**Foreword**

This is a supplement to the IHE Quality, Research and Public Health (QRPH) Technical Framework. Each supplement undergoes a process of public comment and trial implementation before being incorporated into the volumes of the Technical Frameworks.

This supplement is published on August 18, 2017 for trial implementation and may be available for testing at subsequent IHE Connectathons. The supplement may be amended based on the results of testing. Following successful testing it will be incorporated into the QRPH Technical Framework. Comments are invited and may be submitted at [http://www.ihe.net/QRPH_Public_Comments](http://www.ihe.net/QRPH_Public_Comments).

“Boxed” instructions like the sample below indicate to the Volume Editor how to integrate the relevant section(s) into the relevant Technical Framework volume.

Amend Section X.X by the following:

Where the amendment adds text, make the added text **bold underline**. Where the amendment removes text, make the removed text **bold strikethrough**. When entire new sections are added, introduce with editor’s instructions to “add new text” or similar, which for readability are not bolded or underlined.

General information about IHE can be found at [www.ihe.net](http://www.ihe.net).

Information about the IHE QRPH domain can be found at [http://www.ihe.net/IHE_Domains](http://www.ihe.net/IHE_Domains).

Information about the organization of IHE Technical Frameworks and Supplements and the process used to create them can be found at [http://www.ihe.net/IHE_Process](http://www.ihe.net/IHE_Process) and [http://www.ihe.net/Profiles](http://www.ihe.net/Profiles).

The current version of the IHE QRPH Technical Framework can be found at [http://www.ihe.net/Technical_Frameworks](http://www.ihe.net/Technical_Frameworks).
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.20.5 Security Considerations</td>
<td>29</td>
</tr>
<tr>
<td>3.20.6 Protocol Requirements</td>
<td>29</td>
</tr>
<tr>
<td>3.20.6.1 Sample SOAP Messages</td>
<td>30</td>
</tr>
<tr>
<td>3.20.6.1.1 Sample Retrieve Process Definitions SOAP Request</td>
<td>30</td>
</tr>
<tr>
<td>3.20.6.1.2 Sample Retrieve Process Definitions SOAP Response</td>
<td>30</td>
</tr>
<tr>
<td>3.22 Publish Process Definitions [QRPH-22]</td>
<td>31</td>
</tr>
<tr>
<td>3.22.1 Scope</td>
<td>32</td>
</tr>
<tr>
<td>3.22.2 Use Case Roles</td>
<td>32</td>
</tr>
<tr>
<td>3.22.3 Referenced Standards</td>
<td>32</td>
</tr>
<tr>
<td>3.22.4 Interaction Diagram</td>
<td>33</td>
</tr>
<tr>
<td>3.22.4.1 Publish Process Definitions Message</td>
<td>33</td>
</tr>
<tr>
<td>3.22.4.1.1 Trigger Events</td>
<td>33</td>
</tr>
<tr>
<td>3.22.4.1.2 Message Semantics</td>
<td>33</td>
</tr>
<tr>
<td>3.22.4.1.3 Expected Actions</td>
<td>34</td>
</tr>
<tr>
<td>3.22.4.2 Publish Process Definitions Response Message</td>
<td>34</td>
</tr>
<tr>
<td>3.22.4.2.1 Trigger Events</td>
<td>34</td>
</tr>
<tr>
<td>3.22.4.2.2 Message Semantics</td>
<td>34</td>
</tr>
<tr>
<td>3.22.4.2.3 Expected Actions</td>
<td>35</td>
</tr>
<tr>
<td>3.22.5 Security Considerations</td>
<td>35</td>
</tr>
<tr>
<td>3.22.6 Protocol Requirements</td>
<td>35</td>
</tr>
<tr>
<td>3.22.6.1 Sample SOAP Messages</td>
<td>36</td>
</tr>
<tr>
<td>3.22.6.1.1 Sample Publish Process Definitions SOAP Request</td>
<td>36</td>
</tr>
<tr>
<td>3.22.6.1.2 Sample Publish Process Definitions SOAP Response</td>
<td>37</td>
</tr>
<tr>
<td>3.25.1 Scope</td>
<td>37</td>
</tr>
<tr>
<td>3.25.2 Use Case Roles</td>
<td>38</td>
</tr>
<tr>
<td>3.25.3 Referenced Standards</td>
<td>38</td>
</tr>
<tr>
<td>3.25.4 Interaction Diagram</td>
<td>39</td>
</tr>
<tr>
<td>3.25.4.1 Initiate Process Message</td>
<td>39</td>
</tr>
<tr>
<td>3.25.4.1.1 Trigger Events</td>
<td>39</td>
</tr>
<tr>
<td>3.25.4.1.2 Message Semantics</td>
<td>39</td>
</tr>
<tr>
<td>3.25.4.1.3 Expected Actions</td>
<td>40</td>
</tr>
<tr>
<td>3.25.4.2 Initiate Process Response Message</td>
<td>40</td>
</tr>
<tr>
<td>3.25.4.2.1 Trigger Events</td>
<td>40</td>
</tr>
<tr>
<td>3.25.4.2.2 Message Semantics</td>
<td>40</td>
</tr>
<tr>
<td>3.25.4.2.3 Expected Actions</td>
<td>41</td>
</tr>
<tr>
<td>3.25.5 Security Considerations</td>
<td>41</td>
</tr>
<tr>
<td>3.25.6 Protocol Requirements</td>
<td>41</td>
</tr>
<tr>
<td>3.25.6.1 Sample SOAP Messages</td>
<td>42</td>
</tr>
<tr>
<td>3.25.6.1.1 Sample Initiate Process SOAP Request</td>
<td>42</td>
</tr>
<tr>
<td>3.25.6.1.2 Sample Initiate Process SOAP Response</td>
<td>42</td>
</tr>
<tr>
<td>3.26 Retrieve Activities [QRPH-26]</td>
<td>43</td>
</tr>
<tr>
<td>3.26.1 Scope</td>
<td>43</td>
</tr>
<tr>
<td>Page</td>
<td>Section</td>
</tr>
<tr>
<td>------</td>
<td>---------</td>
</tr>
<tr>
<td>140</td>
<td>3.26.2</td>
</tr>
<tr>
<td>143</td>
<td>3.26.3</td>
</tr>
<tr>
<td>144</td>
<td>3.26.4</td>
</tr>
<tr>
<td>145</td>
<td>3.26.4.1</td>
</tr>
<tr>
<td>146</td>
<td>3.26.4.1.1</td>
</tr>
<tr>
<td>147</td>
<td>3.26.4.1.2</td>
</tr>
<tr>
<td>148</td>
<td>3.26.4.1.3</td>
</tr>
<tr>
<td>149</td>
<td>3.26.4.2</td>
</tr>
<tr>
<td>150</td>
<td>3.26.4.2.1</td>
</tr>
<tr>
<td>151</td>
<td>3.26.4.2.2</td>
</tr>
<tr>
<td>152</td>
<td>3.26.4.2.3</td>
</tr>
<tr>
<td>153</td>
<td>3.26.5</td>
</tr>
<tr>
<td>154</td>
<td>3.26.6</td>
</tr>
<tr>
<td>155</td>
<td>3.26.6.1</td>
</tr>
<tr>
<td>156</td>
<td>3.26.6.1.1</td>
</tr>
<tr>
<td>157</td>
<td>3.26.6.1.2</td>
</tr>
<tr>
<td>158</td>
<td>3.27</td>
</tr>
<tr>
<td>159</td>
<td>3.27.1</td>
</tr>
<tr>
<td>160</td>
<td>3.27.2</td>
</tr>
<tr>
<td>161</td>
<td>3.27.3</td>
</tr>
<tr>
<td>162</td>
<td>3.27.4</td>
</tr>
<tr>
<td>163</td>
<td>3.27.4.1</td>
</tr>
<tr>
<td>164</td>
<td>3.27.4.1.1</td>
</tr>
<tr>
<td>165</td>
<td>3.27.4.1.2</td>
</tr>
<tr>
<td>166</td>
<td>3.27.4.1.3</td>
</tr>
<tr>
<td>167</td>
<td>3.27.4.2</td>
</tr>
<tr>
<td>168</td>
<td>3.27.4.2.1</td>
</tr>
<tr>
<td>169</td>
<td>3.27.4.2.2</td>
</tr>
<tr>
<td>170</td>
<td>3.27.4.2.3</td>
</tr>
<tr>
<td>171</td>
<td>3.27.5</td>
</tr>
<tr>
<td>172</td>
<td>3.27.6</td>
</tr>
<tr>
<td>173</td>
<td>3.27.6.1</td>
</tr>
<tr>
<td>174</td>
<td>3.27.6.1.1</td>
</tr>
<tr>
<td>175</td>
<td>3.27.6.1.2</td>
</tr>
<tr>
<td>176</td>
<td>3.28</td>
</tr>
<tr>
<td>177</td>
<td>3.28.1</td>
</tr>
<tr>
<td>178</td>
<td>3.28.2</td>
</tr>
<tr>
<td>179</td>
<td>3.28.3</td>
</tr>
<tr>
<td>180</td>
<td>3.28.4</td>
</tr>
<tr>
<td>181</td>
<td>3.28.4.1</td>
</tr>
<tr>
<td>182</td>
<td>3.28.4.1.1</td>
</tr>
<tr>
<td>183</td>
<td>3.28.4.1.2</td>
</tr>
<tr>
<td>184</td>
<td>3.28.4.1.3</td>
</tr>
</tbody>
</table>
3.28.4.2 Send Process State Alert Response Message ............................................. 55
  3.28.4.2.1 Trigger Events ....................................................................................... 55
  3.28.4.2.2 Message Semantics ............................................................................. 55
  3.28.4.2.3 Expected Actions ................................................................................ 55
3.28.5 Security Considerations .................................................................................. 55
3.28.6 Protocol Requirements ................................................................................... 55
  3.28.6.1 Sample SOAP Messages ........................................................................ 56
  3.28.6.1.1 Sample Send Process State Alert SOAP Request ............................... 57
  3.28.6.1.2 Sample Send Process State Alert SOAP Response .............................. 57
Introduction

Retrieve Process for Execution (RPE) is a profile for collaborative workflow or collaborative process management involving three main actors:

1. the manager of process definitions
2. the manager of runtime processes
3. the performer of process activities

Note that the process definition contains specification for all activities to be performed. An activity defines a specific act, several of which define the entire process.

