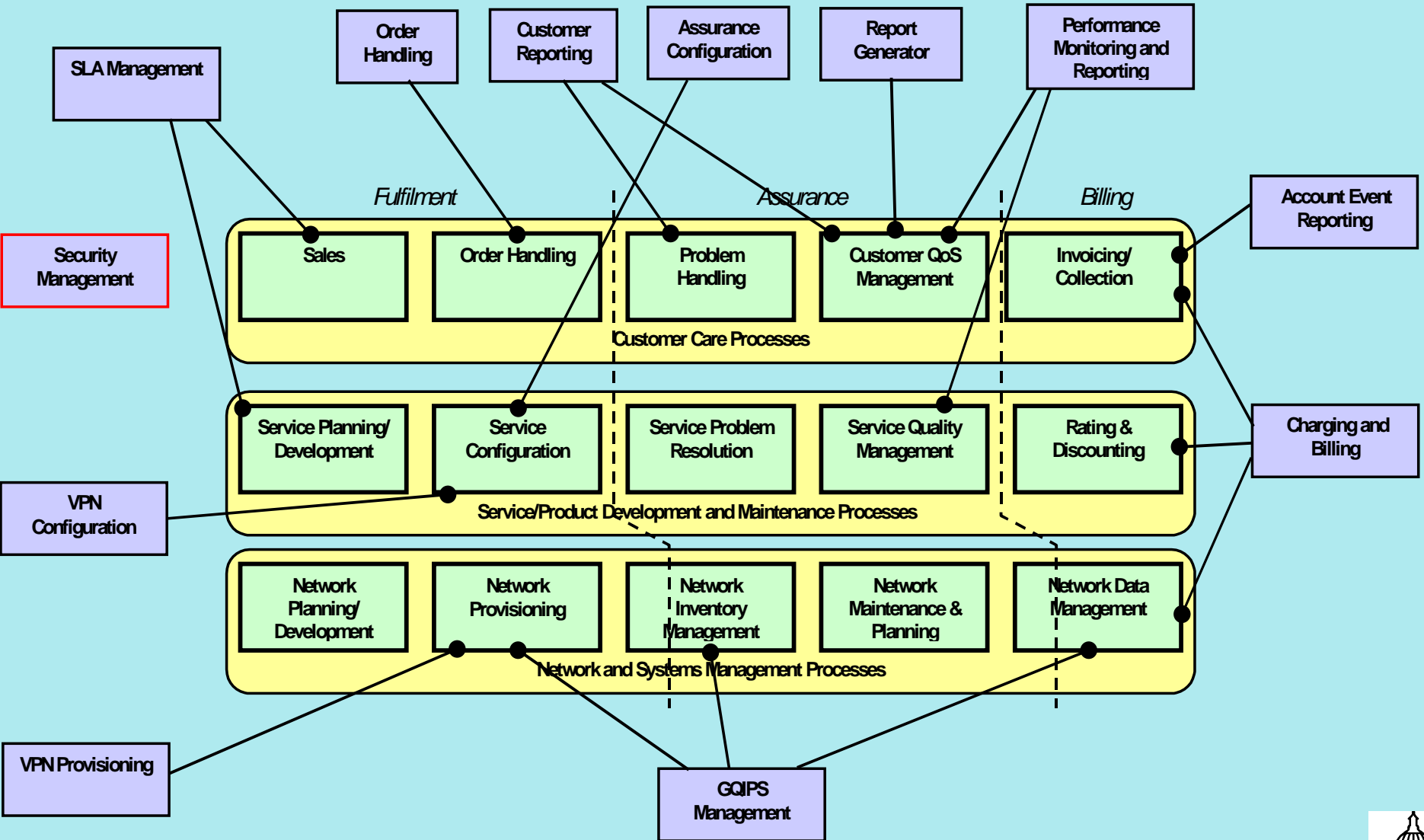
RPs, Contracts and BB definitions:  
• Bindings  
• Implementation independent specs  
• Packaged with analysis and design info  
• Variability and Constraint specs

© FORM Consortium



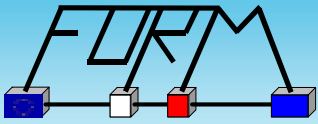


# FORM: Mapping to TM Forum Business Processes

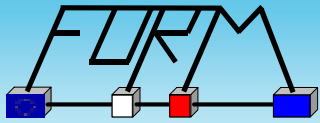


© FORM Consortium





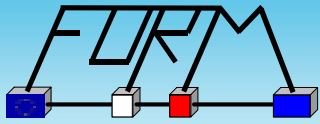
# Policies



# Background

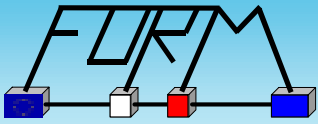
- TMF SLA Group
- Imperial College work
  - Ponder language
- DMTF: PCIM and CIM
- IETF
- IST project Tequila
- Other IST projects (CADENUS, AQUILA, etc)
- Active Networks



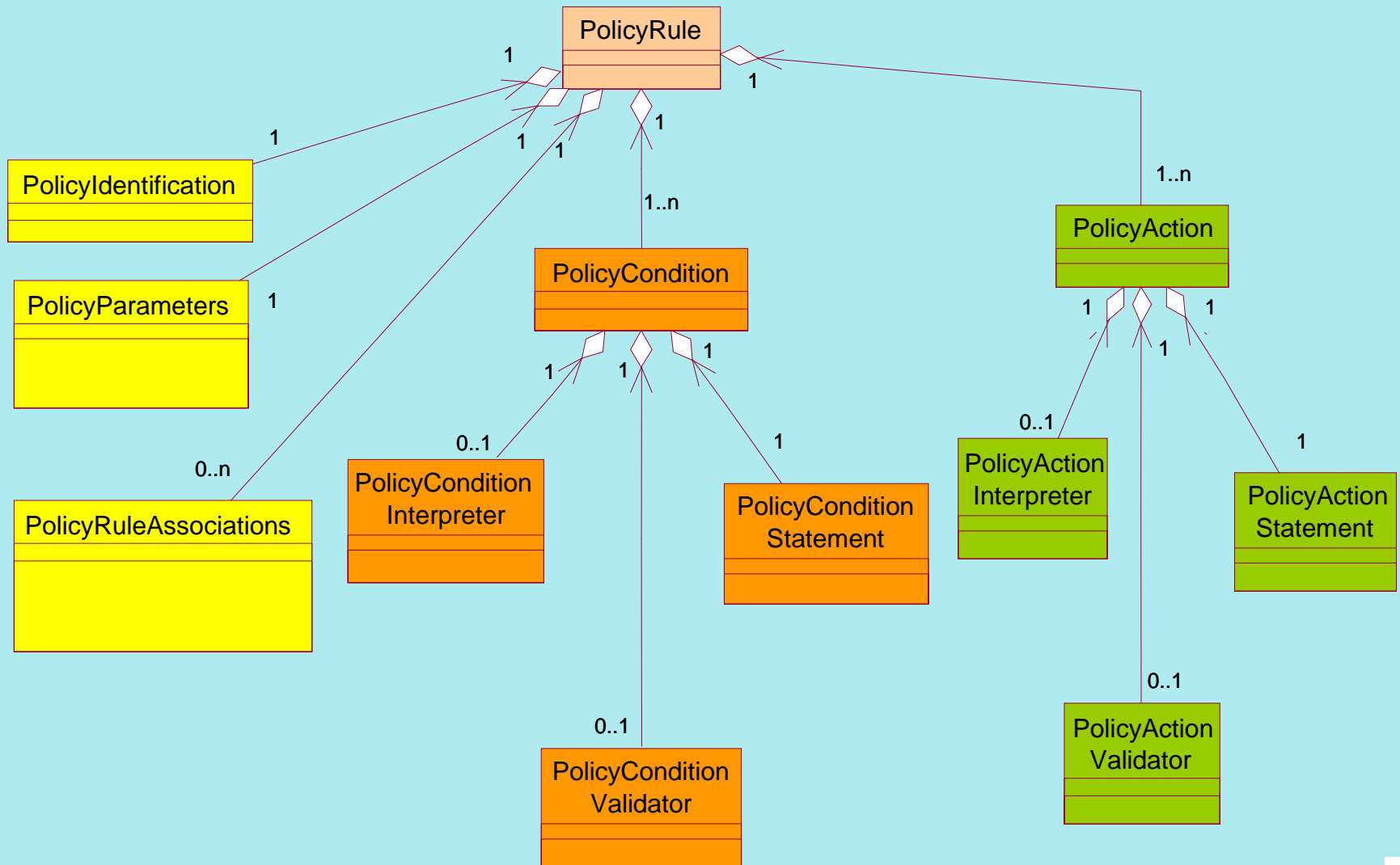


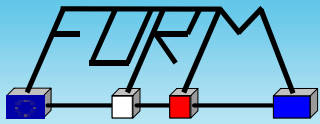
# Policy Issues

- Are policies useful? Why?
- What functionality can be policy-based?
- Generic policy language
- O-O Policies
- Hierarchical policy organisation
- Transformation from high to low level policies
- Transformation from low to high level policies
- Policy meta-model
- Policy semantics
- Security concerns

