



Creating A Single Global Electronic Market

1 ebXML Business Concept 2 Demonstration

3 Tokyo Proof of Concept
4 Proposal Working Draft
5

6 Joint effort between Repository, BP/CC Build
7 team and the tpaML team.
8
9

10 **Version 0.1**

11 ebXML Tokyo POC Proposal v0.1.doc

12 **Author**

13 David RR Webber
14

15 **Contributors**

16 Joe Dalman
17 Lisa Camahan
18 Bruce Peat
19 Duane Nickull



20

Overview

21

This document is a draft proposal whose purpose is to convey the scope, the outline and implementation plan for the Tokyo Business Concept Demonstration (POC) and to solicit additional input.

22

23

24

Scope:

25

The purpose of the Tokyo POC is to build the appropriate, interoperable, ebXML infrastructure that enables ebXML participants to:

26

27

- Prove the viability of the ebXML R&R interfacing specifications to enable the use of the BP/CC analysis and design approach to create a reusable business definition in XML syntax.

28

29

30

- Show the ability of ebXML Business Concepts to be used by a business domain user who is not required to know XML itself.

31

32

- Show a scenario where the ebXML business process components themselves collaborate and show how they connect for documenting and storing Business Concept information that relates to a chosen industry business domain and use case model.

33

34

35

Outline:

36

The implementation itself will consist of the following:

37

- The approach is to create a set of Web forms that show how a business user can interact with ebXML compliant components to define and implement a business interchange.

38

39

- The ebXML components are, in order of use, TPA, RegRep, BP, and CC.

40

41

- The web forms demonstrate interaction of each of the ebXML pieces and allow us to highlight the purpose and mission of each such piece (TPA, RegRep, and BP/CC).

42

43

- Business User: – an analyst, or subject matter expert, using ebXML to define an industry process and the associated ebXML transaction process into an XML syntax format.

44

45

- Registry/Repository: – backend XML interface component that responds to information requests and queries, and then provides an update request interface to store the output that the Business User creates.

46

47

48

49

50

51

51

Implementation Plan:

52

Each of the teams involved will be responsible for providing their own working components. RegRep will create the XML interface specifications and XML syntax, BP/CC will create the Web forms needed, and TPA will create the XML definitions of their control information, and the associated user log-in form. The POC team will then be responsible for creating one or more functioning examples, and any associated presentational materials and documentation.

53

54

55

56

57

58

The Tokyo POC will use open software components and architecture that do not require vendor specific products to implement, and all local source code used in the demo forms and associated XML structures will be made available for release with the demonstration documentation.

59

60

61

62

63

Business Concept Demo Scenario

64

The Tokyo Business Concept scenario allows the details of the XML syntax, the ebXML technical specifications and the Repository mechanisms underpinning the business interactions to be clearly understood and replicated.

65

66

67

The demonstration scenario will feature information from one or more real industry vertical scenarios to ensure relevance. The mechanisms should also show how the underlying XML syntax supports multiple registry/repository interactions.

68

69

70

Each of the teams will be responsible for ensuring that all XML syntax formats are compliant to the ebXML specifications, and also that the interfaces used are also similarly compliant.

71

72

73

Business Process

74

The business process to be modelled should have a theme relevant to Japan to aid connect to local attendees, the audience and press. (US <-> Japan trade interaction within an industry vertical that's more than a simple supply chain model? Thought: product information / labelling reporting within food industry?).

75

76

77

78

The diagram below shows the basic components envisioned for the demonstration. The diagram also depicts only the positive scenario, where the user interactions are successfully completed as expected. Due to the nature and time constraints of the proof of concept, it is unlikely that there will be a need to model the possible errors that may occur in the scenario.

79

80

81

82

83

Business Data

84

The successful operation of the demonstration will be dependent on creating pre-existing sets of XML structures with the necessary control information contained already to set the background of the business scenario. Clearly documenting and showing this control information will be part of the requirements for the demonstration presentation itself.