RPE started as Retrieve Protocol for Execution Profile, a proposed automated mechanism for an Electronic Health Record (EHR) to retrieve a complex set of clinical research instructions (a study protocol) from a research sponsor’s Electronic Data Capture (EDC) system or research site’s Clinical Trials Management System (CTMS) and to execute these instructions within the EHR. Interest has since grown to expand RPE to support processes from other domains involving collaborative workflows like quality, public health and patient care support. RPE now does this consistent with accepted IT standards for process management. In September of 2011, RPE underwent a substantial re-write that required current users of RPE to re-develop their implementations of the profile. The current version harmonizes the original Retrieve Protocol for Execution and the new Retrieve Process for Execution to lessen the burden on users of the profile to conform to the profile.

RPE defines the transport mechanism for process definitions, but does not provide the definitions per se. Additional profiles that define specific processes will complement RPE and bring the process automation capability to completion. One of these profiles is the Clinical Research Process Content (CRPC) Profile.

Open Issues and Questions

No open issues at this time.

Closed Issues

No closed issues at this time.
Volume 1 – Integration Profiles
X Retrieve Process for Execution (RPE) Profile

Retrieve Process for Execution (RPE) is a profile for collaborative workflow or collaborative process management involving three main actors:

1. the manager of process definitions
2. the manager of runtime processes
3. the performer of process activities

X.1 RPE Actors/Transactions

Figure X.1-1 shows the actors directly involved in the Retrieve Process for Execution Profile and the relevant transactions between them. Other actors that may be indirectly involved due to their participation in other profiles are not necessarily shown.

Table X.1-1 lists the transactions for each actor directly involved in the RPE Profile. In order to claim support of this integration profile, an implementation must perform the required transactions (labeled “R”). Transactions labeled “O” are optional. A complete list of options defined by this integration profile, which implementations may choose to support, is listed in Section X.2.
Table X.1-1: RPE Actors and Transactions

<table>
<thead>
<tr>
<th>Actors</th>
<th>Transactions</th>
<th>Optionality</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process State Manager</td>
<td>Retrieve Process Definitions [QRPH-20]</td>
<td>O (Note 1)</td>
<td>QRPH TF-2: 3.20</td>
</tr>
<tr>
<td></td>
<td>Publish Process Definitions [QRPH-22]</td>
<td>O (Note 1)</td>
<td>QRPH TF-2: 3.22</td>
</tr>
<tr>
<td></td>
<td>Retrieve Activities [QRPH-26]</td>
<td>O</td>
<td>QRPH TF-2: 3.26</td>
</tr>
<tr>
<td></td>
<td>Update Activity [QRPH-27]</td>
<td>O</td>
<td>QRPH TF-2: 3.27</td>
</tr>
<tr>
<td></td>
<td>Send Process State Alert [QRPH-28]</td>
<td>R</td>
<td>QRPH TF-2: 3.28</td>
</tr>
<tr>
<td>Process Activity Executor</td>
<td>Retrieve Process Definitions [QRPH-20]</td>
<td>O (Note 2)</td>
<td>QRPH TF-2: 3.20</td>
</tr>
<tr>
<td></td>
<td>Publish Process Definitions [QRPH-22]</td>
<td>O (Note 2)</td>
<td>QRPH TF-2: 3.22</td>
</tr>
<tr>
<td></td>
<td>Retrieve Activities [QRPH-26]</td>
<td>O</td>
<td>QRPH TF-2: 3.26</td>
</tr>
<tr>
<td></td>
<td>Update Activity [QRPH-27]</td>
<td>O</td>
<td>QRPH TF-2: 3.27</td>
</tr>
<tr>
<td></td>
<td>Send Process State Alert [QRPH-28]</td>
<td>R</td>
<td>QRPH TF-2: 3.28</td>
</tr>
</tbody>
</table>

Note 1 – A Process State Manager SHALL implement either the Retrieve Process Definitions or Publish Process Definitions transactions.

Note 2 – A Process Activity Executor SHALL implement either the Retrieve Process Definitions or Publish Process Definitions transactions.

X.1.1 Actor Descriptions and Requirements

X.1.1.1 Process Definition Manager

The Process Definition Manager manages access to a repository of process definitions, allowing for search and retrieval.

An example would be a research sponsor providing access to the clinical research processes it has developed. Another would be a provider of evidence-based clinical practice guidelines.

X.1.1.2 Process State Manager

The Process State Manager manages the initiation and state of runtime process instances. The Process State Manager typically also supports the initiation and lifecycle management of task activities associated with a process while providing the ability for task performers to retrieve and update activities.

An example would be a research sponsor conducting clinical trials in conjunction with its EHR participants. Another would be an EHR performing clinical care in accordance with executable guideline-based processes.

The states of a process from the perspective of RPE actors are indicated below:
All state transitions above are managed by the Process State Manager which can then use the Send Process State Alert transaction to notify the Process Activity Executor.

**X.1.1.3 Process Activity Executor**

The Process Activity Executor performs activities as prescribed by a running process being managed by a Process State Manager. The Process Activity Executor retrieves current activity or task lists, works its list, updating the Process State Manager on activity state until completion. This cycle is repeated until all process activities have been worked and the process itself completes.

An example would be an EHR performing activities as part of a clinical trial being managed by a research sponsor. Another example would be an EHR performing guideline-based care process activities.

The states of an activity from the perspective of RPE actors are indicated below:
The Process State Manager is responsible for transitioning the activity into the “Active” state and possibly forcing it into a non-normal end state, i.e., “Exited” and using the Send Process State Alert transaction to notify the Process Activity Executor.

The Process Activity Executor is responsible for marking the activity “Completed” and using the Update Activity transaction to update the Process State Manager with this change as well as any associated output data.

**X.2 RPE Actor Options**

Options that may be selected for each actor in this profile, if any, are listed in the Table X.2-1.

<table>
<thead>
<tr>
<th>Actor</th>
<th>Options</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process State Manager</td>
<td>Retrieve Process Definitions</td>
<td>Section X.2.1</td>
</tr>
<tr>
<td></td>
<td>Publish Process Definitions</td>
<td>Section X.2.4</td>
</tr>
<tr>
<td></td>
<td>Retrieve Activities</td>
<td>Section X.2.2</td>
</tr>
<tr>
<td></td>
<td>Update Activity</td>
<td>Section X.2.3</td>
</tr>
<tr>
<td>Process Activity Executor</td>
<td>Retrieve Process Definitions</td>
<td>Section X.2.1</td>
</tr>
</tbody>
</table>

*Figure X.1.1.3-1: Process Activity Flow*
The optional transactions extend the capabilities of the participating actor. Each actor can participate in any or all appropriate optional transactions.

**X.2.1 Retrieve Process Definitions Option**

The Retrieve Process Definitions Option allows a Process State Manager or Process Activity Executor to retrieve the process definition. An actor that supports the Retrieve Process Definitions Option shall support the [QRPH-20] transaction.

**X.2.2 Retrieve Activities Option**

The Retrieve Activities Option allows a Process Activity Executor to retrieve the current set of activities it needs to execute as part of processes it is a participant in and managed by the given Process State Manager. An actor that supports the Retrieve Activities Option shall support the [QRPH-26] transaction.

**X.2.3 Update Activity Option**

Update Activity allows a Process Activity Executor to provide an update on activity’s state or data to a Process State Manager for a process it is a participant in. An actor that supports the Update Activity Option shall support the [QRPH-27] transaction.

**X.2.4 Publish Process Definitions Option**

Publish Process Definitions allows a Process Definition Manager to push updates to processes to Process Activity Executors and Process State Managers. An actor that supports the Publish Process Definitions Option shall support the [QRPH-22] transaction.

**X.3 Groupings**

No groupings specified.

**X.4 RPE Process Flow**

**X.4.1 Use Cases**

The clinical research use case below describes the before and after effects of implementing the Retrieve Process for Execution (RPE) Profile for an investigational new drug clinical trial scenario.
Scenario

The setting for the clinical trial use case is a physicians’ practice where patient care is delivered side-by-side with clinical research. The site, Fictitious Medical Group, is a multi-site physician practice, employing over 100 physicians in a variety of specialties. The Fictitious CEO encourages the physicians to participate as site investigators for pharmaceutical-sponsored clinical trials.

Before RPE

Preconditions

1. A Clinical Research Process is defined by a clinical trials expert at Sponsor, a biopharma research company.
2. Fictitious provides support for clinical research activities in the form of a Research Department of twelve dedicated study coordinators, mostly RNs, along with clerical and data-entry support personnel.
3. Fictitious Medical Group uses an Electronic Health Record (EHR) and a number of sponsor-provided Electronic Data Capture (EDC) systems for documenting clinical trial activities.

Clinical Research Site's Involvement:

Fictitious involvement in a clinical study begins when the Research Department receives a request for proposal (RFP) from Sponsor. A Study Coordinator, Patricia Zone, RN, evaluates the RFP for business viability and clinical appropriateness, provides the requested documentation back to the sponsor, and agrees to participate. After being approved as a site for the Sponsor #1234 trial, the Fictitious Medical Group provides the required regulatory documentation to the sponsor.