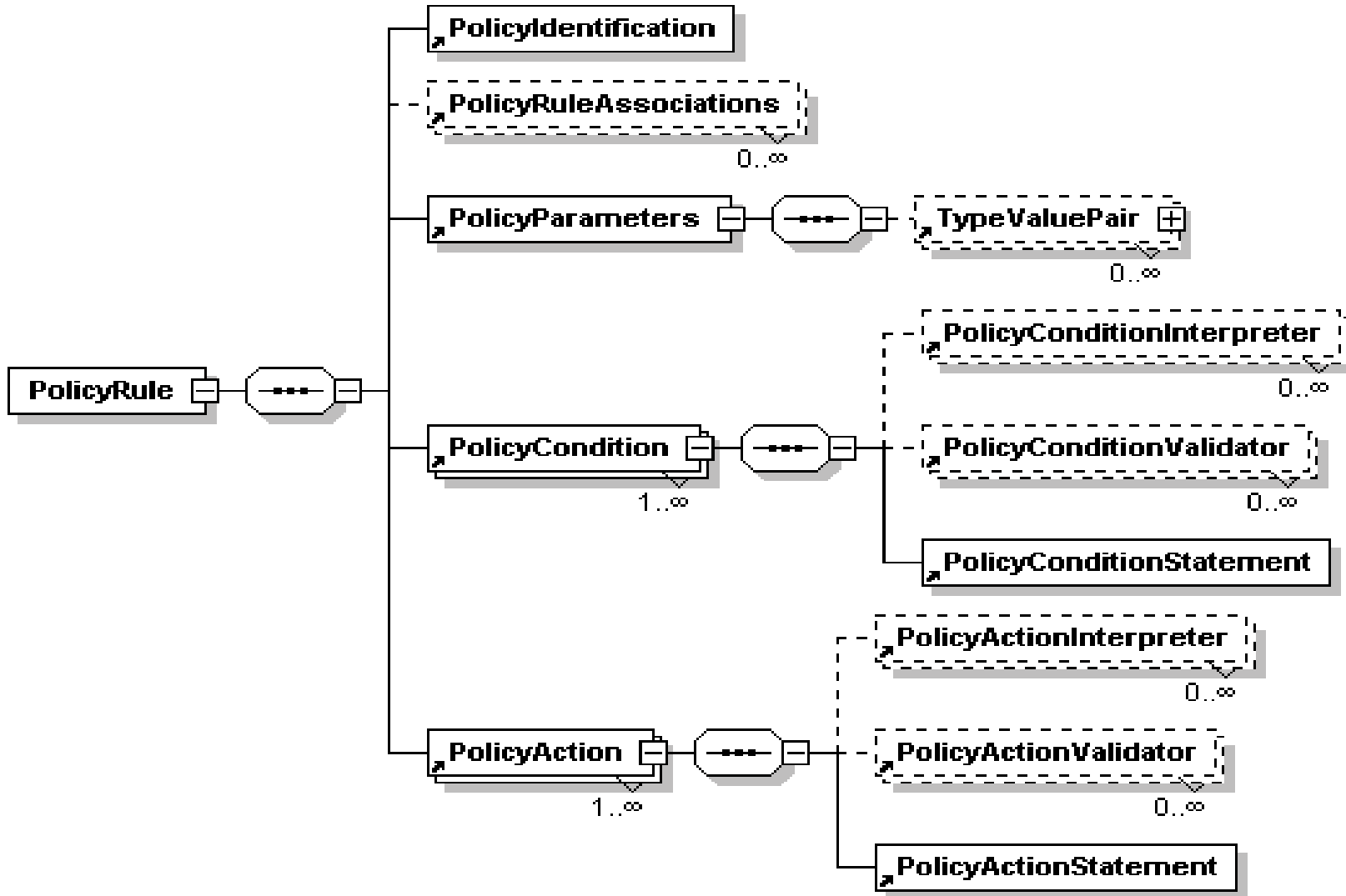


# Generic Policy Model (GPM)

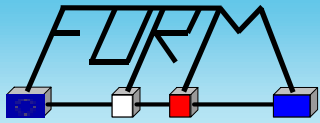




# polML: Generic Policy Model DTD

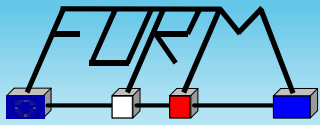


Generated with XMLSpy Schema Editor [www.xmlspy.com](http://www.xmlspy.com)



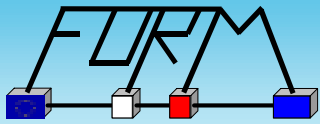
# GPM Features

- Meta-model
- Policy language independent
- Applies to different application areas
- Provides for hierarchical policy organisation
- Provides for policy transformation functions
- Policy semantics are out of the model
- Security information can be added
- Compatible with current IETF/DMTF policy models
- A base for the implementation of policy-independent PDP/PEPs
- Separation between PDP/PEP infrastructure and policy specification, interpretation and enforcement



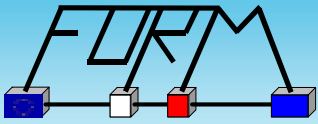
# Relating the GPM to the DMTF/IETF Policies

- Policy Identification
  - Rule usage description
- Policy Parameters
  - Enabled, priority, mandatory, condition list type, sequenced action, policy roles
- Policy Rule Associations
  - Policy roles?
- Policy Condition(s)
  - Policy condition(s), time period
- Policy Action(s)
  - Policy action(s)