85

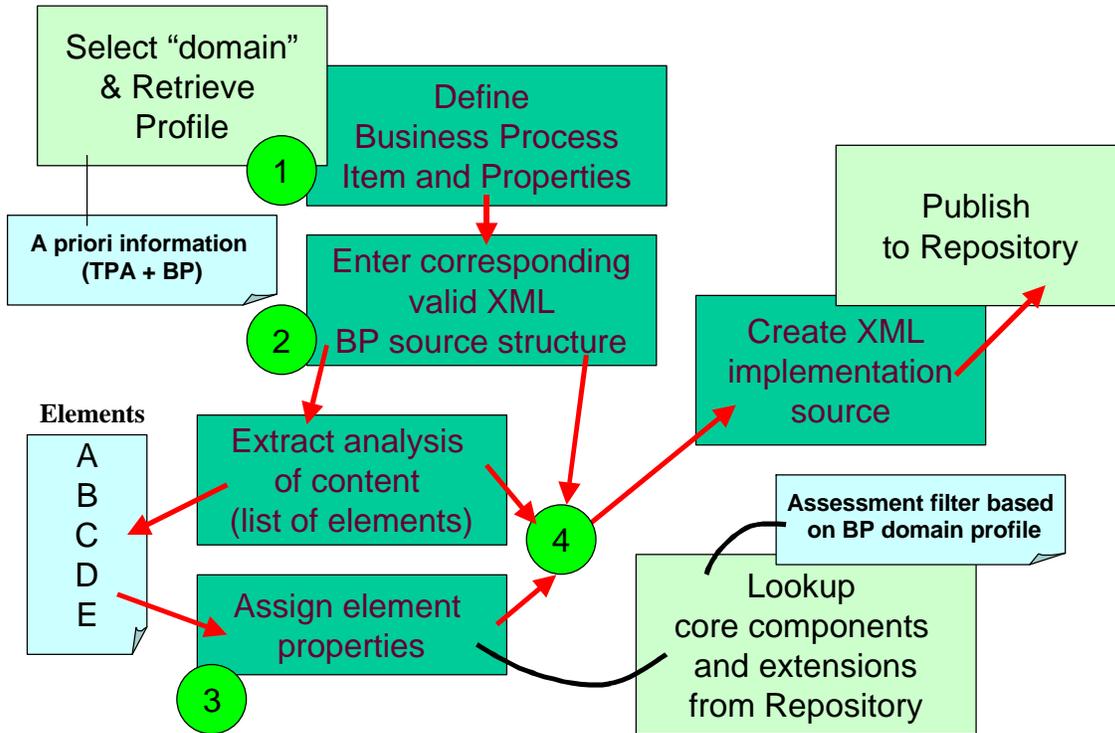
86

87

88

88 Figure 1. The ebXML concept diagram.

Empowering the business users



89

90 **The ebXML Business Concept Technical Details**

91 **Trading Partner Access (TPA) profile**

92 The TPA team will provide a valid XML structure containing user information (item 1 from
93 Figure 1). This will identify the business domain the user is within such as: Food Industry,
94 Car Manufacture, etc. It will also provide the context within that industry (eg: Accounting,
95 Regulatory, Supply, etc). Then it will also provide a profile of the user such as company,
96 security, and privilege level. This information will establish the context of the user access
97 and configure default information to the BP interface form as the next step.

98 This TPA profile will be retrieved when the user first logs into the demonstration.

99 **Registry/Repository**

100 When the user completes the login step, the registry will be referenced to retrieve the TPA
101 profile XML content. Each user has a URN associated (such as 'quikbuy.com:analyst')
102 that allows the registry to identify this information.

103 **Business Process (BP)**

104 The BP form will ask the user to define the business process context of the interchange
105 (item 2 from Figure 1) that is being defined by completing such information as: short
106 description, title, process, intent, context, usage, level, action (next, previous, fail) and an
107 optional method (such as a Java or CICS process URN). It will also ask the user to select
108 (using display file tool control) the specific XML transaction example physical file that
109 represents the transaction being documented. This file will then be loaded into memory
110 within the user interface forms, ready for the next stage.

111 **Core Components (CC)**

112 The CC form will automatically inspect the XML transaction example and create a pop-up
113 list of the constituent elements (technical note: XML parser DOM calls used for this). Each
114 element will be numbered sequentially.

115 A second pop-up list is then created alongside this by querying the RegRep for the already
116 existing core components that match the stated business domain and context (technical
117 note: this RegRep response in an XML structure and information is then also loaded in the
118 XML parser DOM within the user form interface).