Following trial set up, Patricia contacts Corey Jones, a patient at Fictitious, about participating in the trial and Corey agrees to participate as a subject. A number of tasks deal with this individual patient:

1. Obtain proper consent and other documentation from study candidate Corey Jones.
2. Register Corey in the EHR as a candidate in trial #1234, using the EHR’s patient index.
3. Register Corey as a candidate in the EDC system.
4. Schedules Corey’s study screening visits using the EHR scheduling module, and flag the visits as pertaining to the trial #1234.
5. Examine screening results to confirm continued enrollment in the trial.
6. After screening, Patricia obtains and schedules the next set of study activities corresponding to the path (arm) of the Sponsor #1234 trial.
Post conditions

1. Fictitious Medical Group uses an EHR and the EDC system to document the Sponsor #1234 trial activities.

After RPE

Preconditions

1. A Clinical Research Process is defined by the clinical trials expert at Sponsor using a study design tool.
2. The resulting study design definition document is stored and made available for access by a Process Definition Manager (implemented by an EDC or other system).
3. A Process State Manager (implemented by an EDC or other system) is available to deploy, run and manage the execution of the clinical trial process.
4. The Fictitious EHR (or other system) can implement the Process Activity Executor role.

Clinical Research Site's Involvement:

Initial:

1. The Process Definition Manager pushes a list of processes to the Process Activity Executor

Screening:

1. Process Activity Executor uses the Initiate Process transaction to notify the Process State Manager that the site wishes to enter a patient into the study.
2. Process Activity Executor uses the Retrieve Activities transaction to obtain from the Process State Manager the set of screening activities.
3. Process Activity Executor uses the Update Activity transaction to send the Process State Manager the screening results. (Note: if activities have associated RFD forms then Update Activity may be replaced by RFD form retrieval and submission.)
4. Process State Manager uses the Send Process State Alert transaction to notify the Process Activity Executor that the patient passed screening and has been enrolled in the trial.

Treatment:

1. Process Activity Executor uses the Retrieve Activities transaction again to obtain from the Process State Manager the next set of study activities for this patient.
2. Process Activity Executor uses the Update Activity transaction to send the Process State Manager updates of activities as they are performed. (Note: if activities have associated RFD forms then Update Activity may be replaced by RFD form retrieval and submission.)
3. Process Activity Executor can, at any time, use the Send Process State Alert transaction to notify the Process State Manager that the patient has withdrawn from the trial.
4. Process State Manager can at any time use the Send Process State Alert transaction to notify the Process Activity Executor that the trial has been placed on hold.

**X.4.2 Process Flow**

The basic process flow for RPE is shown below:

![Diagram of Basic Process Flow in RPE Profile](image)

**Figure X.4.2-1: Basic Process Flow in RPE Profile**

1. The Process Definition Manager pushes process definitions of interest to the Process Activity Executor.

2. The Process Definition Manager pushes process definitions of interest to the Process State Manager.

3. The Process Activity Executor requests process initiation by the Process State Manager forwarding a given process definition identifier as well as other required data, e.g., a patient identifier, demographics or eligibility criteria.
4. The Process State Manager notifies the Process Activity Executor that the process is actively proceeding or otherwise, e.g., after initial screening of patient data.

5. The Process Activity Executor retrieves the current activity it has to perform.

6. After completion of an activity, the Process Activity Executor sends the Process State Manager the updated activity state and output data.

7. The Process Activity Executor can always notify the Process State Manager of unscheduled events that may affect the process state, e.g., patient withdrawal from a clinical trial.

X.5 RPE Security Considerations

The risk analysis for RPE enumerates assets, threats, and mitigations. The complete risk data is stored and available from IHE.

The purpose of this risk assessment is to notify vendors of some of the risks that they are advised to consider when implementing RPE actors. For general IHE risks and threats, please see ITI TF-1: Appendix L. The vendor is also advised that many risks cannot be mitigated by the IHE profile and instead responsibility for mitigation is transferred to the vendor, and occasionally to the affinity domains, individual enterprises and implementers. In these instances, IHE fulfills its responsibility to notify affected parties through the use of the following sections.

<table>
<thead>
<tr>
<th>Risk Scenario</th>
<th>Type of Impact</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1: Patient is misidentified during process initiation, activity retrieval initiation or update. Demographic or identification information is intermixed with that of another patient.</td>
<td>Loss of Data Integrity resulting in potentially incorrect diagnosis, treatment and/or data analysis, possibly resulting in loss of life or quality of life for the patient and/or errant analysis in the context of research, public health or quality processes.</td>
<td>M</td>
</tr>
<tr>
<td>S2: A malicious attacker may attempt to compromise the system to obtain wrongful access to patient identity, clinical, financial or insurance data.</td>
<td>Loss of Privacy, Legal Liability / Compensation, Loss / Decrease of Funding, Loss of Public Trust or Reputation for Patient, Provider, Agency, Institution or Organization, Accountable Employee Loses Job</td>
<td>M</td>
</tr>
<tr>
<td>S3: Accidental release of personal health information (PHI) protected by consent agreements, regulation or law.</td>
<td>Loss of Privacy, Legal Liability / Compensation, Loss / Decrease of Funding, Loss of Public Trust or Reputation for Patient, Provider, Agency, Institution or Organization, Accountable Employee Loses Job</td>
<td>M</td>
</tr>
</tbody>
</table>

### X.5.1 Recommendations

The high impact risks include:

- mismatch between patient and patient data
- malicious system attack
- inappropriate or premature disclosure of personal health information
- patient and organization identity protection

For risk mitigation, the following profiles from the ITI Technical Framework are referred:

- Audit Trail and Node Authentication (ATNA)
- Basic Patient Privacy Consents (BPPC)
- Cross-Enterprise User Assertion (XUA)
- Consistent Time (CT)
- Enterprise User Authentication (EUA)
- Document Digital Signature (DSG)

This profile includes the mitigations:

**M1**: Ensure additional demographics are available to verify patient identity (e.g., address, other identifiers)

**M2**: Identify actors, ensure secure login and access control to protected data, e.g., using XUA and grouping with EUA for protected applications.

**M3**: Use secured communication for any patient data e.g., as per ATNA.
M4: Audit access to PHI, e.g., as per ATNA.

M5: Obtain patient consent to release protected data, e.g., using BPPC and DSG.

M6: Use pseudonymized identifiers for organizations in transactions as supported by RPE.

These mitigations are transferred to vendors and clients:

T1: Verify demographics other than patient name, birth date and gender.

T2: Providers evaluate and review activity data before submission to ensure data is entered correctly for the correct patient. Providers are cautioned not to use RPE for unmediated treatment or diagnosis, i.e., a doctor must always intervene prior to treatment or diagnosis to ensure that errors that may occur in submission are checked by a human prior to engaging in any treatment or diagnosis of a patient.

T3: Secure internal networks from unauthorized access.

T4: Ensure strong password use for applications with access to PHI.

T5: Employ a restrictive RBAC scheme for applications providing access to PHI.

T6: Obtain patient consents and ensure patient data requiring increased protection is clearly identified.

T7: Workflow processes should be designed to ensure provider to patient communication occurs prior to sharing results with negative implications.

T8: Assign pseudonymized identifiers in processes when it is required to shield real identities.

T9: Actors are advised to consider the usability of their logging and audit repository implementation.

X.6 RPE Version mapping

The following tables specify mappings from the original Retrieve Protocol for Execution to the new Retrieve Process for Execution to assist implementers in translating their original RPE actors, transactions, and parameters to the new actors, transactions, and parameters.

<table>
<thead>
<tr>
<th>Retrieve Protocol for Execution Actor</th>
<th>Retrieve Process for Execution Actor</th>
</tr>
</thead>
<tbody>
<tr>
<td>ProtocolDefManager</td>
<td>Process Definition Manager</td>
</tr>
<tr>
<td>ProtocolExecutor</td>
<td>Process Activity Executor</td>
</tr>
<tr>
<td>ProtocolStateManager</td>
<td>Process State Manager</td>
</tr>
</tbody>
</table>
Table X.6-2: Transaction Mapping

<table>
<thead>
<tr>
<th>Retrieve Protocol for Execution Transaction</th>
<th>Retrieve Process for Execution Transaction</th>
</tr>
</thead>
<tbody>
<tr>
<td>PatientScreeningVisitsScheduled – Retired [QRPH-12]</td>
<td>UpdateActivity [QRPH-27]</td>
</tr>
<tr>
<td>RecordPatientStudyVisit – Retired [QRPH-16]</td>
<td>UpdateActivity [QRPH-27]</td>
</tr>
</tbody>
</table>

Table X.6-3: Parameter Mapping

<table>
<thead>
<tr>
<th>Retrieve Protocol for Execution Parameter</th>
<th>Retrieve Process for Execution Parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>query</td>
<td>processDefinitionIdentifier</td>
</tr>
<tr>
<td>ProtocolDefs</td>
<td>processDefinition</td>
</tr>
<tr>
<td>contentType</td>
<td>processDefinitionLanguage</td>
</tr>
<tr>
<td>study/id</td>
<td>processDefinitionIdentifier</td>
</tr>
<tr>
<td>subjectID</td>
<td>processIdentifier</td>
</tr>
<tr>
<td>schedule</td>
<td>activity</td>
</tr>
<tr>
<td>visit</td>
<td>activity</td>
</tr>
</tbody>
</table>
Appendix A – Actor Summary Definitions

**Process Definition Manager** – A system that manages access to a repository of process definitions allowing for search and retrieval. An example would be a research sponsor providing access to the clinical research processes it has developed. Another would be a provider of evidence-based clinical practice guidelines.

**Process State Manager** – A system that manages the initiation and state of runtime process instances. The Process State Manager typically also supports the initiation and lifecycle management of task activities associated with a process while providing the ability for task performers to retrieve and update activities. An example would be a research sponsor conducting clinical trials in conjunction with its EHR participants. Another would be an EHR performing clinical care in accord with executable guideline-based processes.

**Process Activity Executor** – A system that performs activities as prescribed by a running process being managed by a Process State Manager. The Process Activity Executor retrieves current activity or task lists, works its list, updating the Process State Manager on activity state until completion. This cycle is repeated until all process activities have been worked and the process itself completes. An example would be an EHR performing activities as part of a clinical trial being managed by a research sponsor. Another would be an EHR performing guideline-based care process activities.
Appendix B – Transaction Summary Definitions

Retrieve Process Definitions [QRPH-20] – enables access to one or more process definitions specified by an identifier or other query criteria. This transaction is implemented by the Process Definition Manager and used by both the Process State Manager – to deploy processes it wishes to manage – and the Process Activity Executor – to examine processes it may be interested in becoming an activity participant.