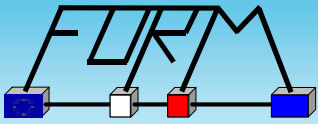


# FORM: Areas of Policy Use

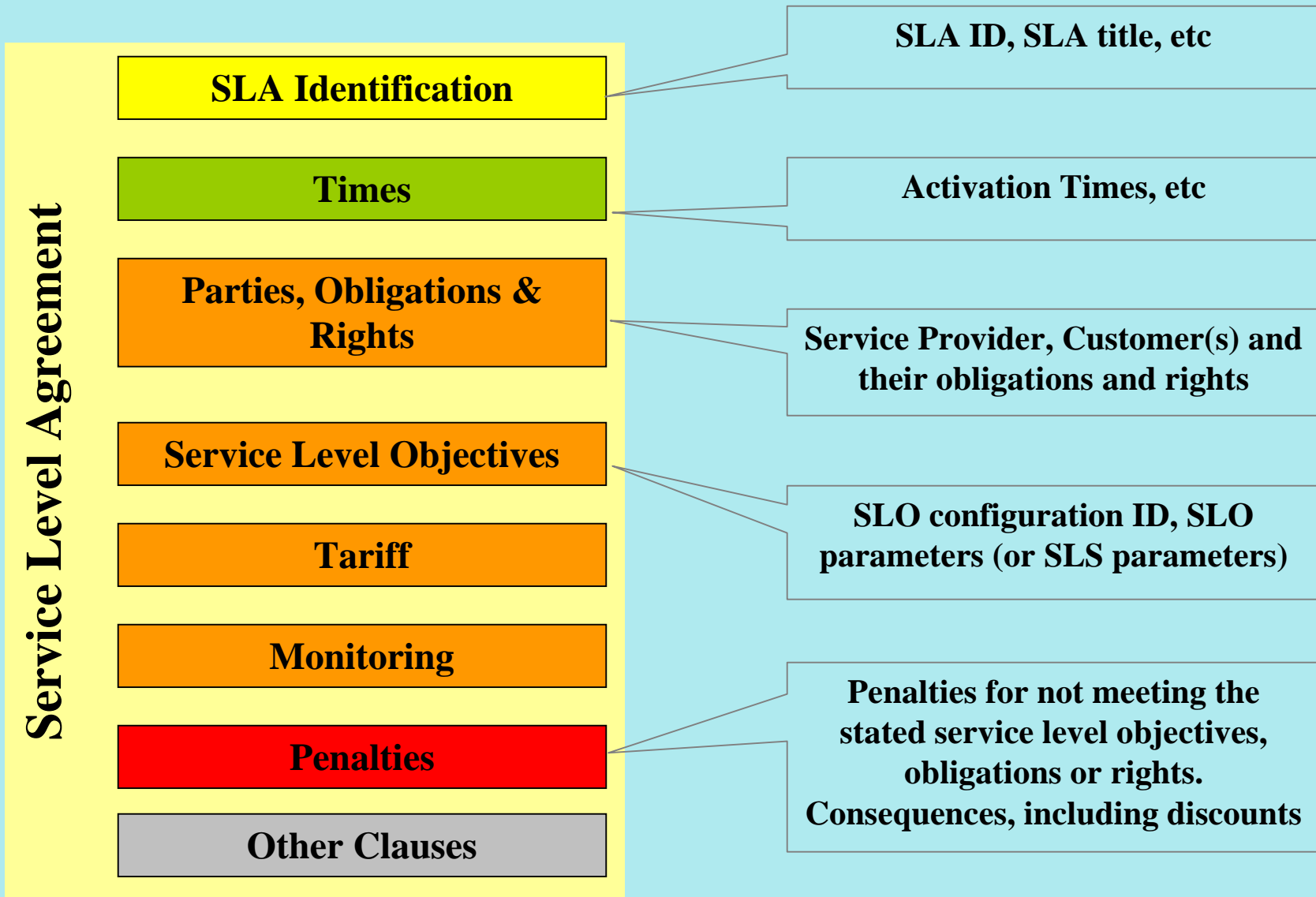
- Automating the SLA negotiation process
- Generalising use of policies for controlling BB behaviour in general
- Policy-based SLS negotiation
- Policy-based network configuration and monitoring (assurance)
- Policy-based configuration of the IPSEC tunnel



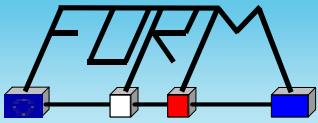
# SLA Negotiation



# Structure of a SLA







# SLA Negotiation Process



**Customer Admin.**



**Service Provider Admin.**

:OH Customer Interface

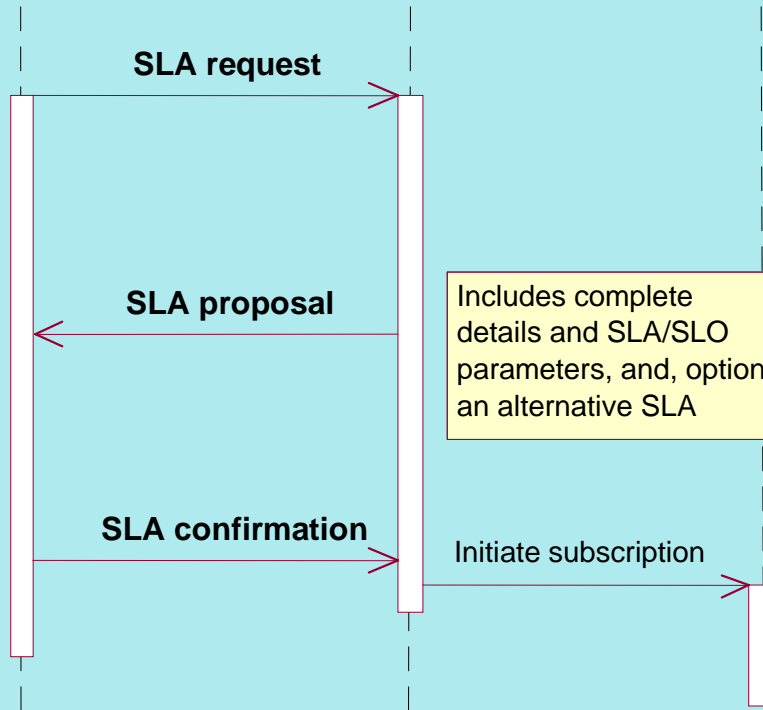
:SLA Negotiation Engine

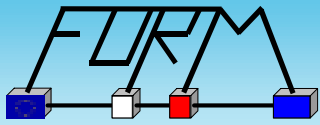
:OH Subscription Manager

Includes SLA/SLO parameters or SLO ID. Customer details are optional here.

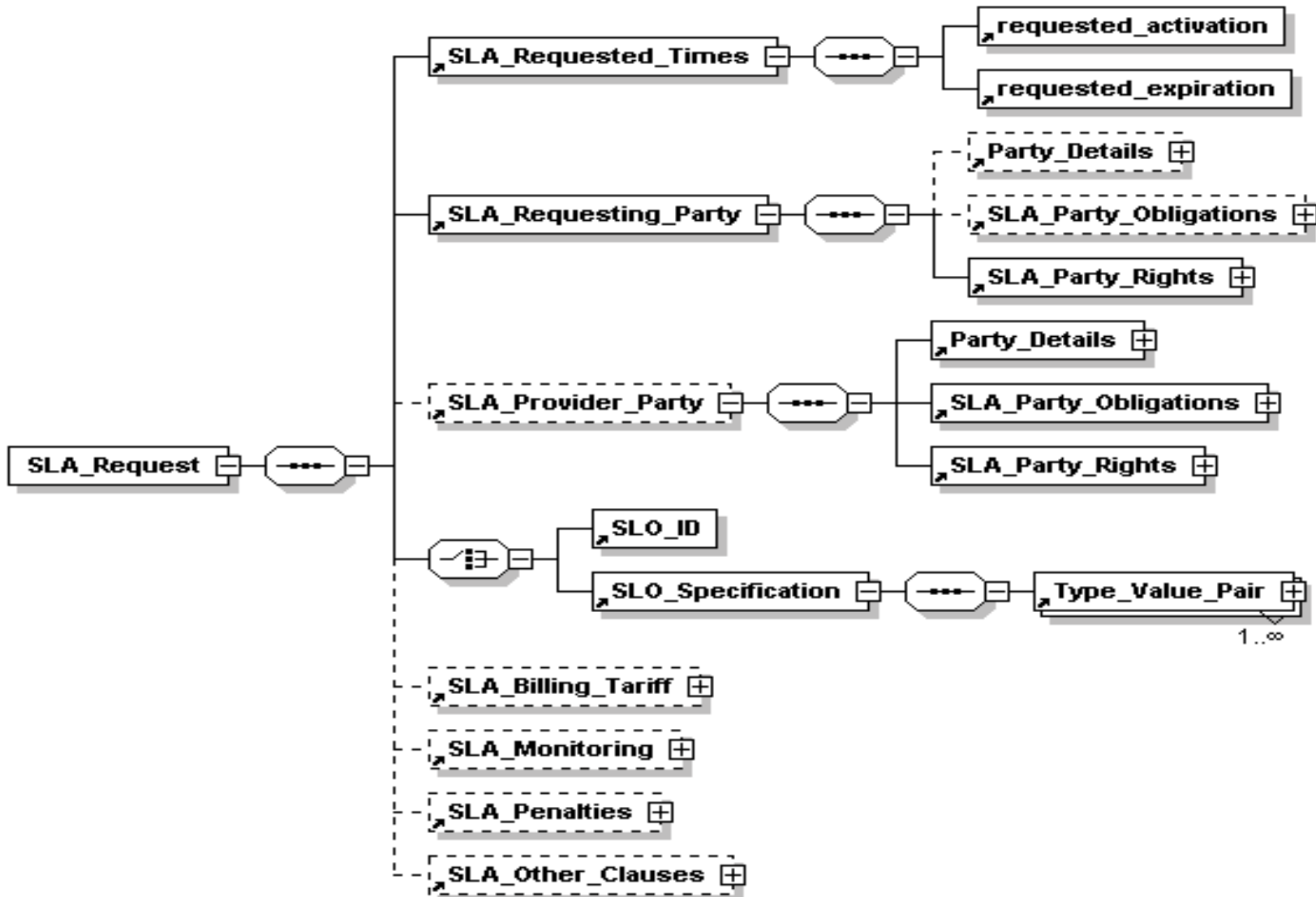
Customer details compulsory. Also, it is stated which of the alternatives is confirmed

