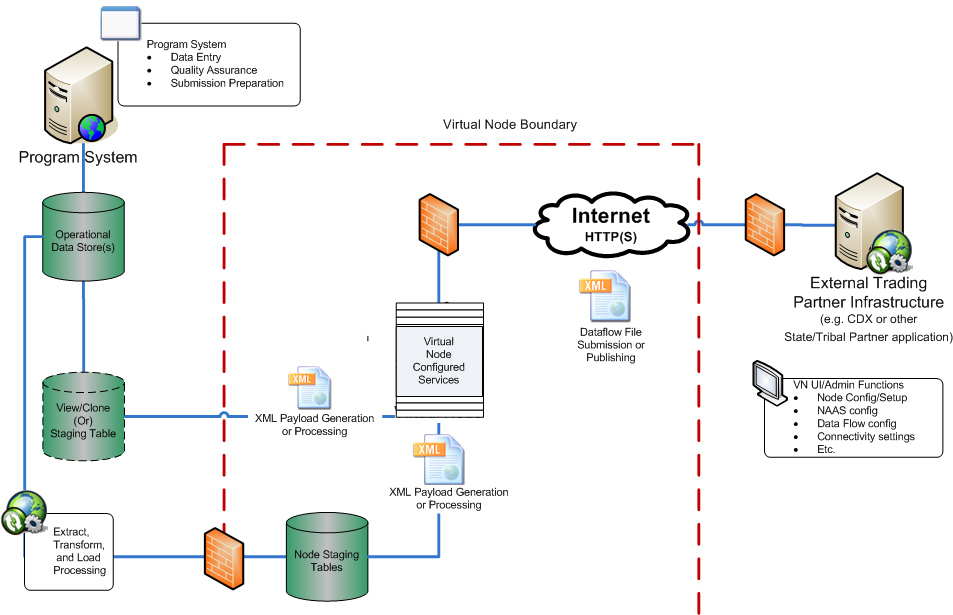
Draft Exchange Network Virtual Node Description and Architecture

This document describes the anticipated architecture and example scenario implementations available to the Exchange Network community as a part of EPA’s Exchange Network Virtual Node services. These services are currently being discussed by the Virtual Node Integrated Project Team (IPT)[[1]](#footnote-1) and may undergo change as members of the IPT continue to discuss detailed business use cases and requirements.

## Draft Virtual Node Architecture Concept

The figure below depicts the proposed architectural approach that has emerged from the Virtual Node IPT team meetings. The bullets that follow the figure discuss some of the key components and concepts depicted in this diagram.



The following key components are depicted in the diagram above.

* **Virtual Node Boundary/Virtual Node:** The dotted line labeled Virtual Node Boundary delineates the components that would need to be managed by the EN Trading Partner (each described in detail below) as compared to the services configured by the EN Trading Partner within the Virtual Node infrastructure. The Virtual Node will meet all applicable EN Node specifications and include features such as role based security access, scheduling, data flow configuration wizards, etc. These features are being discussed by the Virtual Node IPT. The VN UI/Admin Functions and Core Node Services boxes in the lower right diagram apply to the Virtual Node platform.
* **Program System and Operational Data Store:** The Program System and Operational Data Store(s) represent the source system application and data storage. Trading Partners interested in making their data available on the Exchange Network (EN) for use cases that include EN Submits or for data flows related to data publishing (Query or Solicit) require a source data system.
* **View/Clone (Or) Staging Table(s):** In some cases, an EN Trading Partner may want to allow for a connection from the Virtual Node to a View/Clone of the Operational Data store to enable simple, agile, and/or rapid data access for data publishing purposes (rather than require data to be loaded into a staging area within the Virtual Node infrastructure). This component represents the portion of an EN Trading Partner’s architecture that the Virtual Node could optionally connect to directly to provide data access.
* **Extract, Transform and Load (ETL) Processing:** In some cases, EN Trading Partners may have constraints that do not allow for direct access from the Virtual Node to local data stores within their DMZ. In these cases, an ETL process could be used to extract the appropriate data from the Operational Data Store(s) and load the data into Node Staging Tables that are part of the centrally maintained Virtual Node architecture. The ETL process would be created and maintained by the EN Trading Partner.
* **Node Staging Tables:** In cases where an EN Trading Partner requires Virtual Node Staging Tables, these can be hosted in the cloud. The Virtual Node would connect to these tables in order to fulfill EN requests.
* **XML Payload Generation or Processing:** These components represent the process by which XML payloads are either generated or inserted into the applicable datastore(s). For data publishing and traditional national system flows, this component would manage the file generation. In the event that an EN Trading Partner uses the VN to accept incoming transactions, this component would manage the receipt and loading of submissions into the applicable datastore.
* **Trading Partner Infrastructure:** The Trading Partner in the diagram could be any external Exchange Network Node presence or any Trading Partner application embedded with Node Client functionality (e.g., a web page, mobile app, etc.). In the case of an EN Submit or Solicit this component would most likely be a EN Node server that would be able to receive the Submit from the VN implementation. In the case of EN Query, the request could be generated by any application with the ability to generate and consume an EN Query service. In the event that an EN Trading Partner uses the Virtual Node to accept incoming transactions, this component could be either another EN Node server or an application that functions as an EN client and is submitting data to the EN Trading Partners instance of the Virtual Node.

## Implementation Options Discussion

This section of this document is intended to describe some of the envisioned implementation options that the Virtual Node IPT has identified for interested parties who may adopt the usage of the Virtual Node platform. Note that the Virtual Node platform could be used by Exchange Network Trading Partners to orchestrate any set of standard Exchange Network Node transaction patterns. These include: (1) Submit data to other Trading Partners on the EN, (2) Publish data via EN Query and Submit processes, (3) Publish data via VN REST service and (4) Receive data from external EN Trading Partners.

1. **Submit data to other Trading Partners on the EN:** This EN pattern involves a data flow where an EN Trading Partner needs to submit data to another Trading Partner. A standard example of this type of use case would be where a Trading Partner would adopt the Virtual Node to manage the process of one or more of the national system flows. Note that for EN Trading Partners that have adopted staging table and plug-in concepts, the architectural principles endorsed by the IPT allow for the staging tables used for these flows to be either located within the Trading Partners architecture or be managed as part of the Virtual Node architecture.
2. **Publish data via EN Query and Solicit processes:** This EN pattern involves a data flow where an EN Trading Partner requests data via an EN Query or Solicit web service from an EN Trading Partner who is using the Virtual Node. Note that Virtual Node implementers can configure these data flows to connect to data stores within their architecture or to staging tables within the Virtual Node architecture.
3. **Publish data via Virtual Node REST services:** This EN pattern involves a data flow where an EN Trading Partner or the public requests data via a REST service on a Virtual Node which is created automatically when a service is configured. Note that Virtual Node implementers can configure these data flows to connect to data stores within their architecture or to staging tables within the Virtual Node architecture.
4. **Receive data from external EN Trading Partners:** This EN pattern involves a data flow where an EN Trading Partner submits data via an EN Submit web service to an EN Trading Partner who is using the Virtual Node. After the Virtual Node authenticates the requester and validates the payload, Virtual Node implementers can configure these data flows to connect to data stores within their architecture or to staging tables within the Virtual Node architecture in order to receive and stage the data for downstream business processes.

1. <http://www.exchangenetwork.net/virtual-node-ipt/> [↑](#footnote-ref-1)