Initiate Process [QRPH-25] – enables a Process Activity Executor to initiate a new process to be managed by a Process State Manager, e.g., an EHR entering a new patient candidate in a clinical trial being managed by a research sponsor.

Retrieve Activities [QRPH-26] – enables a Process Activity Executor to retrieve the current set of activities it needs to execute as part of a process managed by a Process State Manager.

Update Activity [QRPH-27] – allows a Process Activity Executor to provide an update on activity’s state or data to a Process State Manager for a process it is a participant in.

Send Process State Alert [QRPH-28] – provides the Process State Manager and Process Activity Executor the ability to notify each other of unscheduled events that affect the state of the process, e.g., an EHR patient withdrawing from a clinical trial or, a study being placed on hold.
Glossary

*Add the following terms to the Glossary:*

**Process Definition** – A designed flow of activities involving one or more role-based activity performers, implemented in XML and deployable to a runtime process consumer.

**Process** – A specific instance of a process definition running in a process consumer.

**Activity Definition** – A designed task which is deployable to a runtime activity processor, typically as part of a process definition.

**Activity** – A specific instance of an activity definition created in, and available from, an activity processor.
3 IHE Transactions

3.20 Retrieve Process Definitions [QRPH-20]

This section corresponds to transaction [QRPH-20] of the IHE QRPH Transaction Framework. [QRPH-20] is used by the Process Definition Manager and Process Activity Executor Actors.

3.20.1 Scope

This transaction involves a Process Activity Executor or Process State Manager requesting one or more process definitions from a Process Definition Manager. The Process Activity Executor or Process State Manager has one or more process definition identifiers obtained by means outside the scope of this profile.

The Process Definition Manager returns either a list of matching process definitions or an error response.

3.20.2 Use Case Roles

![Use Case Diagram]

- Process Activity Executor
- Process Definition Manager
- Process State Manager
**Actor:** Process Activity Executor  
**Role:** A system that knows how to execute activities that are part of a process.

**Actor:** Process State Manager  
**Role:** A system that manages the runtime state of a process.

**Actor:** Process Definition Manager  
**Role:** A system that provides a set of process definitions upon request.

### 3.20.3 Referenced Standards
Implementers of this transaction shall comply with all requirements described in:
- ITI TF-2x: Appendix V Web Services for IHE Transactions
- Extensible Markup Language (XML) 1.0 (Second Edition). W3C Recommendation 6 October 2000. [http://www.w3.org/TR/REC-xml](http://www.w3.org/TR/REC-xml)

### 3.20.4 Interaction Diagram

![Interaction Diagram](image-url)

[QRPH-20]
3.20.4.1 Retrieve Process Definitions Message

3.20.4.1.1 Trigger Events

The Process Activity Executor or Process State Manager, based on human decision or the application of a rule for automatic operation, wants to obtain from the Process Definition Manager one or more process definitions matching a set of previously supplied identifiers.

3.20.4.1.2 Message Semantics

The following parameters are specified for the body of this transaction.

<table>
<thead>
<tr>
<th>Table 3.20.4.1-2-1: Retrieve Process Definitions Request Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter Name</td>
</tr>
<tr>
<td>processDefinitionIdentifier (list)</td>
</tr>
<tr>
<td>maxOccurs</td>
</tr>
<tr>
<td>startIndex</td>
</tr>
</tbody>
</table>

Section 3.20.6 describes the Web Services protocol requirements and the format of the message in full detail.

2 HL7 is the registered trademark of Health Level Seven International.
3.20.4.1.3 Expected Actions

Upon receipt of the Retrieve Process Definitions Message, the Process Definition Manager shall parse the request and return the requested process definitions in the Retrieve Process Definitions Response message.

The successful response shall be one or more well-formed XML documents matching the supplied identifiers. Otherwise SOAP faults shall be generated accordingly.

3.20.4.2 Retrieve Process Definitions Response Message

3.20.4.2.1 Trigger Events

The delivery of a set of process definitions is triggered by a Process Definition Manager in response to a Retrieve Process Definitions message.

3.20.4.2.2 Message Semantics

A matching list of one or more process definitions is returned. The format of each process definition is a well-formed XML document.

Table 3.20.4.2.2-1: Retrieve Process Definitions Response Parameter

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Optionality</th>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>@status</td>
<td>R</td>
<td>Indicates whether processing was successful</td>
<td>See Table 3.20.4.2-3.</td>
</tr>
<tr>
<td>processDefinition</td>
<td>R</td>
<td>One process definition per process definition identifier requested.</td>
<td>A complex XML element of type specified per processDefinitionLanguage attribute</td>
</tr>
<tr>
<td>processDefinitionLanguage</td>
<td>R</td>
<td>XML attribute</td>
<td>Defined by content profile.</td>
</tr>
<tr>
<td>processErrorList</td>
<td>O</td>
<td>If errors were encountered in processing (see @status), error list is defined here.</td>
<td>See Table 3.20.4.2-4.</td>
</tr>
</tbody>
</table>

Table 3.20.4.2.2-2: Process Response Status Codes

<table>
<thead>
<tr>
<th>Status</th>
<th>processErrorList element</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>urn:ihe:qrph:rpe:2009:ResponseStatusType:Success</td>
<td>May be present. If present, will contain one or more ProcessError elements with warning severity; none with error severity</td>
<td>Process will be processed.</td>
</tr>
<tr>
<td>urn:ihe:qrph:rpe:2009:ResponseStatusType:PartialSuccess</td>
<td>Present, contains one or more ProcessError elements. At least one has error severity; others may have warning severity.</td>
<td>Process will be processed.</td>
</tr>
<tr>
<td>urn:ihe:qrph:rpe:2009:ResponseStatusType:Failure</td>
<td>Present, contains one or more ProcessError elements. At least one has error severity; others may have warning severity.</td>
<td>Process will not be processed.</td>
</tr>
</tbody>
</table>
Table 3.20.4.2-3: ProcessErrorList Structure

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Optionality</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>@highestSeverity</td>
<td>R</td>
<td>The highest severity error that is included in the processErrorList. Table 3.20.4.1.2-3.</td>
</tr>
<tr>
<td>processError</td>
<td>R [1..*]</td>
<td>See Table 3.20.4.1.2-3.</td>
</tr>
</tbody>
</table>

Table 3.20.4.2-4: ProcessError Structure

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Optionality</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>errorCode</td>
<td>R</td>
<td>Shall be taken from Table 3.20.4.1.2-6 when one of those codes is appropriate. All extensions to the list of error codes shall be unique.</td>
</tr>
<tr>
<td>codeContext</td>
<td>R</td>
<td>Supplies additional detail for the errorCode</td>
</tr>
<tr>
<td>location</td>
<td>O</td>
<td>Supplies the location of the error: module name and line number or stack trace if appropriate.</td>
</tr>
</tbody>
</table>

Table 3.20.4.2-5: Error Codes

<table>
<thead>
<tr>
<th>errorCode</th>
<th>Discussion</th>
</tr>
</thead>
<tbody>
<tr>
<td>RPEInvalidProcessContent</td>
<td>The recipient has rejected this submission because it detected that the content did not match the processDefinitionLanguage provided.</td>
</tr>
<tr>
<td>RPEActivityExecutorError</td>
<td>Internal Error</td>
</tr>
<tr>
<td>RPEDefinitionManagerError</td>
<td>These error codes shall be returned if and only if a more detailed code is not available for the condition being reported. If one of these error codes is returned, the attribute codeContext shall contain details of the error condition that may be implementation-specific.</td>
</tr>
<tr>
<td>RPEStateManagerError</td>
<td>Internal Error</td>
</tr>
<tr>
<td>RPEUnknownProcessContent</td>
<td>The recipient has rejected this submission because it detected that the content processDefinitionLanguage was not one that could be handled by that actor.</td>
</tr>
<tr>
<td>RPEUnknownCandidateId</td>
<td>Receiving system is unable to identify the patient from the provided Patient ID.</td>
</tr>
<tr>
<td>RPEInvalidCandidateDemographics</td>
<td>The patient demographics on file do not match the patient demographics received in the message.</td>
</tr>
<tr>
<td>RPEUnknownProcessDefinitionIdentifier</td>
<td>The requested process definition identifier is not known by the receiving system.</td>
</tr>
<tr>
<td>RPEUnknownProcessIdentifier</td>
<td>The requested process identifier is not known by the receiving system.</td>
</tr>
<tr>
<td>RPEUnknownOrganizationIdentifier</td>
<td>The organization included in the message is not known by the receiving system.</td>
</tr>
<tr>
<td>RPEUnknownActivityIdentifier</td>
<td>The requested activity identifier is not known by the receiving system.</td>
</tr>
</tbody>
</table>
### 3.20.4.2.3 Expected Actions

The Process Activity Executor shall consume the set of process definitions. If a SOAP fault is received then this fault should be handled based on the business rules of the system.

### 3.20.5 Security Considerations

See QRPH TF-1:X.5.

### 3.20.6 Protocol Requirements

The Retrieve Process Definitions Request and Response shall be transmitted using Synchronous Web Services Exchange, according to the requirements specified in ITI TF-2x: Appendix V Web Services for IHE Transactions.