Includes complete details and SLA/SLO parameters, and, optionally, an alternative SLA

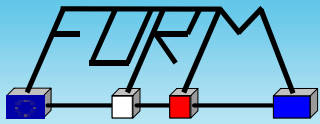




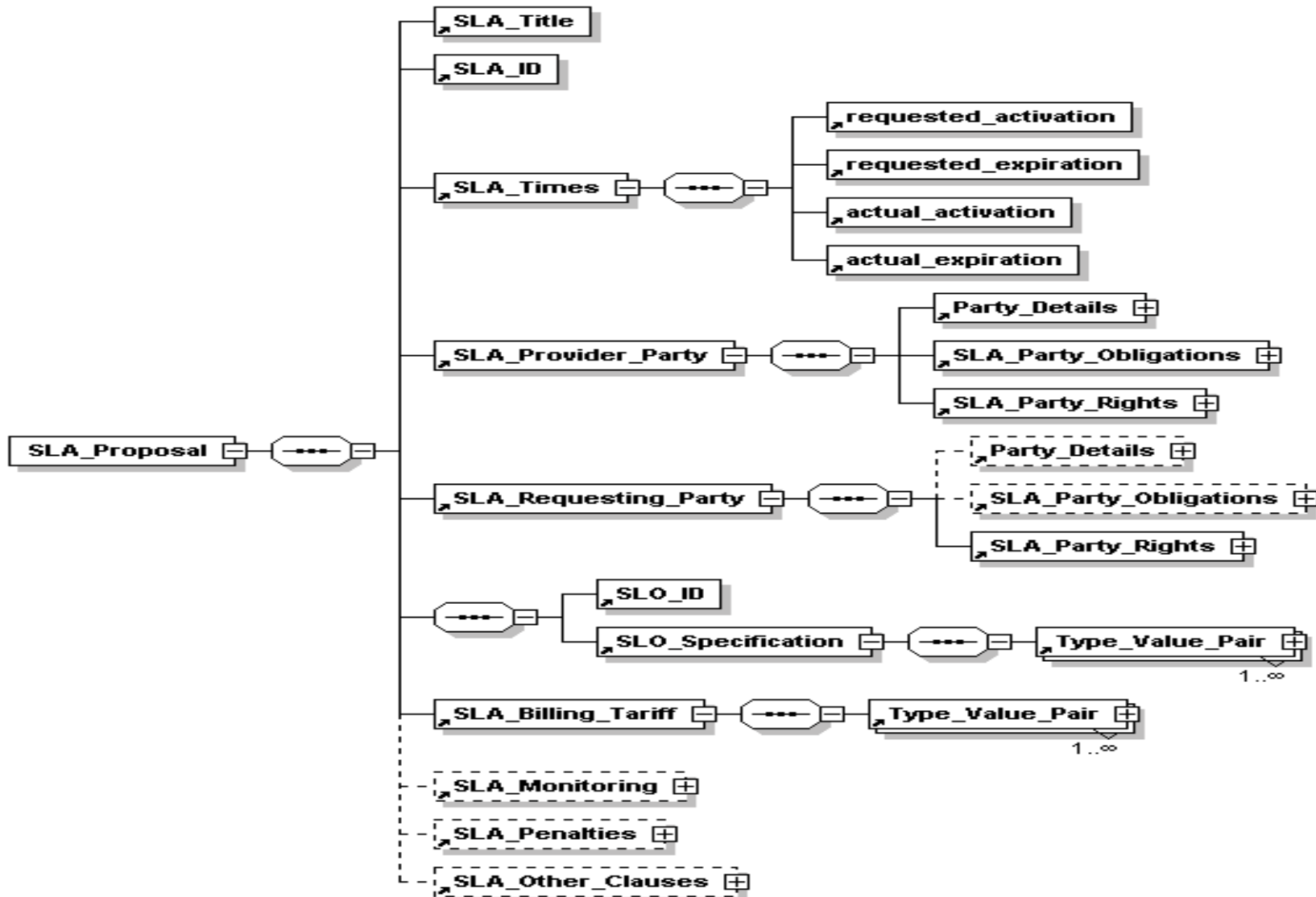
# slaML: SLA Negotiation Request



Generated with XMLSpy Schema Editor [www.xmlspy.com](http://www.xmlspy.com)

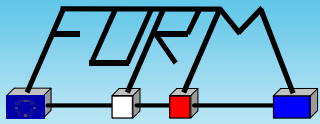


# slaML: SLA Proposal

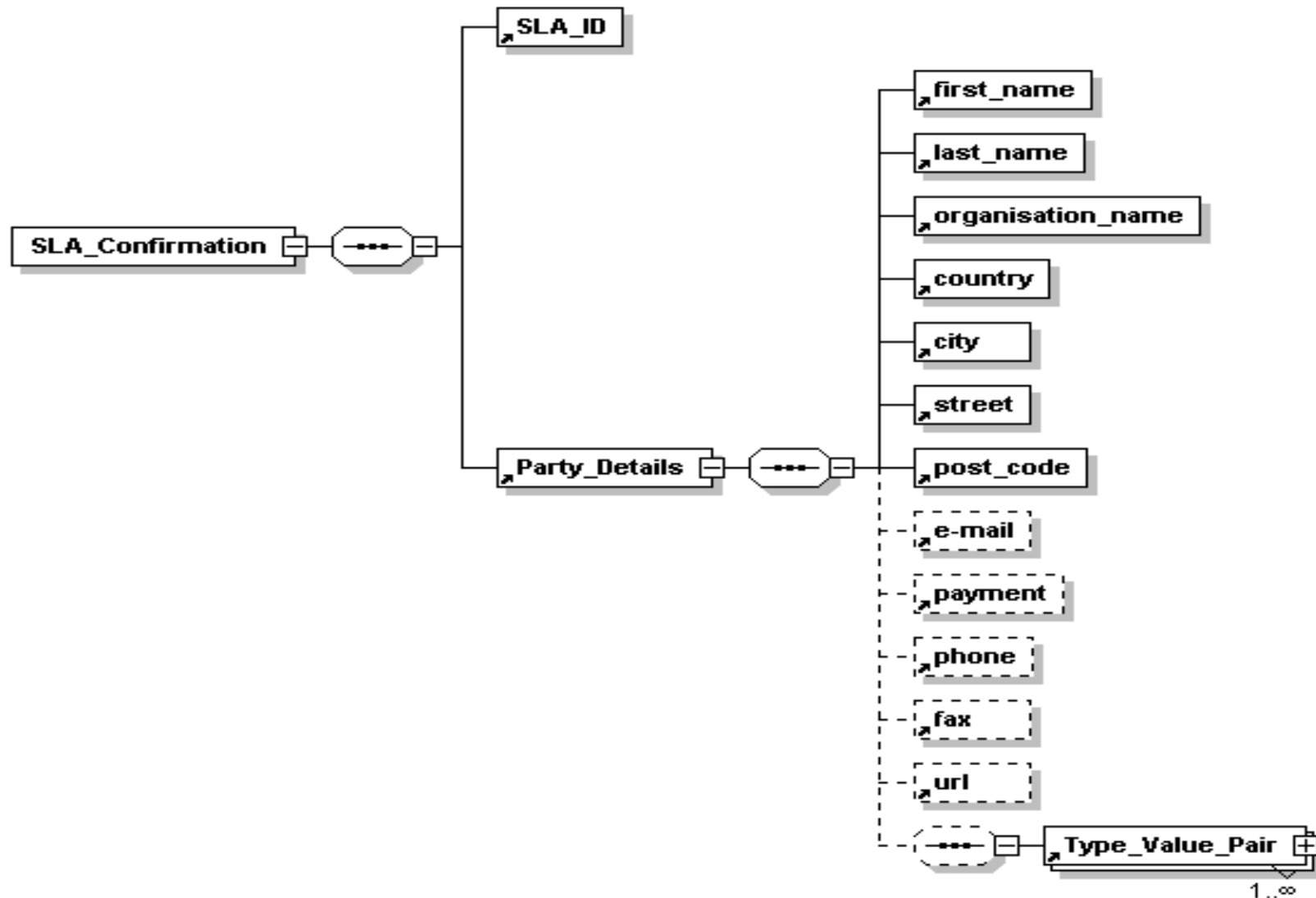


Generated with XMLSpy Schema Editor [www.xmlspy.com](http://www.xmlspy.com)