119 The user selects the matching core components for each of the elements from the
120 business transaction (item 3 from Figure 1). Once this is done, the choices are confirmed,
121 and then the whole information, TPA account, BP, and CC are stored as XML structures
122 by passing an XML based update request to the RegRep (item 4 from Figure 1). This
123 interface detail will be defined by RegRep for storage of the applicable information. The
124 conceptual diagram (figure 2) anticipates that this XML syntax will consist of the RegRep
125 interface envelope, and then three layers, the classification layer (BP/CC), the structure
126 layer (DTD/Schema) and the element layer (CC/business semantics). Development of the
127 layer syntax will require coordination between the teams, just as the TPA syntax and
128 mechanism is also requiring coordination.

129 **Registry/Repository (RegRep)**

130 The RegRep interface will define the interaction XML syntax for the TPA and CC requests.
131 Once the update request has been generated by the CC module, then the final step is to

132 show a RegRep management screen that displays pending change requests, and allows
133 review of the request envelope (who requested and when) and then the actual XML
134 content itself (see addendum below for more detail on interface specifications).
135

136 Security

137 In the interest of simplicity, it is recommended that security not be implemented within the
138 initial demonstration environment.

139 Content Validation

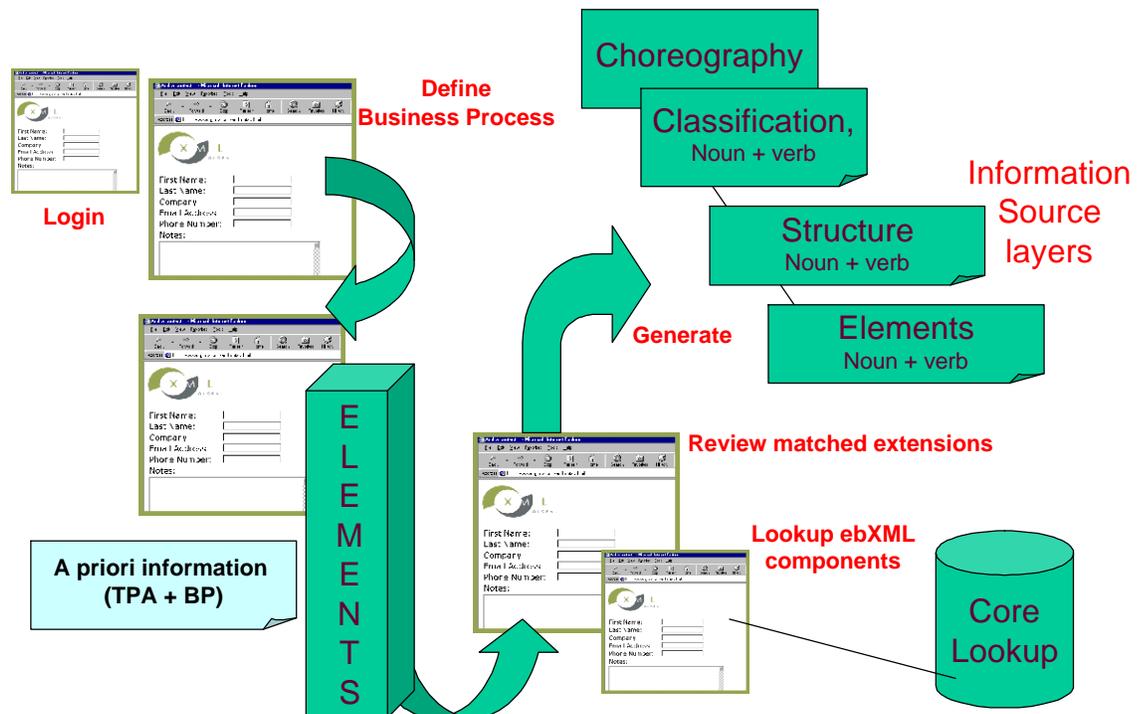
140 All content will be compliant to ebXML specifications and valid well-formed XML document
141 instances.