#### Table 3.20.6-1: WSDL Namespace Definitions

<table>
<thead>
<tr>
<th>Namespace</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ihe</td>
<td>urn:ihe:qrph:rpe:2009</td>
</tr>
<tr>
<td>soap12</td>
<td><a href="http://schemas.xmlsoap.org/wsd/soap12/">http://schemas.xmlsoap.org/wsd/soap12/</a></td>
</tr>
<tr>
<td>wsaw</td>
<td><a href="http://www.w3.org/2005/08/addressing">http://www.w3.org/2005/08/addressing</a></td>
</tr>
<tr>
<td>xsd</td>
<td><a href="http://www.w3.org/2001/XMLSchema">http://www.w3.org/2001/XMLSchema</a></td>
</tr>
</tbody>
</table>

These are the requirements for the Retrieve Process Definitions transaction presented in the order in which they would appear in the WSDL definition:

- The following types shall be imported (xsd:import) in the /definitions/types section:
  - namespace="urn:ihe:qrph:rpe:2009", schema="RPE.xsd"
- The /definitions/message/part/@element attribute of the Retrieve Process Definitions Request message shall be defined as: “ihe:RetrieveProcessDefinitionsRequest”
- The /definitions/message/part/@element attribute of the Retrieve Process Definitions Response message shall be defined as: “ihe:RetrieveProcessDefinitionsResponse”
- The /definitions/portType/operation/input/@wsaw:Action attribute for the Retrieve Process Definitions Request message shall be defined as “urn:ihe:qrph:rpe:2009:RetrieveProcessDefinitions”
- The /definitions/binding/operation/soap12:operation/@soapAction attribute shall be defined as “urn:ihe:qrph:rpe:2009:RetrieveProcessDefinitions”
These are the requirements that affect the wire format of the SOAP message. The other WSDL properties are only used within the WSDL definition and do not affect interoperability. Full sample request and response messages are in Section 3.20.6.1 Sample SOAP Messages.

### 3.20.6.1 Sample SOAP Messages

The samples in the following two sections show a typical request and corresponding response as contained in a SOAP Body. **Note: the SOAP Header should be populated according to the IHE ITI TF-2x: Appendix V: Web Services for IHE Transactions.**

Full WSDLs can be found on the IHE FTP site at: 

#### 3.20.6.1.1 Sample Retrieve Process Definitions SOAP Request

```xml
<soap:Body>
  <rpe:processDefinitionIdentifier root="1.2.3.4" extension="1006"/>
</rpe:RetrieveProcessDefinitionsRequest>
</soap:Body>
```

#### 3.20.6.1.2 Sample Retrieve Process Definitions SOAP Response

```xml
<soap:Body>
<rpe:RetrieveProcessDefinitionsResponse
  <rpe:processDefinition processDefinitionLanguage="http://www.omg.org/spec/BPMN/2.0/">
      <!-- Import schemas and define RPE and HL7 element items for reference by process data object variables -->
      <import importType="http://www.w3.org/2001/XMLSchema" location="RPE.xsd" namespace="urn:ihe:qrph:rpe:2009"/>
      <import importType="http://www.w3.org/2001/XMLSchema" location="HL7-SD/StudyDesign.xsd" namespace="urn:hl7-org:v3"/>
      <itemDefinition id="InitiateProcessRequestItem" structureRef="rpe:InitiateProcessRequest"/>
      <itemDefinition id="RequestContextItem" structureRef="rpe:requestContext"/>
    </definitions>
  </rpe:processDefinition>
</rpe:RetrieveProcessDefinitionsResponse>
</soap:Body>

3.22 Publish Process Definitions [QRPH-22]

This section corresponds to transaction [QRPH-22] of the IHE QRPH Transaction Framework. [QRPH-22] is used by the Process Definition Manager, the Process Activity Executor and the Process State Manager Actors.
3.22.1 Scope
This transaction involves a Process Definition Manager publishing process definitions to a Process Activity Executor or Process State Manager.

3.22.2 Use Case Roles

Actor: Process Activity Executor
Role: A system that knows how to execute activities that are part of a process.

Actor: Process State Manager
Role: A system that manages the runtime state of a process.

Actor: Process Definition Manager
Role: A system that provides a set of process definitions.

3.22.3 Referenced Standards
Implementers of this transaction shall comply with all requirements described in:
- ITI TF-2x: Appendix V Web Services for IHE Transactions
3.22.4 Interaction Diagram

3.22.4.1 Publish Process Definitions Message

3.22.4.1.1 Trigger Events
The Process Definition Manager defines a new process.

3.22.4.1.2 Message Semantics
The following parameters are specified for the body of this transaction.
Table 3.22.4.1.2-1: Publish Process Definitions Request Parameters

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Optionality</th>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>processDefinition</td>
<td>R</td>
<td>One process definition per process definition identifier requested.</td>
<td>A complex XML element of type specified per optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>processDefinitionLanguage attribute</td>
</tr>
<tr>
<td>processDefinitionLanguage</td>
<td>R</td>
<td>XML attribute</td>
<td>Defined by content profile.</td>
</tr>
</tbody>
</table>

Section 3.20.6 describes the Web Services protocol requirements and the format of the message in full detail.

3.22.4.1.3 Expected Actions

Upon receipt of the Publish Process Definitions Message, the Process Activity Executor or Process State Manager SHALL respond with a message indicating that the process was received. That message is defined in Table 3.22.4.1.2-2.

The successful response shall be one or more well-formed XML documents matching the supplied identifiers.

Otherwise SOAP faults shall be generated accordingly.

3.22.4.2 Publish Process Definitions Response Message

3.22.4.2.1 Trigger Events

This message occurs upon receipt of a Publish Process Definitions Request.

3.22.4.2.2 Message Semantics

A matching list of one or more process definitions is returned. The format of each process definition is a well-formed XML document.

Table 3.22.4.1.2-2: Publish Process Definitions Response Parameters

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Optionality</th>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>@status</td>
<td>R</td>
<td>Indicates whether processing was successful</td>
<td>See Table 3.20.4.1.2-3.</td>
</tr>
<tr>
<td>responseCode</td>
<td>R</td>
<td>A code indicating successful receipt of the message.</td>
<td>PROCESS_RECEIVED</td>
</tr>
<tr>
<td>processErrorList</td>
<td>O</td>
<td>If errors were encountered in processing (see @status),</td>
<td>See Table 3.20.4.1.2-4.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>error list is defined here.</td>
<td></td>
</tr>
</tbody>
</table>
3.22.4.2.3 Expected Actions

The Process Activity Executor or Process State Manager consumes the received process information.

3.22.5 Security Considerations

See QRPH TF-1:X.5.

3.22.6 Protocol Requirements

The Publish Process Definitions Request and Response shall be transmitted using Synchronous Web Services Exchange, according to the requirements specified in ITI TF-2x: Appendix V Web Services for IHE Transactions.

<table>
<thead>
<tr>
<th>Table 3.22.6-1: WSDL Namespace Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>ihe</td>
</tr>
<tr>
<td>soap12</td>
</tr>
<tr>
<td>wsaw</td>
</tr>
<tr>
<td>xsd</td>
</tr>
</tbody>
</table>

These are the requirements for the Publish Process Definitions transaction presented in the order in which they would appear in the WSDL definition:

- The following types shall be imported (xsd:import) in the /definitions/types section:
  - namespace="urn:ihe:qrph:rpe:2009", schema="RPE.xsd"
- The /definitions/message/part/@element attribute of the Publish Process Definitions Request message shall be defined as: “ihe:PublishProcessDefinitionsRequest”
- The /definitions/message/part/@element attribute of the Publish Process Definitions Response message shall be defined as: “ihe:PublishProcessDefinitionsResponse”
- The /definitions/binding/operation/soap12:operation/@soapAction attribute shall be defined as “urn:ihe:qrph:rpe:2009:PublishProcessDefinitions”
These are the requirements that affect the wire format of the SOAP message. The other WSDL properties are only used within the WSDL definition and do not affect interoperability. Full sample request and response messages are in Section 3.20.6.1 Sample SOAP Messages.

3.22.6.1 Sample SOAP Messages

The samples in the following two sections show a typical request and corresponding response as contained in a SOAP Body. Note: the SOAP Header should be populated according to the IHE ITI TF-2x: Appendix V: Web Services for IHE Transactions.

Full WSDLs can be found on the IHE FTP site at: ftp://ftp.ihe.net/TF_Implementation_Material/QRPH/RPE

3.22.6.1.1 Sample Publish Process Definitions SOAP Request

Note to the editor: please keep the following format for the sample text – courier new, 8pt, no spacing before and after the paragraph, tab stops every 1/8 of an inch for the first inch.