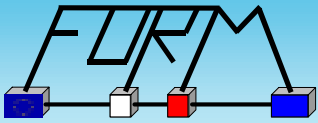


# slaML: SLA Confirmation



Generated with XMLSpy Schema Editor [www.xmlspy.com](http://www.xmlspy.com)





# SLA Negotiation Process (2)



**Customer Admin.**



**Service Provider Admin.**

:OH Customer Interface

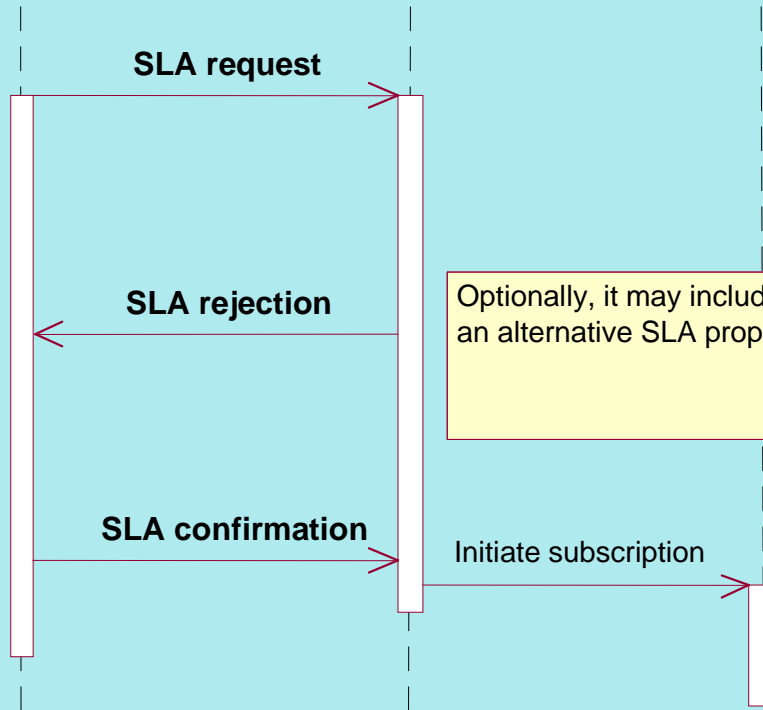
:SLA Negotiation Engine

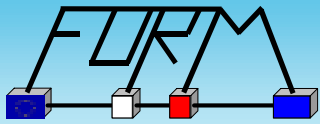
:OH Subscription Manager

Includes SLA/SLO parameters or SLO ID. Customer details are optional here.

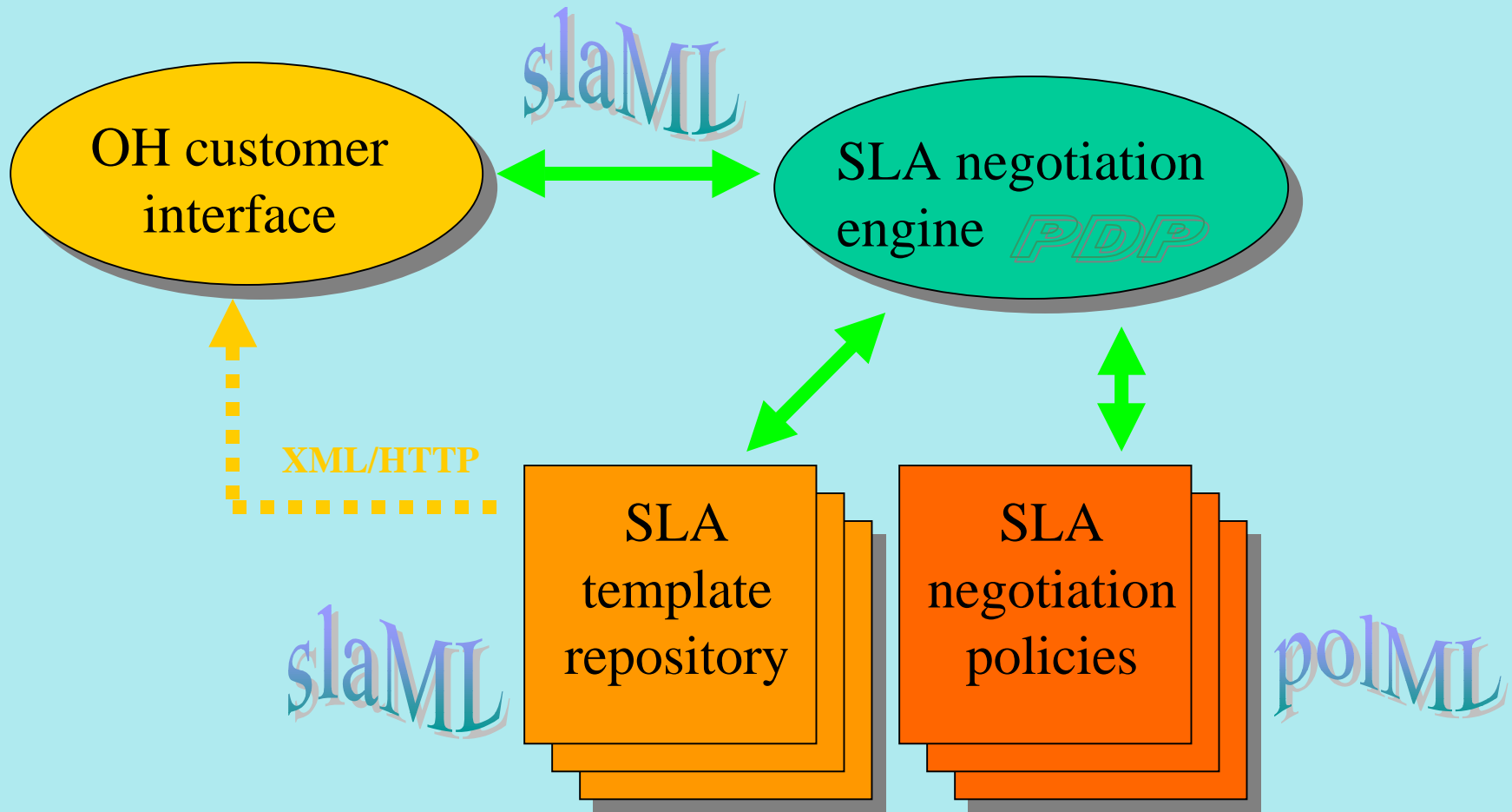
Customer may confirm the rejection or the suggested alternative SLA proposal

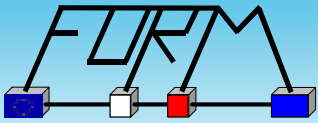
Optionally, it may include an alternative SLA proposal