142 Implementation Environment and Vendor Roles

143 Since it is anticipated that the whole demonstration can be built to run in both IE and
144 Navigator using JavaScript specific vendor products will not be required. The diagram
145 below relates to Figure 1 and shows form based interfaces for each of the components
146 from Figure 1.

147 **Figure 2. Conceptual Demonstration Environment**

HTML forms + JavaScript + XML source



148
149

150 However, several vendors have committed to participating in the initiative, and so this will
151 be at the level of providing staff resources to ensure the components of the demonstration
152 are developed. Another aspect may be to have each vendor provide a single PowerPoint
153 slide that shows how that vendor anticipates using the capabilities of the Business

154 Concept demonstration within their own existing product(s). This will show applicability of
155 the technology and support from vendors in delivering solutions based on these ebXML
156 technology components.

157 Vendors so far indicating participation include: CommerceOne, Sun, Software AG, TIE
158 Systems, and XMLGlobal.

159 **ebXML Business Concept Caveats and Issues**

160 **Demonstration Implementation**

161 The premise is that it is more important to demonstrate the business functionality rather
162 than some network interoperability. The premise is if the concept works, it can be
163 implemented remotely over a network. Networks are known to work!

164

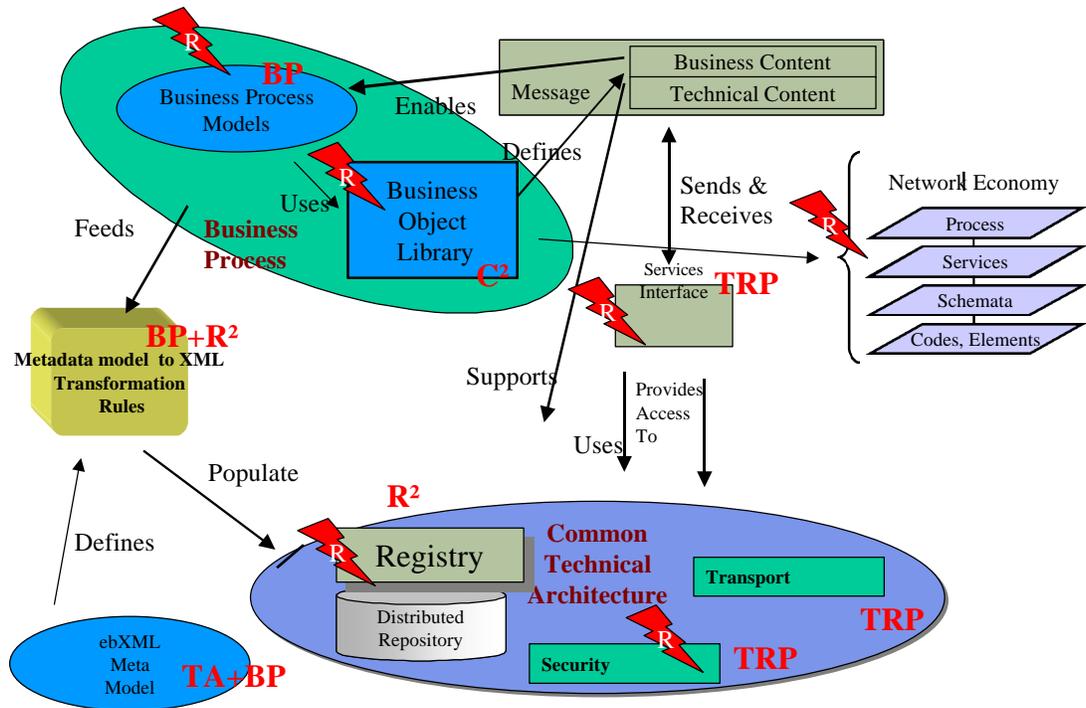
165

165 **Addendum:**

166 **Registry Repository Technical and Specification Notes.**

167 In San Jose we reviewed the potential interface points that the Registry/Repository will
168 need to support. Figure 3 shows these as red markers placed over the ebXML Technical
169 Architecture diagram.

170 **Figure 3. Registry/Repository Interfaces.**



171
172

173 **Summary of interfaces:**

174 A single interface using XML syntax will be provided (technical note: RPC based supported first, others
175 later). The interface will provide XML syntax to access both Registry services and Repository services
176 as either separate or combined interactions.