```xml
<soap:Body>
    <rpe:processDefinition processDefinitionLanguage="http://www.omg.org/spec/BPMN/2.0/"
      xmlns:rpe="urn:ihe:qrph:rpe:2009"
      xmlns:tns="http://www.examples.org"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xmlns:hl7="urn:hl7-org:v3"
      xmlns:rpe="urn:ihe:qrph:rppe:2009" id="_1312929184333">
      <!-- Import schemas and define RPE and HL7 element items for reference by process
      data object variables -->
      <import importType="http://www.w3.org/2001/XMLSchema" location="RPE.xsd"
        namespace="urn:ihe:qrph:rppe:2009"/>
      <import importType="http://www.w3.org/2001/XMLSchema" location="HL7-SD/StudyDesign.xsd"
        namespace="urn:hl7-org:v3"/>
      <itemDefinition id="InitiateProcessRequestItem" structureRef="rpe:InitiateProcessRequest"></itemDefinition>
      <itemDefinition id="RequestContextItem" structureRef="rpe:requestContext"></itemDefinition>
      <itemDefinition id="PatientDataItem" structureRef="rpe:patientData"></itemDefinition>
      <itemDefinition id="EligibilityCriterionItem" structureRef="hl7:eligibilityCriterion"></itemDefinition>
      <itemDefinition id="ObservationItem" structureRef="hl7:observation"></itemDefinition>
      <itemDefinition id="SubstanceAdministrationItem" structureRef="hl7:substanceAdministration"></itemDefinition>
      <message id="InitiateProcessRequest" itemRef="tns:InitiateProcessRequestItem"
        name="Initiate Process Request"/>
      <process isExecutable="false" id="_6" name="IHE RPE Study Design">
        <documentation id="Title">IHE RPE Study Design</documentation>
        <documentation id="Description">A simple single arm example consisting of
          Screen, Treatment and Followup epochs with 3 activities total:
          a screening visit, a treatment visit and a followup visit 6 months later.</documentation>
```
<ioSpecification>
<definitions>

<dataInput isCollection="true" name="Eligibility Criteria" id="EligibilityCriteria" itemSubjectRef="tns:EligibilityCriterionItem"/>
<dataInput name="Request Context" id="RequestContext" itemSubjectRef="tns:RequestContextItem"/>
<dataInput name="Patient Data" id="PatientData" itemSubjectRef="tns:PatientDataItem"/>
<dataInput name="Substance Administration" id="SubstanceAdministration" itemSubjectRef="tns:SubstanceAdministrationItem"/>

<dataOutput isCollection="true" name="Screening Observations" id="ScreeningObservations" itemSubjectRef="tns:ObservationItem"/>
<dataOutput isCollection="true" name="Treatment Observations" id="TreatmentObservations" itemSubjectRef="tns:ObservationItem"/>
<dataOutput isCollection="true" name="Follow-up Observations" id="FollowUpObservations" itemSubjectRef="tns:ObservationItem"/>

...<rpe:processDefinition>
</definitions>
</rpe:processDefinition>
</rpe:PublishProcessDefinitionsResponse>
</soap:Body>

3.22.6.1.2 Sample Publish Process Definitions SOAP Response

Note to the editor: please keep the following format for the sample text – courier new, 8pt, no
spacing before and after the paragraph, tab stops every 1/8 of an inch for the first inch.

3.25 Initiate Process [QRPH-25]

This section corresponds to transaction [QRPH-25] of the IHE QRPH Transaction Framework.
[QRPH-25] is used by the Process State Manager and Process Activity Executor Actors.

3.25.1 Scope

This transaction involves a Process Activity Executor requesting the initiation of a process with a
Process State Manager.

In its request the Process Activity Executor supplies an initial context, including:
• an identifier for the process definition of interest
• an organizational identifier
• a patient identifier
• an endpoint reference for callback notifications (alerts)
• other optional patient data (such as demographics)

3.25.2 Use Case Roles

**Actor:** Process Activity Executor  
**Role:** A system that knows how to execute activities that are part of a process.  
**Actor:** Process State Manager  
**Role:** A system that manages the runtime state of a process.

3.25.3 Referenced Standards

See Section 3.20.3.
### 3.25.4 Interaction Diagram

![Interaction Diagram](image)

### 3.25.4.1 Initiate Process Message

#### 3.25.4.1.1 Trigger Events

The Process Activity Executor, based on human decision or the application of a rule for automatic operation, wants to initiate a process with a Process State Manager.

#### 3.25.4.1.2 Message Semantics

The following parameters are specified for the body of this transaction.

**Table 3.25.4.1.2-1: Initiate Process Request Parameters**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Optionality</th>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>patient</td>
<td>R</td>
<td>Basic demographics for the patient.</td>
<td>Simple patient definition, of type rpe:PatientType.</td>
</tr>
<tr>
<td>candidateID</td>
<td>R</td>
<td>List of patient identities.</td>
<td>HL7 element of type InstanceIdentifier.</td>
</tr>
<tr>
<td>name</td>
<td>O</td>
<td>Patient name</td>
<td>HL7 patient name type</td>
</tr>
<tr>
<td>address</td>
<td>O</td>
<td>Patient address</td>
<td>HL7 address type</td>
</tr>
<tr>
<td>dob</td>
<td>O</td>
<td>Patient date of birth</td>
<td>HL7 datetime type</td>
</tr>
<tr>
<td>processDefinitionIdentifier</td>
<td>R</td>
<td>Identifier for the process definition to be initiated.</td>
<td>HL7 element of type InstanceIdentifier.</td>
</tr>
<tr>
<td>processState</td>
<td>R</td>
<td>Initial state of the process</td>
<td>An XML element of type enumerated string.</td>
</tr>
<tr>
<td>organizationIdentifier</td>
<td>O</td>
<td>Identifier for the involved organization.</td>
<td>HL7 element of type InstanceIdentifier.</td>
</tr>
<tr>
<td>requestorReference</td>
<td>O</td>
<td>The Process Activity Executor callback endpoint for the Process State</td>
<td>WS Addressing EndpointReferenceType</td>
</tr>
</tbody>
</table>
Section 3.25.6 describes the Web Services protocol requirements and the format of the message in full detail.

### 3.25.4.1.3 Expected Actions

The Process State Manager parses the Initiate Process request and, if well-formed, initiates an instance of the given process and acknowledges the request with a returned process context to be used in subsequent transactions.

Otherwise SOAP faults shall be generated accordingly.

### 3.25.4.2 Initiate Process Response Message

#### 3.25.4.2.1 Trigger Events

The message is triggered by a Process State Manager receiving an Initiate Process request from a Process Activity Executor and after successful initiation of the process.

#### 3.25.4.2.2 Message Semantics

The following output parameters are the body of the response.

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Optionality</th>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>@status</td>
<td>R</td>
<td>Indicates whether processing was successful</td>
<td>See Table 3.20.4.1.2-3.</td>
</tr>
<tr>
<td>processIdentifier</td>
<td>R</td>
<td>Identifier for process instance</td>
<td>HL7 element of type InstanceIdentifier.</td>
</tr>
<tr>
<td>processState</td>
<td>R</td>
<td>Initial state of the process</td>
<td>An XML element of type enumerated string.</td>
</tr>
<tr>
<td>assignedOrganizationIdentifier</td>
<td>O</td>
<td>Assigned identifier for organization.</td>
<td>HL7 element of type InstanceIdentifier.</td>
</tr>
<tr>
<td>patient</td>
<td>O</td>
<td>Provided if additional identifiers are assigned to the patient.</td>
<td>Simple patient definition, of type rpe:PatientType.</td>
</tr>
</tbody>
</table>

---

4 CCD is the registered trademark of Health Level Seven International.
3.25.4.2.3 Expected Actions

The Process Activity Executor shall consume the response code based on the business rules of the system. If a SOAP fault is received then this fault should also be handled based on the business rules of the system.

3.25.5 Security Considerations

See QRPH TF-1:X.5.

3.25.6 Protocol Requirements

The Initiate Process request and response shall be transmitted using Synchronous Web Services Exchange, according to the requirements specified in ITI TF-2x: Appendix V Web Services for IHE Transactions.

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Optionality</th>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>processErrorList</td>
<td>O</td>
<td>If errors were encountered in processing (see @status), error list is defined here.</td>
<td>See Table 3.20.4.1.2-4.</td>
</tr>
</tbody>
</table>

Table 3.25.6-1: WSDL Namespace Definitions

<table>
<thead>
<tr>
<th>ihe</th>
<th>urn:ihe:qrph:rpe:2009</th>
</tr>
</thead>
<tbody>
<tr>
<td>soap12</td>
<td><a href="http://schemas.xmlsoap.org/wsdl/soap12/">http://schemas.xmlsoap.org/wsdl/soap12/</a></td>
</tr>
<tr>
<td>wsaw</td>
<td><a href="http://www.w3.org/2005/08/addressing">http://www.w3.org/2005/08/addressing</a></td>
</tr>
<tr>
<td>xsd</td>
<td><a href="http://www.w3.org/2001/XMLSchema">http://www.w3.org/2001/XMLSchema</a></td>
</tr>
</tbody>
</table>

These are the requirements for the Initiate Process transaction presented in the order in which they would appear in the WSDL definition:

- The following types shall be imported (xsd:import) in the /definitions/types section:
  - namespace="urn:ihe:qrph:rpe:2009", schema="RPE.xsd"
- The /definitions/message/part/@element attribute of the Initiate Process Request message shall be defined as: “ihe:InitiateProcessRequest”
- The /definitions/message/part/@element attribute of the Initiate Process Response message shall be defined as: “ihe:InitiateProcessResponse”
- The /definitions/portType/operation/input/@wsaw:Action attribute for the Initiate Process request message shall be defined as “urn:ihe:qrph:rpe:2009:InitiateProcess”
• The /definitions/portType/operation/output/@wsaw:Action attribute for the Initiate Process response message shall be defined as: “urn:ihe:qrph:rpe:2009:InitiateProcessResponse”

• The /definitions/binding/operation/soap12:operation/@soapAction attribute shall be defined as “urn:ihe:qrph:rpe:2009:InitiateProcess”

These are the requirements that affect the wire format of the SOAP message. The other WSDL properties are only used within the WSDL definition and do not affect interoperability. Full sample request and response messages are in the Section 3.25.6.1 Sample SOAP Messages.

3.25.6.1 Sample SOAP Messages

The samples in the following two sections show a typical request and corresponding response as contained in a SOAP Body. Note: the SOAP Header should be populated according to the IHE ITI TF-2x: Appendix V: Web Services for IHE Transactions and in accord with Section 3.25.5 Security Considerations.

Full WSDLs can be found on the IHE FTP site at: ftp://ftp.ihe.net/TF_Implementation_Material/QRPH/RPE

3.25.6.1.1 Sample Initiate Process SOAP Request

Note to the editor: please keep the following format for the sample text – courier new, 8pt, no spacing before and after the paragraph, tab stops every 1/8 of an inch for the first inch.

```xml
<soap:Body>
    <rpe:patient>
      <rpe:candidateID root="" extension="" />
      <rpe:name>
        <hl7:given>John</hl7:given>
        <hl7:family>Smith</hl7:family>
      </rpe:name>
      <rpe:address />
      <rpe:dob value="19990101" />
    </rpe:patient>
    <rpe:processDefinitionIdentifier root="1.2.3.4" extension="1006" />
    <rpe:processState>ACTIVE</rpe:processState>
    <rpe:organizationIdentifier root="1.2.3.5" extension="CIS_CE_CB" />
    <rpe:requestorReference>
      <wsa:Address>http://example.com/fabrikam/acct</wsa:Address>
    </rpe:requestorReference>
  </rpe:InitiateProcessRequest>
</soap:Body>
```

3.25.6.1.2 Sample Initiate Process SOAP Response

Note to the editor: please keep the following format for the sample text – courier new, 8pt, no spacing before and after the paragraph, tab stops every 1/8 of an inch for the first inch.