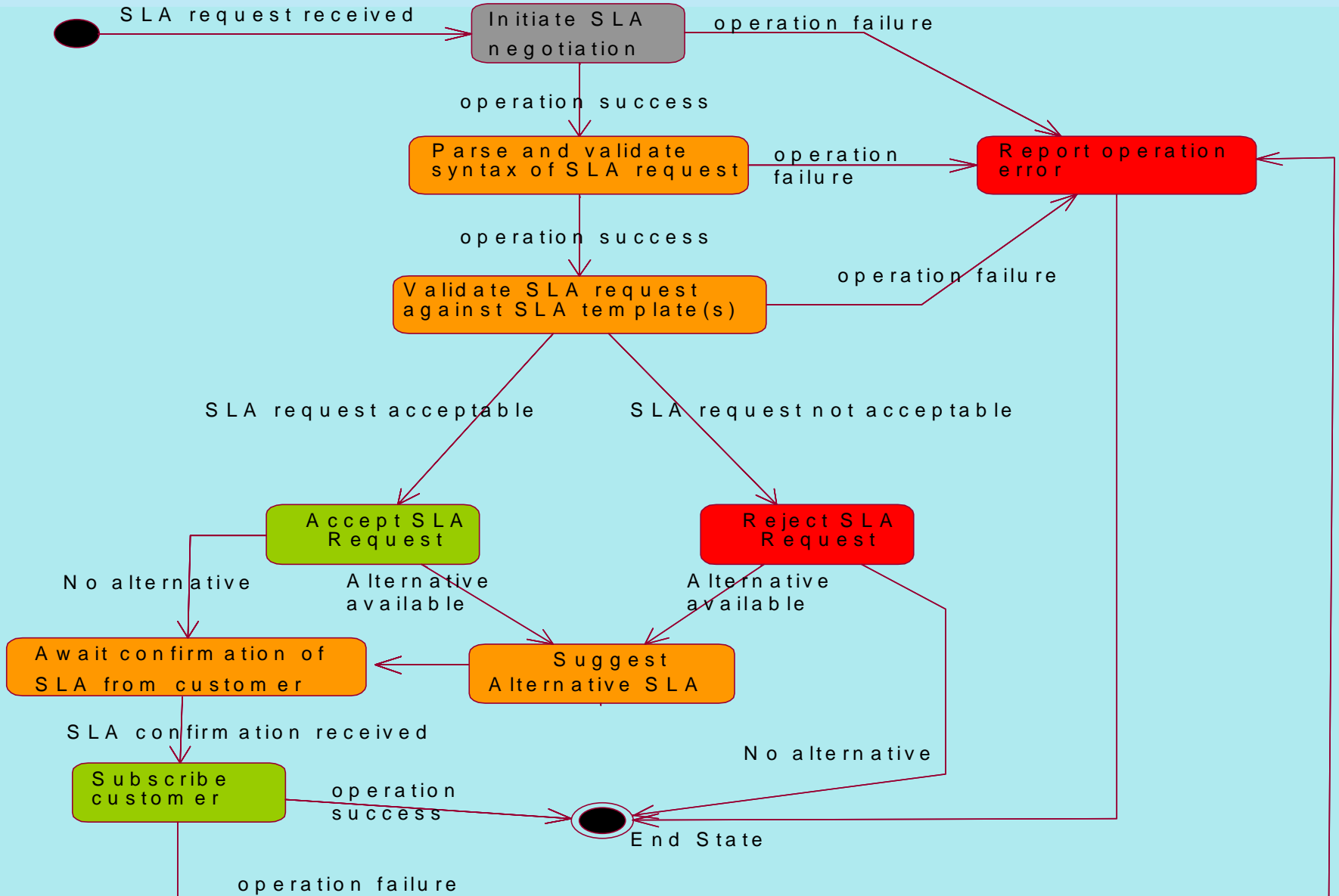


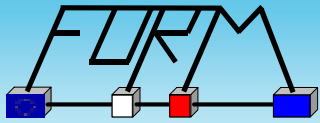
# Main SLA Negotiation Components





# Operation of the Negotiation Engine

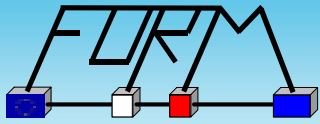




# SLA Negotiation Policy Examples

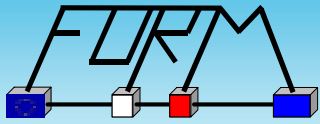
- SLA Negotiation Policies
  - If *SLASuccessfullyProccessed* then (*AcceptSLA*, *ConcludeSLANegotiation*)
  - If *SLANegotiationConcluding* then *CheckForAlternativeSLA*
  - If *SLOParameterRejected(bw)* then *CreateAlternativeSLA(bw, 2Mbps)*
- SLO Processing Policies
  - If *SLOParameter(bw) >= 2Mbps* then *RejectSLOParameter(bw)*





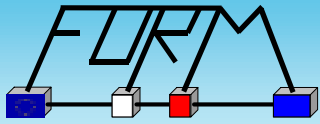
# Technology Issues

- XML for policy specification
  - Portable policies
  - Human readable policies
  - Interoperability with applications/databases
  - Definition of meta-policies as DTDs
  - Efficient policy processing, presentation and transformation (XPath, XSLT)
  - No O-O
  - Semantics?
- XML over HTTP for SLA negotiation
  - Efficient negotiation using a web browser, bypassing firewalls
  - SLA in human-readable form

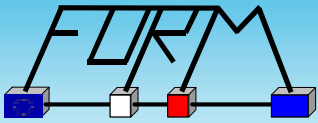


# Open Issues and Further Work

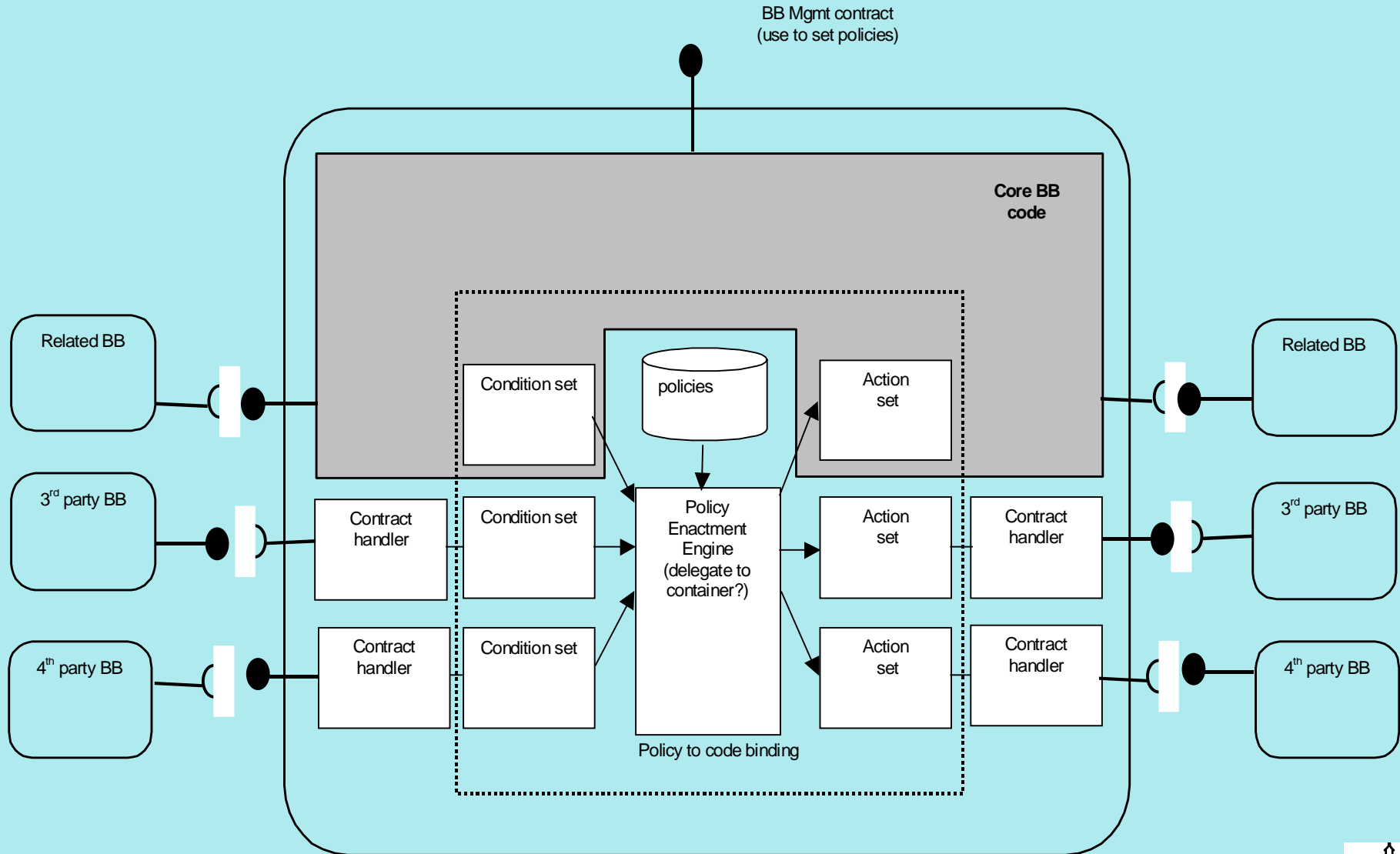
- Work on a policy specification language, parsing and enforcement components
- Develop a generic PDP/PEP based on the Generic Policy Model
- Develop a SLA policy negotiation engine based on the generic PDP/PEP engine
- Provide a policy transformation function for specific SLA policies
- IETF/DMTF policy parsers and enforcers
- Investigate using a generic PDP/PEP for IETF/DMTF policy processing
- Address security concerns