177 The ebXML component customers for these interactions will be:

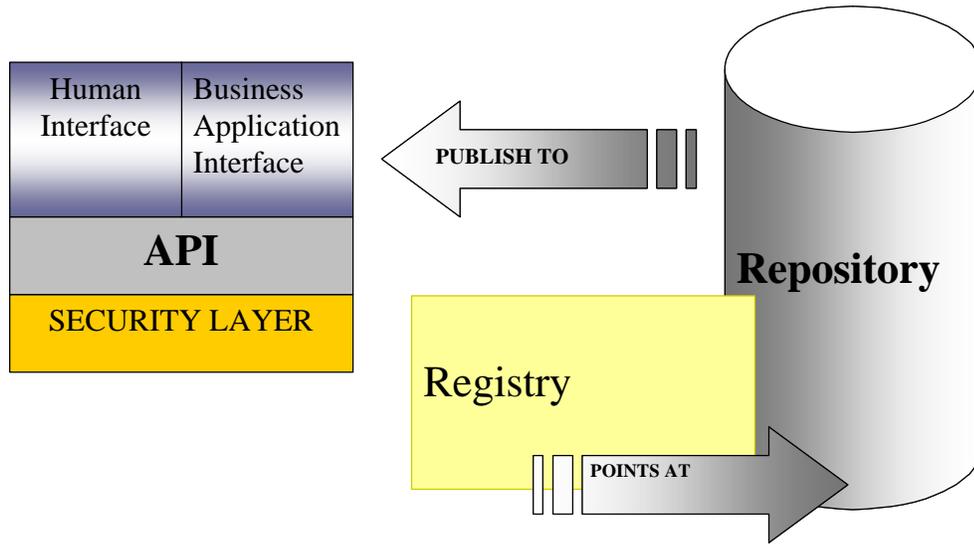
- 178 -Transport (TRP related)
- 179 -Local software process (batch / GUI queries) (CC related)
- 180 -Business Process Coordinator (BP/CC related)
- 181 -Discovery / Security (TPA related).

182
183
184 The interchange XML formats will support both query and update requests. The output returned in
185 response to a query will not be a generalized output set. Instead, specific discrete sets of response
186 information structures will be developed specifically for the ebXML concept demonstration interactions.

187

187

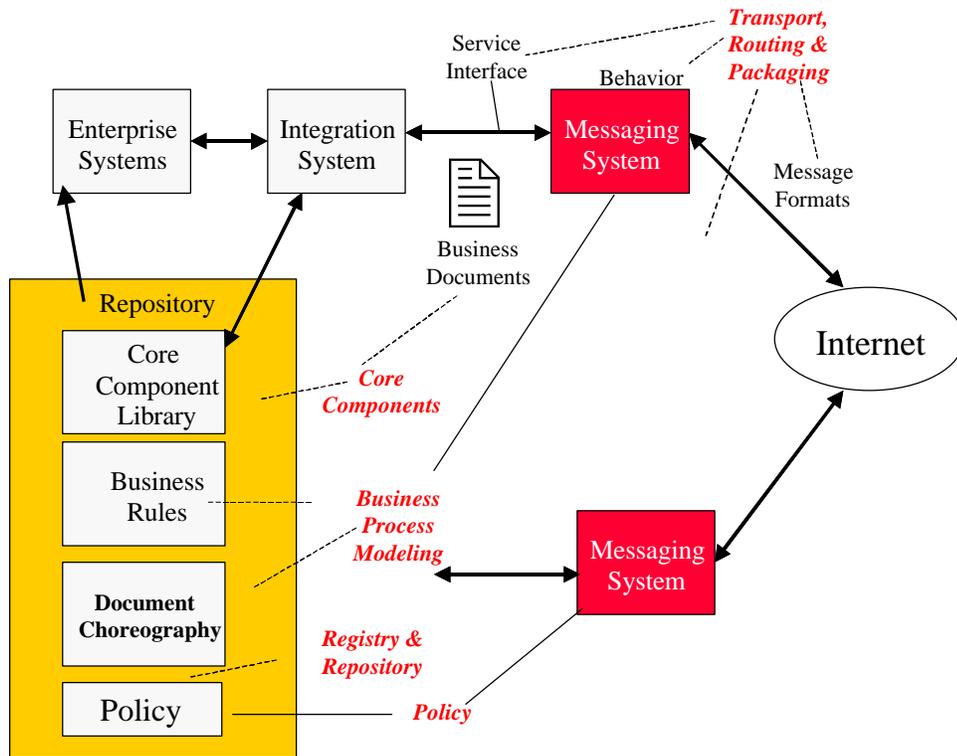
Figure 4. Query Interface Interaction Model



188

189

Figure 5. Architecture Interactions - Transaction Transport Delivery.