```xml
<soap:Body>
```

3.26 Retrieve Activities [QRPH-26]

This section corresponds to transaction [QRPH-26] of the IHE QRPH Transaction Framework. [QRPH-26] is used by the Process State Manager and Process Activity Executor Actors.

3.26.1 Scope

The Retrieve Activities transaction enables a Process Activity Executor to retrieve the current set of activities it needs to execute as part of processes in which it is a participant and which are managed by the given Process State Manager.

In its request, the Process Activity Executor supplies the current process context, per Table 3.25.4.2.2-1.

3.26.2 Use Case Roles

**Actor:** Process Activity Executor

**Role:** A system that knows how to execute activities that are part of a process.

**Actor:** Process State Manager

**Role:** A system that manages the runtime state of a process.

3.26.3 Referenced Standards

See Section 3.20.3.
3.26.4 Interaction Diagram

3.26.4.1 Retrieve Activities Message

3.26.4.1.1 Trigger Events
The Process Activity Executor, based upon human decision or application of a rule for automatic operation, wants to retrieve the current set of process activities it needs to perform from a Process State Manager.

3.26.4.1.2 Message Semantics
The current process context (see Table 3.25.4.2.2-1) is the sole input. Section 3.26.6 describes the Web Services protocol requirements and the format of the message in full detail.

3.26.4.1.3 Expected Actions
The Process State Manager parses the Retrieve Activities Request message and, if well-formed, returns the current set of activities for the given organization and patient. If no activities are outstanding an empty response is returned. Otherwise SOAP faults shall be generated accordingly.

3.26.4.2 Retrieve Activities Response Message

3.26.4.2.1 Trigger Events
The message is triggered by a Process State Manager in response to a Retrieve Activities Request from a Process Activity Executor.
3.26.4.2.2 Message Semantics

A list of zero or more process activities is returned. The format of each activity is a well-formed XML document of the type supported by the process of which they are part.

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Optionality</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>@status</td>
<td>R</td>
<td>Indicates whether processing was successful</td>
</tr>
<tr>
<td>activity</td>
<td>R</td>
<td>Current process activity</td>
</tr>
<tr>
<td>activityLanguage</td>
<td>R</td>
<td>XML attribute</td>
</tr>
<tr>
<td>processIdentifier</td>
<td>R</td>
<td>Identifier for process instance</td>
</tr>
<tr>
<td>processState</td>
<td>R</td>
<td>Initial state of the process</td>
</tr>
<tr>
<td>assignedOrganizationIdentifier</td>
<td>O</td>
<td>Assigned identifier for organization.</td>
</tr>
<tr>
<td>patient</td>
<td>O</td>
<td>Provided if additional identifiers are assigned to the patient.</td>
</tr>
<tr>
<td>processErrorList</td>
<td>O</td>
<td>If errors were encountered in processing (see @status), error list is defined</td>
</tr>
</tbody>
</table>

3.26.4.2.3 Expected Actions

The Process Activity Executor shall consume the list of activities based on the business rules of the system. If a SOAP fault is received then this fault should also be handled based on the business rules of the system.

3.26.5 Security Considerations

See QRPH TF-1:X.5.

3.26.6 Protocol Requirements

The Retrieve Activities Request and Response messages shall be transmitted using Synchronous Web Services Exchange, according to the requirements specified in ITI TF-2x: Appendix V Web Services for IHE Transactions.

<table>
<thead>
<tr>
<th>Table 3.26.6-1: WSDL Namespace Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ihe</strong></td>
</tr>
<tr>
<td>soap12</td>
</tr>
<tr>
<td>wsaw</td>
</tr>
</tbody>
</table>
These are the requirements for the Retrieve Activities transaction presented in the order in which they would appear in the WSDL definition:

- The following types shall be imported (xsd:import) in the /definitions/types section:
  - namespace="urn:ihe:qrph:rpe:2009", schema="RPE.xsd"
- The /definitions/message/part/@element attribute of the Retrieve Activities request message shall be defined as: “ihe:RetrieveActivitiesRequest”
- The /definitions/message/part/@element attribute of the Retrieve Activities response message shall be defined as: “ihe:RetrieveActivitiesResponse”
- The /definitions/portType/operation/input/@wsaw:Action attribute for the Retrieve Activities request message shall be defined as “urn:ihe:qrph:rpe:2009:RetrieveActivities”
- The /definitions/portType/operation/output/@wsaw:Action attribute for the Retrieve Activities response message shall be defined as: “urn:ihe:qrph:rpe:2009:RetrieveActivitiesResponse”
- The /definitions/binding/operation/soap12:operation/@soapAction attribute shall be defined as “urn:ihe:qrph:rpe:2009:RetrieveActivities”

These are the requirements that affect the wire format of the SOAP message. The other WSDL properties are only used within the WSDL definition and do not affect interoperability. Full sample request and response messages are in Section 3.26.6.1 Sample SOAP Messages.

### 3.26.6.1 Sample SOAP Messages

The samples in the following two sections show a typical request and corresponding response as contained in a SOAP Body. Note: the SOAP Header should be populated according to the IHE Appendix V: Web Services for IHE Transactions and in accord with Section 3.26.5 Security Considerations.

Full WSDLs can be found on the IHE FTP site at: ftp://ftp.ihe.net/TF_Implementation_Material/QRPH/RPE

#### 3.26.6.1.1 Sample Retrieve Activities SOAP Request

*Note to the editor: please keep the following format for the sample text – courier new, 8pt, no spacing before and after the paragraph, tab stops every 1/8 of an inch for the first inch.*

```xml
<soap:Body>
  <rpe:RetrieveActivitiesRequest xmlns:rpe="urn:ihe:qrph:rpe:2009">
    <rpe:patient/>
    <rpe:candidateID root="" extension="" />
    <rpe:name/>
  </rpe:RetrieveActivitiesRequest>
</soap:Body>
```
3.26.6.1.2 Sample Retrieve Activities SOAP Response

Note to the editor: please keep the following format for the sample text – courier new, 8pt, no spacing before and after the paragraph, tab stops every 1/8 of an inch for the first inch.

3.27 Update Activity [QRPH-27]

This section corresponds to transaction [QRPH-27] of the IHE QRPH Transaction Framework. [QRPH-27] is used by the Process State Manager and Process Activity Executor Actors.

3.27.1 Scope

Update Activity allows a Process Activity Executor to provide an update on an activity’s state or data to a Process State Manager for a process in which it is a participant.

In its request, the Process Activity Executor supplies the current process context, per Table 3.25.4.2.2-1, and the updated activity.
3.27.2 Use Case Roles

Actor: Process Activity Executor
Role: A system that knows how to execute activities that are part of a process.

Actor: Process State Manager
Role: A system that manages the runtime state of a process.

3.27.3 Referenced Standards
See Section 3.20.3.

3.27.4 Interaction Diagram

3.27.4.1 Update Activity Message

3.27.4.1.1 Trigger Events
The Process Activity Executor, based upon human decision or application of a rule for automatic operation, wants to update the Process State Manager with the current state or data associated with an activity it is performing.
### 3.27.4.1.2 Message Semantics

The following parameters are specified for the body of this transaction.

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Optionality</th>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>activity</td>
<td>R</td>
<td>One or more process activities.</td>
<td>A complex XML element representing an activity within a process as specified per the activityLanguage attribute</td>
</tr>
<tr>
<td>activityLanguage</td>
<td>R</td>
<td>XML attribute</td>
<td>Defined by content profile.</td>
</tr>
<tr>
<td>processIdentifier</td>
<td>R</td>
<td>Identifier for process instance</td>
<td>HL7 element of type Instanceldentifier.</td>
</tr>
<tr>
<td>processState</td>
<td>R</td>
<td>State of the process</td>
<td>An XML element of type enumerated string.</td>
</tr>
<tr>
<td>assignedOrganizationIdentifier</td>
<td>O</td>
<td>Assigned identifier for organization.</td>
<td>HL7 element of type Instanceldentifier.</td>
</tr>
<tr>
<td>patient</td>
<td>O</td>
<td>Provided if additional identifiers are assigned to the patient.</td>
<td>Simple patient definition, of type rpe:PatientType.</td>
</tr>
</tbody>
</table>

Section 3.27.6 describes the Web Services protocol requirements and the format of the message in full detail.

### 3.27.4.1.3 Expected Actions

The Process State Manager parses the Update Activity request and if well-formed, updates the given activity for the state and/or data supplied.

If the activity is updated as a result of this transaction the Process State Manager acknowledges the request with a response code of “ACTIVITY_UPDATED”.

Otherwise SOAP faults shall be generated accordingly.

### 3.27.4.2 Update Activity Response Message

#### 3.27.4.2.1 Trigger Events

The message is triggered by a Process State Manager receiving an Update Activity request from a Process Activity Executor.

#### 3.27.4.2.2 Message Semantics

A responseCode XML element of type string is returned which confirms successful update of the activity with a value of “ACTIVITY_UPDATED”.
3.27.4.2.3 Expected Actions

The Process Activity Executor shall consume the response code based on the business rules of the system. If a SOAP fault is received then this fault should also be handled based on the business rules of the system.

3.27.5 Security Considerations

See Section QRPH TF-1:X.5

3.27.6 Protocol Requirements

The Update Activity request and response shall be transmitted using Synchronous Web Services Exchange, according to the requirements specified in ITI TF-2x: Appendix V Web Services for IHE Transactions.

<table>
<thead>
<tr>
<th>Table 3.27.6-1: WSDL Namespace Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>ihe</td>
</tr>
<tr>
<td>soap12</td>
</tr>
<tr>
<td>wsaw</td>
</tr>
<tr>
<td>xsd</td>
</tr>
</tbody>
</table>

These are the requirements for the Update Activity transaction presented in the order in which they would appear in the WSDL definition:

- The following types shall be imported (xds:import) in the /definitions/types section:
  - namespace=”urn:ihe:qrph:rpe:2009”, schema=”RPE.xsd”

- The /definitions/message/part/@element attribute of the Update Activity request message shall be defined as: “ihe:UpdateActivityRequest”

- The /definitions/message/part/@element attribute of the Update Activity response message shall be defined as: “ihe:UpdateActivityResponse”

- The /definitions/portType/operation/input/@wsaw:Action attribute for the Update Activity request message shall be defined as “urn:ihe:qrph:rpe:2009:UpdateActivity”

- The /definitions/portType/operation/output/@wsaw:Action attribute for the Update Activity response message shall be defined as: “urn:ihe:qrph:rpe:2009:UpdateActivityResponse”

- The /definitions/binding/operation/soap12:operation/@soapAction attribute shall be defined as “urn:ihe:qrph:rpe:2009:UpdateActivity”

These are the requirements that affect the wire format of the SOAP message. The other WSDL properties are only used within the WSDL definition and do not affect interoperability. Full sample request and response messages are in the Section 3.27.6.1 Sample SOAP Messages.
3.27.6.1 Sample SOAP Messages

The samples in the following two sections show a typical request and corresponding response as contained in a SOAP Body. *Note the SOAP Header should be populated according to the IHE Appendix V: Web Services for IHE Transactions and in accord with Section 3.27.5 Security Considerations.*

Full WSDLs can be found on the IHE FTP site at: [ftp://ftp.ihe.net/TF_Implementation_Material/QRPH/RPE](ftp://ftp.ihe.net/TF_Implementation_Material/QRPH/RPE)

3.27.6.1.1 Sample Update Activity SOAP Request

*Note to the editor: please keep the following format for the sample text – courier new, 8pt, no spacing before and after the paragraph, tab stops every 1/8 of an inch for the first inch.*