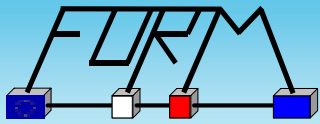


# Backup Slides

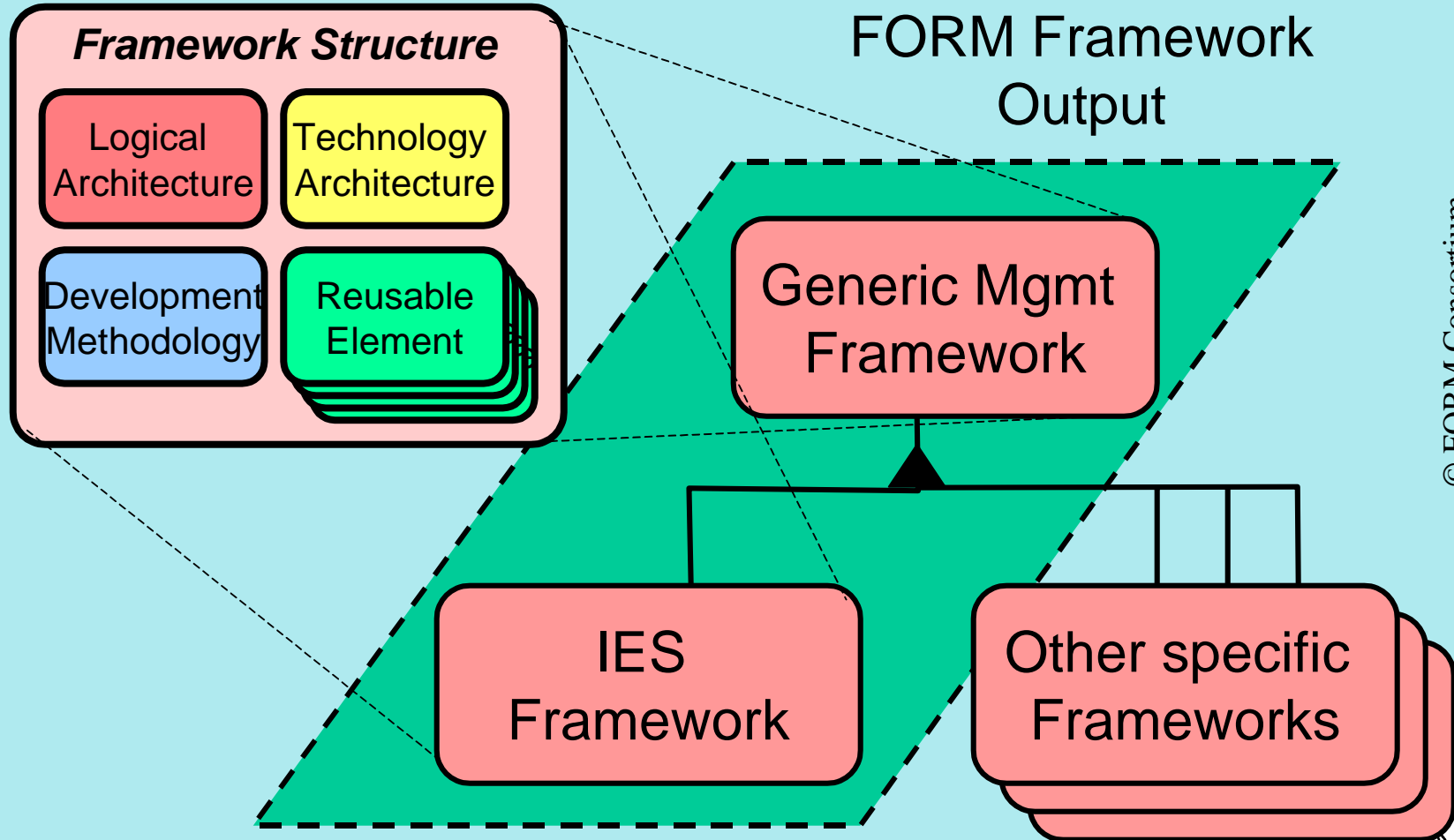


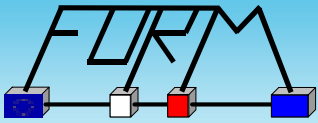
# Policy-based BB behaviour



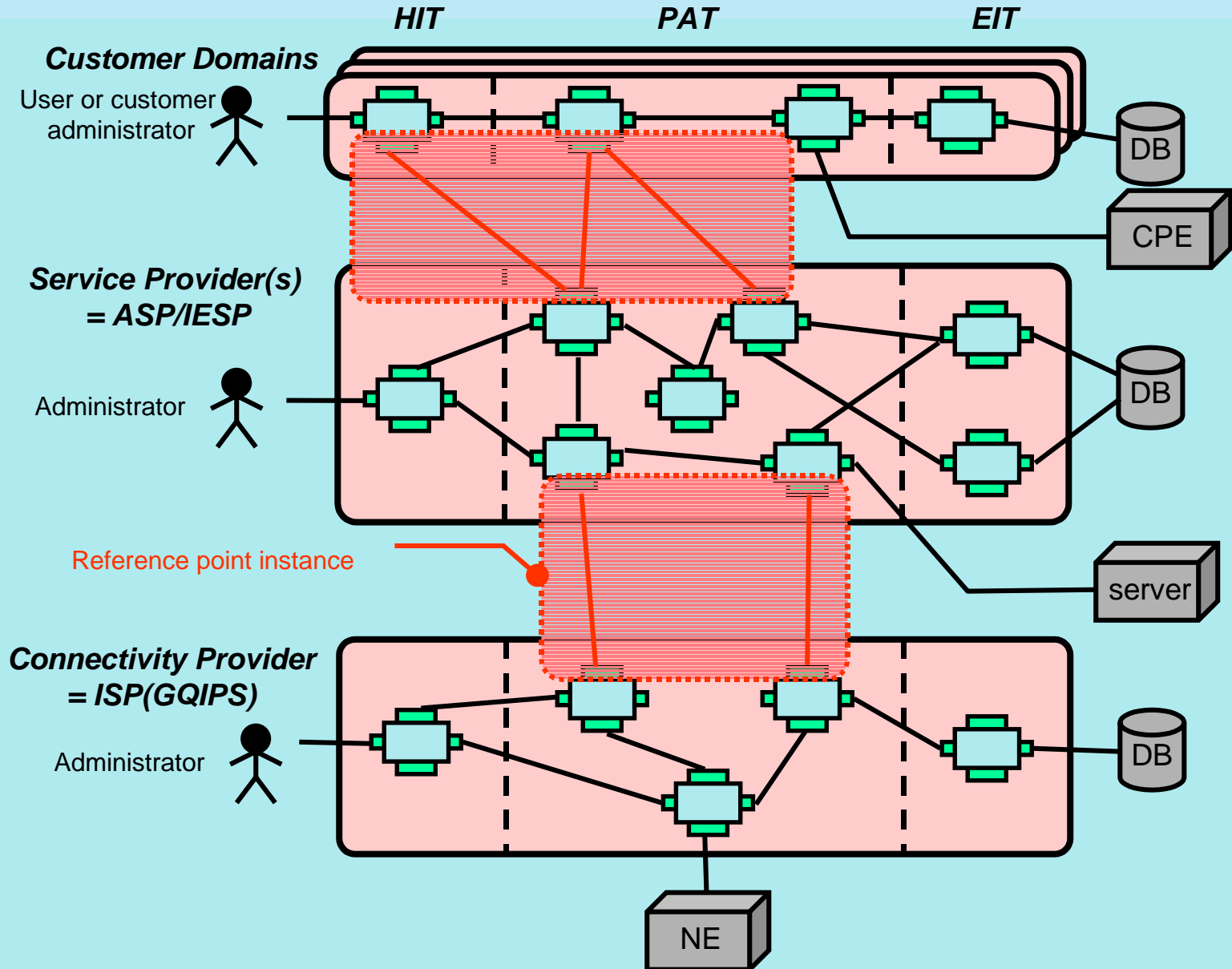


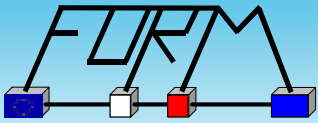
# FORM: Scope of the Open Development Framework