190

191

192

192

Figure 6. Registry/Repository Query Use Case.

193

194

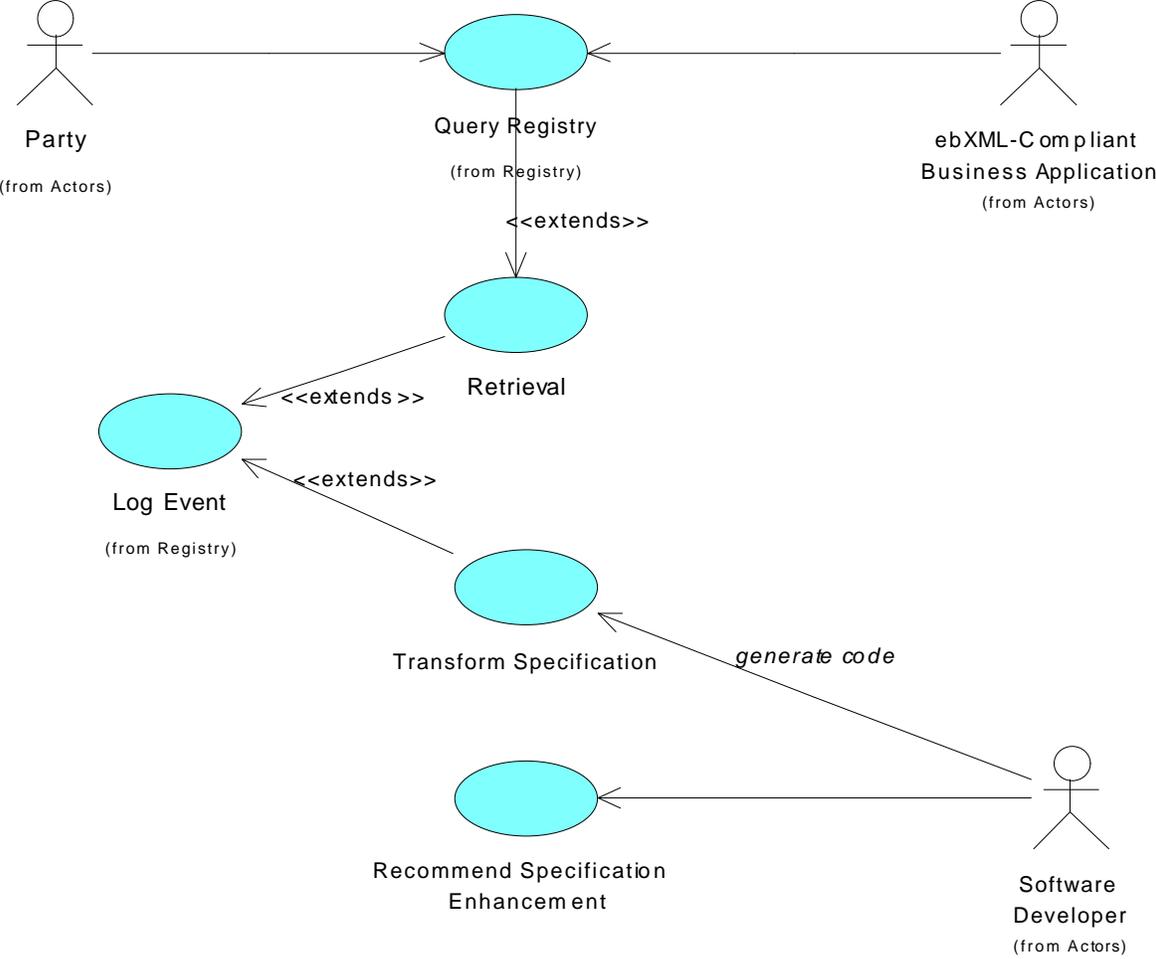
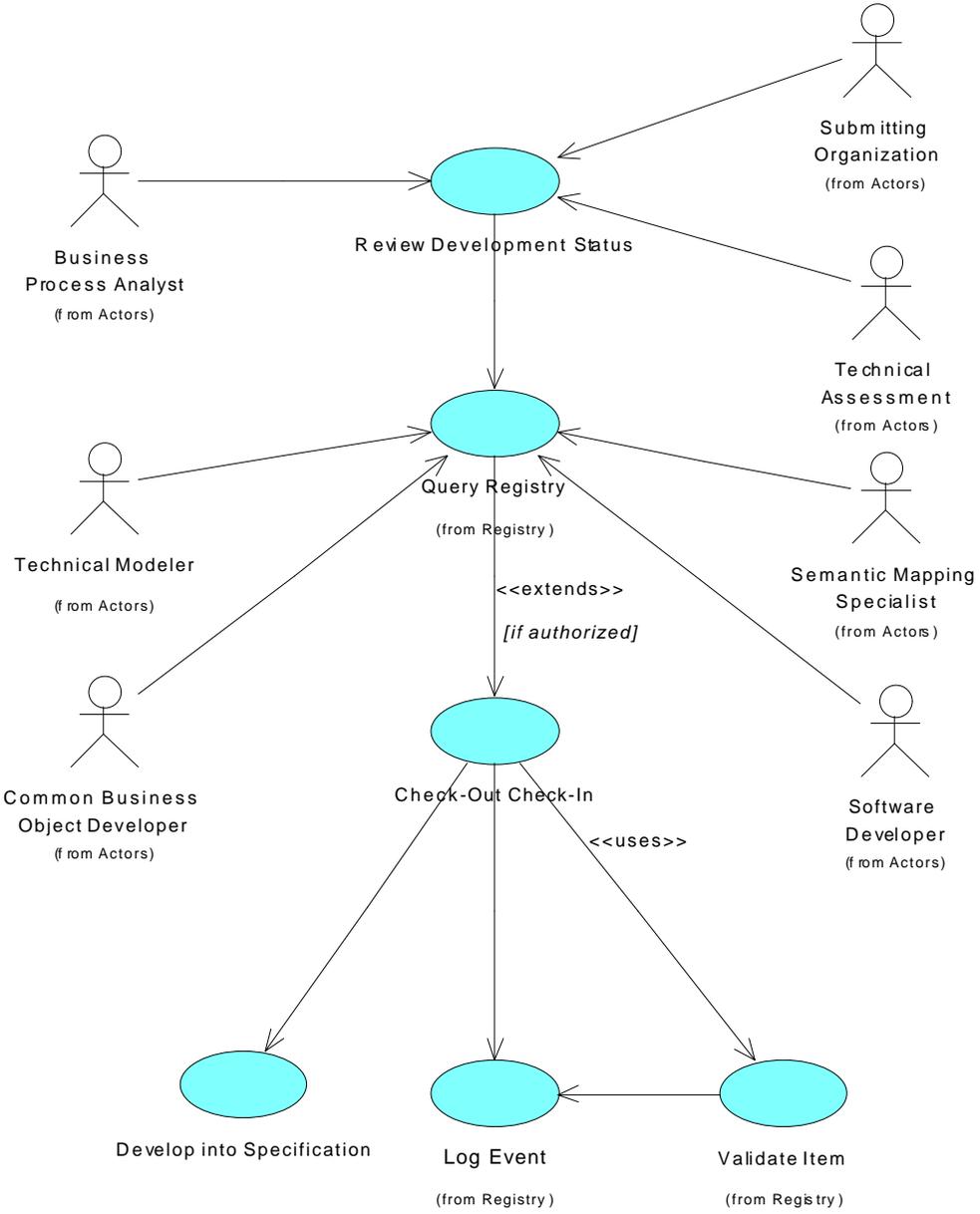


Figure 7. Registry/ Repository Query Actors.



Summary.

The items provided here provide linkage between the Part 1 and Part 2 RegRep Specification documents and the Tokyo POC approach.