```xml
<soap:Body>
    <rpe:patient>
      <rpe:candidateID root="" extension="" />
      <rpe:name>
        <hl7:given>John</hl7:given>
        <hl7:family>Smith</hl7:family>
      </rpe:name>
      <rpe:address />
      <rpe:dob value="19990101" />
    </rpe:patient>
    <rpe:processIdentifier root="1.2.3.5" extension="1006-1" />
    <rpe:processDefinitionIdentifier root="1.2.3.4" extension="1006" />
    <rpe:processState>ACTIVE</rpe:processState>
    <rpe:assignedOrganizationIdentifier root="1.2.3.5" extension="CIS_CE_CB" />
    <rpe:activity activityLanguage="http://www.omg.org/spec/BPMN/2.0/">
      <userTask xmlns="http://www.omg.org/spec/BPMN/20100524/MODEL" implementation="##unspecified" name="Screening Activity" id="ScreeningActivity">
        <dataOutputAssociation id="ScreeningObservations_DataOutput">
          <targetRef>ScreeningObservations</targetRef>
          <assignment>
            <from><![CDATA[
              <hl7:observation xmlns:hl7="urn:hl7-org:v3" classCode="OBS" moodCode="CRT">
                <id root="1.2.5.2.3.4" extension="DBP" />
                <code code="8462-4" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC" displayName="BP Diastolic">
                  <originalText>Diastolic Blood Pressure</originalText>
                </code>
                <statusCode code="active"/>
                <value xsi:type="PQ" value="120" unit="mmHg"/>
              </hl7:observation>
            ]]>></from>
            <to>ScreeningObservations</to>
          </assignment>
        </dataOutputAssociation>
      </userTask>
    </rpe:activity>
  </rpe:UpdateActivityRequest>
</soap:Body>
```
3.27.6.1.2 Sample Update Activity SOAP Response

Note to the editor: please keep the following format for the sample text – courier new, 8pt, no spacing before and after the paragraph, tab stops every 1/8 of an inch for the first inch.

3.28 Send Process State Alert [QRPH-28]

This section corresponds to transaction [QRPH-28] of the IHE QRPH Transaction Framework. [QRPH-28] is used by the Process State Manager and Process Activity Executor Actors.

3.28.1 Scope

This transaction involves either:

- a Process Activity Executor alerting a Process State Manager of a change that affects the state of a process it initiated,

OR

- a Process State Manager alerting a Process Activity Executor of a change associated with a process it initiated with the Process State Manager, in which case the alert contains.

In either case, the request passed by the actor supplies the current process context as per Table 3.25.4.2.2-1 and a process state value.
3.28.2 Use Case Roles

**Actor:** Process Activity Executor  
**Role:** A system that knows how to execute activities that are part of a process.

**Actor:** Process State Manager  
**Role:** A system that manages the runtime state of a process.

3.28.3 Referenced Standards

See Section 3.20.3.

3.28.4 Interaction Diagram
3.28.4.1 Send Process State Alert Message

3.28.4.1.1 Trigger Events

Either:

- the Process Activity Executor, based upon human decision or application of a rule for automatic operation, wants to alert the Process State Manager of a change that affects a process it initiated

  OR

- the Process State Manager, based upon human decision or application of a rule for automatic operation, wants to alert the Process Activity Executor of a change associated with a process it initiated with the Process State Manager.

3.28.4.1.2 Message Semantics

The following parameters are specified for the body of this transaction.

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Optionality</th>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>patient</td>
<td>O</td>
<td>Basic demographics for the patient.</td>
<td>Simple patient definition, of type rpe:PatientType.</td>
</tr>
<tr>
<td>processDefinitionIdentifier</td>
<td>R</td>
<td>Identifier for the process definition to be initiated.</td>
<td>HL7 element of type InstanceIdentifier.</td>
</tr>
<tr>
<td>processIdentifier</td>
<td>R</td>
<td>Identifier for process instance</td>
<td>HL7 element of type InstanceIdentifier.</td>
</tr>
<tr>
<td>processState</td>
<td>R</td>
<td>The process state change being alerted.</td>
<td>An XML element of type enumerated string.</td>
</tr>
</tbody>
</table>
Section 3.28.6 describes the Web Services protocol requirements and the format of the message in full detail.

### 3.28.4.1.3 Expected Actions

The Process Activity Executor or Process State Manager parses the Send Process State Alert request and if well-formed acknowledges the request with a response code of “ALERT_RECEIVED”.

Otherwise SOAP faults shall be generated accordingly.

### 3.28.4.2 Send Process State Alert Response Message

#### 3.28.4.2.1 Trigger Events

The message is triggered by a Process Activity Executor or Process State Manager receiving a well-formed Send Process State Alert request.

#### 3.28.4.2.2 Message Semantics

A responseCode XML element of type string is returned to confirm receipt of the alert with a value of “ALERT_RECEIVED”.

#### 3.28.4.2.3 Expected Actions

The Process Activity Executor or Process State Manager shall consume the response code based on the business rules of the system. If a SOAP fault is received then this fault should also be handled based on the business rules of the system.

### 3.28.5 Security Considerations

See QRPH TF-1:X.5.

### 3.28.6 Protocol Requirements

The Send Process State Alert request and response shall be transmitted using Synchronous Web Services Exchange, according to the requirements specified in ITI TF-2x: Appendix V Web Services for IHE Transactions.
These are the requirements for the Send Process State Alert transaction presented in the order in which they would appear in the WSDL definition:

- The following types shall be imported (xsd:import) in the /definitions/types section:
  - namespace="urn:ihe:qrph:rpe:2009", schema="RPE.xsd"

- The /definitions/message/part/@element attribute of the Send Process State Alert request message shall be defined as: “ihe:SendProcessStateAlertRequest”

- The /definitions/message/part/@element attribute of the Send Process State Alert response message shall be defined as: “ihe:SendProcessStateAlertResponse”

- The /definitions/portType/operation/input/@wsaw:Action attribute for the Send Process State Alert request message shall be defined as: “urn:ihe:qrph:rpe:2009:SendProcessStateAlert”


- The /definitions/binding/operation/soap12:operation/@soapAction attribute shall be defined as “urn:ihe:qrph:rpe:2009:SendProcessStateAlert”

These are the requirements that affect the wire format of the SOAP message. The other WSDL properties are only used within the WSDL definition and do not affect interoperability. Full sample request and response messages are in Section 3.28.6.1 Sample SOAP Messages.

### 3.28.6.1 Sample SOAP Messages

The samples in the following two sections show a typical request and corresponding response as contained in a SOAP Body.

*Note: The SOAP Header should be populated according to the ITI TF-2x: Appendix V Web Services for IHE Transactions and in accord with Section 3.28.5 Security Considerations.*

Full WSDLs can be found on the IHE FTP site at

ftp://ftp.ihe.net/TFIMPLEMENTATION_MATERIAL/QRPH/RPE
3.28.6.1.1 Sample Send Process State Alert SOAP Request

Note to the editor: please keep the following format for the sample text – courier new, 8pt, no spacing before and after the paragraph, tab stops every 1/8 of an inch for the first inch.

Process State Manager to Process Activity Executor (notification of patient enrollment):

```xml
<soap:Body>
    <rpe:patient>
      <rpe:candidateID root="" extension="" />
      <rpe:name>
        <hl7:given>John</hl7:given>
        <hl7:family>Smith</hl7:family>
      </rpe:name>
      <rpe:address />
      <rpe:doe value="19990101" />
      <rpe:processIdentifier root="1.2.3.5" extension="1006-1" />
      <rpe:processDefinitionIdentifier root="1.2.3.4" extension="1006" />
      <rpe:processState>ACTIVE</rpe:processState>
      <rpe:assignedOrganizationIdentifier root="1.2.3.5" extension="CIS_CE_CB" />
    </rpe:patient>
  </rpe:SendProcessStateAlertRequest>
</soap:Body>
```

3.28.6.1.2 Sample Send Process State Alert SOAP Response

Note to the editor: please keep the following format for the sample text – courier new, 8pt, no spacing before and after the paragraph, tab stops every 1/8 of an inch for the first inch.

```xml
<soap:Body>
  </rpe:SendProcessStateAlertResponse>
</soap:Body>
```