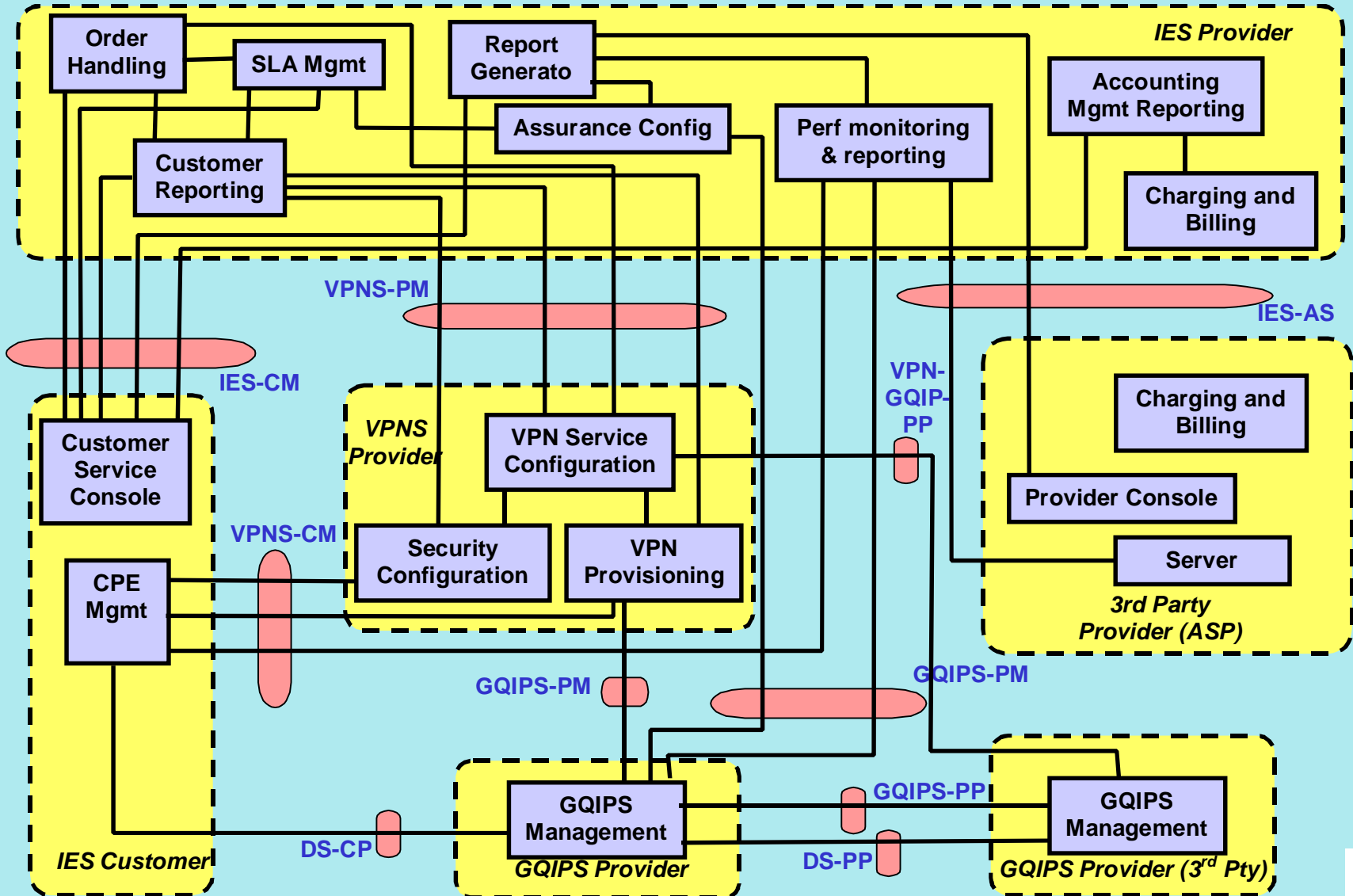


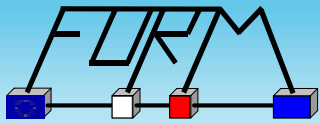
# Architecture Bindings



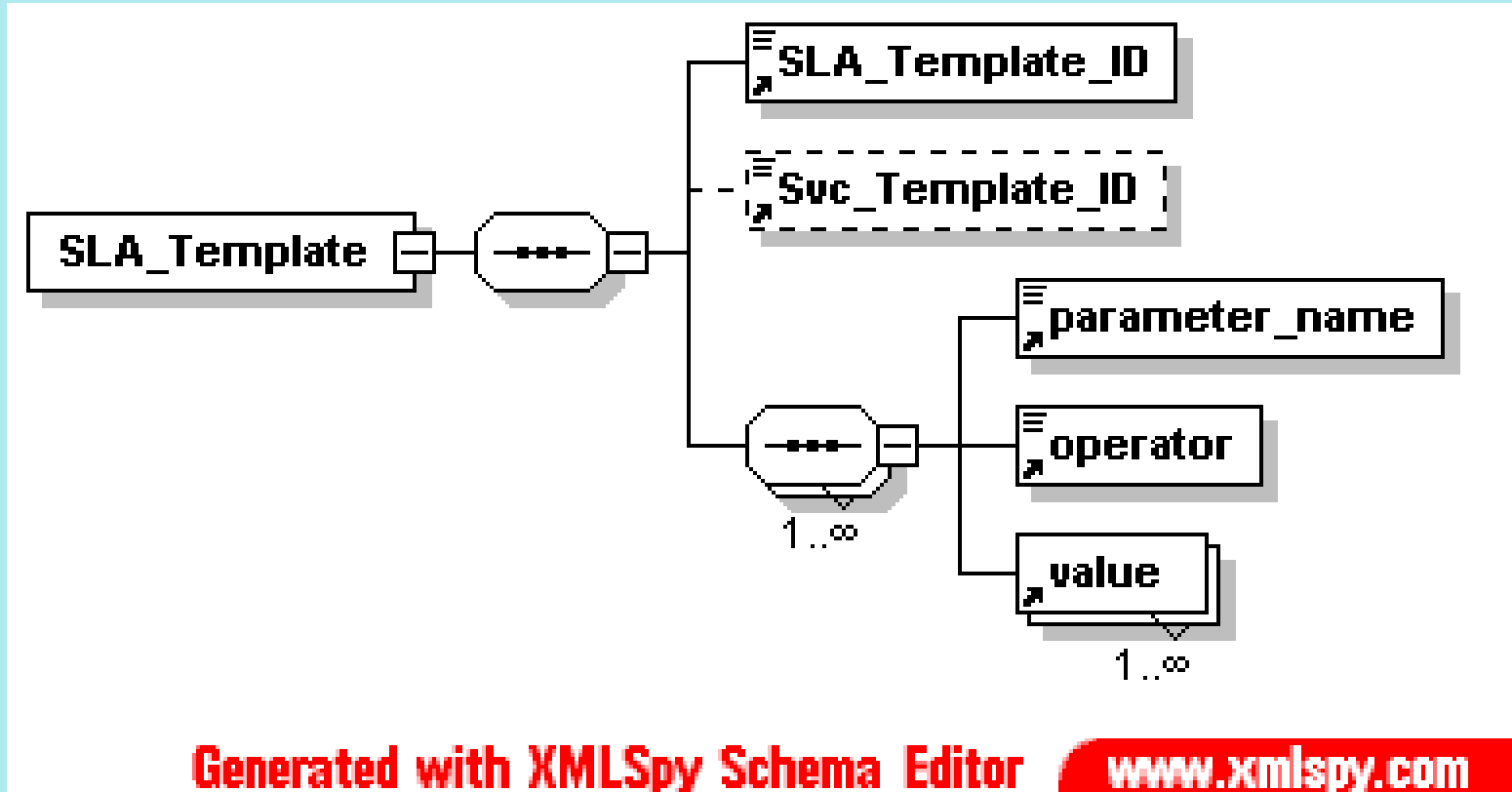


# Refined Business Processes





# slaML: SLA Template



Generated with XMLSpy Schema Editor

[www.xmlspy.com](http://www.xmlspy.com)